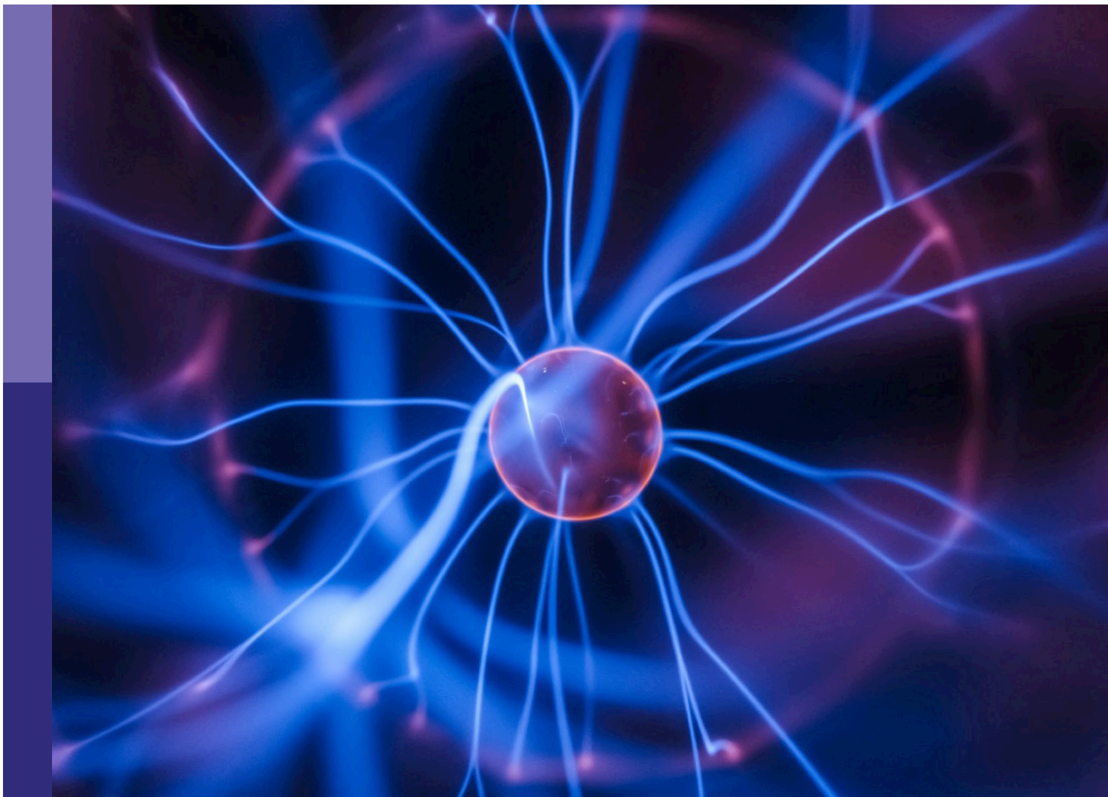
frontiers | Research Topics

Interdisciplinary approaches to the structure and performance of interdependent autonomous human machine teams and systems (A-HMT-S)

Edited by
William Frere Lawless, Donald Sofge and Daniel M. Lofaro

Published in
Frontiers in Physics



ISSN 1664-8714
ISBN 978-2-83251-930-1
DOI 10.3389/978-2-83251-930-1

About Frontiers

Frontiers is more than just an open access publisher of scholarly articles: it is a pioneering approach to the world of academia, radically improving the way scholarly research is managed. The grand vision of Frontiers is a world where all people have an equal opportunity to seek, share and generate knowledge. Frontiers provides immediate and permanent online open access to all its publications, but this alone is not enough to realize our grand goals.

Frontiers journal series

The Frontiers journal series is a multi-tier and interdisciplinary set of open-access, online journals, promising a paradigm shift from the current review, selection and dissemination processes in academic publishing. All Frontiers journals are driven by researchers for researchers; therefore, they constitute a service to the scholarly community. At the same time, the *Frontiers journal series* operates on a revolutionary invention, the tiered publishing system, initially addressing specific communities of scholars, and gradually climbing up to broader public understanding, thus serving the interests of the lay society, too.

Dedication to quality

Each Frontiers article is a landmark of the highest quality, thanks to genuinely collaborative interactions between authors and review editors, who include some of the world's best academicians. Research must be certified by peers before entering a stream of knowledge that may eventually reach the public - and shape society; therefore, Frontiers only applies the most rigorous and unbiased reviews. Frontiers revolutionizes research publishing by freely delivering the most outstanding research, evaluated with no bias from both the academic and social point of view. By applying the most advanced information technologies, Frontiers is catapulting scholarly publishing into a new generation.

What are Frontiers Research Topics?

Frontiers Research Topics are very popular trademarks of the *Frontiers journals series*: they are collections of at least ten articles, all centered on a particular subject. With their unique mix of varied contributions from Original Research to Review Articles, Frontiers Research Topics unify the most influential researchers, the latest key findings and historical advances in a hot research area.

Find out more on how to host your own Frontiers Research Topic or contribute to one as an author by contacting the Frontiers editorial office: frontiersin.org/about/contact

Interdisciplinary approaches to the structure and performance of interdependent autonomous human machine teams and systems (A-HMT-S)

Topic editors

William Frere Lawless — Paine College, United States
Donald Sofge — Naval Research Laboratory, United States
Daniel M. Lofaro — Naval Research Laboratory, United States

Citation

Lawless, W. F., Sofge, D., Lofaro, D. M., eds. (2023). *Interdisciplinary approaches to the structure and performance of interdependent autonomous human machine teams and systems (A-HMT-S)*. Lausanne: Frontiers Media SA.
doi: 10.3389/978-2-83251-930-1

Table of contents

frontiers | Frontiers in Physics

TYPE Editorial
PUBLISHED 01 March 2023
DOI 10.3389/fphy.2023.1150796

Editorial: Interdisciplinary approaches to the structure and performance of interdependent autonomous human machine teams and systems

W. F. Lawless[1]*, Donald A. Sofge[2], Daniel Lofaro[2] and Ranjeev Mittu[2]

[1]Paine College, Augusta, GA, United States, [2]United States Naval Research Laboratory, Washington, DC, United States

KEYWORDS

interdependency, data dependency, autonomy, human-machine teams, systems

Check for updates

OPEN ACCESS

EDITED AND REVIEWED BY
Alex Hansen,
Norwegian University of Science and Technology, Norway

*CORRESPONDENCE
W. F. Lawless,
w.lawless@icloud.com

SPECIALTY SECTION
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

RECEIVED 25 January 2023
ACCEPTED 13 February 2023
PUBLISHED 01 March 2023

CITATION
Lawless WF, Sofge DA, Lofaro D and Mittu R (2023), Editorial: Interdisciplinary approaches to the structure and performance of interdependent autonomous human machine teams and systems.
Front. Phys. 11:1150796.
doi: 10.3389/fphy.2023.1150796

Editorial on the Research Topic
Interdisciplinary approaches to the structure and performance of interdependent autonomous human machine teams and systems

Our Research Topic seeks to advance the physics of autonomous human-machine teams with a mathematical, generalizable model [1]. However, limited team science exists (e.g., aircrews; in [2]). Why? Team science has been hindered by relying on observing how "independent" individuals act and communicate (*viz.*, i.i.d. data; [3,4]), but independent data cannot reproduce the interdependence observed in teams [5]. In agreement, the National Academy of Sciences stated: The "performance of a team is not decomposable to, or an aggregation of, individual performances" ([6], p. 11), evidence of non-factorable teams and data dependency, requiring random searches to find well-fitted teammates, all characterized by fewer degrees of freedom and reduced entropy from interdependence. We review what else we know about a physics of autonomous human-machine teams.

First, we argue that state-dependency [7] rescues traditional social science from its current validation (e.g., "implicit" bias; [8,9]) and replication crises ([10]; e.g., attempts to reduce bias are "dispiriting" [11]), caused by assuming that cognition subsumes individual behavior, needing only independent data (i.i.d.) for teams. The result: Traditional models include large language models like OpenAI's ChatGPT and game theory. Strictly cognitive, ChatGPT and two-person games are assumed to easily connect to reality, but ChatGPT skeptics exist ([12]; [13]); and in *Science* [14], real-world multi-agent approaches are "currently out of reach for state-of-the-art AI methods." Previewed in *Science*, "real-world, large-scale multiagent problems … are currently unsolvable" [15].

Second, to describe interdependence between cogition and behavior, Bohr, the quantum pioneer [16,17]) borrowed "complementary" from psychologist, William James [18]. Later, but long before the Academy's 2021 report, Schrödinger [19] wrote that entanglement meant "the best possible knowledge of a *whole* does not necessarily include the best possible knowledge of all its *parts,* even though they may be entirely separate." [20] borrowed

Schrödinger's state-dependency to found social psychology; and engineers [21] to found Systems Engineering, a concept mostly abandoned until resurrected by the Academy's 2015 report on team interdependence [5].

Third, generalizing the Academy's 2021 claim while adhering to the laws of thermodynamics, data dependency arises when individuals become teammates, reducing degrees of freedom as a team coheres. With coherence, entropy decreases, increasing the power of a team's productivity; however, when interviewed as individuals, coherence is lost.

Fourth, testing for data dependence in teams has proved successful. Treating the structure of a team as key for autonomous agents, assuming a team's size matches a problem [22], with [23] "invisible hand" as baseline, team structure ranges from a group of individuals to a coherent team, generating from least to maximum team power. Several barriers lie ahead; e.g., the tradeoff between structure and performance may be a mathematical cul-de-sac, yet one we have generalized to multiple phenomena [1,24–26]: uncertainty and conflict (where logic fails [27]); deception; blue-red team challenges; emotion; vulnerability; innovation; and mergers (*viz.*, random searches for team fittedness).

Fifth, by exploiting data dependency, uncertainty reduced inside of bounded spaces may recover rational choice [28], game theory and [29] bounded rationality: For example, cross-examination in a courtroom is the greatest means to discovering truth [30], a bounded space with strict rules (judges) where opposing officers (lawyers) facing uncertainty compete to persuade an audience (jury) of each's interpretation of reality; legal appeals further reduce uncertainty with an "informed assessment of competing interests" [31]. Generalizing, we see that a blue team's decision under uncertainty on the battlefield challenged by an AI-assisted red-team might prevent future tragedies [32]; and why machine learning and game theory require controlled contexts.

Finally, for now, interdisciplinary explorations include social science (e.g., bidirectional trust [33]) and philosophy (e.g., ethics). Citing UN Secretary General Antonio Guterres, the Editors of the *New York Times* [34]recommended that "humans never completely surrender life and decision choices in combat to machines." However, from [35],"Autonomous weapons already ... [may] start their own war ... [but] No theory for this encroaching world yet exists." Uncertain of the next step, our success has confirmed the limits of a team science built on observing independent individuals; open science is critical to advance the science of autonomy; and an interdisciplinary approach to the physics of teamwork may master autonomous human-machine teams and offer guidance to prevent future wars.

Next, we introduce the published articles for our Research Topic.

Ira Moskowitz uses Riemannian distance for a cost metric to improve multi-agent team efficiency. With an idealized model of the problem's geometry, he found solutions that may satisfy. Specifically, a combination of increasing skill and interdependence may optimize the probability of multi-agent teams reaching the correct conclusion to a problem confronted.

William Lawless proposes that a science of interdependent agents is necessary for autonomous human-machine teams. As evidence, a case study of the Uber pedestrian fatality in 2018 finds that the Uber car and its operator were both independent. But with an open approach, he discovers a tradeoff in a team's structural entropy and productivity.

Robert Hunjet's team consider bidirectional communication between humans and AI swarms to improve efficiency. To reduce ambiguity, they design a language used by Australian aborigines, the Jingulu, naming it JSwarm. It allows them to separate semantics from syntax. They provide an example in real-time with shepherding, planning human studies next.

Rino Falcone and Cristiano Castelfranchi investigate social interaction primitives in a dependence network of agents to model subjective valuations of trustworthiness when performing tasks. Their model allows a comparison of reality and subjective beliefs in preparation for autonomous collaboration with humans. They observe objective relationships emerge between agents, and they plan a future simulation.

Fred Petry and his team briefly review game theory for autonomy across several successful applications. They focus on Nash and Stackleberg equilibria and social dilemmas. They find that the use of "best responses" may create a negative result. In some situations, cooperation may violate moral rules, a result that has created lively discussions among practitioners about autonomy.

Krishna Pattipati's team simulate autonomous multi-agent systems with path planning algorithms for interdependent agents to produce intelligent courses of action under uncertainty (their derived generalized recursions subsume the well-known Sum-product, Max-product, Dynamic Programming, and joint Reward/Entropy maximization approaches as special cases). Using unified probabilistic inference and dynamic programming, communication rules, and factor graphs in reduced normal form produce optimal decisions subject to agent schedules, predicting that bounded rationality and human biases can be overcome.

Tony Gillespie wants to ensure trust of autonomous human-machine teams when decision-making transfers between humans and machines. He identifies three key Research Topic and important questions for human trust and acceptance of autonomous entity actions; describes teams as hierarchical control systems for responsibilities and actions with practical solutions; and presents three applications of his technique.

Ryan Quandt questions assumptions as human-machine teams approach autonomy: that interactions depend on how AI is housed, positioned, and navigates society. Behaviors in these settings reveal whether human and machine act and communicate jointly. Experiments should be performed and interpreted so that the successes of teams help society (and AI) to understand their actions.

Nicolas Hili's team notes that paper and pens are still used for modeling systems, partly because Computer-Aided Systems Engineering whiteboard tools remain problematic. New CASE tools improved applications, but without explainability. Instead, by separating handwritten text from geometrical symbols, they validate a human-machine interface for sketching that captures system models using interactive whiteboards with explainability.

Ashok Goel's team studies robots tasked with assembling objects by manipulating parts, a complex problem prone to failure. They use meta reasoning, robotic principles and dual encoding of state expectations, finding that low-level information or high-level expectations alone produces poor results. They outline a multi-level robotic system for assembling objects having six degrees of freedom.

Ibrahim et al. review safety for human-machine teams in uncertain or safety-critical contexts, highlighting trust for their safe and effective operation. They use Autonomous Ground Vehicles to explore examples of interdependent teaming, communication and trust between humans and autonomous systems, emphasizing that context influences trust for these systems.

Tom McDermott and Dennis Folds describe an information model that distributed human and machine teams can interpret for decisions under complexity (military hierarchical command and control structures; Rules of Engagement; Commander's Intent; and Transfer of Authority language). They use Construal Level Theory with progressive disclosures across real-time mission planning and control systems, demonstrated for simulated military mine countermeasures.

Mito Akiyoshi applies social science to interacting humans to guide the emergence of trust for Autonomous Human Machine Teams and Systems in real world contexts. She integrates these theoretical perspectives: the ecological theory of actors and tasks; theory of introducing social problems for civics; and political economy developed in the sociological study of markets.

Matthew Johnson's team generalizes the effects of interdependence for adaptability and team effectiveness, finding it critical to human-machine team success. To help engineers move beyond models of individuals, they operationalize interdependence with formal structure and activity graphs to address complexity. They provide an example of an adversarial domain that exploits interdependence for effective, adaptive management. social and experiential aspects to be accounted for in the design of autonomous systems.

Ariel Greenberg and Julie Marble (https://www.frontiersin.org/articles/10.3389/fphy.2022.1080132/full) provide an overview of the conceptual foundations of teaming between people and intelligent machines. They examine the original meaning of relevant interpersonal terms as a basis from which to enrich their translated usage in the context of human-machine teaming, highlighting social and experiential aspects to be accounted for in the design of autonomous systems.

Author contributions

All authors listed have made a substantial, direct, and intellectual contribution to the work and approved it for publication.

Acknowledgments

For the many past years of summer research support, the first author thanks RM, Branch Head, Information and Decision Sciences, Information Technology Department, Naval Research Laboratory, Washington, DC; for his many years of editing and other support, the first author thanks his colleague DS, Computer Scientist and Roboticist, who leads the Distributed Autonomous Systems Group at the Navy Center for Applied Research in Artificial Intelligence (NCARAI), Naval Research Laboratory, Washington, DC; and for her unwavering support over many years of conceptual discussions and constructive debates, the first author thanks Kim Butler, North Augusta, SC.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

1. Lawless WF. Toward a physics of interdependence for autonomous human-machine systems: The case of the Uber fatal accident. *Front Phys* (2022). doi:10.3389/fphy.2022.879171

2. Bisbey TM, Reyes DL, Traylor AM, Salas E. Teams of psychologists helping teams: The evolution of the science of team training. *Am Psychol* (2019) 74(3):278–89. doi:10.1037/amp0000419

3. Schölkopf B, Locatello F, Bauer S, Ke NR, Kalchbrenner N, Goyal A, et al. (2021), Towards causal representation learning, arXiv

4. Shannon CE. A mathematical theory of communication. *Bell Syst Tech J* (1948) 27(379–423):623–56. doi:10.1002/j.1538-7305.1948.tb00917.x

5. Cooke NJ, Hilton ML, Enhancing the effectiveness of team science. *Authors: Committee on the science of team science; board on behavioral, cognitive, and sensory sciences; division of behavioral and social sciences and education*. Washington (DC): National Research CouncilNational Academies Press (2015).

6. Endsley MR. *Human-AI teaming: State-of-the-Art and research needs*. Washington, DC: The National Academies of Sciences-Engineering-MedicineNational Academies Press (2021).

7. Davies P. Does new physics lurk inside living matter? *Phys Today* (2021) 73(8):34–40. doi:10.1063/PT.3.4546

8. Blanton H, Klick J, Mitchell G, Jaccard J, Mellers B, Tetlock PE. Strong claims and weak evidence: Reassessing the predictive validity of the IAT. *J Appl Psychol* (2009) 94(3):567–82. doi:10.1037/a0014665

9. Leach CW (2021), Editorial, journal of personality and social psychology: Interpersonal relations and group processes. Available from: https://www.apa.org/pubs/journals/features/psp-pspi0000226.pdf, [Retrieved 11 15 2021].

10. Nosek B. Estimating the reproducibility of psychological science. *Science* (2015) 349(6251):943. doi:10.1126/science.aac4716

11. Paluck EL, Porat R, Clark CS, Green DP. Prejudice reduction: Progress and challenges. *Annu Rev Psychol* (2021) 72:533–60. doi:10.1146/annurev-psych-071620-030619

12. Klein E. *"A skeptical take on the A.I. revolution. The A.I. expert Gary Marcus asks,"What if ChatGPT isn't as intelligent as it seems?* (2023). New York Times, retrieved 1/7/2023. Available at: https://www.nytimes.com/2023/01/06/opinion/ezra-klein-podcast-gary-marcus.html.

13. Zumbrun J. ChatGPT Needs Some Help with Math Assignments. 'Large language models' supply grammatically correct answers but struggle with calculations. *Wall Street J* (2023).

14. Perolat B, De Vylder B, Hennes D, Tarassov E, Strub F, de Boer V, et al. Mastering the game of Stratego with model-free multiagent reinforcement learning. *Science* (2022) 378(6623):990–6. also, see the Research Highlight by Yury Suleymanov, same issue. doi:10.1126/science.add4679

15. Suleymanov Y, De Vylder B, Hennes D, Tarassov E, Strub F, de Boer V, et al. Mastering the game of Stratego with model-free multiagent reinforcement learning. *Science* (2022) 378(6623):990–6. doi:10.1126/science.add4679

16. Bohr N. Science and the unity of knowledge. In: L Leary, editor. *The unity of knowledge*. New York: Doubleday (1955). p. 44–62.

17. Pais A. Niels bohr's times. In: *Physics, philosophy, and polity*. Oxford, UK: Clarendon Press (1991).

18. James W. *The principles of psychology*. New York, United States: Dover Publications (1950).

19. Schrödinger E. Discussion of probability relations between separated systems. *Proc Cambridge Phil Soc* (1935) 3132(555–563):446–51.

20. Lewin K. *Field theory of social science. Selected theoretical papers. Darwin Cartwright*. New York: Harper and Brothers (1951).

21. Walden DD, Roedler GJ, Forsberg KJ, Hamelin RD, Shortell TM. Systems Engineering Handbook. A guide for system life cycle processes and activities. In: *Prepared by international council on system engineering (INCOSE-TP-2003-002-04)*. 4th ed. Hoboken, NJ: John Wiley and Sons (2015).

22. Cummings J. *Team science successes and challenges*. Bethesda MD: National Science Foundation Sponsored Workshop on Fundamentals of Team Science and the Science of Team Science (2015).

23. Smith A. *An inquiry into the nature and causes of the wealth of nations*. Chicago: University of Chicago Press (1977).

24. Lawless WF, Risk determination versus risk perception: A new model of reality for human–machine autonomy. *Informatics* (2022) 9:30. doi:10.3390/informatics9020030

25. Lawless WF. Interdependent autonomous human–machine systems: The complementarity of fitness, vulnerability and evolution. *Entropy* (2022) 24(9):1308. doi:10.3390/e24091308

26. Lawless WF. Autonomous human-machine teams: Reality constrains logic, but hides the complexity of data dependency. *Invited, Spec Issue Data Sci Finance Econ* (2022) 2(4):464–99. doi:10.3934/DSFE.2022023

27. Mann RP. Collective decision making by rational individuals. *PNAS* (2018) 115(44):E10387–E10396. doi:10.1073/pnas.1811964115

28. Sen A. The formulation of rational choice. *Am Econ Rev* (1994) 84(2):385–90.

29. Simon HA. *Bounded rationality and organizational learning*. Technical Report AIP 107. Pittsburgh, PA: CMU (1989).

30. White J. *California v. Green, 399 U.S. 149* (1970). U.S. Supreme Court Publisher. Available at: https://www.supremecourt.gov/.

31. Ginsburg RB (2011), American electric power Co., inc, Available at: http://www.supremecourt.gov/opinions/10pdf/10-174.pdf (Accessed 11 May 2017).

32. DoD (2021), Kirby and air force lt. Gen. Sami D. Said hold a press briefing. Pentagon Press Secretary John F. Available from https://www.defense.gov/News/Transcripts/Transcript/Article/2832634/pentagon-press-secretary-john-f-kirby-and-air-force-lt-gen-sami-d-said-hold-a-p/. [Retrieved 11 3 2021].

33. Lawless WF, Sofge DA. The intersection of robust intelligence and trust: Hybrid teams, firms and systems. In: WFR LawlessMittu, D Sofge, S Russell, editors. *Autonomy and artificial intelligence: A threat or savior?* New York: Springer (2017). p. 255–70.

34. Editors (2019), "Ready for weapons with free will? New York times, Available from: https://www.nytimes.com/2019/06/26/opinion/weapons-artificial-intelligence.html.

35. Kissinger H (2022), "Henry Kissinger's guide to avoiding another world war. Ukraine has become a major state in Central Europe for the first time in modern history," Available from: https://thespectator.com/topic/henry-kissinger-guide-avoiding-another-world-war/. [Retrieved 12 27 2022].

ORIGINAL RESEARCH
published: 25 April 2022
doi: 10.3389/fphy.2022.861633

A Cost Metric for Team Efficiency

Ira S. Moskowitz *

Naval Research Laboratory, Information Management and Decision Architectures Branch, Washington, DC, United States

We use a Riemannian metric as a cost metric when it comes to the optimal decisions that should be made in a multi-agent/Team scenario. The two parameters of interest to us are Team skill and Team interdependence, which are modeled as Wiener process drift and the inverse of Wiener process diffusion, respectively. The underlying mathematics is presented, along with some approximating rules of thumb. It is noteworthy that the mathematics points to, what seems at first, counter-intuitive paradigms for Team performance. However, in reality the mathematics shows a subtle interplay between the factors affecting Team performance.

Keywords: agents, team, Wiener process, Brownian motion, multi-agent system

OPEN ACCESS

Edited by:
William Frere Lawless,
Paine College, United States

Reviewed by:
Sergio Da Silva,
Federal University of Santa Catarina, Brazil
Stephen Russell,
Jackson Memorial Hospital, United States

****Correspondence:***
Ira S. Moskowitz
ira.moskowitz@nrl.navy.mil

Specialty section:
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

Received: *24 January 2022*
Accepted: *28 February 2022*
Published: *25 April 2022*

Citation:
Moskowitz IS (2022) A Cost Metric for Team Efficiency.
Front. Phys. 10:861633.
doi: 10.3389/fphy.2022.861633

1 INTRODUCTION

We are concerned here with how a multi-agent System (MAS) [2], or Team, reaches the successful conclusion of a task. In Team science, an important parameter for success is interdependence [1, 9, 10, 12]. The Team may be human, machine, or a hybrid. However, our mathematical assumptions implicitly assume that the Team is very machine-like in its behavior and discounts the vagaries of human psychology, e.g., [5]. We address this further at the conclusion of this article.

In [13] it was shown how to model Team behavior as (1-dimensional) Brownian motion [4] (starting at a point Z on the line). In particular, we proposed using Brownian motion $\mathscr{B}(t)$ with (high) drift μ, for (high) Team skill and (low) diffusion σ, for (high) interdependence, arbitrarily starting at a point Z, $0 \le Z \le A$. We consider that the Team has succeeded if it reaches point A before 0, and the Team has failed if it reaches 0 before A.

The drift, as mentioned, models Team skill. By way of motivation (using humans), imagine we have a restaurant kitchen crew (building on the restaurant Team given in [11]). We would like all the Team members to have the most skill possible; this would go into the calculation of the drift μ. High skill mapping to high μ. We would also like the Team to work together, hence we desire a high interdependence. Interdependence is the inverse of the diffusion, thus a Team with high interdependence has low diffusion σ, and a Team with each Team member acting independently of the other has high diffusion σ. This leads to the question of which is better—high drift μ, or low diffusion σ? Again, let us go back to our kitchen crew example. If everyone in the kitchen is skilled, but working independently of the others, the result will be a disaster. The dessert will be served before the main course, wine will be served after dessert, etc. Thus, skill alone does not lead to optimal success. On the other hand, consider a kitchen crew with no skill, but working together hand in glove. The results here are also less than optimal—very bad food served in an efficient manner. What is needed is a combination of both factors for optimal Team success, and that is what our idealized mathematics show.

Definition 1. W*e say that a stochastic process* $\mathscr{W}_t, t \ge 0$ *is a Wiener proces*s [8] if

- $\mathscr{W}_0 = 0$.
- *With probability 1, the function* $t \to \mathscr{W}_t$ *is continuous in t.*
- *The stochastic process* $\{\mathscr{W}_t\}, t \ge 0$*, has stationary, independent increments.*

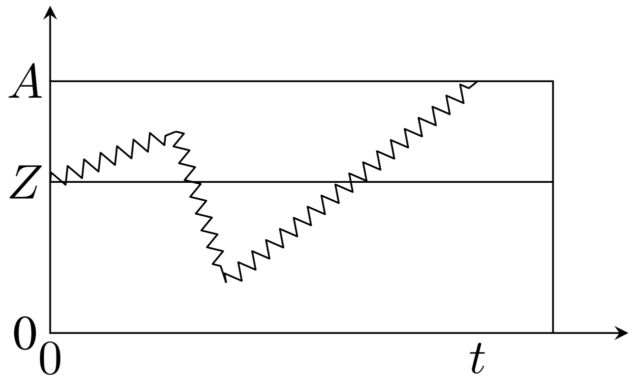


FIGURE 1 | Brownian motion, $\mathscr{B}(t)$, starting at Z and with absorbing boundaries at A (top) and 0 (bottom).

- *The increment, $\mathscr{W}_{t+s}-\mathscr{W}_s$, has the distribution of the standard normal random variable, N (0, t) (this latter part of the definition tells us that $\mathscr{W}_t$ has the distribution of N (0, t)).*

Definition 2. From [13, 15, 17, 18], we say that $\mathscr{B}(t)$ is Brownian motion with drift μ and diffusion σ, that starts at Z, 0 ≤ Z ≤ A, if

$$\mathscr{B}(t) = \mu t + \sigma \mathscr{W}(t) + Z. \tag{1}$$

Let $P_Z(\partial = 0)$ be the probability that $\mathscr{B}(t)$ hits the bottom boundary first (Team failure), then $P_Z(\partial = A) = 1 - P_Z(\partial = 0)$ is the probability that it hits the top boundary first (Team success). **Figure 1** is an example of such a sample path. These probabilities are derived from stopping probabilities ([3]), and we use L'Hôpital's rule for $\mu = 0$.

$$P_Z\Big(\partial = 0\Big) = \begin{cases} \dfrac{e^{-\frac{2A\mu}{\sigma^2}} - e^{-\frac{2Z\mu}{\sigma^2}}}{e^{-\frac{2A\mu}{\sigma^2}} - 1}, & \text{if } \mu \neq 0 \\ 1 - \dfrac{Z}{A}, & \mu = 0 \end{cases} \tag{2}$$

and

$$P_Z\Big(\partial = A\Big) = \begin{cases} \dfrac{e^{-\frac{2Z\mu}{\sigma^2}} - 1}{e^{-\frac{2A\mu}{\sigma^2}} - 1}, & \text{if } \mu \neq 0 \\ \dfrac{Z}{A}, & \mu = 0. \end{cases} \tag{3}$$

For now, we will concentrate on **Equation 3**. Keep in mind that as $\sigma \to 0$, then $P_Z(\partial = A) \to 1$, if $\mu > 0$, and that $P_Z(\partial = A) \to 0$, if $\mu < 0$ (of course, if $Z > A$, this would not hold).

For $Z = 0$, we have that $P_Z(\partial = A) = 0$ since 0 is an absorbing boundary—that is, if we start at 0 we are done. Also, for $Z = A$, we have that $P_Z(\partial = A) = 1$, which also makes sense—if we start at A, we never leave A.

Now say that we need to make an assessment—is it better to modify the drift μ or modify the diffusion σ to increase $P_Z(\partial = A)$? Also, what is the cost of this modification, and how do we measure the cost?

2 FIRST STEPS

We start by defining our manifold **B** and its Riemannian structure.[1]

2.1 Our Manifold B

We would like to know the costs of changing $P_Z(\partial = A)$ as we vary μ and σ. We consider $\mathscr{B}(t)$ in terms of its two parameters, μ and σ.

With this in mind, we define a 2-dimensional Riemannian manifold **B**, homeomorphic to $\mathbb{R} \times \mathbb{R}^+$, and with a global μ, σ chart. We give **B** the Riemannian metric

$$ds^2 = d\mu \otimes d\mu + \frac{1}{\sigma^2} d\sigma \otimes d\sigma. \tag{4}$$

This metric captures the fact that for σ fixed, the difference in μ is simply the standard L_1 distance between them, and that it is independent of the diffusion value. However, the diffusion is also independent of the drift value, but as we attempt to make the diffusion (standard deviation) smaller, it costs more and more, until we approach ∞ at the Dirac distribution.

Note 1—We have chosen to give an infinitesimal distance between points (μ, σ) and $(\mu + d\mu, \sigma + d\sigma)$ and then extend it to a global distance. The Riemannian metric ds^2 captures the fact that changes in μ are Euclidean straight-line distance, whereas changes in σ are based on the inverse of the variance. This concept aligns with how normal distributions differ. We further note that this result is also similar to the Fisher information of the normal distribution (a normalized Poincaré upper-half-plane). What is important about our Riemannian metric is that only the $d\sigma^2$ is modified from the standard Euclidean metric. Again, this emphasizes the fact that changing the mean of the normal distribution is strictly Euclidean, whereas if we attempt to lower the variance, it requires much more "power," and in the limit approaches infinite power. This approach agrees with our thinking that total interdependence (exactly the opposite of independent behavior) has a diffusion of 0, where as totally uncorrelated behavior has infinite diffusion [13, 6.4.2].—

Thus, **B** has the first fundamental form

[1] In this article, we had to make a choice between readability for the non-expert in differential geometry and exact precision with respect to Riemannian geometry. We hope that we have achieved a happy middle ground, and we assure the interested reader that any of the missing fine points can be found in the literature (e.g. [20]).

$$\left[g_{ij}\right] = \begin{pmatrix} E & F \\ F & G \end{pmatrix}$$
$$= \begin{pmatrix} 1 & 0 \\ 0 & \frac{1}{\sigma^2} \end{pmatrix}, \text{ where } i, j \text{ are indexed independently over } \mu, \nu. \tag{5}$$

Assume we are at $p \in \mathbf{B}$, where $p = (p_\mu, p_\sigma)$, and with tangent vectors $\hat{U} = u_1 \frac{\widehat{\partial}}{\partial \mu} + u_2 \frac{\widehat{\partial}}{\partial \sigma}$, $\hat{W} = w_1 \frac{\widehat{\partial}}{\partial \mu} + w_2 \frac{\widehat{\partial}}{\partial \sigma}$ at p, where $\frac{\widehat{\partial}}{\partial \mu}, \frac{\widehat{\partial}}{\partial \sigma}$ are the canonical basis for the tangent space at p.

The inner product between them is

$$\langle \hat{U}, \hat{W} \rangle := u_1 w_1 + \frac{1}{(p_\sigma)^2} u_2 w_2.$$

The norm of a vector $\hat{W}$ is

$$\|\hat{W}\| := \sqrt{\langle \hat{W}, \hat{W} \rangle}.$$

Say $c(t)$ is a smooth curve in $\mathbf{B}$, c: $(a, b) \to \mathbf{B}$, then there is the velocity vector field (on the curve) denoted as $\dot{c}(t)$. This velocity vector field assigns to each point $c(t')$ on the curve $c(t)$ the velocity vector (which is also a tangent vector of M) of the curve at $c(t')$ expressed as $\dot{c}(t')$ (keep in mind that this is multi-dimensional). That is, $c(t) = (c_\mu(t), c_\sigma(t))$, and $\dot{c}(t) = \dot{c}_\mu(t) \frac{\widehat{\partial}}{\partial \mu} + \dot{c}_\sigma(t) \frac{\widehat{\partial}}{\partial \sigma}$ (where the raised dot symbol is the usual differentiation with respect to t, and $\frac{\widehat{\partial}}{\partial \mu}, \frac{\widehat{\partial}}{\partial \sigma}$ are understood to be the canonical tangent space basis vectors at the point $c(t) \in M$). To simplify notation, we can express this as $\dot{c}(t) = \langle \dot{c}_\mu(t), \dot{c}_\sigma(t) \rangle$.

We define the **length** of $c(t)$, denoted as $L(c)$, as

$$L(c) := \int_a^b \|\dot{c}(\tau)\| d\tau = \int_a^b \sqrt{\left[\dot{c}_\mu(\tau)\right]^2 + \frac{\left[\dot{c}_\sigma(\tau)\right]^2}{\left[c_\sigma(\tau)\right]^2}}\, d\tau. \tag{6}$$

Given two points, $p, q \in \mathbf{B}$, and $c(t)$, any smooth curve between them (this can be relaxed to include piece-wise smooth, but not of class C^∞), we define the **distance** between them as

$$d(p, q) := \inf L(c). \tag{7}$$

2.2 Team Geometry

Our metric is modeled on the hyperbolic metric in the Poincaré half-plane model. An important difference is that E does not depend on the σ value. The change in drift is independent of the diffusion value which we feel is the correct way to model Team action. Furthermore, the μ distance is linear with respect to μ. This choice assumes that only the change of Team skill matters, not the values it ranges between. However, the change in diffusion, which is independent of drift, does depend on the different diffusion (interdependence values) that the Team is choosing between. This approach makes sense in terms of a normal distribution. Going from a normal distribution $N(\mu, 10)$ to $N(\mu, 9)$ requires much less change in the distribution itself than going from $N(\mu, 1)$ to $N(\mu, 0.9)$, and again going from $N(\mu, 0.5)$ to $N(\mu, 0.45)$.

We see in **Figure 2** that as $\sigma \to 0^+$, the difference in the normal plots is more severe. This behavior is in contrast to changing μ, which has the effect of shifting the graph to the left or right, but not changing its shape. We use a Riemannian manifold because it gives us the means to modify the metric for other models of Team behavior. This approach is accomplished by adjusting the g_{ij} in **Equation 5**.

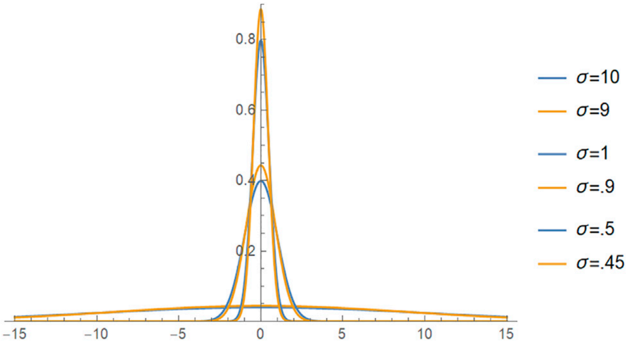


FIGURE 2 | $N(0, \sigma^2)$ for three groups. g The bottom pair is $\sigma = 10, 9$; the middle pair is $\sigma = 1, 0.9$; and the top two, which are the most different are $\sigma = 0.5, 0.45$.

2.3 Curvature and Geodesics

We start by considering the Gaussian (sectional) curvature K of $\mathbf{B}$ as a function of the first fundamental form.

First, using **Equation 5**, we consider the easily obtainable matrices (the sub-index indicates the partial differentiation with respect to that index) for $\mathbf{B}$.

$$\begin{pmatrix} E_\mu & F_\mu \\ F_\mu & G_\mu \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \text{and} \begin{pmatrix} E_\sigma & F_\sigma \\ F_\sigma & G_\sigma \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & \frac{-2}{\sigma^3} \end{pmatrix}. \tag{8}$$

From [14, Eq. (9.22), Eq. (9.33)] we use the Brioschi formula for a Riemannian 2-manifold in general with generic parameters μ, ν, which, for arbitrary E and G, and when $F = 0$ [14, Eq. 9.25], becomes

$$K = \frac{-1}{\sqrt{EG}} \left\{ \frac{\partial}{\partial \mu} \left(\frac{1}{\sqrt{E}} \frac{\partial \sqrt{G}}{\partial \mu} \right) + \frac{\partial}{\partial \sigma} \left(\frac{1}{\sqrt{G}} \frac{\partial \sqrt{E}}{\partial \mu} \right) \right\} \tag{9}$$

$$= \frac{-1}{2\sqrt{EG}} \left\{ \frac{\partial}{\partial \mu} \left(\frac{G_\mu}{\sqrt{EG}} \right) + \frac{\partial}{\partial \sigma} \left(\frac{E_\sigma}{\sqrt{EG}} \right) \right\}. \tag{10}$$

For $\mathbf{B}$, $G_\mu = 0$ and $E_\sigma = 0$, we find that $K = 0$.

Now we move on to the geodesics of $\mathbf{B}$. First we have to find the Christoffel (tensor) symbols (symmetric in the lower indicies). We define these on a local patch of a Riemannian 2-manifold, M, in general with generic parameters μ, σ.

$$\Gamma^\mu_{\mu\mu} = \frac{GE_u + FE_\sigma - 2FF_u}{2(EG - F^2)} \tag{11}$$

$$\Gamma^\mu_{\mu\sigma} = \frac{GE_\sigma - FG_u}{2(EG - F^2)} \tag{12}$$

$$\Gamma^\mu_{\sigma\sigma} = \frac{-FG_\sigma - GG_u + 2GF_\sigma}{2(EG - F^2)} \tag{13}$$

$$\Gamma^{\sigma}_{\mu\mu} = \frac{-FE_u - EE_\sigma + 2EF_u}{2\left(EG - F^2\right)} \tag{14}$$

$$\Gamma^{\sigma}_{\mu\sigma} = \frac{EG_u - FE_\sigma}{2\left(EG - F^2\right)} \tag{15}$$

$$\Gamma^{\sigma}_{\sigma\sigma} = \frac{EG_\sigma + FG_u - 2FF_\sigma}{2\left(EG - F^2\right)}. \tag{16}$$

Thus, for our manifold **B**, we have that all of the Christoffel symbols are 0, except for $\Gamma^{\sigma}_{\sigma\sigma} = \frac{-1}{\sigma}$.

Definition 3. *For* $t \in (0, 1)$, *a smooth curve* $c(t) = (c_1(t), c_2(t))$, $\dot{c}(t) = (\dot{c}_1(t), \dot{c}_2(t))$ *in a Riemannian manifold with* ∇ *being the Levi-Civita* connection [20] is a ***geodesic*** *if*

$$\nabla_{\dot{c}(t)}\dot{c}(t) = 0. \tag{17}$$

In general, one does not need to restrict t to the unit interval, but we have done this as a convenience. In general, geodesics are unique up to an affine parametrization; without loss of generality, we have fixed this by setting the t interval to [0, 1]. Note that by the existence and uniqueness theorem for ordinary differential equations (ODEs), we can find a unique geodesic if we also include the vector values at c (0) and $c'(0)$. (It turns out for **B** that this follows directly.)

We do not want to get into too many of the details of the ∇ operator above. It is covariant differentiation, which is the directional derivative of the vector field $\dot{c}(t)$ in the direction $\dot{c}(t)$ with adjustments for curvature K. Details can be readily found in the literature (e.g. [14]). Since the one local coordinate system (patch) we have given for **B** suffices, the geodesic equation reduces to

$$\ddot{c}_\mu(t) + \sum_{i,j\in\{\mu,\sigma\}} \Gamma^{\mu}_{ij}\dot{c}_i(t)\dot{c}_j(t) = 0, \text{ and} \tag{18}$$

$$\ddot{c}_\sigma(t) + \sum_{i,j\in\{\mu,\sigma\}} \Gamma^{\sigma}_{ij}\dot{c}_i(t)\dot{c}_j(t) = 0, \tag{19}$$

which, using the above values of the Christoffel symbols, simplifies to

$$\ddot{c}_\mu(t) = 0, \text{ and} \tag{20}$$

$$\ddot{c}_\sigma(t) - \frac{(\dot{c}_\sigma)^2}{c_\sigma(t)} = 0. \tag{21}$$

Trivially, we find that

$$c_\mu(t) = at + b.$$

To obtain $c_\sigma(t)$, we need to solve a non-linear second order ODE, so we simplify notation and use the auxiliary variable $w = \dot{c}_\sigma$, which gives $\ddot{c}_\sigma = \frac{dw}{dc_\sigma}\dot{c}_2 = \frac{dw}{dc_\sigma}w$. Now we perform the usual trickery, but check our answer at the end.

$$\begin{aligned} \ddot{c}_\sigma &= \frac{(\dot{c}_\sigma)^2}{c_\sigma} \\ \frac{dw}{dc_\sigma}w &= \frac{w^2}{c_\sigma} \\ \frac{dw}{w} &= \frac{dc_\sigma}{c_\sigma} \\ w &= \beta c_\sigma \\ \dot{c}_\sigma &= \beta c_\sigma \\ c_\sigma(t) &= \alpha e^{\beta t}. \end{aligned}$$

which when we check does solve **Eq. 21** for $c_\sigma(t)$ in its most general form. Thus,

$$c(t) = \left(at + b, \alpha e^{\beta t}\right). \tag{22}$$

Theorem 1. The constants a, b, α, β uniquely fix the geodesic.
Proof. Say there are two geodesics $c, \bar{c}$: $[0, 1] \to \mathbf{B}$ such that

$$c(t) = \left(at + b, \alpha e^{\beta t}\right) \text{, and}$$
$$\bar{c}(t) = \left(\bar{a}t + \bar{b}, \bar{\alpha}e^{\bar{\beta}t}\right).$$

Assume they are the same geodesic; then by evaluating the geodesic at $t = 0$, we have that

$$b = \bar{b}, \alpha = \bar{\alpha}.$$

Now using the above and evaluating the geodesics at $t = 1$, we have that

$$a = \bar{a}, \beta = \bar{\beta}.$$

□

So all we have to do now is to determine the four constants in the geodesic curve to uniquely specify it. As noted above, if we specify $c(0) = (\mu_0, \sigma_0)$ and $\dot{c}(0) = (\aleph, \beth)$, then simple calculations show that we uniquely fix the geodesic as

$$c(t) = \left(\aleph t + \mu_0, \nu_0 e^{\frac{\beth}{\sigma_0}t}\right). \tag{23}$$

However, we are interested in the boundary value problem to see if knowing $c(0)$, $c'(0)$ also gives us a unique solution. In general, for geodesics on an arbitrary Riemannian manifold, this result need not be true. By way of example, consider the geodesics (where the locus is a great circle) on S^2. Given $c(0)$, $c(1)$, there are infinitely many geodesics that satisfy the conditions (they just keep wrapping around). What is different in our situation, however, is that the geodesics never go back on themselves (this is seen by looking at the form of $c(t)$). If we have that $c(0) = (\mu_0, \sigma_0)$ and $c(1) = (\mu_1, \sigma_1)$, then simple calculations show that these boundary conditions uniquely fix the geodesic as

$$\begin{aligned} c(t) &= \left(\mu_0 + (\mu_1 - \mu_0)t,\ \sigma_0 e^{t\ln\left(\frac{\sigma_1}{\sigma_0}\right)}\right) \\ &= \left(\mu_0 + (\mu_1 - \mu_0)t,\ \sigma_0\left(\frac{\sigma_1}{\sigma_0}\right)^t\right). \end{aligned} \tag{24}$$

Thus, for the geodesics c: $[0, 1] \to \mathbf{B}$, we find that a solution exists and, given $c(0)$ and $c(1)$, that the geodesic is uniquely expressed as in **Equation 24**.

Equations 6 and **7** tell us how to obtain a topology based on the metric distance. This topology makes **B** homeomorphic to the upper half-plane with its standard topology. (Note though that **B** is not isometric to the upper half-plane with the standard Euclidean metric.) Since the latter space is complete, so is **B**. By the Hopf-Rinow theorem [20], given an initial point $p = (x_0, y_0)$, and a final point $q = (x_1, y_1)$, there exists a geodesic $c(t)$ between them such that $c(0) = p$, $c(1) = q$ and $L(c) = d(p, q)$. Given **Equation 24**, we have shown how to uniquely construct such a geodesic; therefore, the

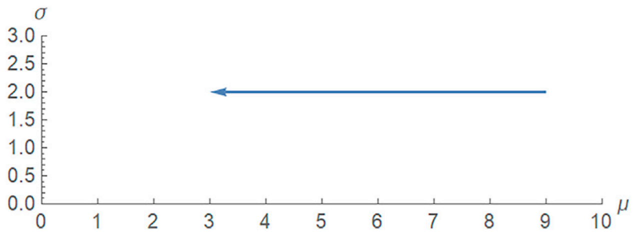


FIGURE 3 | Geodesic starting at p = (9, 2) and ending at q = (3, 2).

geodesic from **Equation 24** has the property that its length is the distance between the points.

The message from this result is that, given two points p, q on **B**, if we find the geodesic between them (remember we only use as a domain [0, 1]), then the length of that geodesic is the distance between them. This result is similar to what occurs in the Poincaré upper half-plane. We also note that this result would not work on S^2 because the boundary values do not uniquely determine the geodesic—as discussed above, the geodesics on S^2 can wrap around themselves, which does not happen on **B** or the Poincaré upper half-plane. Therefore, this leaves us with the following corollary to the above theorem.

Corollary 1.1. Given two points in $p = (\mu_0, \nu_0) \in \mathbf{B}$, and $q = (\mu_1, \sigma_1) \in \mathbf{B}$, there is a unique geodesic c: $[0, 1] \rightarrow \mathbf{B}$ between them such that $c(0) = p$, $c(1) = q$. Furthermore, $c(t)$ is length minimizing, that is, $L(c) = d\,(p, q)$. The geodesic is as given in **Equation 24**.

Now let us examine the length of our geodesic $c(t)$ between $p = (\mu_0, \sigma_0)$ and $q = (\mu_1, \sigma_1)$. By **Equation 6**, we have that. $L\,(c) = \int_0^1 \sqrt{(\mu_1 - \mu_0)^2 + [\ln(\frac{\sigma_1}{\sigma_0})]^2}\, d\tau$, thus,

Corollary 1.2. Given two points in $p = (\mu_0, \sigma_0) \in \mathbf{B}$, $q = (\mu_1, \sigma_1) \in \mathbf{B}$, the length of the unique geodesic $c(t)$ between them is

$$\sqrt{(\mu_1 - \mu_0)^2 + \left[\ln\left(\frac{\sigma_1}{\sigma_0}\right)\right]^2}. \tag{25}$$

Let us see what some of these geodesics look like (their traces).

Example 1: Let us examine the case where σ is held constant between two points. Let $p = (\mu_1, \sigma)$, $q = (\mu_2, \sigma)$. The geodesic between them is

$$c\,(t) = (\mu_0 + (\mu_1 - \mu_0)t, \sigma).$$

We illustrate this result with $p = (9, 2)$, $q = (3, 2)$ in **Figure 3**. When ν is fixed, we are in a standard Euclidean metric, the length of the geodesic is its distance, and it follows from **Eqs 6**, **7** that $d\ (p,\ q) = |\mu_1 - \mu_0|$,

The result is a horizontal straight line of length six. When ν is fixed, our geometry is standard Euclidean geometry. Example 2: Let us now look at the opposite situation, when we hold μ fixed and vary σ.

The trace of this geodesic, shown in **Figure 4**, is simply a vertical line, however, its length is not its Euclidean length of 2—0.5 = 1.5, rather its length is $|\ln(.5/2)| = 1.39$. The geometry here is far from Euclidean. But now let us consider the geodesic starting at $p = (3, 0.2)$ and ending at $q = (3, 0.05)$. Again, its length is not the Euclidean length of 0.15; rather, its length is $|\ln(.05/.2)| = 1.39$, the same as the first part of this example. Let us summarize these two examples. For **B**, the length of a geodesic connecting two points with fixed σ is simply their Euclidean distance. However, the length of a geodesic in **B** connecting two points with the same μ only depends on their ratio, the distance being the absolute value of the natural log of the ratio.

Let us look at the general geodesics a bit more. In **Figure 5**, we see paths of the geodesics that start at (μ_0, σ_0) and end at (μ_1, σ_1). These representative samples, along with **Figures 3**, **4**, show the general shape of the geodesics.

An interesting question becomes: Given a point (μ_0, σ_0), what is the locus of points (μ, σ) distance D from this point? From **Equation 25**, we easily have that

$$\sigma = \sigma_0 e^{\pm\sqrt{D^2 - (\mu - \mu_0)^2}}. \tag{26}$$

From this result, we see that $\mu \in [\mu_0 - D, \mu_0 + D]$ and $\sigma \in [\sigma_0 e^{-D}, \sigma_0 e^{D}]$.

We plot in **Figure 6** the locus of points for σ as a 2-valued "function" of μ, with distance 2 from the point (1.5, 3) and when Z=.6 (recall that we have normalized A to 1).

Let us go back to **Eq. 3** and see how $P_{.6}\,(\eth\ = 1)$ varies as we look at all of the points at a set distance from (μ_0, σ_1).

We start by summarizing some of the results from [16, Sec 6.2.2] that discuss how $P_Z\,(\eth\ = A)$ behaves.

- $P_Z\,(\eth\ = A)$ is an increasing function of μ.
- For $\mu > 0$, $P_Z\,(\eth\ = A)$ is a decreasing function of σ.
- For $\mu < 0$, $P_Z\,(\eth\ = A)$ is an increasing function of σ.

Since the center of our 2-ball is (1.5, 3), let us examine the two points with extreme μ values of -0.5 and 3.5. We find that

$$P_{.6}\Big(\eth\ = 1\Big)\Big|_{(-.5,3)} = .587 < P_{.6}\Big(\eth\ = 1\Big)\Big|_{(1.5,3)} = .639$$
$$< P_{.6}\Big(\eth\ = 1\Big)\Big|_{(3.5,3)} = .690,$$

and

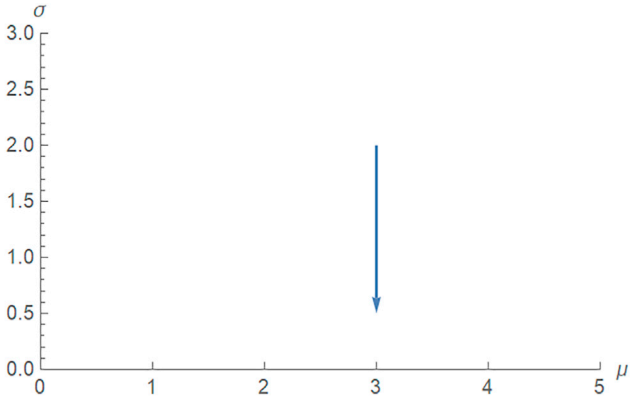


FIGURE 4 | Geodesic starting at p = (3, 2) and ending at q = (3, 0.5).

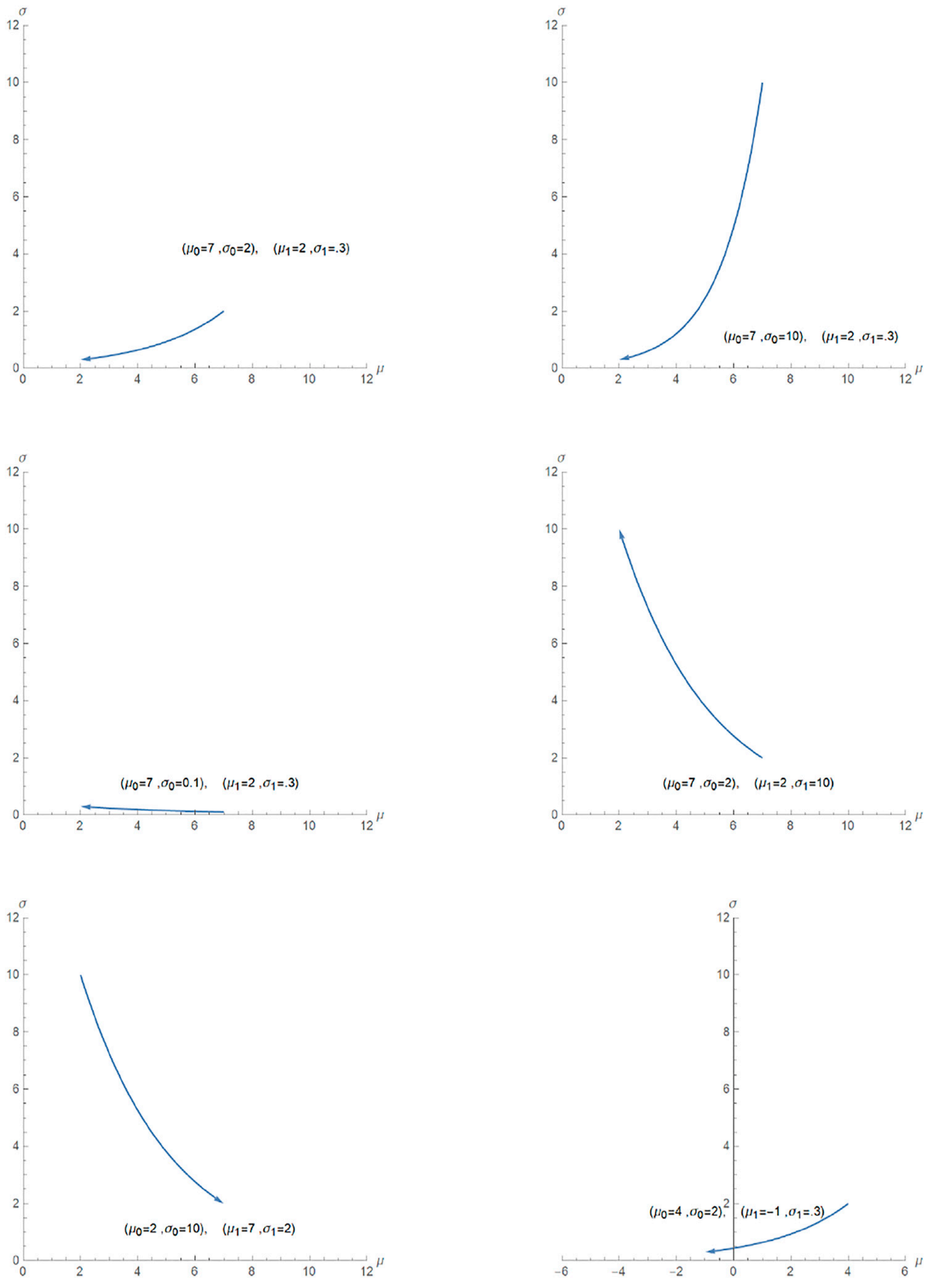


FIGURE 5 | Various geodesics.

$$P_{.6}\left(\eth = 1\right)|_{(3.5,.406)} = .999 > P_{.6}\left(\eth = 1\right)|_{(1.5,3)} = .639$$
$$> P_{.6}\left(\eth = 1\right)|_{(3.5,22.17)} = .601.$$

Keep in mind that for μ fixed at 3.5, as σ grows, the boundary probability approaches $0.6 = Z/A = 0.6/1$. This results in the values at the four "corners" of the metric-circle. One might think that the south pole is the highest probability. Let us plot $P_Z(\eth = A) = 1$ as a 2-valued function of μ. That is, we plot $P_Z(\eth = A) = 1$ as a function of μ with $\sigma = \sigma_0 e^{\sqrt{D^2-(\mu-\mu_0)^2}}$ (which corresponds to the top red semi-circle) and $\sigma = \sigma_0 e^{-\sqrt{D^2-(\mu-\mu_0)^2}}$ (which corresponds to the bottom blue semi-circle). The range of μ is $\mu \in [\mu_0 - D, \mu_0 + D]$. The point $(\mu, e^{\sqrt{D^2-(\mu-\mu_0)^2}})$ has a lesser probability than $(\mu, e^{-\sqrt{D^2-(\mu-\mu_0)^2}})$, since for μ fixed, the smaller σ becomes, the greater the probability when μ is positive.

Please note when comparing **Figures 6**, **7** that the blue and red regions have shifted.

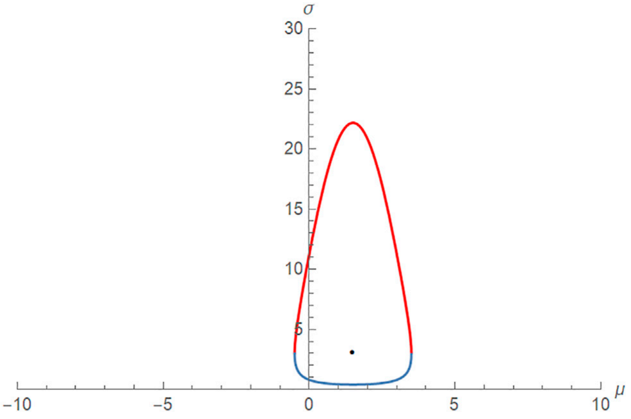


FIGURE 6 | Geodesic locus: Points at a distance 2 from (1.5, 3) with A = 1, Z =0.6. This is our metric "circle."

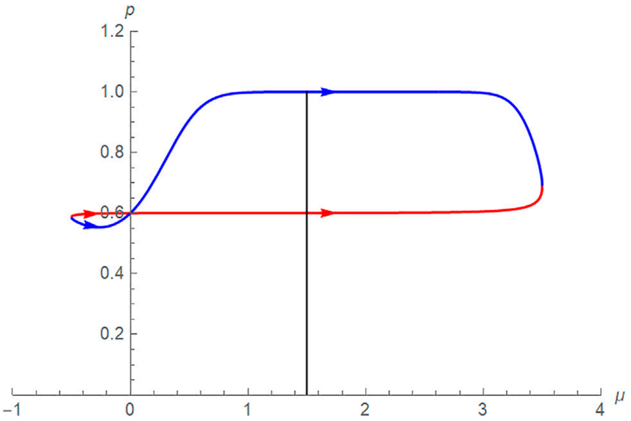


FIGURE 7 | $P_{.6}\,(\eth\ = 1)$ of point distance 2 from $(\mu, \sigma) = (1.5, 3)$ as μ increases (indicated by arrow direction) from −0.5 to 3.5. We see that for negative drift, the probabilities have an inverse behavior.

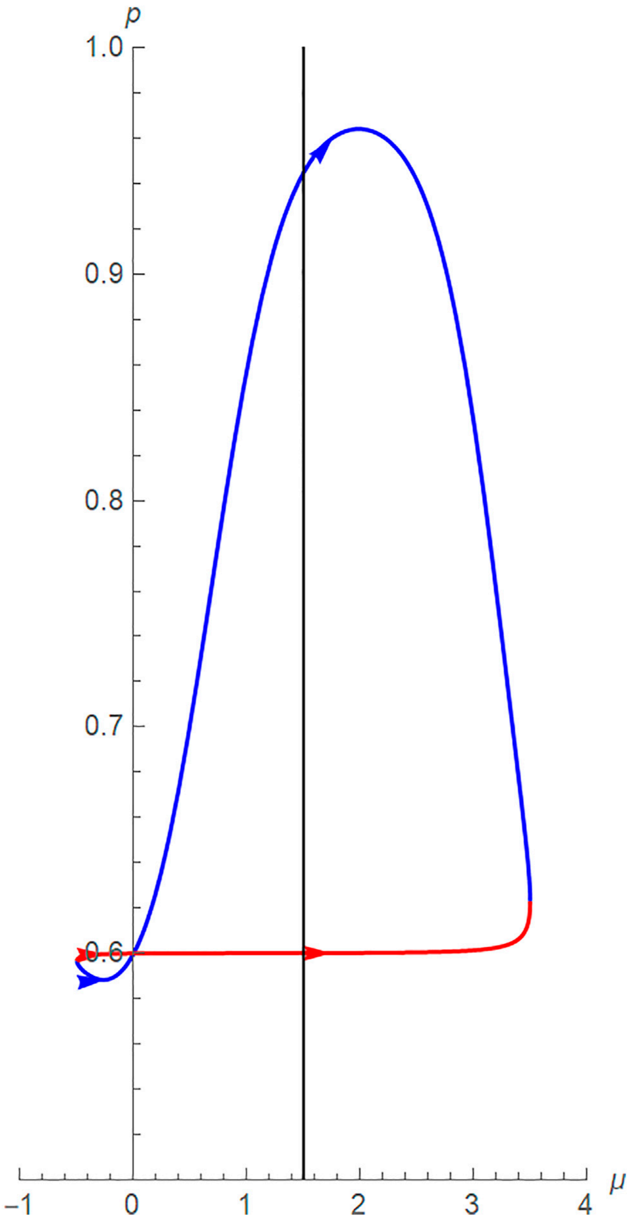


FIGURE 8 | $P_{.6}\,(\eth\ = 1)$ of points distance 2 from $(\mu, \sigma) = (1.5, 6)$ as μ increases (indicated by the arrow's direction) from −0.5 to 3.5.

From **Figure 7**, one might infer that the maximum probability occurs at the bottom corner $(\mu, \sigma) = (1.5, 0.406)$. Further numerical analysis shows that this is not true; the actual maximum occurs closer to when $\mu = 2$ (with the corresponding $\sigma > 0.406$). Let us use a different example to show this result better, as in **Figure 8**. Here, it is much more obvious that the maximum probability does not occur at the south pole of the metric circle.

We also see that the minimum probability does not occur for the smallest μ; rather, it too is a combination of a small μ, but with a larger σ.

In **Figure 9**, we have combined the plots from **Figures 6**, **7**. That is, **Figure 6**, which is a plot of a 2-valued function of σ against μ, is sketched in $(\mu, \sigma, 0)$ space. In **Figure 7**, since σ is now a 2-valued function of μ, we see that the probability $P_{.6}\,(\eth\ = 1)$ (of points distance 2 from the center $(1.5, 3)$) is a 2-valued function of μ and lives naturally in (μ, σ, p) space. In other words, the points on the top of a point of distance 2 is on the top plot. The red curves correspond to $\sigma = 3e^{\sqrt{2^2-(\mu-1.5)^2}}$, and the blue curves correspond to $\sigma = 3e^{-\sqrt{2^2-(\mu-1.5)^2}}$. We see that the red probability hovers around 0.6, whereas the blue approaches, very closely in fact, to a probability of 1.

3 SURFACE GEOMETRY

Let us move away from points a certain Riemannian distance from a point and consider the surface and the level sets of $P_Z\,(\eth\ = A)$. As before, we will normalize A to be 1, and let

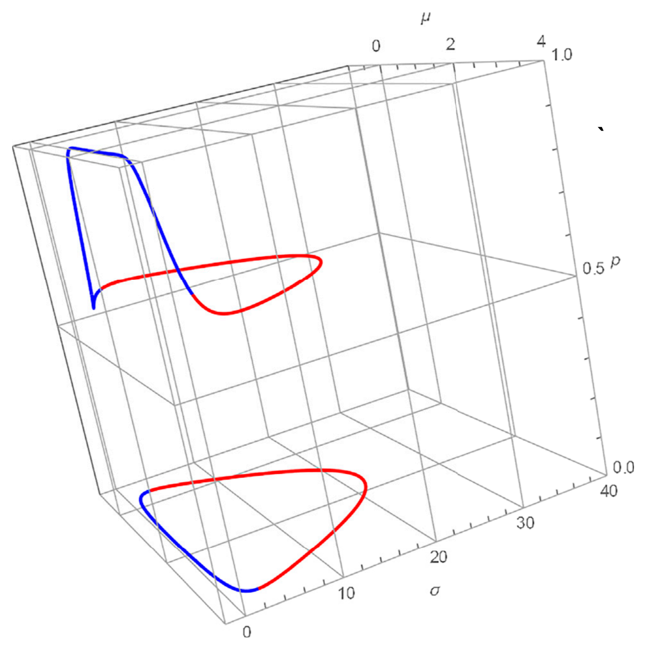


FIGURE 9 | Combo.

$Z = 0.6$. The behavior of $P_Z(\eth = A)$ as functions of μ and σ has been analyzed in [13], so we will not repeat the results from there. The plot of $P_{.6}(\eth = 1)$ is given in **Figure 10**.

Let us consider the level sets of $P_{.6}(\eth = 1)$; in fact, this holds for the level sets of $P_Z(\eth = A)$ for $Z \in (0, 1)$.

Theorem 2. *The level sets of* $P_Z(\eth = 1), 0 < Z < 1$ *correspond to constant values of* μ/σ^2.

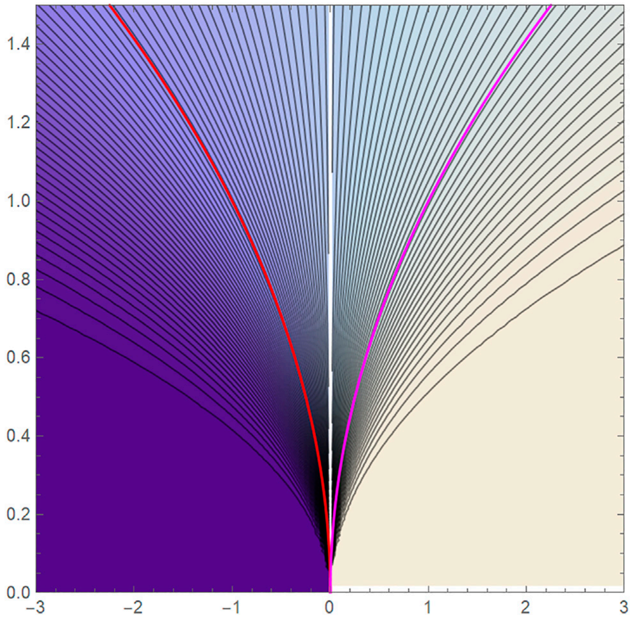


FIGURE 11 | Level sets of $P_{.6}(\eth = 1)$, horizontal axis is μ, vertical axis is σ.

Proof. (⇐) If $\mu/\sigma^2 = \mu'/\sigma'^2 = k$, then it is obvious that $P_Z(\eth = 1) = \frac{e^{-\frac{2Z\mu}{\sigma^2}}-1}{e^{-\frac{2\mu}{\sigma^2}}-1} = \frac{e^{-2Zk}-1}{e^{-2k}-1} = \frac{e^{-\frac{2Z\mu'}{\sigma'^2}}-1}{e^{-\frac{2\mu'}{\sigma'^2}}-1}$. (⇒) By (16, Cor 3.1), we have that for $C > D > 0$, that $\frac{e^{-Dk}-1}{e^{-Ck}-1}$ is an increasing function of x. Let $C = 2$, $D = 2Z$, we have that $f(k) = \frac{e^{-2Zk}-1}{e^{-2k}-1}$ is an increasing function of k and the result follows. □

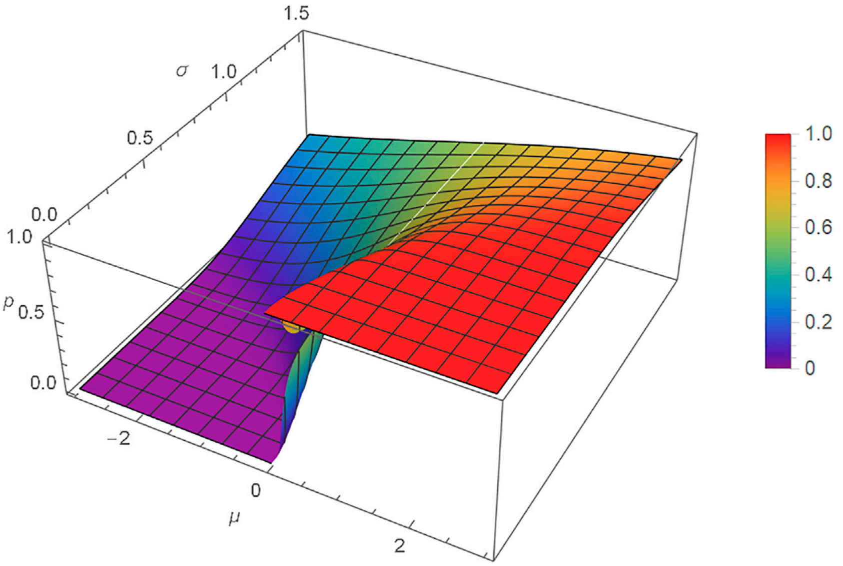


FIGURE 10 | Plot of $P_{.6}(\eth = 1)$ for $\mu \in (-3, 3)$, $\sigma \in (0, 1.5)$. The limited range is due to the fact that the plot is extremely close to 1 for large μ and small σ. The function is continuous and is equal to 0.6 when $\mu = 0$.

Let us look at the level sets of this surface in **Figure 11**. We see 100 level sets going from left to right in ascending order from 0.01 to 0.99 in approximate steps of 0.01. The middle level set is white and that corresponds to $P_{.6}(\partial = 1) = .6$ which occurs when $\mu = 0$.

Keep in mind that every level set of $P_{.6}(\partial = 1)$ corresponds to the curve given by $\mu/\sigma^2 = k$. For $k < 0$, we have the level set on the left hand side (LHS) of **Figure 11**. For $k = 0$, we have the vertical white line at $\mu = 0$, and for $k > 0$ we have the level sets on the right hand side (RHS) of **Figure 11**. Note that we specifically illustrated the level set corresponding to $\mu/\sigma^2 = -1$ which is equivalent to $\sigma = -\sqrt{\mu}$ and corresponds to the red level set on the LHS of the figure; and corresponds to $P_{.6}(\partial = 1) = .363$. We also illustrated the level set corresponding to $k = 1$, which is the purple curve on the RHS of the figure and corresponds to $P_{.6}(\partial = 1) = .808$. Of course, now we see why the maximum values that we discussed above are not at μ corresponding to the center of the metric circle, but to the right. This result occurs because the level curve that is tangent to the plot is where the maximum is found. This result can be analyzed with Lagrange multiplier theory, a direction we will pursue in future work. It suffices for this article to show that the trade-off between μ and σ is non-trivial.

4 IMPACT ON TEAMS AND MULTI-AGENT SYSTEMS

We have learned from the mathematics that the decision to attempt to increase skill or to increase interdependence is not trivial. The best answer is a complicated mathematical expression. We also could have looked at the time to obtain the correct answer, but this is even more complicated and will also be addressed in future work.

Presently, our problem boils down to the probability of reaching the correct answer by using the Riemannian distance described in this article—which gets a Team to the highest new probability of success by staying within the distance constraints on skill and interdependence.

Of course, general rules of thumb can be derived by studying the geometry of the question in hand, and near-optimal solutions may be good enough to satisfy a user.

Recall from [16], for $\mu > 0$ (the situations we have been looking into), the lower the diffusion, the greater the interdependence. We note that our mathematics shows that *to optimize Team performance, it takes a combination of increasing the drift/skill* $\mu > 0$ *and lowering the diffusion* σ (*increasing interdependence*) *to optimize the probability of the Team of multi-agents of reaching the correct conclusion to a problem that it confronts.*

5 CONCLUDING REMARKS

Presently, to avoid complicated Riemannian geometric discussions, it is best to use rules of thumb that can be derived from the various plots of the Teams in question. This present work continues the theme emphasized by Lawless [9, 10] of the importance of interdependence, but also has thrown the skill issue into the mathematical mix. This point is not to say that others have ignored skill, rather that their focus was in the interesting and not completely understood topic of interdependence. It was our desire in this article to present a framework incorporating more of the mathematics for decision making.

Future work needs to be done on this topic. We have presented an idealized mathematical model. If the Teams are not simply multi-agents systems, but rather human, or human-machine hybrid teams, our model must be tempered by human factors. Humans do not act as automatons. These ideas are discussed in detail in the beautiful books by Kahneman [6, 7], and also [21]. A good overview of Kahneman's Nobel prize work in behavioral economics can be found in [5]. In fact, from [7] we can take the concept of noise and view that in terms of diffusion. As we rely more and more on hybrid teams, we must factor in a behavioral economics type approach to Team decision making. How this relates to the mathematics holds promise as a new research area. Furthermore, Teams are often subject to the wisdom of crowds [21], or the stupidity of crowds [19] (this work involves ants, which might be better representative agents than humans when attempting to model a machine), and the mathematical model we have presented does not incorporate such human factors. Of course, future work could include looking at and measuring these factors for an actual Team/MAS.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding author.

AUTHOR CONTRIBUTIONS

The author confirms being the sole contributor of this work and has approved it for publication.

ACKNOWLEDGMENTS

We thank Bill Lawless, Ruth Heilizer and the reviewers for their efforts. We are grateful to a reviewer and Ciara Sibley for pointing out, and discussing, the relevant research of Daniel Kahneman.

REFERENCES

1. National Research Council. Enhancing the Effectiveness of Team Science. In: Cooke NJ Hilton ML, editors. *Committee on the Science of Team Science; Board on Behavioral, Cognitive, and Sensory Sciences; Division of Behavioral and Social Sciences and Education; National Research Council*. Washington, DC: The National Academies Press (2015).
2. Springer. *Autonomous Agents and Multi-Agent Systems*. Berlin, Germany: Springer Journal (1998-2021).
3. Feller W. *An Introduction to Probability Theory and its Applications, Vols.1&2*. Hoboken, NJ, USA: Wiley (1950/1968).
4. Einstein A. Über die von der molekularkinetischen Theorie der Wärme geforderte Bewegung von in ruhenden Flüssigkeiten suspendierten Teilchen. *Ann Phys* (1905) 322:549–60. doi:10.1002/andp.19053220806
5. Kahneman D. Maps of Bounded Rationality: Psychology for Behavioral Economics. *Am Econ Rev* (2003) 3(5):14491475. doi:10.1257/000282803322655392
6. Kahneman D. *Thinking, Fast and Slow*. New York, NY, USA: Farrar, Straus and Giroux (2013).
7. Kahneman D, Olivier S, Cass S. *Noise: A Flaw in Human Judgment*. New York City: Little, Brown Spark (2021).
8. Lalley S, Mykland P. *Lecture Note Statistics 313: Stochastic Processes II*. New York City: Spring (2013). Available from: https://galton.uchicago.edu/lalley/Courses/313/. (Accessed October 13, 2021).
9. Lawless WF. The Entangled Nature of Interdependence. Bistability, Irreproducibility and Uncertainty. *J Math Psychol* (2017) 78:51–64. doi:10.1016/j.jmp.2016.11.001
10. Lawless WF. The Physics of Teams: Interdependence, Measurable Entropy and Computational Emotion. *Front Phys* (2017) 2017. doi:10.3389/fphy.2017.00030
11. Lawless WF. Quantum-Like Interdependence Theory Advances Autonomous Human-Machine Teams (A-HMTs). *Entropy* (2020) 22(11):1227. doi:10.3390/e22111227
12. Moskowitz IS, Lawless W, Hyden P, Mittu R, Russell S. *A Network Science Approach to Entropy and Training*. Palo Alto, California, U.S.: AAAI Spring Symposia Series, AAAI Press, 2015.
13. Moskowitz IS. Agent Team Action, Brownian Motion and Gambler's Ruin. *Eng Artificially Intell Syst* (2021) 2021. doi:10.1007/978-3-030-89385-9_6
14. Moskowitz IS, Russell S, Lawless W. An Information Geometric Look at the Valuing of Information. In *Chapter 9. Human-Machine Shared Contexts*. Editors Lawless W. Amsterdam, Netherlands: Elsevier (2020). doi:10.1016/b978-0-12-820543-3.00009-2
15. Moskowitz IS, Brown NL, Goldstein Z. A Fractional Brownian Motion Approach to Psychological and Team Diffusion Problems. In: Lawless W, et al, editors. *Ch. 11, Systems Engineering and Artificial Intelligence*. Berlin, Germany: Springer (2021). doi:10.1007/978-3-030-77283-3_11
16. Moskowitz IS. Agent Team Action, Brownian Motion and Gambler's Ruin. Chaper 11. In: Lawless W, et al, editors. *Lecture Notes in Computer Science (LNCS)*. Berlin, Germany: Springer (2021).
17. Ratcliff R. A Theory of Memory Retrieval. *Psychol Rev* (1978) 85(2):59–108.
18. Ratcliff R, Smith PL, Brown SD, McKoon G. Diffusion Decision Model: Current Issues and History. *Trends Cogn Sci* (2016) 20(4):260–81. doi:10.1016/j.tics.2016.01.007
19. Sasaki T, Granovskiy B, Mann RP, Sumpter DJ, Pratt SC, Pratta SC. Ant Colonies Outperform Individuals when a Sensory Discrimination Task Is Difficult but Not when it Is Easy. *Proc Natl Acad Sci U S A* (2013) 110:13769–73. doi:10.1073/pnas.1304917110
20. Spivak M. *A Comprehensive Introduction to Differential Geometry. Volumes 1-5*. 3rd ed. Los Angeles: Publish or Perish (1999).
21. Surowiecki J. *The Wisdom of Crowds*. New York City: Anchor (2005).

Conflict of Interest: The author declares that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

REVIEW
published: 23 June 2022
doi: 10.3389/fphy.2022.879171

Toward a Physics of Interdependence for Autonomous Human-Machine Systems: The Case of the Uber Fatal Accident, 2018

*William Lawless**

Department of Mathematics and Psychology, Paine College, Augusta, GA, United States

Computational autonomy has begun to receive significant attention, but neither the theory nor the physics is sufficiently able to design and operate an autonomous human-machine team or system (HMS). In this physics-in-progress, we review the shift from laboratory studies, which have been unable to advance the science of autonomy, to a theory of autonomy in open and uncertain environments based on autonomous human systems along with supporting evidence in the field. We attribute the need for this shift to the social sciences being primarily focused on a science of individual agents, whether for humans or machines, a focus that has been unable to generalize to new situations, new applications, and new theory. Specifically, the failure of traditional systems predicated on the individual to observe, replicate, or model what it means to even be the social is at the very heart of the impediment to be conquered and overcome as a prelude to the mathematical physics we explore. As part of this review, we present case studies but with a focus on how an autonomous human system investigated the first self-driving car fatality; how a human-machine team failed to prevent that fatality; and how an autonomous human-machine system might approach the same problem in the future. To advance the science, we reject the aggregation of independence among teammates as a viable scientific approach for teams, and instead explore what we know about a physics of interdependence for an HMS. We discuss our review, the theory of interdependence, and we close with generalizations and future plans.

Keywords: interdependence, autonomy, teams, systems, human-machine

Edited by:
Víctor M. Eguíluz,
Spanish National Research Council (CSIC), Spain

Reviewed by:
Sergio Da Silva,
Federal University of Santa Catarina, Brazil
Frédéric Vanderhaegen,
University of Valenciennes and Hainaut-Cambresis, France
Martin Baumann,
University of Ulm, Germany

***Correspondence:**
William Lawless
w.lawless@icloud.com

Specialty section:
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

Received: *18 February 2022*
Accepted: *23 May 2022*
Published: *23 June 2022*

Citation:
Lawless W (2022) Toward a Physics of Interdependence for Autonomous Human-Machine Systems: The Case of the Uber Fatal Accident, 2018. Front. Phys. 10:879171. doi: 10.3389/fphy.2022.879171

1 INTRODUCTION. DIFFERENT SCHOOLS OF THOUGHT AND CONTROVERSIES

Aim: The U.S. National Academy of Sciences [1,2] has determined that interdependence among teammates is the critical ingredient for a science of autonomous human-machine teams and systems (HMS), but that teams cannot be disaggregated to determine why or how they work together or to replicate them, stymying the development of a mathematics or physics of autonomous human-machine teams. Yet, surprisingly, the social sciences, which began with the Sophists over two millennia ago, still aggregate individuals to study “the nature and properties of the social world” [3]. Our aim is to overcome this barrier that has precluded the study of the “social world” to construct a physics of autonomy.

The social disruption posed by human-machine systems is more likely to be evolutionary, but it poses a dramatic change that Systems Engineers, social scientists and AI researchers must be prepared to manage; however, the theory to bring about this disruption we suspect will prove to be revolutionary. Designing synergistic interactions of humans and machines that holistically give rise to intelligent and autonomous systems requires significant shifts in thinking, modeling, and practice, beginning with changing the unit of analysis from independent humans or programmable machines to interdependent teams and systems that cannot be disaggregated. The case study of a fatality caused by a self-driving Uber car highlighted in our review barely scratches the surface to prepare readers for this disruption. The case study is a simple model of a machine and its human operator that formed a two-agent team involved in a fatal accident; but the human teams subsequently involved in the analysis of this fatality serve as a tool to measure how far we must go intellectually to accommodate an HMS.

We hope to provide a sufficient overview of the literature for interested readers to begin advanced studies to expand their introduction to the physics of autonomy as it currently exists. We conclude with a discussion of generalizations associated with our theory of autonomy, and, in particular, the entropy production associated with state-dependent [4] changes in an autonomous HMS [5]. In our review, we contrast closed model approaches to solving autonomy problems with open model approaches. A closed model is self-contained; the only uncertainty it is able to study is in the complexity created within its own model. Open models contain natural levels of uncertainty, competition or conflict, and sometimes all three.

At its most basic level, in contrast to closed systems, the case studies explore the fundamental tool of debate used for millennia by autonomous human teams confronting uncertainty. They led us to conclude that machines using artificial intelligence (AI) to operate as members of an HMS must be able to tell their human partners whenever the machines perceive a change in the context or emotion that affects their team's performance (context change may not be detectable by machine learning alone, which is context dependent; see [6]); in turn, AI machines must be able to understand the humans interacting with them in order to assess their contributions to a team's performance from their perspective as team members (i.e., how can an HMS improve a team's choices; how can an HMS improve the effectiveness of a team's performance; etc.; in [7]). The human and AI members of the team must be able to develop goals, learn, train, work and share experiences together. As part of a team, however, once these AI governed machines learn what humans want them to learn, they will know when the human members of their team are either complacent or malicious in the human's performance of the human's roles [8], a capability thought to be possible over the next few years [9]; in that case, or if a machine detects an elevated level of emotions, the machine must be able to express its reservations about a team's decisions or processes to prevent a mistake. There is even more to be mined in the future from the case studies we review. Specifically, if a human or machine is a poor team member, what exists in the AI Engineering[1] toolbox, the social science armature, physics, or elsewhere to aid a team in the selection of a new member of a system? Furthermore, how is the structure of an HMS with a poorly performing team member, human or machine, related to the performance of the team?

In what follows, the primary problem is to better manage the state of interdependence between humans and their machine teammates. The National Academy of Sciences review of human teams in 2015 [1] renewed interest in the study of interdependence, but the Academy was not clear about its implications, except that it existed in the best performing human teams [10], that led them to conclude that a team of interdependent teammates will likely be more productive than the same collection of individuals who perform as independent individuals [1,7]. The new National Academy of Sciences report on "Human-AI Teaming," commissioned by the U.S. Air Force, also discusses the value of interdependence, but without physics [2].

2 FUNDAMENTAL CONCEPTS, ISSUES AND PROBLEMS

In this section, we briefly review the definitions of terms used in this review (autonomy; rational; systems; closed and open systems; machine learning; social science).

2.1 Definitions: Autonomy. Autonomous Human-Machine Teams

Autonomy. Autonomous systems have intelligence tools to respond to situations that were not programmed or anticipated during design; e.g., decisions; self-directed behavior; human proxies ([2], p. 7). Autonomous human-machine systems work together to fulfill their design roles as teammates without outside human interaction in open systems, which include uncertainty, competition or conflict. Partially autonomous systems, however, require human oversight.

Autonomous human-machine teams occur in states of interdependence between humans and machines, both types able to make decisions together ([2], p. 7). Like autonomous human teams, they likely will be guided by rules (e.g., rules of engagement; rules for business; norms; laws; etc.). Theories of autonomy include human-machine symbiosis, arising only under interdependence, and addressed below.

2.2 Definitions: The Rational

Systems engineering. A concept is rational when it can be studied with reason or in a logical manner. A rational approach for traditional system engineering problems is considered to be the hallmark of engineering, such as a self-driving car. Paraphrased, from IBM's Lifecycle Management,[2] the rational approach consists of several steps: determining the requirements to solve a problem; design and modeling; managing the project; quality

[1] A new discipline is being proposed by Systems Engineers, and titled, AI Engineering [112].

[2] https://www.ibm.com/docs/en/elm/6.0?topic=overview-rational-solution-sse.

control and validation; and then to integrate across disciplines to assure the success of a solution:

> Requirements engineering: Solicit, engineer, document, and trace the requirements to determine the needs of the stakeholders involved. Build the work teams (e.g., considering the solutions available from engineering, software, technology, policy, etc.) that are able to adapt as change occurs in the needs and designs to be able to deliver the final product.
>
> Architecture design and modeling: Model visually to validate requirements, design architectures, and build the product.
>
> Project management: Integrate planning and execution, automate workflows, and manage change across engineering disciplines and development teams, including: iteration and release planning; change management; defect tracking; source control; automation builds; reporting; and customizing the process.
>
> Quality management and testing: Collaborate for quality control, automated testing, and defect management.
>
> Connect engineering disciplines: Visualize, analyze, and organize the system product engineering data with the tools that are available (viz., design and operational metrics and performance goals).

From the *Handbook of Systems Engineering* [11], engineering system products must be a transdisciplinary process; must include product life cycles (e.g., manufacturing; deployment; use; disposal); must be validated; must consider the environment it operates within; and must consider the interrelationships between the elements of the system and users.

2.3 Definitions: Artificial Intelligence and Machine Learning

At its simplest, AI is a rational approach to make systems more intelligent or "smart" by incorporating "rules" that perform specific tasks inside of a closed system (e.g., hailing a ride from Point A to Point B on a software platform from an Uber driver; [3] connecting the nearest available Uber driver and estimating costs and fees; gaining a mutual agreement between the customer and Uber driver). However, AI must also address human-machine teams operating in open systems ([2], p. 25). Machine learning (ML) is a subset of AI used to train an algorithm that learns from a "correct," tagged or curated data set to operate, say, a self-driving car being driven to learn while in a laboratory, on a safe track, or over a well-trodden, closed path in the real world (e.g., in the case of the Uber self-driving car, it was in its second loop along its closed path at the time of the fatality; in [12]; for more on Uber's self-driving technology, [4] see https://www.uber.com/us/en/atg/technology/and https://www.uber.com/us/en/atg/). In contrast to ML, deep learning (DL) is a subset of ML that may construct neural networks for classification with multiple bespoke layers and may entail embedded algorithms for training each layer; also, DL assumes a closed system.

2.4 Definitions: Social Science, Especially Its Application to Autonomous Teams

Social science studies "the nature and properties of the social world" [3]. It primarily observes individuals in social settings by aggregating data from individuals, and by statistical convergence processes on the data collected. We address its strengths and weaknesses regarding autonomous human-machine teams.

Social science: Strengths. Social science has several strengths that can be applied to autonomous human-machine teams. For example, the study of the cockpit behavior of commercial airline pilots separates the structure of teams from their performance [13], which we adopt. Functional autonomous human-machine teams cannot be disaggregated to see why they work ([2], p. 11), an important finding that supports the physics model we later propose. Cummings [10] found that the worst performing science teams were found to be interdisciplinary, suggesting poor structural fits, agreeing with Endsley [2], findings similar to Lewin's [14] claim that the whole is greater than the sum of its parts, which we adopt and explain. And it is similar to the emergence of synergy in systems engineering [11], which we also adopt, including symbiosis (mutual benefit).

Social science: Weaknesses. Traditional social science has little guidance to offer to the new science of autonomous systems [15]. The two primary impediments with applying traditional social science to an HMS are, first, its use of closed systems (e.g., a laboratory) to study solutions to the problems faced instead of the open field where the solutions must operate ([2], p. 56); and second, the reliance by social scientists on the implicit beliefs of individuals as the cause of observed behaviors, or the implicit behaviors of individuals derived from aggregated beliefs.

Implicit beliefs or behaviors are rational. Either works well for limited solutions to closed problems (e.g., game theory). The difficulty with implicit beliefs or behaviors is their inability to generalize. First, with the goal of behavioral control [16], by adopting implicit beliefs, physical network scientists, game theorists [17], Inverse Reinforcement Learning (IRL) scientists (e.g. [18]) and social scientists (e.g. [19]) have dramatically improved the accuracy and reliability of applications that predict human behavior but only in situations where alternative beliefs are suppressed, in low risk environments, or in highly certain environments; e.g., the implicit preferences based on the actual choices made in games [17] do not agree with the preferred choices explicitly stated beforehand ([20], p. 33). In these behavioral models, beliefs have no intrinsic value.

In contrast, second, often based on surveys or mental tasks, cognitive models discount the value of behavior [21], improving the correlations between cognitive concepts and cognitive beliefs about behavior, but not actual behavior; e.g., self-esteem beliefs correlate strongly with beliefs about academics or work (e.g. [22]),

[3] https://www.feedough.com/uber-business-model/

[4] On December 7th, Uber sold its self-driving unit to Aurora Innovation Inc. [113].

TABLE 1 | The failure of concepts to generalize to build new theory: The case of social science in closed systems. Column two contrasts the leading concepts in social science by its founding social scientist(s); Column three shows the scientist who toppled the leading concept. The table highlights the inability of social scientists to build new theory from prior findings.

Leading theory	Leading theory and theorist	Theory invalidated by:
Self-Esteem	Diener [22]; hailed by the American Psychological Association	[23]
Ego-Depletion	[25]	[26]
Implicit Attitudes Theory (racism)	[27]	[28]
Superforecasters	[30]	[32]
[31], PNAS	Shu et al. [31]; includes Ariely and Bazerman, the chief developer and promoter of the "Honesty" scale	Berenbaum [29], Editor in Chief, PNAS, retracted

but not with actual academics or work (e.g. [23]). Implicit behavior models, however, lead to predictions that either fail on their face or from a lack of replication. These failures are reviewed in **Table 1**. They have led to Nosek's [24] replication project in an attempt to avoid searing headlines of retractions; however, Nosek's replication project does not overcome the problem with generalizations.

Table 1 illustrates that "findings" in traditional social science with data from individuals in closed systems cannot be generalized to new findings. From the first row of **Table 1**, proposed by Diener [22], high self-esteem has been hailed by the American Psychological Association (APA) as the best psychological state that an individual can achieve, but the concept was found to be invalid by Baumeister et al. [23]. Later, Baumeister developed ego-depletion theory [25], a leading concept in social psychology until it was found to be invalid by [26]. Implicit attitudes theory, the concept undergirding implicit racism, was proposed by Greenwald et al. [27], but later found to be invalid by Tetlock's team [28]. The next failure to generalize, is the leading theory in social psychology developed by Tetlock for predictions to be made by the public and businesses known as superforecasting; however, the first two forecasts made by his highly trained international superforecasters were that the United Kingdom's Brexit would not occur in 2016, and that Donald Trump would not be elected President; both happened. After years of giving TED talks on the value of honesty, [5] Ariely published his new "honesty" scale in the Proceedings of the National Academy of Sciences (PNAS), which recently was retracted by the Editor of PNAS [29]. Apparently, Ariely fabricated the data for his honesty scale.

Social science includes economics. As an example from economics of the same problem with models of closed systems of individual beliefs, Rudd [33] has written that "Mainstream economics is replete with ideas that "everyone knows" to be true, but that are actually arrant non-sense." Rudd focuses on inflation, concluding that expectations (based on surveys) of inflation are not related to the inflation that actually occurs. Rudd's conclusion is part of an ongoing series of arguments about the causes of inflation. For example, from two Nobel Laureates, first has been "the failure of many economists to get inflation right" [34]; and second, inflation has also been attributed to the fear of a wage-price spiral driving expectations [35]. But other economists such as Larry Summers, a leading economist, have predicted that the extraordinary fiscal expenditures during the pandemic would likely cause inflation [36]. Summers was initially contradicted by a proponent of the new economics, known as Modern Monetary Theory, but MMT, which holds that inflation is unlikely from excessive government expenditures, is now on the defensive from the existence of rapidly rising government expenditures associated with inflation [37].

As another example from economics, Leonard [38] concludes that the Federal Reserve's Open Market Committee (FOMC) often misunderstands the US economy by failing to produce intended results, partly due to its adherence to consensus seeking (also known as group think or minority control, often under the auspices of a strong leader of the FOMC; in [5]). As a last example this time with economic game theory, the use of war games in the fleet results in "preordained proofs," per retired General Zinni (in [39]); that is, choose a game for a given context to obtain a desired outcome.

2.5 Definitions. Open Systems and Interdependence

A different model than closed systems of individuals is needed to be replaced by open systems of teams [2]. The approaches to design and operation in the future, however, must also include autonomy and the autonomous operations of human and machine teams and systems. [6] That likely means that these models for autonomy must address conflict and uncertainty, both of which impede or preclude the rational approaches that are only understood in closed systems [40], like game theory [17].

[5] See Ariely giving a TED talk on "How to change your behavior for the better," including honesty at https://www.ted.com/talks/dan_ariely_how_to_change_your_behavior_for_the_better.

[6] To meet the digital future, the Systems Engineering Research Center (SERC) is developing a roadmap for Systems Engineering; e.g., https://www.researchgate.net/publication/340649785_AI4SE_and_SE4AI_A_Research_Roadmap.

As an overly simplistic example: Should the well-trained arm of a human implicated in a fatality be taken to court (a human comparison with ML and DL), the actions of the arm would have to be explained. In this simple case, explanations would have to be provided in a court of law by the human responsible for the arm (e.g. [15]). Explainability for machines is hoped to be provided eventually in real time by AI, but it is neither available today nor at the time of Uber's fatal accident (e.g., [41, 42]). In the case of Uber's pedestrian fatality, the explanation was provided at great expense and after more than a year of intense scrutiny by the National Transportation Safety Board [43].

Another example further prepares us to move beyond the case study. By following simple rules, the work output from a team of uniform workers digging a ditch, or machines in a military swarm, can be aggregated; e.g., "many hands make light work" [1]. With Shannon's [44] rules of communication between two or more independent agents, it is straight forward to model. In contrast, a team or system constituted with orthogonal roles (e.g., a small restaurant with a waiter, cashier and cook), while more common, is much harder to model for the important reason that their perceptions of reality are different. Consider a bi-stable illusion that generates two orthogonal or incommensurable interpretations (e.g., the bi-stable two-faces candlestick illusion). A human perceiving one interpretation of the illusion cannot perceive its bi-stable counterpart simultaneously [45]. Thus, the information collected from two or more workers coordinating while in orthogonal roles can lead to zero correlations, precluding the convergence to a single story from occurring [46]; e.g., despite over a century of being the most successful theory with prediction after successful prediction, quantum theory does not abide by intuition and it resists a rational interpretation (e.g. [47]).

A distinction is necessary. Quantum mathematics is logical, rational and generalizable, however, the interpretations derived from its results are neither logical, rational nor generalizable, an important distinction.

More relevant to our case study, rational approaches, beliefs and behaviors, whether implicit or observed, fail in the presence of uncertainty [48] or conflict [40], exactly where interdependence theory thrives [5]. Facing uncertainty, interdependence theory predicts that free humans engage in debate to exploit the bistable views of reality that naturally exist to explore the tradeoffs that test or search for the best paths going forward, bringing to bear experience, goals, ability to negotiate, fluidity of the situation, all interdependently integrated to confront the uncertainty faced. Thus, in the development of human-machine systems, an environment for interdependence, shared experience, and team learning from training is necessary. This idea also extends to actual teaming; the human and machine must continually test and reevaluate their interdependence *via* jointly developed knowledge/skills/abilities. In particular, reducing the uncertainty faced by a team or system requires that human and machine teammates are both able to explain to each other, however imperfectly, their past actions, present status, and future plans in causal terms [41,42].

Literature. For the human-autonomous vehicle interaction, there are generally accepted concepts and theories in the literature for what technical prerequisites have to happen for humans and machines to become team players; e.g. mutual predictability, directability, shared situation awareness and calibrated trust in automation [49, 50]. In aviation, fly-by wire systems have been implemented that enable human-machine interaction, some of which are being tested in vehicles (e.g., conduct by wire, H-mode; [7] in [51]; and [52]).

2.6 Preliminary Implications for Theory

Lewin [14] founded the discipline of social psychology. His key contribution identified the importance of interdependence in what was then known as "group dynamics." Two of his followers developed a full theory of interdependence centered around Von Neumann's and Morgenstern's [53] idea of games [54,55], but the lead contributor, Thibaut, died and Lewin's student, Kelley [56], gave up on being able to account for why human preferences established before a game failed to predict the choices made during actual games. Jones [20] asserted that ". . . most of our lives are conducted in groups and most of our life-important decisions occur in contexts of social interdependence," but that the study of interdependence in the laboratory was "bewildering." Subsequently, assuming that only i.i.d. data was of value in the replication of experiments (where i.i.d. stands for "independent and identically distributed" data; in [57]), Kenny et al. [58] devised a method to remove the statistical effects of interdependence from experimental data, somewhat akin to treating quantum effects as "pesky" [46]. After the National Academy of Sciences [1] renewed interest in the study of interdependence in 2015, the Academy's review of human-machine teams concluded that the interdependence among team members precluded the attribution of a team's performance to the "disaggregation" of its contributing members ([2], p. 11), directly contradicting Von Neumann's theory of automata, but directly supporting our physics of interdependence. How, then, can it be studied is the goal of this article.

3 CURRENT RESEARCH GAPS. A CASE STUDY

Purpose. The purpose of this case study is to explore some of the implications of teamwork, such as metrics of structure or performance of teams, that can be applied in an AI engineered system by reviewing one of the first autonomous human-machine systems that failed, resulting in a fatal accident. We then attribute the cause of the accident to a lack of interdependence between the human operator and the machine.

[7] https://link.springer.com/referenceworkentry/10.1007/978-3-319-12352-3_60?noAccess=true.

3.1 NTSB Report

The following is summarized from the NTSB [59] report on automation.[8]

On 18 March 2018, a 49-year-old female pedestrian walking a bicycle was fatally struck by a 2017 Volvo XC90 Uber vehicle operating an Automated Driving System (ADS) then under development by Uber's Advanced Technologies Group (ATG).

At the time of the pedestrian fatality, the ATG-ADS had used 1 lidar and 8 radars to measure distance; several cameras for detecting vehicles, pedestrians, reading traffic lights and classifying detected objects; various sensors that had been recently calibrated for telemetry, positioning, monitoring of people and objects, communication, acceleration and angular rates. It also had a human-machine interface (HMI) tablet and a GPS used solely to assure that the car was on an approved and pre-mapped route before engaging the ADS. The ADS allowed the vehicle to operate at a maximum speed of 45 mph (p. 7), to travel only on urban and rural roads, and under all lighting and weather conditions except for snow accumulation. The ADS system was easily disengaged; until then, almost all of its data was recorded (the exception noted below of lost data occurred whenever an alternative determination of an object was made by ADS; e.g., shifting from an "object" in the road to an oncoming "vehicle" ahead).

The ADS constructed a virtual environment from the objects that its sensors detected, tracked, classified and then prioritized based on fusion processes (p. 8). ADS predicted and detected any perceived object's goals and paths as part of its classification system. However, if classifications were made and then changed as happened in this case (e.g., from "object" to "vehicle" and back to "object"), the prior tracking history was discarded, a flaw since corrected; also, pedestrians outside of a crosswalk were not assigned a predicted track, another flaw since corrected.

When ADS detected an emergency (p. 9), it suppressed any action for one second to avoid false alarms. After the 1-s delay, the car's self-braking or evasion could begin, a major flaw since corrected (p. 15). If a collision could not have been avoided, an auditory warning was to be given to the operator at the same time that the vehicle was to be slowed (in the case study, the vehicle may have also begun to slow because an intersection was being approached).

Using the recorded data to replay the accident, before impact, radar first detected the pedestrian 5.6 s before impact; lidar made its first detection at 5.2 s, classified the object as unknown and static, changed to a static vehicle at 4.2 s on a path predicted to be a miss, reclassified to "other" and static but back again to vehicle between 3.8 and 2.7 s, each re-classification discarding its previous prediction history for that object; then a bicycle but static and a miss at 2.6 s; then unknown, static and a miss at 1.5s; then a bicycle and an unavoidable hazard at 1.2 s, the categorization of a hazard immediately initiating "action suppression"; after the 1 s pause, finally an auditory alert was sounded at 0.2 s; the operator took control at 0.02 s before impact; and the operator selected brakes at 0.7 s after impact.

3.2 NTSB Notes

- The indecisiveness of the ADS was partly attributed to the pedestrian not being in a crosswalk, a feature the system was not designed to address (p. 12), since corrected.
- The ADS failed to correctly predict the detected object's path, and only determined it to be a hazard at 1.2 s before impact, causing any action to be suppressed for 1.0 s but, and as a consequence of the impact anticipated in the shortened time-interval remaining before impact, exceeding the ADS design specifications for braking and thus not enacted; after this self-imposed 1.0s delay, an auditory alert was sounded (p. 12).
- For almost 20 min before impact, the HMI presented no requests for its human operator's input (p. 13), likely contributing to the human operator's sense of complacency.

3.3 NTSB Lessons Learned

Several lessons were learned and discussed in the NTSB report.

- The operator was distracted by her personal cell phone ([12], p. v); [9] the pedestrian's blood indicated that she was impaired from drugs and that she violated Arizona State's policy by jaywalking.
- Uber had inadequate safety risk assessments of its procedures, ineffective oversight in real-time of its vehicle operators to determine whether they were being complacent, and exhibited overall an inadequate safety culture (p. vi; see also [60]).
- The Uber ADS was functionally limited, unable to correctly classify the object as a pedestrian, to predict her path, or to adequately assess its risk until almost impact.
- The ADS's design to suppress action for 1 s to avoid false alarms increased the risk of driving on the roads and prevented the brakes from being applied immediately to avoid a hazardous situation. Volvo's ADS was partially disabled to prevent conflicts with its radar which operated on the same frequency as the radar for Uber's ATG-ADS (p. 15).
- By disconnecting the Volvo car's own safety systems, however, Uber increased its systemic risk by eliminating the redundant safety systems for its ADS, since corrected (p. vii).
- According to NTSB's decision, although the National Highway Traffic Safety Administration (NHTSA) had published a third version of its automated vehicles policy, NHTSA provided no means to a self-driving company of evaluating its vehicle's ADS to meet national or State safety regulations, or to provide a company with the detailed guidance to design an adequate ADS to operate safely. NTSB recommended that safety assessment reports submitted to NHTSA, voluntary at the time of NTSB's final report, be made mandatory (p. viii) and uniform across all states; e.g., Arizona had taken no action by the time that NTSB's final report was published.

[8] In this section, page numbers in parenthesis refer to the NTSB [59] report.

[9] In this section, page numbers in parenthesis refer to the NTSB [12] report.

3.4 Three More Case Studies

Several other case studies could be addressed; e.g., Tesla's advanced driver Autopilot failed to detect a truck's side as it entered the roadway [61]; a distracted Tesla driver's autopilot drove through a stop sign, but the car did not alert its distracted driver [62]; and, for the first time, vehicle manslaughter charges have been filed against the driver of a Tesla for misusing its autopilot by running a red light and crashing into another car, killing two persons [63]. These case studies signal that a new technology has arrived and that we must master its arrival with physics to generalize it to an autonomous HMS that is safe and effective.

3.5 Current Research Gaps Summarized

Putting aside issues important to NTSB and the public, from a human-machine team's perspective, by both being independent of each other, the Uber car and its human operator formed an inferior team [15]. Human teams are autonomous, the best being highly interdependent [10], and not exclusively context dependent (currently, however, machine learning models are context dependent, operating in fully defined and carefully curated certain contexts; in [6]). For technology and civilization to continue to evolve [64], what does autonomy require for future human-machine teams and systems? Facing uncertain situations, the NTSB report indirectly confirmed that no single human or machine agent can determine context alone, nor, presently, unravel by themselves the cause of an accident as complex as the Uber fatality (see also [15]); however, resolving uncertainty requires at a minimum a collective goal, a shared experience, and a state of interdependence that integrates these with information from the situation; moreover, autonomy needs the ability to adapt to rapid changes in context [2], and, overall, to operate safely and ethically as an autonomous human-machine system resolves the uncertainty it faces. We know that the findings of Cummings [10] contradict Conant's [65] generalization of Shannon to minimize the interdependence occurring in teams and organizations. And to reduce uncertainty and increase situation awareness, trust and mutual understanding in an autonomous system necessitates that human and machine teammates are able to explain to, or debate with, each other, however imperfectly, their views of reality in causal terms [41,42]. To operate interdependently, humans and machines must share their experiences in part by training, operating and communicating together. To prevent fatalities like those reflected in the case studies requires interdependence. Otherwise, functional independence will lead to more mistakes like those explored by the NTSB about the Uber self-driving car's pedestrian fatality.

3.5.1 A Deeper Analysis

In summary, despite interdependence having originated in social science, by focusing on the i.i.d. data derived from independent individuals in closed system experiments [57], the different schools in social science have been of limited help in advancing the science of interdependence. For example, if the members of a team when interdependent are more productive than the same individuals in a team but who act independently of each other [1,2,7,10], then studying how to increase or decrease the quantity of interdependence and its effects becomes a fundamental issue. However, although Lewin [14] founded social psychology to study interdependence in groups, an Editorial by the new editor of the Journal of Personality and Social Psychology (JPSP): Interpersonal Relations and Group Processes, *JPSP* seeks to publish articles that reflect that "our field is becoming a nexus for social-behavioral science on individuals in context" [66]; Leach's shift towards independence further removes the Journal's founding vision away from the theory of interdependence established by Lewin [14]. Fortunately, the Academy has rejected this regressive shift [1,2].

4 POTENTIAL DEVELOPMENTS. THE MOVE TOWARDS A THEORY OF INTERDEPENDENT AUTONOMY

In systems engineering, structures have here-to-fore been treated as static, physical objects more often designed by computer software with solutions identified by convergence (i.e., Model-Based Engineering, Slide-16; in [67]). In this model, function is a structure's use (SL16), and dynamics is a system's behavior over time (SL26). However, we have found that the structure of an autonomous team is not fixed; e.g., adding redundant, unnecessary members to a fluid team adversely reduces the interdependence between teammates and a team's productivity [5]. In fact, in business mergers, it is common for teams to discard or replace dysfunctional teammates to reach an optimum performance, the motivation for organizations sufficiently free to be able to gain new partners to improve competitiveness, or to spin-off losing parts of a complex business.

4.1 Potential Developments. The Move Towards a Theory of Interdependent Autonomy

We next consider whether there is a thermodynamic advantage in the structure of an autonomous human-machine participants in a team interdependent on their team's performance.

To better make the point, we begin with a return to the history of interdependence, surprisingly by a brief discussion of quantum theory. In 1935 (p. 555), Schrödinger wrote about quantum theory by describing entanglement:

> . . . the best possible knowledge of a *whole* does not necessarily include the best possible knowledge of all its *parts,* even though they may be entirely separate and therefore virtually capable of being 'best possibly known' . . . The lack of knowledge is by no means due to the interaction being insufficiently known . . . it is due to the interaction itself. . . .

Similarly, Lewin [14], the founder of Social Psychology, wrote that the "whole is greater than the sum of its parts."

Likewise, from the *Systems Engineering Handbook* [11], "A System is a set of elements in interaction" [68] where systems ". . . often exhibit emergence, behavior which is meaningful only when attributed to the whole, not to its parts" [69].

There is more. Returning to Schrödinger (p. 555),

> Attention has recently been called to the obvious but very disconcerting fact that even though we restrict the disentangling measurements to *one* system, the representative obtained for the *other* system is by no means independent of the particular choice of observations which we select for that purpose and which by the way are *entirely* arbitrary.

If the parts of a team are not independent [8], and if the parts of a whole cannot be disaggregated ([2], p. 11), does a state of interdependence among the orthogonal, complementary parts of a team confer an advantage to the whole [15]? An answer comes from the science of human teams: Compared to a collection of the same but independent scientists, the members of a team of scientists when interdependent are significantly more productive [1,10]. Structurally, to achieve and maintain maximum interdependence, a team must not have superfluous teammates [5]; that is, a team must have the least number of teammates necessary to accomplish its mission. The physics of an autonomous whole, then, means a loss of independence among its parts; i.e., the independent parts must fit together into a "structural" whole, characterized by a reduction in the entropy produced by the team's structure [5]. Thus, for an autonomous whole to be greater than the sum of its individual parts, unlike most practices in social science (the exception being commercial airliner teams; in [13]) or systems engineering, structure and function must be treated interdependently [5].

For a ground state, when a team's structure is stable and existing at a low state of emotion, **Eq. 1** captures Lewin's [14] notion that the whole, *S*, is greater than the sum of its parts, (S_i), and System Engineering's conjecture of the emergence of synergy, both occurring when the whole produces less entropy than the sum of its parts:

$$S_{Whole} \le \sum\nolimits_{i=1}^{n} S_i \tag{1}$$

In contrast, an excited state occurs with internal conflict in a structure [70], when teammates are independent of each other, or when emotion courses through a team as happened with the tragic drone strike in Afghanistan on 29 August 2021 [71], then the whole becomes less than the sum of its parts as all of a team's free energy is consumed by individuals heedless of their rush to judgment, captured by **Eq. 2**:

$$S_{Whole} \ge \sum\nolimits_{i=1}^{n} S_i \tag{2}$$

Interdependence theory guides us to conclude that the intelligent interactions of teammates requires that the teammates be able to converse in a bidirectional causal language that all teammates in an autonomous system can understand; viz., intelligent interactions guide the team to choose teammates that best fit together. In the limit as the parts of a whole become a whole [15], the entropy generated by an autonomous team's or system's whole structure must drop to a minimum to signify the well-fitted team, allowing the mission of the best teams to maximize performance (maximum entropy production, or MEP; in [72]); e.g., by overcoming the obstacles faced [73]; by exploring solution space for a patent [74]; or by merging with another firm to reduce a system's vulnerability. [10] In autonomous systems, characterizing vulnerability in the structure of a team, system or an opponent was the job that Uber failed to perform in a safety analysis of its self-driving car; instead, it became the job that NTSB performed for the Uber team. But as well, the Uber self-driving car and its operator never became a team, remaining as independent parts of a whole (viz., **Eq. (2)**); nor did the Uber car recognize that its operator had become complacent and that the Uber car needed to take an action to protect itself, its human operator and the pedestrian it was about to strike by stopping safely [9]. Unfortunately, even with intelligence being designed into autonomous cars, vehicles are still being designed as tools for human drivers and not as collaborative human-machine teams. Until the car and driver collaboratively learn, train, work and share experiences together interdependently, such mistakes will continue to occur.

This review fits with a call for a new physics of life to study "state-dependent dynamics" (e.g., an example may be quantum biology; in [4]), another call for a new science of social interaction [75], for how humans interact socially with machines (e.g., the CASA paradigm, where human social reactions to computers was studied, in [76–78])), and another to move beyond i.i.d. data [57] in the pursuit of a new theory of information value [79]. The problems with applying social science and Shannon's information theory to teams and systems are becoming clearer as part of an interdisciplinary approach to a new science of autonomous human-machine teams and systems, leading us to focus on managing the positive and negative effects of interdependence. One of the end results, for which we strive in the future, is the new science of information value [80].

In sum, as strengths of interdependence, we have proposed that managing interdependence with AI is critical to the mathematical selection, function and characterization of an aggregation of agents engineered into an intelligent, well-performing unit, achieving MEP in a complementary tradeoff with structure, like the focusing of a telescope. Once that state occurs, disaggregation for analysis of how the parts contribute to a team's success is not possible ([2], p. 11). Interdependence also tells us that each person or machine must be selected in a trial-and-error process, meaning that the best teams cannot be replicated, but they can be identified [5]; and, second, the information for a successful, well-fitted team cannot be obtained in static tests but is only available from the dynamics afforded by the competitive situations in the field able to stress a team's structure as it performs its functions autonomously; i.e., not every good idea for a new structure succeeds in reality

[10] For example, Huntington Ingalls Industries has purchased a company focused on autonomous systems [114].

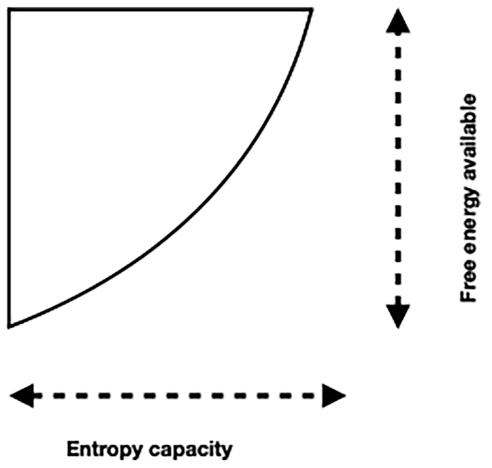


FIGURE 1 | An open systems' notional diagram of free energy abstracted from Gibbs.[11] From it, we see that an organization provides its team with sufficient Helmholtz free energy (the ordinate) "from an external source . . .(to maintain its) dissipative structure" [85] by offsetting its waste and products produced (the abscissa). We illustrate with a notional diagram of free energy abstracted from Gibbs (closed systems).

(e.g., a proposed health venture became "unwieldy," in [81]). This conclusion runs contrary to matching theory (e.g. [82]) and rational theory [40]. But it holds in the face of uncertainty and conflict for autonomous systems (cf [48]), including autonomous driving (e.g. [83]).

4.2 A Brief Model of Team Entropy Introduced, With Generalizations

In **Figure 1**, we have proposed a model of the entropy production by teams from the application of the free energy available to an autonomous human-machine team to be able to conduct its teamwork and perform its mission (e.g., [46,84]).

Interdependence between structure and performance implies that a limited amount of free energy is available for a team to care for its teammates and perform its mission. For an open-system model of teams, we propose that interdependence between structure and performance uses the free energy available to create a trade-off between uncertainty in the entropy produced by the structure of an autonomous human-machine system and uncertainty in its performance [46,84]:

$$\Delta(structure)*\Delta(per\,formance) \sim C \tag{3}$$

In **Eq. 3**, uncertainty in the entropy produced by a team's structure times uncertainty in the entropy produced by the performance of the team is approximately equal to a constant, *C*. As structural costs minimize, the emergence of synergy occurs as the team's performance increases to a maximum, increasing its power.

The predictions from **Eq. 3** are counterintuitive. Applying it to concepts and action results in a tradeoff: as uncertainty in a concept converges to a minimum, the overriding goal of social scientists, uncertainty in the behavioral actions covered by that concept increase exponentially, rendering the concept invalid, the result that has been found for numerous concepts; e.g., self-esteem [23]; implicit attitudes [28]; ego-depletion [26]. These problems with concepts have led to the widespread demand for replication [24]. But the demand for replication more or less overlooks the larger problem with the lack of generalizability arising from what amounts to the use of strictly independent data collected from individual agents [57], which we have argued, cannot recreate the social effects being observed or captured.

In contrast, with **Eq. 3**, interdependence theory generalizes to several effects. To illustrate, we briefly discuss authoritarianism; risk perception; mergers; deception; rational; and vulnerability and emotion.

4.2.1 Authoritarianism

Authoritarians attempt to reduce social noise by minimizing structural effects under their control. However, instead of seeking the best teams with trail and error processes, authoritarians and gangs seek the same effect by suppressing alterative views, social strife, social conflict, etc. Consequently, these systems are unable to innovate [74]. Two examples are given by China and Amazon: Enforced cooperation in China increases its systemic vulnerability to risk and its need to steal innovations (e.g., [86,87]). Similarly, monopolies increase their organizational vulnerability to risk and their need to *steal* innovations from their clients (e.g., Amazon, in [88]).

4.2.2 Risk Determination vs. Risk Perception

Applying **Eq. 1** first to the risk determination of an uncertain event and then to the subjective risk perception of the same event, not surprisingly, the two risks may not agree. For example, Slovic et al. [89] found large differences between the risks determined by experience and calculations vs. the perceived risks associated with nuclear wastes. In the case of the tragic drone attack in Afghanistan by the US Air Force that killed an innocent man and several children, the risk assessment was driven by risk perceptions that led to an emotional rush to judgment, leading to a tragic result [71].

Human observers can generate an infinite spectrum of possible interpretations or risk perceptions, including non-sensical and even dangerous ones as experienced by DoD's [71] unchallenged decision to launch what became its very public and tragic drone attack. Humans have developed two solutions to this quandary: suppress all but the desired perception, e.g., with authoritarian leader's or monopolist's rules that preclude action except theirs [90] or battle-test the risk perceptions in a competitive debate between the chosen perception and its competing alternative perceptions, deciding the best with majority rules [91]. DoD [71] attributed its failure to its own suppression of alternative interpretations. After its failed drone attack, the Air Force concluded that it needed to test both risk assessments and risk perceptions before launching new drone

[11] http://esm.rkriz.net/classes/ESM4714/methods/free-energy.html

attacks. The Air Force concluded that one way to test these judgments is to debate the decision before the launch of a drone (i.e., with the use of "red teams").

4.2.3 Mergers

Several reviews of mergers and acquisitions over the years have found mostly failures; e.g., Sirower [92] concluded that two-thirds of mergers were ultimately unsuccessful. As an example of failure, America Online (AOL) acquired Time Warner in 2001, a megamerger that almost doubled the size of AOL, but the merged firm began to fail almost immediately. From **Eq. 3**, the entropy generated by its new structure could not be minimized by AOL and in fact grew, leading to an extraordinary drop in performance in 2002, the loss of Time Warner in 2009, and a depleted AOL's acquisition by Verizon in 2015. In comparison, Apple, one of the most successful companies in the world, acquires a company or more every few weeks, usually as a faction of a percent of Apple's size, the new firms quickly absorbed [93]. From this comparison, we conclude that it is not possible to determine how a new teammate will work out, requiring a trial and error process, the best fit characterized by **Eq. 3** as a reduction in entropy signifying the fit, most likely when a target company provides a function not available to a firm but that complements it orthogonally [46].

Numerous other examples exist. UPS plans to spin-off its failing truck business [94]. Fiat Chrysler's merger with PSA to form Stellantis in 2021 was designed to better compete in its market [95]. Facebook's merger now plans to shore up its vulnerability after Apple revised its privacy policy, which adversely affected Facebook's advertisement revenue [96]. Mergers can also be forced by a government, but the outcome may not be salutary (e.g., Didi's ride-hailing business has been forced by the Chinese government to allow its representatives to participate in Didi's major corporate decisions; in [97]).

4.2.4 Deception

Equation 3 tells us that the best deceivers do not stand out, but instead, fit into a structure as if they belong. One of the key means of using deception is to infiltrate into a system, especially in computational or cyber-security systems [98]. If done by not disturbing the structure of a system or team, deception applied correctly will not increase the structural entropy generated by a team or system, allowing a spy to practice its trade undetected. From Tzu [99], to enact deception: "Engage people with what they expect; it is what they are able to discern and confirms their projections. It settles them into predictable patterns of response, occupying their minds while you wait for the extraordinary moment—that which they cannot anticipate."

4.2.5 Rational

There is another way to address "rational." As we alluded earlier, the rational can also be formal knowledge [100]. When the "rational" is formal knowledge, it is associated with the effort for logical, analytical reasoning versus the easy non-analytical path of recognition afforded by intuition, which may be incorrect. As iterated before, humans approach naïve intuition or perception by challenging it, leading to a debate, with the best idea surviving the test [84]. But, in addition, and as a generalization from Shannon [44] that we accept, knowledge produces zero entropy [65]. We generalize Conant's concept by applying it to the structure of a perfect team; we have found that, in the limit, the perfect team's structure minimizes the production of entropy by minimizing its degrees of freedom [46]. If we look at the production of entropy as a tradeoff, minimizing the entropy wasted on its own structure allows, but does not guarantee, that a team has more free energy available to maximize its production of entropy (MEP) in the performance of its mission.

4.2.6 Vulnerability, Internal Conflict and Emotion

The effect of conflict in a team illuminates a team's structural vulnerability (e.g. [70]). Internal conflict in a team is essential to identifying vulnerability by a team's opponents during a competition. Regarding the team itself, training is a means to identify and repair (with mergers, etc.) self-weaknesses in a team to prevent it from being exploited by an opponent.

The open conflict between Apple and Facebook provides an excellent example of targeting a structural vulnerability in Facebook by Apple and publicized during the aggressive competition between these two firms. As reported in the *Wall Street Journal* [101],

> Facebook Inc. will suffer damage to its core business when Apple Inc. implements new privacy changes, advertising industry experts say, as it becomes harder for the social-media company to gather user data and prove that ads on its platform work. The core of Facebook's business, its flagship app and Instagram, would be under pressure, too. The Apple change will require mobile apps to seek users' permission before tracking their activity, restricting the flow of data Facebook gets from apps to help build profiles of its users. Those profiles allow Facebook's advertisers to target their ads efficiently. The change will also make it harder for advertisers to measure the return they get for the ads they run on Facebook—how many people see those ads on mobile phones and take actions such as installing an app, for example.

5 DISCUSSION. THE GAPS IN A THEORY OF INTERDEPENDENCE AND AUTONOMY

Interdependence is an unsolved problem that requires more than traditional social science and systems engineering. It is a hard problem. Jones [20] found that a study of interdependence in the laboratory caused "bewildering complexities." Despite his reservations about interdependence, we review our findings and those from the literature that point to the best path going forward to adopt the physics based approach offered by the phenomenon of interdependence in autonomous teams and systems.

Interdependent teams cannot be disaggregated to rationally approach the parts of a team from an individual performer's perspective ([2], p. 11). But we can observe how teams perform with the team as the unit of analysis, we can reduce redundancy to improve interdependence and performance, and we can add better teammates and replace inferior teammates to improve performance [74,102]. This means that a rational approach on paper to building a team is bound to produce poor results. A trial and error method to see what works in the field is the best approach and that can only be determined by a reduction in structural entropy production with, as part of the tradeoff, increases in maximum entropy production measured by the team's overall performance.

From the National Academy of Sciences [2], solutions to autonomous human-machine problems must be found in the field and under the conditions where the autonomy will operate. There, free choices should govern as opposed to the forced choices offered to participants in games. There, teams and organizations must be free to discard or replace the dysfunctional parts of a team, and free to choose the best choice available among the replacement candidates to test whether a good fit occurs. There, vulnerability is also a concern when competing against another team, tested by selling a company's stock short [103–105]. [12] Namely, a vulnerability is characterized by an increase in structural entropy production [84]. There, emotion becomes a factor: as a vulnerability is exposed, emotion increases above a ground state; e.g., the recent financial sting reported by Meta's Facebook from its billions of dollars in losses caused by Apple's new privacy advertisement policy (in [106]).

When minority control by authoritarian leaders impedes the reorganization of structures designed to maximize performance, it is likely to reduce innovation; e.g., by reducing interdependence after adding redundant workers (e.g. [107]); by constraining the choices available to teams and systems [84]; or by reducing the education available to a citizenry (as in the Middle Eastern North African countries plus Israel, where we found that the more education across a free citizenry, the more innovation a country experienced; in [74]). Authoritarian control (by a gang, a monopoly, a country) can suppress the many supported by a group by using forced cooperation to implement its rules, but the more followers that are forced to cooperate, the more that innovation is impeded.

In contrast, Axelrod [105] concludes based on game theory that competition reduced social welfare: "the pursuit of self-interest by each [participant] leads to a poor outcome for all." This outcome can be avoided, Axelrod argued, when sufficient punishment exists to discourage competition. Perc et al. [109] agree that " we must learn how to create organizations, governments, and societies that are more cooperative and more egalitarian ... " Contradicting Axelrod, Perc and others, we have found the opposite, that the more competitive is a county, the better is its human-development, its productivity, and its standard of living [74,102,110]. For example, China's forced cooperation across its system of communes promulgated by its Great Leap Forward program was modeled after [111]:

> the Soviet model of industrialization in China [which failed]. . . . The inefficiency of the communes and the large-scale diversion of farm labour into small-scale industry disrupted China's agriculture seriously, and three consecutive years of natural calamities added to what quickly turned into a national disaster; in all, about 20 million people were estimated to have died of starvation between 1959 and 1962.

By rejecting Axelrod's and China's use of punishment to enforce its minority control, when an interdependence between culture and technology is allowed to freely exist, free expression "reflects interdependent processes of brain-culture co-evolution" [64].

Our study is not exhaustive (e.g., due to the limitations of space, we left out: factorable tensors, implying no interdependence; orthogonality, precluding individuals from being able to multitask; competition generates bistable information; perturbations collapse teams with redundancy or otherwise poorly structured and operated teams; etc.; we also had plans to apply our physics in **Eq. 3** to the U.S. Army's Multi Domains Operations, or MDO, to show that, based on interdependence theory, MDO would be an inferior application for autonomous human-machine teams because its agents are independent, precluding synergy or power from team arrangements). Thus, we have much to study in the future.

6 CONCLUSION

We conclude that the cause of the Uber self-driving car accident was the lack of interdependence between the human operator and the machine. The case studies and theory indicate that no synergy arises when teammates remain independent of each other (**Eqs 1**, **3**). Internal conflict causes a vulnerability in a team that can be exploited by an opponent. In contrast, for autonomy to occur, an HMS must have shared experiences by training, operating, and communicating together to control each other. When that happens, when a structure of a team is stable and producing minimum entropy in a state of interdependence, synergy occurs (a mutually beneficial symbiosis). Self-awareness of each other and of the team must be built during training and continued during operations. Interdependence requires situation awareness of the environment of each other and of the team's performance; trust; sustainable attention; mutual understanding; and communication devices all come in to play (for a review of bidirectional trust, see [8]).

Traditional social science is weakest when it has little to say to improve states of interdependence, strongest when it contributes to its advancement. By not sidestepping the physics of what is occurring in the physical reality of a team, we conclude that a state of interdependence cannot be disaggregated into elements that can then be summed by states of independence to recreate the

[12] https://www.investopedia.com/terms/s/shortsale.asp.

interdependent event being witnessed [2]. This happens because interdependence reduces the degrees of freedom for a whole, precluding a simple aggregation of the parts for the whole. A human-machine awareness of each other in a team, and of the team as a team, a human-machine team sharing coordination among its teammates, a human-machine team collaborating together as a team, all are necessary.

In closing, social systems based on closed models used to study independent agents are unable to contribute to the evolution posed by autonomous human-machine systems. The data derived from these models are subjective, whether based on game theory (e.g. [108]), rational choices (e.g. [40]), or superforecasts (e.g. [30]). To be of value, subjective interpretations must be tested, challenged or debated. As we have portrayed in this review, the disruption to social theory posed by human-machine systems is more likely to be revolutionary, a dramatic change that autonomy scientists working with human-machine teams and systems must be prepared to contribute, to manage and to live with.

6.1 Conclusion. The Contribution of the Manuscript to the Literature

This manuscript contributes to the literature by applying basic concepts from physics to an autonomous HMS. **Equation 3** is a metric for the tradeoffs between an autonomous human-machine team's structure and its performance. We also recognize that interdependence is a phenomenon in nature that can be modeled with physics like any other natural phenomenon. We recognize that an autonomous HMS cannot occur with independent agents. And we have postulated that one of the contributions by future machines in an autonomous HMS is by monitoring the emotional states among its human teammates with alerts about distorted situational awareness, by providing an open awareness of their emotional states, and by impeding their haste to decide.

AUTHOR CONTRIBUTIONS

The author confirms being the sole contributor of this work and has approved it for publication.

AUTHOR CONTRIBUTIONS

The author confirms being the sole contributor of this work and has approved it for publication.

FUNDING

Parts of this manuscript were written while the author was a Summer Fellow at the Naval Research Laboratory, DC, funded by the Navy's Office of Naval Research.

ACKNOWLEDGMENTS

An abbreviated version of this manuscript was published in the Interactive AI Magazine;[13] and an earlier version of this article was invited by the editors of the revised Systems Engineering Handbook (to be published in 2023). I acknowledge the contributions of: Thomas Shortell (AI Systems Working Group, Chair, INCOSE;[14] Systems Engineering, Lockheed Martin Co.; thomas.m.shortell@lmco.com); Tom McDermott (AI Systems Working Group, Chair, INCOSE; Deputy Director, Systems Engineering Research Center, Stevens Institute of Technology in Hoboken, NJ; tamcdermott42@gmail.com); Ranjeev Mittu (Branch Head, Information Management and Decision Architectures Branch, Information Technology Division, U.S. Naval Research Laboratory in Washington, DC; ranjeev.mittu@nrl.navy.mil); and Donald Sofge (Lead, Distributed Autonomous Systems Group, Navy Center for Applied Research in Artificial Intelligence, U.S. Naval Research Laboratory, DC; donald.sofge@nrl.navy.mil).

REFERENCES

1. NJ Cooke ML Hilton, editors. *Enhancing the Effectiveness of Team Science. National Research Council*. Washington (DC): National Academies Press (2015).
2. Endsley MR. Human-AI Teaming: State-Of-The-Art and Research Needs. In: *The National Academies of Sciences-Engineering-Medicine*. Washington, DC: National Academies Press (2021). Accessed 12/27/2021 from https://www.nap.edu/catalog/26355/human-ai-teaming-state-of-the-art-and-research-needs.
3. Epstein B. In: EN Zalta, editor. *Social Ontology, the Stanford Encyclopedia of Philosophy*. Stanford, CA: The Metaphysics Research Lab (2021). Accessed 4/1/2022 from https://plato.stanford.edu/entries/social-ontology/.
4. Davies P. Does New Physics Lurk inside Living Matter? *Phys Today* (2021) 73(8). doi:10.1063/PT.3.4546
5. Lawless WF. The Interdependence of Autonomous Human-Machine Teams: The Entropy of Teams, but Not Individuals, Advances Science. *Entropy* (2019) 21(12):1195. doi:10.3390/e21121195
6. Peterson JC, Bourgin DD, Agrawal M, ReichmanGriffiths DTL, Griffiths TL. Using Large-Scale Experiments and Machine Learning to Discover Theories of Human Decision-Making. *Science* (2021) 372(6547):1209–14. doi:10.1126/science.abe2629
7. Cooke NJ, Lawless WF. Effective Human-Artificial Intelligence Teaming. In: WF Lawless, R Mittu, DA Sofge, T Shortell, TA McDermott, editors. *Engineering Science and Artificial Intelligence*. Springer (2021). doi:10.1007/978-3-030-77283-3_4
8. WF Lawless, R Mittu, D Sofge, S Russell, editors. *Autonomy and Artificial Intelligence: A Threat or Savior?* New York: Springer (2017).
9. Sofge D, Lawless W, Mittu R. AI Bookie. *AIMag* (2019) 40(3):79–84. doi:10.1609/aimag.v40i3.5196
10. Cummings J. *Team Science Successes and Challenges*. Bethesda MD: National Science Foundation Sponsored Workshop on Fundamentals of Team Science and Science of Team Science (2015).
11. DD Walden, GJ Roedler, KJ Forsberg, RD Hamelin, TM Shortell, editors. *Systems Engineering Handbook. Prepared by International Council on System Engineering (INCOSE-TP-2003-002-04)*. 4th ed. Hoboken, NJ: John Wiley & Sons (2015).

[13] https://interactiveaimag.org/updates/reports/symposium-reports/reports-of-the-association-for-the-advancement-of-artificial-intelligences-2020-fall-symposium-series/.

[14] INCOSE: International Council on Systems Engineering; https://www.incose.org.

12. NTSB. *Collision between Vehicle Controlled by Developmental Automated Driving System and Pedestrian, Tempe, AZ, March 18, 2018. National Transportation Safety Board (NTSB), Accident Report*. Washington, DC: NTSB/HAR-19/03. PB2019-101402 (2019).
13. Bisbey TM, Reyes DL, Traylor AM, Salas E. Teams of Psychologists Helping Teams: The Evolution of the Science of Team Training. *Am Psychol* (2019) 74(3):278–89. doi:10.1037/amp0000419
14. Lewin K. In: D Cartwright, editor. *Field Theory of Social Science*. New York: Harper and Brothers (1951).
15. Lawless WF, Mittu R, Sofge D, Hiatt L. Artificial Intelligence, Autonomy, and Human-Machine Teams - Interdependence, Context, and Explainable AI. *AIMag* (2019) 40(3):5–13. doi:10.1609/aimag.v40i3.2866
16. Liu Y-Y, Barabási A-L. Control Principles of Complex Systems. *Rev Mod Phys* (2016) 88(3):035006. doi:10.1103/RevModPhys.88.035006
17. Amadae SM. *Rational Choice Theory. Political Science and Economics*. London, UK: Encyclopaedia Britannica (2017). from https://www.britannica.com/topic/rational-choice-theory. Accessed 12/15/2018.
18. Arora S, Doshi P. *A Survey of Inverse Reinforcement Learning: Challenges, Methods and Progress*. Ithaca, NY. arXiv: 1806.06877v3 (2020).
19. Gray NS, MacCulloch MJ, Smith J, Morris M, Snowden RJ. Violence Viewed by Psychopathic Murderers. *Nature* (2003) 423:497–8. doi:10.1038/423497a
20. Jones EE. Major Developments in Five Decades of Social Psychology. In: DT Gilbert, ST Fiske, G Lindzey, editors. *The Handbook of Social Psychology, Vol. I*. Boston: McGraw-Hill (1998). p. 3–57.
21. Thagard P. Cognitive Science. In: EN Zalta, editor. *The Stanford Encyclopedia of Philosophy*. Department of Philosophy, Stanford University (2019). Accessed 11/15/2020 from https://plato.stanford.edu/archives/spr2019/entries/cognitive-science.
22. Diener E. Subjective Well-Being. *Psychol Bull* (1984) 95(3):542–75. doi:10.1037/0033-2909.95.3.542
23. Baumeister RF, Campbell JD, Krueger JI, Vohs KD. Exploding the Self-Esteem Myth. *Sci Am* (2005) 292(1):84–91. doi:10.1038/scientificamerican0105-84
24. Nosek B. Estimating the Reproducibility of Psychological Science. *Science* (2015) 349(6251):943.
25. Baumeister RF, Vohs KD. Self-Regulation, Ego Depletion, and Motivation. *Social Personal Psychol Compass* (2007) 1. doi:10.1111/j.1751-9004.2007.00001.x
26. Hagger MS, Chatzisarantis NLD, Alberts H, Anggono CO, Batailler C, Birt AR, et al. A Multilab Preregistered Replication of the Ego-Depletion Effect. *Perspect Psychol Sci* (2016) 11(4):546–73. doi:10.1177/1745691616652873
27. Greenwald AG, McGhee DE, Schwartz JLK. Measuring Individual Differences in Implicit Cognition: The Implicit Association Test. *J Personal Soc Psychol* (1998) 74(6):1464–80. doi:10.1037/0022-3514.74.6.1464
28. Blanton H, Jaccard J, Klick J, Mellers B, Mitchell G, Tetlock PE. Strong Claims and Weak Evidence: Reassessing the Predictive Validity of the IAT. *J Appl Psychol* (2009) 94(3):567–82. doi:10.1037/a0014665
29. Berenbaum MR. Signing at the Beginning Makes Ethics Salient and Decreases Dishonest Self-Reports in Comparison to Signing at the End. *PNAS* (2021) 118(38):e2115397118. doi:10.1073/pnas.2115397118
30. Tetlock PE, Gardner D. *Superforecasting: The Art and Science of Prediction*. New York City, NY: Crown (2015).
31. Shu LL, Mazar N, Gino F, Ariely D, Bazerman MH. Signing at the Beginning Makes Ethics Salient and Decreases Dishonest Self-Reports in Comparison to Signing at the End. *Proc Natl Acad Sci U.S.A* (2012) 109:15197–200. doi:10.1073/pnas.1209746109
32. Lawless W F. Risk Determination versus Risk Perception: A New Model of Reality for Human-Machine Autonomy. *Informat* (2022) 9(2):30. doi:10.3390/informatics9020030
33. Rudd JB. *Why Do We Think that Inflation Expectations Matter for Inflation?* Washington, D.C: Federal Reserve Board (2021). doi:10.17016/FEDS.2021.062
34. Krugman P. *When Do We Need New Economic Theories?* New York City, NY: New York Times (2022). Accessed 2/8/2022 from https://www.nytimes.com/2022/02/08/opinion/economic-theory-monetary-policy.html.
35. Shiller RJ. *Inflation Is Not a Simple Story about Greedy Corporations*. New York City, NY: New York times (2022). Accessed 2/8/2022 from https://www.nytimes.com/2022/02/08/opinion/dont-blame-greed-for-inflation.html.
36. Smialek J. *Why Washington Can't Quit Listening to Larry Summers*. New York City, NY: New York Times (2021). Accessed 2/8/2022 from https://www.nytimes.com/2021/06/25/business/economy/larry-summers-washington.html.
37. Smialek J. *Is This what Winning Looks like? Modern Monetary Theory*. New York City, NY: New York Times (2022). Accessed 2/8/2022 from https://www.nytimes.com/2022/02/06/business/economy/modern-monetary-theory-stephanie-kelton.html.
38. Leonard C. *The Lords of Easy Money*. New York City, NY: Simon & Schuster (2022).
39. Augier M, Barrett SFX. *General Anthony Zinni (Ret.) on Wargaming Iraq, Millennium Challenge, and Competition*. Washington, DC: Center for International Maritime Security (2021). Accessed 10/21/2021 from https://cimsec.org/general-anthony-zinni-ret-on-wargaming-iraq-millennium-challenge-and-competition/.
40. Mann RP. Collective Decision Making by Rational Individuals. *Proc Natl Acad Sci U S A* (2018) 115(44):E10387–E10396. doi:10.1073/pnas.1811964115
41. Pearl J. Reasoning with Cause and Effect. *AI Mag* (2002) 23(1):95. doi:10.1609/aimag.v23i1.1612
42. Pearl J, Mackenzie D. *AI Can't Reason Why*. New York City: Wall Street Journal (2018). Accessed Apr. 27, 2020. from https://www.wsj.com/articles/ai-cant-reason-why-1526657442.
43. NTSB. *Preliminary Report Released for Crash Involving Pedestrian*. Washington, DC: Uber Technologies, Inc. (2019). Accessed 5/1/2019 from https://www.ntsb.gov/news/press-releases/Pages/NR20180524.aspx.
44. Shannon CE. A Mathematical Theory of Communication. *Bell Syst Tech J* (1948) 27:379–423. , 623–656. doi:10.1002/j.1538-7305.1948.tb01338.x
45. Eagleman DM. Visual Illusions and Neurobiology. *Nat Rev Neurosci* (2001) 2(12):920–6. doi:10.1038/35104092
46. Lawless WF. Quantum-Like Interdependence Theory Advances Autonomous Human-Machine Teams (A-HMTs). *Entropy* (2020) 22(11):1227. doi:10.3390/e22111227
47. Weinberg S. Steven Weinberg and the Puzzle of Quantum Mechanics. In: N David Mermin, J Bernstein, M Nauenberg, J Bricmont, S Goldstein, editors. *In Response to: The Trouble with Quantum Mechanics from the January 19, 2017 Issue*. New York City: The New York Review of Books (2017).
48. Hansen LP. Nobel Lecture: Uncertainty outside and inside Economic Models. *J Polit Economy* (2014) 122(5):945–87. doi:10.1086/678456
49. Walch M, Mühl K, Kraus J, Stoll T, Baumann M, Weber M. From Car-Driver-Handovers to Cooperative Interfaces: Visions for Driver-Vehicle Interaction in Automated Driving. In: G Meixner C Müller, editors. *Automotive User Interfaces*. Springer International Publishing (2017). p. 273–94. doi:10.1007/978-3-319-49448-7_10
50. Christoffersen K, Woods D. 1. How to Make Automated Systems Team Players. *Adv Hum Perform Cogn Eng Res* (2002) 2:1–12. doi:10.1016/s1479-3601(02)02003-9
51. Winner H, Hakuli S. Conduct-by-wire–following a New Paradigm for Driving into the Future. *Proc FISITA World Automotive Congress* (2006) 22:27.
52. Flemisch FO, Adams CA, Conway SR, Goodrich KH, Palmer MT, Schutte PC. *The H-Met-Aphor as a Guideline for Vehicle Automation and Interaction*. Washington, DC: NASA/TM-2003-212672 (2003).
53. Von Neumann J, Morgenstern O. *Theory of Games and Economic Behavior*. Princeton: Princeton University Press (1953). (originally published in 1944).
54. Thibaut JW, Kelley HH. *The Social Psychology of Groups*. New York: John Wiley & Sons (1959).
55. Kelley HH, Thibaut JW. *Interpersonal Relations. A Theory of Interdependence*. New York: Wiley (1978).
56. Kelley HH. Lewin, Situations, and Interdependence. *J Soc Issues* (1991) 47:211–33. doi:10.1111/j.1540-4560.1991.tb00297.x
57. Schölkopf B, Locatello F, Bauer S, Ke NR, Kalchbrenner N, Goyal A, et al. *Ithaca, NY: Towards Causal Representation Learning*. arXiv, Accessed 7/6/2021 (2021).
58. Kenny DA, Kashy DA, Bolger N, Gilbert DT, Fiske ST, Lindzey G. Data Analyses in Social Psychology. In: *Handbook of Social Psychology*. 4th ed. Boston, MA: McGraw-Hill (1998). p. 233–65.

59. NTSB. *Vehicle Automation Report*. Washington, DC: National Transportation Safety Board (2019). Accessed 12/3/2020 from https://dms.ntsb.gov/pubdms/search/document.cfm?docID=477717anddocketID=62978andmkey=96894.
60. NTSB. *Inadequate Safety Culture' Contributed to Uber Automated Test Vehicle Crash*. Washington, DC: National Transportation Safety Board (2019). Accessed 4/11/2020 from https://www.ntsb.gov/news/press-releases/Pages/NR20191119c.aspx.
61. Hawkins AJ. *Tesla's Autopilot Was Engaged when Model 3 Crashed into Truck, Report States*. New York City: The Verge (2019). Accessed 3/27/2022 from https://www.theverge.com/2019/5/16/18627766/tesla-autopilot-fatal-crash-delray-florida-ntsb-model-3.
62. Boudette NE. *'It Happened So Fast': Inside a Fatal Tesla Autopilot Accident*. New York City, NY: New York Times (2021). Accessed 3/27/2022 from https://www.nytimes.com/2021/08/17/business/tesla-autopilot-accident.html.
63. NPR. *A Tesla Driver Is Charged in a Crash Involving Autopilot that Killed 2 People*. Washington, DC: NPR (2022). Accessed 3/27/2022 from https://www.npr.org/2022/01/18/1073857310/tesla-autopilot-crash-charges?t=1647945485364.
64. Ponce de León MS, Bienvenu T, Marom A, Engel S, Tafforeau P, Alatorre Warren JL, et al. The Primitive Brain of Early Homo. *Science* (2021) 372(6538):165–71. doi:10.1126/science.aaz0032
65. Conant RC. Laws of Information Which Govern Systems. *IEEE Trans Syst Man Cybern* (1976) SMC-6:240–55. doi:10.1109/tsmc.1976.5408775
66. Leach CW. Editorial. *J Personal Soc Psychol* (2021) 120(1):30–2. doi:10.1037/pspi0000226
67. Douglass BP. *"Introduction to Model-Based Engineering?" Senior Principal Agile Systems Engineer*. Mitre (2021). Accessed 2/13/2022 from https://www.incose.org/docs/default-source/michigan/what-does-a-good-model-smell-like.pdf?sfvrsn=5c9564c7_4.
68. Bertalanffy L. *General System Theory: Foundations, Development, Applications, Rev*. New York: Braziller (1968).
69. Checkland P. *Systems Thinking, Systems Practice*. New York: Wiley (1999).
70. Vanderhaegen F. Heuristic-based Method for Conflict Discovery of Shared Control between Humans and Autonomous Systems - A Driving Automation Case Study. *Robotics Autonomous Syst* (2021) 146:103867. doi:10.1016/j.robot.2021.103867
71. DoD. Pentagon Press Secretary John F. Kirby and Air Force Lt. Gen. Sami D. Said Hold a Press Briefing (2021). Accessed 11/3/2021 from https://www.defense.gov/News/Transcripts/Transcript/Article/2832634/pentagon-press-secretary-john-f-kirby-and-air-force-lt-gen-sami-d-said-hold-a-p/.
72. Martyushev L. Entropy and Entropy Production: Old Misconceptions and New Breakthroughs. *Entropy* (2013) 15:1152–70. doi:10.3390/e15041152
73. Wissner-Gross AD, Freer CE. Causal Entropic Forces. *Phys Rev Lett* (2013) 110(168702):168702–5. doi:10.1103/PhysRevLett.110.168702
74. Lawless WF. Towards an Epistemology of Interdependence Among the Orthogonal Roles in Human-Machine Teams. *Found Sci* (2019) 26: 129–42. doi:10.1007/s10699-019-09632-5
75. Baras JS. *Panel. New Inspirations for Intelligent Autonomy*. Washington, DC: ONR Science of Autonomy Annual Meeting (2020).
76. Nass C, Fogg BJ, Moon Y. Can Computers Be Teammates? *Int J Human-Computer Stud* (1996) 45(6):669–78. doi:10.1006/ijhc.1996.0073
77. Nass C, Moon Y, Fogg BJ, Reeves B, Dryer DC. Can Computer Personalities Be Human Personalities? *Int J Human-Computer Stud* (1995) 43(2):223–39. doi:10.1006/ijhc.1995.1042
78. Nass C, Steuer J, Tauber ER. Computers Are Social Actors. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. New York City: ACM (1994). p. 72–8. doi:10.1145/191666.191703
79. forthcoming Blasch E, Schuck T, Gagne OB. Trusted Entropy-Based Information Maneuverability for AI Information Systems Engineering. In: WF Lawless, J Linas, DA Sofge, R Mittu, editors. *Engineering Artificially Intelligent Systems*. Springer's Lecture Notes in Computer Science (2021).
80. Moskowitz IS, Brown NL, Goldstein Z. A Fractional Brownian Motion Approach to Psychological and Team Diffusion Problems. In: WF Lawless, R Mittu, DA Sofge, T Shortell, TA McDermott, editors. *Systems Engineering and Artificial Intelligence*. Springer (2021). Chapter 11. doi:10.1007/978-3-030-77283-3_11
81. Herrera S, Chin K. *Amazon, Berkshire Hathaway, JPMorgan End Health-Care Venture Haven*. New York City: Wall Street Journal (2021). Accessed 1/5/2021 from https://www.wsj.com/articles/amazon-berkshire-hathaway-jpmorgan-end-health-care-venture-haven-11609784367.
82. McDowell JJ. On the Theoretical and Empirical Status of the Matching Law and Matching Theory. *Psychol Bull* (2013) 139(5):1000–28. doi:10.1037/a0029924
83. Woide M, Stiegemeier D, Pfattheicher S, Baumann M. Measuring Driver-Vehicle Cooperation. *Transportation Res F: Traffic Psychol Behav* (2021) 83: 773 424–39. doi:10.1016/j.trf.2021.11.003
84. Lawless WF, Sofge D. Interdependence: A Mathematical Approach to the Autonomy of Human-Machine Systems. In: *Proceedings for Applied Human Factors and Ergonomics 2022*. Springer (2022).
85. Prigogine I. Ilya Prigogine. Facts. The Nobel Prize in Chemistry 1977 (1977). Accessed 2/10/2022 from https://www.nobelprize.org/prizes/chemistry/1977/prigogine/facts/.
86. Baker G. *Michael Hayden Says U.S. Is Easy Prey for Hackers. Former CIA and NSA Chief Says 'shame on Us' for Not Protecting Critical Information Better*. New York City: Wall Street Journal's editor in chief (2015). from http://www.wsj.com/articles/michael-hayden-says-u-s-is-easy-prey-for-hackers-1434924058. Accessed 6/22/2015
87. Ratcliffe J. *U.S. Director of National Intelligence: "China Is National Security Threat No. 1"*. New York City: Wall Street Journal (2020). Accessed 4/4/2022 from https://www.wsj.com/articles/china-is-national-security-threat-no-1-11607019599.
88. Mattioli D. *Amazon Scooped up Data from its Own Sellers to Launch Competing Products*. New York City: Wall Street Journal (2020). Accessed 10/18/2021 from https://www.wsj.com/articles/amazon-scooped-up-data-from-its-own-sellers-to-launch-competing-products-11587650015?mod=article_inline.
89. Slovic P, Flynn JH, Layman M. Perceived Risk, Trust, and the Politics of Nuclear Waste. *Science* (1991) 254:1603–7. doi:10.1126/science.254.5038.1603
90. Lawless WF, Bergman M, Feltovich N. Consensus-Seeking Versus Truth-Seeking. *ASCE Practice Periodical Hazard Toxic Radioact Waste Manage* (2005) 9(1):59-70.
91. Lawless WF, Akiyoshi M, Angjellari-Dajcic F, Whitton J. Public Consent for the Geologic Disposal of Highly Radioactive Wastes and Spent Nuclear Fuel. *Int J Environ Stud* (2014) 71(1):41-62.
92. Sirower ML. *The Synergy Trap*. New York: Free Press (2007).
93. Leswing K. *How Apple Does MandA*. Englewood Cliffs, NJ: CNBC (2021). Accessed 4/4/2022 from https://www.cnbc.com/2021/05/01/how-apple-does-ma-small-and-quiet-with-no-bankers.html.
94. Smith J, Ziobro P. *UPS to Sell Freight Trucking Business to TFI*. New York City: Wall Street Journal (2021). Accessed 1/26/2021 from https://www.wsj.com/articles/ups-to-sell-freight-trucking-business-to-tfi-for-800-million-11611592797.
95. Naughton N. *Fiat Chrysler, PSA Aim to Complete Trans-Atlantic Merger in Mid-january*. New York City: Wall Street Journal (2021). Accessed 1/5/2021 from https://www.wsj.com/articles/fiat-chrysler-psa-shareholders-to-vote-on-trans-atlantic-tie-up-11609764732.
96. Tweh B. *Facebook Says Apple's Privacy Changes Hurt Digital Ad Measurement*. New York City: Wall Street Journal (2021). Accessed 2/13/2022 from https://www.wsj.com/articles/facebook-says-apples-privacy-changes-hurt-digital-ad-measurement-11632341624.
97. Webb Q, Wei L. *Chinese Ride-Hailing Giant Did I Could Get State Investment*. New York City: Wall Street Journal (2021). Accessed 9/4/2021 from https://www.wsj.com/articles/city-of-beijing-leads-plan-for-state-investment-in-didi-the-embattled-ride-hailing-giant-1163068135.
98. Lawless WF, Mittu R, Moskowitz IS, Sofge D, Russell S. Cyber-(in)security, Revisited. In: P Dasgupta, JB Collins, R Mittu, editors. *Adversary Aware Learning Techniques and Trends in Cyber Security*. Switzerland: Springer Nature (2020).
99. Tzu S. In: RD Sawyer, editor. *The Art of War*. New York: Basic Books (1994). First published in the 5th Century, BC.

100. Kryjevskaia M, Heron PRL, Heckler AF. Intuitive or Rational? Students and Experts Need to Be Both. *Phys Today* (2022) 74(8):28–34. doi:10.1063/PT.3.4813
101. Haggin P, Hagey K, Schechner S. *Apple's Privacy Change Will Hit Facebook's Core Ad Business*. New York City: Wall Street Journal (2021). Accessed 1/30/2021 https://www.wsj.com/articles/apples-privacy-change-will-hit-facebooks-core-ad-business-heres-how-11611938750.
102. Lawless WF. The Physics of Teams: Interdependence, Measurable Entropy, and Computational Emotion. *Front Phys* (2017) 5:30. doi:10.3389/fphy.2017.00030
103. Platt HD, Fuller A. Theory of Short Selling. *J Asset Manag* (2002) 5(1). Accessed 2/18/2022 from. doi:10.2139/ssrn.301321
104. Farooq F. *10 Most Successful Short Sellers of All Time*. Sunnyvale, CA: Yahoo.com (2021). Accessed 2/12/2022 from https://www.yahoo.com/video/10-most-successful-short-sellers-141904957.html.
105. Kennon J. The Basics of Shorting Stock, the Balance (2021). Accessed 2/18,2022 from https://www.thebalance.com/the-basics-of-shorting-stock-356327.
106. Bobrowsky M. *Facebook Feels $10 Billion Sting from Apple's Privacy Push*. New York City: Wall Street Journal (2022). Accessed 2/6/2022 from https://www.wsj.com/articles/facebook-feels-10-billion-sting-from-apples-privacy-push-11643898139.
107. Lawless WF, Sofge DA. The Intersection of Robust Intelligence and Trust. In: WFR LawlessMittu, D Sofge, S Russell, editors. *Autonomy and Artificial Intelligence: A Threat or Savior?* New York: Springer (2017). p. 255–70.
108. Axelrod R. *The Evolution of Cooperation*. New York: Basic (1984).
109. Perc M, Jordan JJ, Rand DG, Wang Z, Boccaletti S, Szolnoki A. Statistical Physics of Human Cooperation. *Phys Rep* (2017) 687:1–51. doi:10.1016/j.physrep.2017.05.004
110. Friedman M. *Capitalism and freedom*. Chicago: New York City: University of Chicago Press (1982).
111. Britannica. The Editors of Encyclopaedia. In: *Great Leap Forward*. London, UK: Encyclopedia Britannica (2020). https://www.britannica.com/event/Great-Leap-Forward (Accessed February 18, 2022).
112. Llinas J. Motivations for and Initiatives on AI Engineering. In: WF Lawless, R Mittu, DA Sofge, T Shortell, TA McDermott, editors. *Systems Engineering and Artificial Intelligence*. Springer (2021). doi:10.1007/978-3-030-89385-9_1
113. Somerville H. *Uber Sells Self-Driving-Car Unit to Autonomous-Driving Startup*. Wall Street Journal (2020). Accessed 12/8/2020 from https://www.wsj.com/articles/uber-sells-self-driving-car-unit-to-autonomous-driving-startup-11607380167.
114. Shelbourne M. *HII Purchases Autonomy Company to Bolster Unmanned Surface Business*. Annapolis, MD: USNI News (2021). Accessed 1/5/2021 from https://news.usni.org/2021/01/04/hii-purchases-autonomy-company-to-bolster-unmanned-surface-business.

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

frontiers | Frontiers in Physics

ORIGINAL RESEARCH
published: 14 July 2022
doi: 10.3389/fphy.2022.944064

JSwarm: A Jingulu-Inspired Human-AI-Teaming Language for Context-Aware Swarm Guidance

Hussein A. Abbass[1]*, *Eleni Petraki*[2] *and Robert Hunjet*[3]

[1]School of Engineering and IT, University of New South Wales, Canberra, NSW, Australia, [2]Faculty of Education, University of Canberra, Canberra, ACT, Australia, [3]Defence Science and Technology Group, Canberra, NSW, Australia

Bi-directional communication between humans and swarm systems begs for efficient languages to communicate information between the humans and the Artificial Intelligence (AI)-enabled agents in a manner that is most appropriate for the context. We discuss the criteria for effective teaming and functional bi-directional communication between humans and AI, and the design choices required to create effective languages. We then present a human-AI-teaming communication language inspired by the Australian Aboriginal language of Jingulu, which we call JSwarm. We present the motivation and structure of the language. An example is used to demonstrate how the language operates for a shepherding swarm guidance task.

Keywords: human-AI teaming, human-swarm teaming, teaming languages, jingulu, human-swarm languages

OPEN ACCESS

Edited by:
William Frere Lawless,
Paine College, United States

Reviewed by:
Jaelle Scheuerman,
United States Naval Research Laboratory, United States
Michael Mylrea,
University of Miami, United States

****Correspondence:***
Hussein A. Abbass
hussein.abbass@gmail.com

Specialty section:
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

Received: *14 May 2022*
Accepted: *16 June 2022*
Published: *14 July 2022*

Citation:
Abbass HA, Petraki E and Hunjet R (2022) JSwarm: A Jingulu-Inspired Human-AI-Teaming Language for Context-Aware Swarm Guidance. Front. Phys. 10:944064. doi: 10.3389/fphy.2022.944064

1 INTRODUCTION

Natural languages are very rich and complex. Human languages have been in place for around 10,000 years and have served humans effectively and efficiently. Even within a single language, there could be many different varieties or codes, used by specific speaker groups, or maintained for particular contexts. English for specific purposes is a discipline of language teaching that focuses on the teaching of English for various professional or occupational contexts, such as English for nursing or English for legal purposes. These natural languages could be too inefficient for an artificial intelligence (AI) enabled agent designed for a particular task or use, due to the languages being highly-complex, thus, creating a space of ambiguity or unnecessary complexity. There is a significant amount of research in computational linguistics that could help and guide the design of human-friendly languages for distributed artificial intelligence (AI) systems to enable humans and AI-enabled agents to work together in a teaming arrangement. Each relationship-type among a group of agents shape the subset of the language required to allow agents to negotiate meanings and concepts associated with the particular domain where the relationship-type belongs. Moreover, understanding the principles for computational efficiency in natural languages has been the subject of inquiry by computational linguists. By identifying the minimum set of rules (ie grammar) governing a language, linguists discover the DNA-equivalent of, and morphogenesis for, human languages. Be it through learning or direct encoding of this minimum set, the concept of computational efficiency can contribute to an assurance process for a proper coverage of the semantic space required for, and requirements to reduce impermissible sentences during, an interaction.

Unsurprisingly, culture shapes language and vice-versa [1, 2]. This has led to the diversity of human languages available today, which vary in grammar, vocabulary and complexity of meaning. Similar to human systems, in artificial systems language also reflects cultural and social networks. To situate the contribution of this paper in the current literature, **Figure 1** depicts a high-level

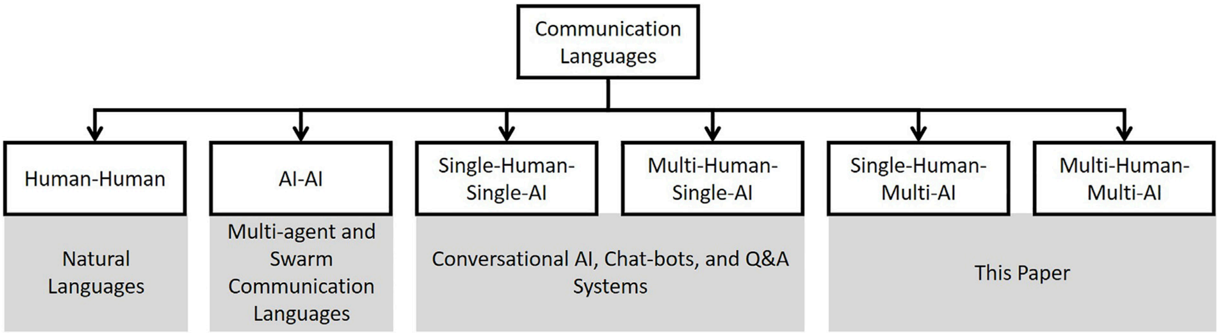


FIGURE 1 | Classification of literature on communication languages and contribution of this paper.

classification of the research done on human-AI languages. The figure presents six dimensions or lenses that one can see the literature. These dimensions are formed from the perspective of who is interacting with whom. We will discuss below three research directions with particular relevance to the current work and focus particularly on human-designed languages for different forms of human-machine interaction. The discussion together with the figure attempts to compress the wide variety of contributions made in this space for centuries.

Human-Human languages have a very long history, with studies that could be traced back to the ancient Greeks. AI-AI languages have its roots in recent literature, and have been a fruitful research area, whereby the languages could be emerging or designed. The literature on the languages required at the interface between humans and machines, or more specifically in this paper human-AI systems, has witnessed different categories of methods to approach the topic. The current literature can generically be categorised into three research directions, each with their own cultural traits. The first research direction aims at designing "computer programming" languages (see for example [3]), aiming at affording a human with a language to program a group of robots. This class of languages allow a human to encode domain knowledge in algorithmic form for robots to function and can be seen as human's means to communicate to the machine. The second research direction focuses on languages required for communication between an AI and another, or between an AI and a human. In this branch, work on designing languages to allow communication among a group of artificial agents has been primarily dominated by the multi-agent literature [4] and more recently the swarm systems literature [5]. When a human interacts with an AI, conversational AI [6, 7], chatbots and Questions and Answer (Q&A) systems [8] dominate the recent literature using data-driven approaches and neural-learning [9]. The third research direction shifts focus away from human-design of the communication language to the emergence of communication and language in a group of agents. A reasonably large body of evolutionary and developmental robotics literature [10] has dedicated significant efforts into this research direction. These three research directions could carry some relevance across all dimensions in **Figure 1**, but clearly the amount of relevance is not uniform.

The focal point of this paper is human-AI teaming, especially within the context of distributed AI systems capable of synchronising actions to generate an outcome, or what we call AI-enabled swarm systems. In particular, our aim is to design a computationally efficient human-friendly language for human-AI teaming that is also appropriate for human-swarm interaction and swarm-guidance. The design is inspired by the Jingulu language [11], an Aboriginal language spoken in parts of Australia and demonstrated on a swarm guidance approach known as shepherding [12]. Briefly, the shepherding problem is inspired by sheepdogs mustering sheep. The shepherding (teaming) system comprises of a swarm (analogous to sheep) to be guided, an actuator agent (analogous to a sheepdog in biological herd mustering) with the capacity to influence the swarm, an AI-shepherd (analogous to sheepdog cognition) with the capacity to autonomously guide the actuator agent (sheepdog body) to achieve a mission, and a human-team (analogous to farmers) with the intent to move the swarm. To achieve this goal, the human team interacts with the AI-shepherd and is required to monitor, understand, and command it when necessary, as well as take corrective actions when the AI-shepherd deviates from the human team's intent. We assume that the AI-shepherd is more clever than a sheepdog and is performing the role of the human-shepherd in a biological mustering setting. Biological swarms such as sheep herds have been shown to be appropriately modelled by attraction and repulsion rules amongst the swarm members. The special characteristic of the Jingulu language is that the language has only three main verbs: do, go and come. Such a structure is most efficient for communication in attraction-repulsion equations-based distributed AI-enabled swarm systems as this paper shows.

An introduction to Jingulu and swarm shepherding is presented in **Section 2**. We then discuss the requirements for computational efficiency when designing human-AI teaming languages and propose an architecture in **Section 4**. A computationally-efficient human-friendly language for human-AI swarm teams is then presented in **Section 5**, followed by a discussion on the assurance of human-AI teaming language in **Section 6**. Conclusions are drawn and future work is discussed in **Section 7**.

2 BACKGROUND

2.1 Aboriginal Languages

Aboriginal people are the traditional owners of the Australian land. Different tribes occupy different parts of the island. Australian Aboriginal languages display unique syntactic properties, and one property in particular is called the non-configurationality free word order [13]. We offer a simple introduction to some basic linguistic features to explain this property.

Syntactic relations describe the minimal components of a simple sentence that usually consists of subject-verb-object, which we will abbreviate as SVO. The presence of SVO is a universal property in the organisation of sentences despite the existence of variation in the order. Some languages follow SVO, others VSO, and others SOV. SVO are syntactic positions that are occupied by noun phrases (NPs), verb phrases (VPs), and NPs, respectively. English is an SVO language.

[My daughter] [ate] [her ice cream]
[Subject] [Verb] [Object]
[Pronoun Noun] [Verb] [Pronoun noun]

Non–configurationality describes a principle that applies to languages whose sentence structure imposes fewer restrictions in the order of syntactic relations. Greek is a non–configurational language that allows SVO swapping in a sentence.

[E kore mou] [efage] [To pagoto ths]
[The daughter my] [ate] [The ice cream her]
[Article, noun, pronoun] [Verb (tense/person)] [Article noun pronoun]
[Subject] [Verb] [Object]
[efage] [E kore mou] [To pagoto ths]
[V] [S] [O]
[To pagoto ths] [E kore mou] [efage]
[O] [S] [V]

Most configurational and non-configurational languages impose restrictions on the constituent order, that is the order of words that forms the subject or the verb phrase or the object phrase. For example, in English, the order of the subject [my daughter] needs to follow [pronoun + noun] order and not vice versa. This group of words always moves together as a phrase (constituent) and cannot be separated.

However, many Aboriginal languages are not only non–configurational but also display free word order within the constituent phrases. This means that a noun phrase, that is a group of words that might fill the position of the subject, for example, or the object, can be split in the sentence. Such languages express meaning, using inflectional morphology such as prefixes and suffixes that might indicate, person, gender, tense, aspect, which are limited in the English language.

This flexibility has also been found in Jiwarli and Walpiri, two heavily studied Aboriginal languages. Jiwarli language (no longer spoken), used to be part of the Pilbara region in Western Australia. Walpiri, is an Aboriginal language spoken in the Northern Territory:

Example from Walpiri [14]:
[Kurdu-jarra-rlu] [ka-pala] [maliki wajili-pi-nyi] [wita-jarra-rl]
[child-(two)-] [(Present)] [Dog chase] [Small]

Two small children are chasing the dog. OR *Two children are chasing the dog and they are small.*

2.2 The Jingulu Language

The Jingili people live in the western Barkly Tablelands of the Northern Territory in the town of Elliott. We consulted the grammar of Jingulu (the language of the Jingili's people) in Pensalfini's dissertation and subsequent book written on the grammar of the Jingulu language [11].

Similar to many Aboriginal languages, Jingulu displays free constituent order.

1. *Uliyija-nga ngllnja-ju karalu. (SVO)*
sun-ERG.f burn-do ground
The Sun is burning the ground.
2. *Uliyijanga karalu ngunjaju. (SOV)*
3. *Ngu njaju uliyijnnga karalu. (VSO)*

Jingulu shows also free word order within the Noun Phrases as seen below.

The Noun phrase in English 'that stick' would never be separated, however in Jingulu, they seem to can be separated.

[Ngunll] [maja-mi] [ngnrru] [darrangku.]
[that] [get] [(Me)] [stick]

Get me that stick.

Jingulu has many interesting features that we will not cover in this paper. Instead, we will focus on the most prominent feature of Jingulu, that it is a language with only three primary verbs: do, go, and come. We will refer to them as light verbs. We are not aware of any other Aboriginal languages that display this structure and as such this is perhaps unique for Jingulu.

We argue that this feature makes Jingulu an ideal natural language for representing spatial movements between entities and the exchange of communication messages, including commands, among the agents. To explain this further, we borrow three examples from [15]. The use of FOC in the following text indicates 'contrastive focus', which is a linguistic marker to represent where focus is placed in a sentence. This is not a feature of Jingulu per se, it is part of the linguistic characterisation linguists use to mark attention in sentences.

Example 1 [15][p.228]

Kirlikirlika darra-ardi jimi-rna urrbuja-ni.
galah eat-go that(n)-FOC galah_grass-FOC
"Galahs eat this grass."

Example 2 [15][p.229]

Aja(-rni) ngaba-nya-jiyimi nginirniki(-rni)?
what(-FOC) have-2sg-come this(n)(-FOC)
"What's this you're bringing."

Example 3 [15][p.229]

Ngaja-mana-ju.
see-3MsglO-do
"He is looking at us."

In the previous examples, the use of do, go and come represents stationarity at current location, and departure away from and arrival to current location, respectively. In the first example, eating the grass indicates that the subject needs to move away from the subject's current location by "going" to the grass. In the second example, questioning what a person brings depicts a picture of something coming from the subject's current location to our location. In the third example, "looking at us" does not require any movements, the act could be performed without a change of location. Despite this simplicity, what is truly powerful in this representation is the acknowledgement of the abstract concept of a space, that does not have to be physical in nature. For example, the space could be a space of ideas where an idea might come to a person or a person can go to an idea. The explicit spatial representation is so powerful in the structure of the language.

One distinctive aspect of the Jingulu language is the structure of the verb. Take for example "see-do," which is the third example above. Prior researchers to Pensalfini explained "do" as an inflectional element representing the final tense-aspect marker. Pensalfini, however, defines it as the "verbal head"; the core syntactic verb. "See" in the sentence is the normalised verb object; a category-less element and root of the verb. The root does not bear any syntactic information, only semantic one. The three semantically bleached light verbs, however, play syntatic and semantic roles. Jingulu has the smallest inventory of inflecting verbs, that is 3, that have complex predicates amongst the northern Australian languages [11].

The verbal structure can, therefore, be described as: *Root (See) + Light-verb (Do).*

Pensalfini saw the final element as the "true syntactic verb," which encodes inflectional properties such as tense, mood, and aspect, as well as "distinctly verbal notions such as associated motion. These elements fall into three broad classes, corresponding to the English verbs "come" (3.3), "go" (3.4) and "do/be" (3.5)."

While the language may appear to be complex or primitive, depending on perspectives, from a human-human communication perspective, the above discussion demonstrates very powerful linguistic features in the Jingulu language that we will use for human-AI teaming in a human-swarm context. In particular, the above structure sees the light verb as a semantic carrier; that is, it is the vehicle that carries the meaning created by the root. This vehicle offers spatio-temporal meaning, while the root offers context. These features will be explained later on, in this manuscript.

2.3 Shepherding

A swarm is a group of decentralised agents capable of displaying synchronised behaviors despite the simple logic they adopt to make decisions in an environment. Members in a swarm do not necessarily synchronise their behavior intentionally. However, for an observer, the repeated patterns of the coordinated actions they display is synchronised in the behavior space. It is this synchronisation that creates observable patterns in the dynamics that allow members in the swarm to either appear in certain formations or act to generate a larger impact than the impact that could have been generated by any of the individuals in isolation.

The Boids (Bird-oids or Bird-Like-Behavior) model by Reynolds [16] is probably the most common demonstration of swarming in the academic literature. The model relies on three simple rules, whereby each agent is (rule 1) attracted to and (rule 2) aligns its direction with its neighbor, and (rule 3) repulses away from very nearby agents. Following these three simple attraction-repulsion equations, the swarm displays complex collective dynamics.

While the collective boids can swarm, real-world use of swarming calls for methods to guide the swarm [17–22]. There are several ways to guide the swarm from the inside by having an insider influence [23] with a particular intent and knowledge of goals. Another approach is to leave the swarm untouched and to guide them externally with a different agent that is specialized in swarm guidance. This approach mimics the behavior of sheepdogs, where a single sheepdog (the guiding agent) can guide a large number of sheep (the swarm). Indeed, the sheep are modelled with two of the boids rules (attraction to neighbors and repulsion from very nearby agents), with the addition of a third rule to repulse away from sheepdogs. A number of similar models exist to implement this swarm-guidance approach [24–26].

Multiple sheepdogs could be used to herd the sheep, and they could themselves act as a swarm leading to a setup of swarm-on-swarm interaction. The implicit assumption in these models is that the number of sheepdogs is far less than the number of sheep; otherwise the problem could become uninteresting and even trivial. In the simplest single-sheepdog model, the sheepdog switches between two behaviors: collecting a sheep when a sheep is outside the cluster zone of the herd and driving the herd when all sheep are collected within the cluster zone of the herd. When more sheepdogs are used, rules to spread them into formations and/or coordinate their actions are introduced [24, 25, 27–31].

One aim of different shepherding models is to increase the controllability of the sheepdog as demonstrated by the number of sheep it can collect. The two most recent models in the literature are the one by [26] and an improvement on it by [32]. Both models use attraction and repulsion forces and smooth movements by adjusting the velocity vector in the previous timestep with new intent. The latter model improved the former with a number of adjustments. One is to skill the artificial sheepdog to use a circular path to reach a collection or a driving point and to avoid dispersing sheep on the way. Sheep have more realistic behaviors when they do not detect sheepdogs.

In nature, the sheep do not continue to group in the absence of a sheepdog. Instead, they continue to do whatever their natural instinct motivates them to do (eating, sleeping, etc); thus, the latter model does not introduce a bias of continuous attraction to neighbors in the absence of a sheepdog effect. The latter model also defined a sheep's neighborhood based on different sensing ranges. This is more realistic than the former model which fixes the number of closest sheep; ensuring every sheep always having a fixed number of neighbors regardless of where these neighbors are. Other changes were introduced, which overall improved the success rate of the guidance provided by the sheepdog. In the remainder of this paper, we use El-Fiqi et al. [32]'s model.

A fundamental principle in modelling the shepherding problem is the cognitive asymmetry of agents, where sheep are the simplest agents cognitively with simple survival goals. Sheepdogs have more complex cognition than sheep as they need to be able to autonomously execute a farmer's intent. The cognition of farmers/shepherds, however, is more complex than sheepdogs due to their role which requires them to have a higher intent with abilities to understand the capabilities of sheepdogs and commanding them to perform certain tasks. In our shepherding-inspired system, we separate between the sheepdog as an actuator and the cognition of a sheepdog. We call the latter the AI, representing the cognitive abilities of a sheepdog to interact and execute human intent.

We present the basic and abstract shepherding model introduced by [32]. While this model has evolved in our own research to more complex versions, it suffices to explain the basic ideas in the remainder of the paper. We will use our generalised notations for shepherding to be consistent with the notations used in our group's publications.

The set of sheep are denoted as $\Pi = \{\pi_1, \ldots, \pi_i, \ldots, \pi_N\}$, where N is the total number of sheep, while the sheepdog agents are denoted as $B = \{\beta_1, \ldots, \beta_j, \ldots, \beta_M\}$, where M is the number of sheepdogs. The agents have a set of behaviours available to choose from; the superset of behaviours is denoted as $\Sigma = \{\sigma_1, \ldots, \sigma_K\}$, where K is the number of behaviours available in the system. Agents occupy a bounded squared environment of length L. Each sheep can sense another sheep in the sensing range of $R_{\pi\pi}$ and can sense a dog in the sensing range of $R_{\pi\beta}$. The global centre of mass (GCM) for sheep is denoted by $\Gamma^t_{\pi_i}$. The flock of sheep in this environment has two states; they are either collected or not. Sheep are collected when all sheep are located within a radius $f(N)$ of their GCM. The radius is calculated as:

$$f(N) = R_{\pi\pi} N^{\frac{2}{3}} \tag{1}$$

If all sheep are within distance of $f(N)$ of their GCM, then they are collected and are ready to be driven as a herd to the goal. The sheepdog moves to the driving point which is located behind the herd on the ray from the goal to the GCM. If the sheep are not collected, the sheepdog needs to identify the furthest sheep to GCM and move to a collection point to collect that sheep by influencing it to move towards the GCM of the herd. By alternating between these two behaviours, in a simple obstacle-free environment, the sheepdog should be able to collect the sheep successfully.

All actions in the basic and abstract shepherding model are represented using velocity vectors; however, these vectors are called force vectors due to the fact that if the agents are actual vehicles, the desired velocities need to be transformed into forces that cause agents to move. For consistency with the shepherding literature, we will call them (proxies of) force vectors. Below is a list of all force vectors used in this basic model.

- Sheep-Sheepdog Repulsive Force $F^t_{\pi_i\beta_j}$: repulsion of π_i agent away from β_j agents at time t.
- Sheep-Sheep Repulsive Force $F^t_{\pi_i\pi_{-i}}$: repulsion of π_i agent away from other $\pi_{k\neq i}$ agent at time t.
- Sheep Attraction to Local Herd $F^t_{\pi_i\Lambda^t_{\pi_i}}$: attraction to Local Centre of Mass for the neighbours of a π_i agent at time t.
- Sheep Local Random Movements $F^t_{e\pi_i}$: jittering movements by the π_i agent at time t.
- Sheep Total Force Vector $F^t_{\pi_i}$: movement vector of the π_i agent at time t.
- Sheepdog Attraction to Driving Point $F^t_{\beta_j d}$: driving vector of the β_j agent at time t.
- Sheepdog Attraction to Collecting Point $F^t_{\beta_j c}$: collection vector of the β_j agent at time t.
- Sheepdog Local Random Movements $F^t_{e\beta_j}$: jittering movements by the β_j agent at time t.
- Sheepdog Total Force Vector $F^t_{\beta_j}$: movement vector of the β_j agent at time t.

The total forces acting on the sheep and sheepdog, respectively, are formed by a weighted sum of the individual forces. The weights are explained in **Table 1**. The equations for total forces are included below.

$$F^t_{\pi_i} = W\pi\Lambda \times F^t_{\pi_i\Lambda^t_{\pi_i}} + W_{\pi\pi} \times F^t_{\pi_i\pi_{-i}} + W_{\pi\beta} \times F^t_{\pi_i\beta_j} + W_{e\pi_i} \times F^t_{e\pi_i} + W_{\pi_v} \times F^{t-1}_{\pi_i}$$

$$F^t_{\beta_j} = W_{\beta_j c} \times F^t_{\beta_j c} + W_{\beta_j d} \times F^t_{\beta_j d} + W_{e\beta_j} \times F^t_{e\beta_j}$$

3 HUMAN-SWARM TEAMING

Human-human teaming, albeit still a challenging topic, seems natural to the extent that most humans would only focus on the external/behavioural traits required to generate effective teams. The compatibility among humans has hardly been questioned; all humans have a brain with similar structure and while their mental models of the world could be different–thus, requiring alignment for effective teams–the internal physiological machines are similar in the manner they operate. When we discuss teaming among different species, especially when one species is biological (humans, dogs, sheep, etc) and the other is in-silico (computers controlling UGVs, UAVs, etc), some of the factors taken for granted in human-human teaming need to be scrutinised and looked at with a great level of depth.

A particular focus in this paper is the alignment of representation language on all levels of operations inside a machine. Interestingly, in a human, neurons form the nerves

TABLE 1 | The Library of behaviors observed or performed by the sheepdog.

Desires	Weight vector
Sheep desire to cluster	$W\pi\Lambda$
Sheep desire to avoid collision	$W_{\pi\pi}$
Sheep desire to avoid sheepdog	$W_{\pi\beta}$
Sheep desire to stay where they are	$W_{e\pi_i}$
Sheep desire to maintain velocity at $t - 1$	W_{π_υ}
Sheepdog desire to collect astray sheep	$W_{\beta_j c}$
Sheepdog desire to drive collected sheep	$W_{\beta_j d}$
Sheepdog desire to desire to avoid sheep on the way to collection or driving points	$W\beta_\pi$
Sheepdog desire to rest	$W_{e\beta_j}$

that sense, the nerves that control the joints and actuators, and the basic mechanical unit for thinking. While these neurons perform different functions, they work with similar principles. It seems within humans, the representational unit in a nervous system, the neuron, is the unified and smallest representational unit for sensing, deciding and acting. The representational unit of concern in this paper sits at a higher level than the physiological neuron; it sits on the level of thinking and decision making, where it takes the form we called 'force vectors'.

We will differentiate between three representations as showing in **Figure 2**. The first representation, we call *control-representation*, is one where the action-production logic for an agent is ready for hardware/body/form/shape execution. Control systems at their lower level are executed in a CPU, GPU, FPGA or a neural network chip where the output gets transmitted to actuators. They mostly come in calculus forms. The mathematics of the control system, even for simple ones, are not necessarily on the level to be explainable to a general user. What is important on this level is computational efficiency from sensing to execution to ensure that the agent acts in a timely manner, and the assurance of performance to ensure that the agent acts correctly.

The second representation, we call *reasoner-representation*, is where the agent needs to make inferencing to connect its high level goals and low-level control, create an appropriate set of actions, and select the right courses of actions to achieve its goals or purpose. On this level, the agent performs functions that govern the overall logic that connects its mission to its actions.

The third representation, we call *communication-representation*, is where the agent needs to transform its internal representation for reasoning and action production to external statements to be communicated to other agents, including humans, in its environment. This representation is key for agents to exchange knowledge, negotiate meaning, and be transparent to gain trust of others in their eco-system.

Take for example the artificial *sheepdog*, which we will assume to be a ground autonomous vehicle. Its objective is to collect all dispersed sheep outside the paddock into the paddock area. It senses the environment through its onboard sensors and/or through communication messages received from the larger system it is operating within. Through sensing, it needs the following information to be able to complete its mission: location of sheep, location of goal (paddock), location and size of obstacles in the environment, and location of other *dogs* in the environment. The *dog* may receive all information about all entities in an accurate and precise form as it is the case in a perfect simulated world, or it may receive incomplete or ambiguous information with noise as is the case in a realistic environment.

On a cognitive-level, the *dog* needs to decide which sheep it needs to direct its attention to, how it will get to them, how it will influence them to get them to move, where to take them, and what to do next until the overall mission is complete. The timescale on which the cognitive level works on is moderate. We will quantify this later in the paper. Meanwhile, whatever the cognitive-level decides, it needs to be transformed into movements and actions.

The control representation takes the relevant subset of the sensed information and the immediate waypoints the *dog* needs to move to and generates control vectors for execution. It needs to transform the required positions decided on by the cognitive level into a series of acceleration and orientation information steps that get transmitted to its actuators (for example, joints and/or wheels). The time scale this level operates on is shorter than what the cognitive-level operates on.

Additionally, the dog may need to communicate with its (human or AI) handler, explaining what it is doing and/or obtaining instructions. The timescale in which the communication operates could vary, and the system needs to be able to adjust this time scale based on the cognitive agents it is teaming with. For example, the communication could occur more frequently if the dog is interacting with another AI than if it was interacting with a human.

Each representational language defines what is representable (capacity), and thus, what is achievable (affordance), using such representation. For example, if the control system is linear, the advantages include: being easy to analyse and being easy to prove/disprove its stability. We equally understand its disadvantages for example in requiring a complete system identification exercise prior to the design of the controller and its inability to adapt when context changes. When two or more representational languages interface with one another, their differences generate challenges and vulnerabilities. We will illustrate this point with the three representational languages discussed for the artificial sheepdog.

The reasoner-representation relies on propositional calculus. A set of propositions can be transformed to an equivalent binary integer programming problem. If the communication-representation is in unrestricted natural language, clearly many sentences exchanged at the interface level will not be interpreted properly by the reasoner. In the same manner, when the control-representation is a stochastic non-linear system, some actions produced by the controller may not be interpretable by the reasoner. This requirement for equivalence at the interface between the three representations impose constraints on which representation to select. Meanwhile, it ensures that actions are interpretable at all levels; thus, what the agent does at the control or cognitive levels can be explained to other agents in the environment, and requests from other agents in the environment can be executed as they are by the agent. Moreover, due to the equivalence in the capacity of the representation language at each level, mappings between different representations are direct mappings; thus, they are efficient.

Last, but not least, once a system is assured on one level, due to the equivalence of the representation language on all three levels, the system can be easily assured on another level. For example, take the case where a system is assured that it will not request or accept an unethical request. Considering that what the agent executes at the control level is equivalent to the request it receives on the communication level, if we guarantee that the mappings between levels are correct and complete, we can induce that the control level will not produce an unethical behaviour.

Before departing from the discussion above, the representation language plays a dual role in a system. On the one hand, it constraints the system's capacity to perform. As we explained above, a linear system can only be guaranteed to perform well under sever assumptions of linearity. On the other hand, the representation language equally constrains affordance. This may not be so intuitive because affordance is the opportunities that the environment offers an individual agent to do. An agent is unable to tap into opportunities where the representation language constrains its action set or the quality of actions. Due to the trade-offs discussed above, the decision on which representation language to use needs to be risk-based to analyse the vulnerability and remedies of the design choices made during that decision.

In the next section, we will present the requirements for the human-machine teaming problem, including a formal representation of the problem, before presenting the Jingulu swarm language in the following section.

4 HUMAN-AI TEAMING LANGUAGE REQUIREMENTS

In human-AI teaming, different categories of information assist in the efficiency of the teaming arrangements and the ability of the system to adapt [33]. However, these capabilities will not materialise unless there is a language that allows this information to flow and to be understood by the humans and the swarm. In this section, we focus primarily on the requirements for this language.

4.1 Human-AI Teaming Language Requirements

The discussion and example presented in the previous section illustrate the scope of each of the three levels. From this scope, we will draw and justify the requirements for the Human-Swarm Language as follows:

1. Contextual Relevance: The representation languages need to be appropriate for the particular mission the agent is assigned to do. The representation needs to represent, and when necessary, enable the explanation of, the sensorial information in the context within which the agent operates, the logic used for action production, the actions produced at a particular level, the intent of the agent, and the measures that the agent uses to assess its performance. Put simply, the representation serves the context; everything the context requires should be representable by the chosen language at each level.
2. Computational Efficiency: The reasoner-representation works in the middle between the control-representation and the communication-representation. The three representations need to allow the clocks of the three levels to serve each other's frequencies. For example, if the reasoner needs to produce a plan on 0.2 Hz (ie a plan each 5 s), and the controller is running on 1 Hz, while the communication system needs to explain the decisions made in the system on 0.05 Hz, the representation on each level needs to be computationally efficient to allow each level to work on these timescales without latencies. In other words, if it takes 25 s to produce a sentence on the communication layer, the agent will not be able to catch up with the speed of action-production. This latter case will force the agent to be selective in what aspects of its actions it needs to explain, which could generate cognitive gaps in the understanding of other agents in the environment.
3. Semantic Equivalence: A reasoner that is producing a plan that can't be transformed intact[1] to the communication-representation will put the dog in a situation that the handler can't understand. Similarly, if the control level is relying on highly non-linear and inseparable functions, it could be very difficult to exactly explain its actions through the communication-level. Representation is a language, and the three representational levels need to be able to map the meanings they individually produce to each other. Semantic equivalence is a desirable feature, which would ensure that any meaning produced on one level has sentences on other levels that can reproduce it without introducing new meaning (ie correctness) or excluding some of the meaning (ie completeness).
4. Direct Syntactic Mapping: The easier it is to map each sentence on one level to a sentence on a different level, the less time it will take to translate between different levels. The direct mapping of syntactic structures from one level to another contributes to achieving the two requirements of computational efficiency and semantic equivalence.

In the next sub-section, we will present formal notations to demonstrate the mappings from the internals of shepherding and swarm guidance equations to the external transparent representation enabling the Jingulu-Swarm based communication language. These mappings are essential to ensure that the requirements above have been

[1] While we acknowledge that human communication contains and tolerates ambiguity, we argue that given the current state of technological advances in artificial intelligence systems, it is still difficult to allow for ambiguity to prevail in a human-AI interaction. Therefore, we are constraining the space currently to a bounded set of statements, where meaning is exact; thus, interpretation can be done intact if we further assume that the communication channels do not introduce further noise causing ambiguity in received messages.

taken into account during the design and have been met during the implementation phase of the system.

4.2 Interpretability, Explainability and Assurance of Human-Swarm Systems

Abbass et al. [34]. presented a formal definition of transparency towards bi-directional communication in human-swarm teaming systems. The concept of transparency was based on three dimensions, interpretability, explainability and predictability. We will quote these definitions here again for completeness and to allow us to expand on them for human-AI teaming.

Let $\mathbb{L}$ be a set of languages, where each language $\mathcal{L}_i \in \mathbb{L}$ is a set of sentences, $\mathbb{S}_\beta$. In this world view, any sentence in any language will be a member of the superset $\mathbb{L}$; while noting that communicative sentences by an agent are interpretations of internal knowledge statements $\mathbb{K}_\beta$ of the agent. The mapping from internal knowledge representation $\mathbb{K}_\beta$ to a sentence $\mathbb{S}_\beta$ is achieved by a transformation function $\mathbb{R}$. The reverse is achieved by $\mathbb{R}^{-1}$. Below is the set of definitions quoted from [34].

Definition 4.1 (Interpretability). $\mathbb{I}$ is an interpretation function that maps a sentence in one language to a sentence in a second, potentially the same, language; that is,

$$\mathbb{I}: \mathbb{S}_m \to \mathbb{S}_n,\ \mathbb{S}_m \in \mathcal{L}_i,\ \mathbb{S}_n \in \mathcal{L}_j,\ \mathcal{L}_i, \mathcal{L}_j \in \mathbb{L} \tag{2}$$

Definition 4.2 (Explainability). $\mathbb{E}$ *is an explanation function iff, given a hypothesis* $\mathbb{S}^{c_0 \to e}$*, there exists,*

$$\mathbb{E}: \mathbb{S}^{c_0 \to e} \to \{\mathbb{S}^{c_0}, \mathbb{S}^{c_0 \to c_1}, \mathbb{S}^{c_1 \to c_2}, \ldots, \mathbb{S}^{c_{k-1} \to c_k}, \mathbb{S}^{c_k \to e}\}$$

$$where\ \ \mathbb{R}^{-1}(\mathbb{S}^{c_i}) \in \mathbb{K}_\beta \tag{3}$$

Definition 4.3 (Predictability). $\mathbb{P}$ is a prediction function that takes a subset of axioms and facts, and projects them through induction or abduction onto a different set of axioms or facts; that is,

$$\mathbb{P}: \mathbb{S}^c \to \mathbb{S}^e \tag{4}$$

Definition 4.4 (Transparency). $\mathbb{T}$ is a transparency function that decides on the level of interpretability, explainability and predictability that will be visible from one agent to another agent, then implicitly or explicitly forms the language it will use to communicate this information to the other agents; that is,

$$\mathbb{T}: \{\mathbb{I}, \mathbb{E}, \mathbb{P}\} \to \mathcal{L}_j \tag{5}$$

Figure 3 depicts the coupling of the internal decision making of an agent with the above definitions, leading to a transparent human-AI teaming setting. Without loss of generality, we will assume in our example that agents have complete and certain information. The relaxation of this assumption does not change the modules in the conceptual diagram in **Figure 3**, but rather, it changes the design choices and complexity of implementing each module.

In **Figure 3**, an agent senses two types of states, its internal agent states reflecting its self-awareness, and the environment's states representing the states of other agents and the space it is located within. In shepherding, the sheepdog needs to sense its own position location (its own state), and the states of the environment, which consists of the position locations of sheep and the goal. The sheepdog needs to decide on its goal. This goal-setting module could choose the goal by listening to a human commanding the sheepdog or the sheepdog could have its own internal mechanism for goal setting as in autonomous shepherding. The goal could be mustering, where the aim is to herd the sheep to a goal location.

Based on the goal of the sheepdog, its state and the state in the environment, the agent needs to select an appropriate behaviour. The behavioural database contains two main behaviours in this basic model: a collecting and a driving behaviour based on the radius calculated in **Eq. 1**. Once a behaviour is selected, a planner is responsible for sequencing the series of local movements by the sheepdog to achieve the desired behaviour. For example, if the behaviour is to "drive," the sheepdog needs to reach the driving point using a path that does not disturb the sheep then modulate its force vectors on the sheep to drive them to the goal location. The planner will generate a series of velocity vectors for the sheepdog to follow. While the planner has an intended state, in a realistic setup, the desired state by the planner may be different from the actual state achieved in the environment due to many factors including noise in the actuators, terrain, weather, or energy level. The state update function is the oracle that takes the actions of the agents and updates the states of the agent and the environment.

The above description explains how the sheepdog makes decisions. However, an external agent, be it a human or artificial, needs to operate effectively as a teammate. Therefore, the sheepdog needs to be able to communicate the rationale of its decisions. The three factors for transparency mentioned above are critical in this setting [35]. Explainability provides teammates the reasons why certain goals, behaviours, and actions were selected at a particular point of time. Predictability offers the information for teammates to be ready for future actions of the sheepdog; thus, it reduces surprises which could negatively impact an agent's situation awareness and trust. The sheepdog is operating with force vectors as explained above. However, an external teammate may not understand these force vectors or may get overloaded when a sheepdog storms it with a large number of force vectors. This is where the force vectors generated by the explainability and predictability module need to be transformed into a language that the agent can use to interact with other agents. This language needs to be bidirectional; that is, humans and artificial agents need to be able to exchange sentences in this language that they can transform them into their own internal representation. In the case of sheepdogs, the plain English sentences need to be transformed to force vectors and vice-versa.

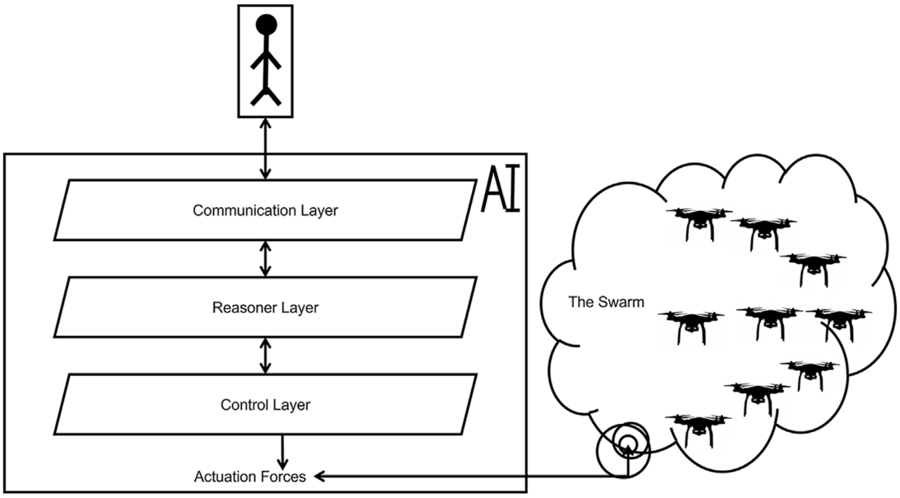


FIGURE 2 | Generic three-layer representation of AI.

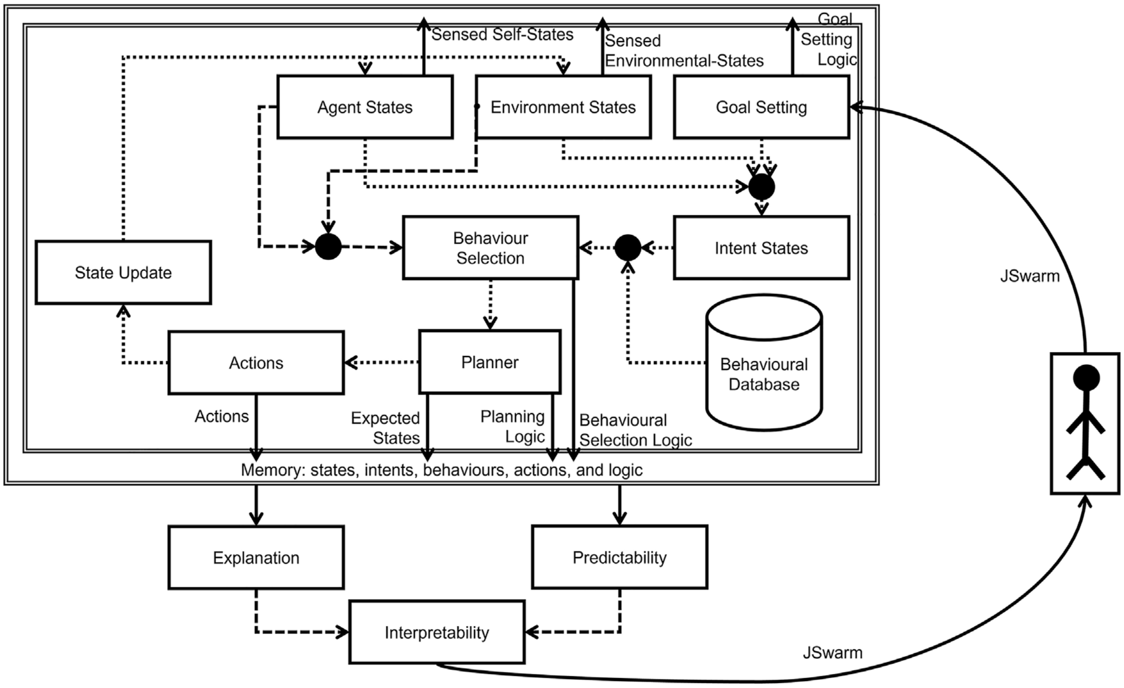


FIGURE 3 | Jingulu inspired Human-AI teaming.

5 JSWARM: A JINGULU-INSPIRED HUMAN-SWARM LANGUAGE

Similar to Boids, the shepherding system transforms all swarm actions into attraction and repulsion equations. In principle, a whole mission can be encoded in this system. Let us revisit sheep herding as a mission example to illustrate the application and efficiency of the JSwarm language. We recall herding as a library of behaviors that gets activated based on the sheep's and the dog's understanding of a situation. We will use the definitions of a context and a situation as per [36], where a context is "the minimum set of information required by an entity to operate autonomously and achieve its mission's objectives," while a situation is defined as "a manifestation of invariance in a

TABLE 2 | The Library of behaviors observed or performed by the sheepdog.

Situation	Behavior	Force vector	Sentence
Sheepdog detected	Sheep Attraction to Center of Mass	$F^{t}_{\pi_i \Lambda^{t}_{\pi_i}}$	COME to CM
Sheepdog detected	Sheep Repulsion from Nearby Sheep	$F^{t}_{\pi_i \pi\ _i}$	GO away from Nearby Sheep
Sheepdog detected	Sheep Repulsion from sheepdog	$F^{t}_{\pi_i \beta_j}$	GO away from sheepdog
Sheepdog not detected	Sheep perform small local random movements	$F^{t}_{\pi_i \epsilon}$	GO Random Direction and Steps
Always	Sheep move towards velocity at $t - 1$	$F^{t-1}_{\pi_i}$	GO velocity(t-1)
Astray sheep detected	Sheepdog Attraction to Collection Point	$F^{t}_{\beta_j c}$	COME to Collection Point
Sheep Collected	Sheepdog Attraction to Driving Point	$F^{t}_{\beta_j d}$	COME to Driving Point
Sheep detected on the path	Sheepdog Repulsion from nearby sheep	$F^{t}_{\beta_\pi}$	GO away from nearby sheep
Chaos or natural dynamics	Sheepdog remains at current location	$F^{t}_{\beta_j \epsilon}$	DO nothing

subset of this minimal information set over a period of time." In the herding example, different contexts and situations may activate different behaviors. We will illustrate these by categorizing all behaviors as either attraction, repulsion or no-movements. We will commence our description of the language by mapping all attractions, repulsions and no-movements forces to sentences with the verbs "COME," "GO," and "DO," respectively. We will assume for simplicity that the dog has complete accurate knowledge of all agents in the environment, including how the sheep moves. This is only for convenience for this first iteration of the language to avoid adding unnecessary complexity.

Table 2 summarises the basic behaviors in shepherding. It is worth noting that any other required behavior could still be explained using the attraction-repulsion system. For example, if the agent wishes to speak, words are directed towards an audience and thus, the audience becomes the attraction point or the agent becomes the repulsion point of the message. When an agent wishes to drop a parcel, the target location of the parcel becomes the attraction point and the agent becomes the repulsion point. When an agent wishes to eat, the food-store becomes the repulsion point of the food and the agent becomes the attraction point. In each of these examples, there is a frame of reference, such as the agent, where the world is seen from that agent's perspective. In this world view, every behavior in a mission of any type could be encoded as movements in a space. For example, this is the underlying fundamental concept of a transition in a state space, where a transition is a movement from one state/location in a space to another. When modelling flow of ideas, an idea is either sent to an agent (attracted to the agent), created by an agent (the doing of an agent), or brainstormed by an agent (doing of an agent). The attraction-repulsion system assumes that things move in one or more spaces. While the information on space and time are not required in the simple shepherding example, it is very important to include information on space and time in our description of the language when agents operates in different spaces or on different timescales to ensure that the language is general enough to capture these complexities.

Each agent has its context encoded in a state-vector, representing the super-set of all spaces an agent needs to be aware of. For example, in shepherding, three spaces are important, the physical space affording an agent with information on the location of each other agent in the environment, the group space providing information on the state of groups and members in the groups (whether the herd is clustered or not, there is astray sheep or not, if so, how many stray sheep) and the behavioral space representing the type of perceived or real behavior an agent or a group of agents are performing. The previous three spaces are sub-spaces of larger spaces as shown in **Figure 4**. Decisions are made by transforming sensorial information into particular spaces that an agent operate on. We call these the embedding spaces, a concept familiar to AI researchers working with natural language processing. The physical space is a subset of the embedding space. The behavioural space could be seen as the externalisation of actions generated in the cognitive space. The grouping aspects in shepherding are just a subset of the social space that could extend to social ties and relationships.

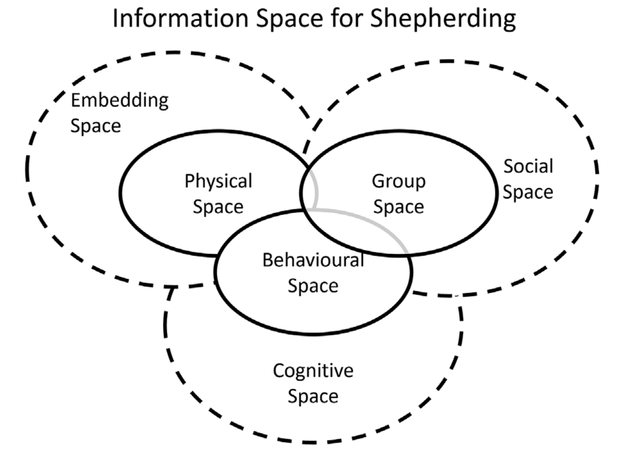


FIGURE 4 | The information spaces AI-enabled agents and humans operate within in the shepherding problem.

The cognitive level of an agent may focus on generating movements in the group and behavioral spaces. For example, the sheepdog would want to collect the astray sheep so that the

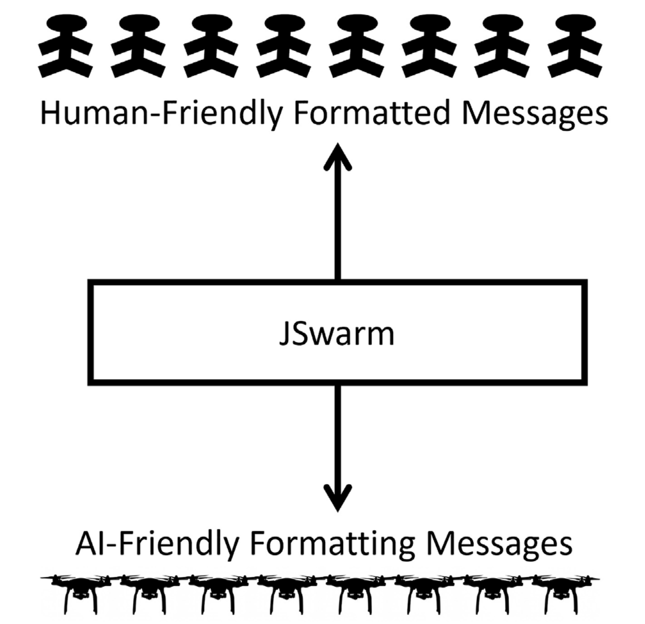

Human-Friendly Formatted Messages
JSwarm
AI-Friendly Formatting Messages

FIGURE 5 | JSwarm as mid-layer between AI-enabled agents and humans.

group of sheep is clustered together. These movements can be achieved through a plan that will then generate a sequence of actions requiring the sheepdog to move in the physical space. The plan and its associated actions need to be executed within a particular time-frame; thus, sentences may need to be parameterised with time.

The previous discussion offers the rationale for the design choice of the JSwarm langauge, which is presented formally next.

JSwarm has three types of sentences: sentences to communicate behaviors, sentences to communicate intent, and sentences to communicate state information. The state sentences are relevant when the AI communicates to humans its states. The behavioural systems are more relevant for action-production. The intent statements are crucial for commanding and communicating goals to the AI. **Figure 5** depicts the position of JSwarm as a formalism that sits between humans and AI agents. Each of the three sentence types has a structure explained below. We describe it in a predicate-logic-like syntax as an intermediate representation for humans and AI-enabled agents.

Behavior: Subject.Verb(Space.TimeDelay.
TimeDuration).SupportingVerb(Space.
TimeDelay.TimeDuration).Object
Intents: Subject.DO.SupportingVerb.Object
States:
Subject.DO.SupportingVerb(Space.TimeDuration).Object

We will present below examples to cover the space of each sentence type mentioned above. We assume all spaces represented in Cartesian coordinates and time in seconds.

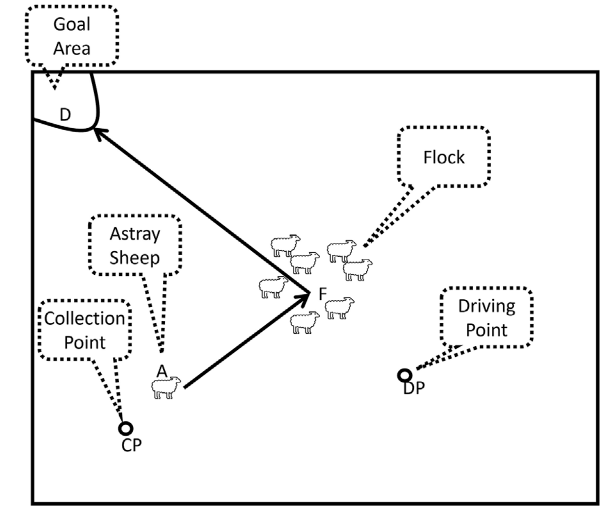


FIGURE 6 | A depiction of the environment for the shepherding scenario.

- Behaviour: *Subject.Verb(Space.TimeDelay. TimeDuration).SupportingVerb(Space.TimeDelay. TimeDuration).Object*
  - Sheep4.Go((x = 50,y = 70).0.50).Escape(NULL.0.30).Dog1
    Now (0 delays), Sheep Sheep4 needs to go to location (50,70) within 50 s and escape dog Dog1 for 30 s.
  - Sheep2.Come((x = 20,y = 15).0.30).Group(NULL.0.10).Herd1
    Within 30 s from now, sheep Sheep2 needs to arrive at location (20,15) to group within 10 s with herd Herd1.
  - Dog1.Come((x = 40,y = 40).0.30).Collecting(NULL.0.20).Sheep5
    Within 30 s from now, dog Dog1 needs to start moving to arrive at location (40,40) and spend 20 s collecting sheep Sheep5.
  - Dog1.Come((x = 70,y = 20).0.30).Driving(NULL.0.15).Herd0
    Within 30 s from now, dog Dog1 needs to start moving to arrive at location (70,20) to drive Herd0 for 15 s.
  - Herd1.Do(NULL.0.0).Sitting(NULL.0.20)
    Herd Herd1 needs to stay in its current location for 20 s.
  - Dog1.Do(NULL.10.0).Sitting(NULL.0.30)
    Dog Dog1 needs to wait for 10 s then sit in its location for 30 s.
- Intents: *Subject.DO.SupportingVerb.Object*
  - Dog1.Do.Herd.Herd1
    Dog Dog1 needs to herd group Herd1.
  - Herd1.Do.Escape.Dog1
    Herd Herd1 needs to escape dog Dog1.
- States: *Subject.SupportingVerb(TimeDuration).Object*
  - Dog1.Do.Collecting(120).Many
    Dog Dog1 is collecting many sheep for 2 min.
  - Dog1.Do.Collecting(160).Sheep5
    Dog Dog1 is collecting sheep Sheep5 for 3 min.
  - Dog1.Do.Driving(60).All
    Dog Dog1 is driving all sheep for 1 min.
  - Dog1.Do.Driving(30).Herd0
    Dog Dog1 is driving herd Her0 for 30 s.

- Dog1.Do.Patrolling(3600).All
  Dog Dog1 is patrolling all sheep for an hour.
- Sheep1.Do.Foraging(1,200)
  Sheep Sheep1 is foraging for 20 min.
- Sheep1.Do.Escaping(60).Herd1
  Sheep Sheep1 is escaping herd Her1 for 1 min.

JSwarm is designed to be a transparent human-swarm language in a manner consistent with **Eq. 5** and the definition of transparency given in the previous section. As shown in **Figure 5**, JSwarm sits at the middle layer between humans and AI-enabled swarms. We provided in this paper the direct mappings from JSwarm to force vectors and vice-versa. This form of interpretability is both sound and complete. The example presented in the following section will demonstrate explainability, where the logs of the JSwarm is a series of expressions providing the intents and actions taken by the AI-enabled swarm. Predictability relies on the mental model formed within an agent's brain (including computers) for an agent to predict another. Thanks to the 1-to-1 mapping in JSwarm, and the simplicity of the abstract shepherding problem presented in this paper, predictability is less of a concern. While we argued that JSwarm is transparent in the abstract shepherding problem, our future work will test this hypothesis in more complex environments.

6 HUMAN-SWARM LANGUAGE DEMONSTRATION

The JSwarm language could generate significant amount of sentences due to its ability to work on the level of atomic action. It could also generate very comprehensive sentences due to its ability to work on the behavioural space. The design relies on the definitions provided in the previous section, where interpretability is the mapping from the equations of shepherding to JSwarm syntax presented in this section, and explainability is the outcome of the sequence of expressions produced by JSwarm to explain the sequence of behaviours presented by an agent. In this section, we will present a scenario for shepherding to demonstrate the use of the JSwarm language.

Consider a case of a 100 × 100 m paddock with the goal situated at the top left corner, the sheep are spread around the centre point with an astray sheep at location (20,20), and the dog at the goal location. The dog has complete and accurate situation awareness of the location of all sheep. Its internal logic determines that it needs to activate its collecting behaviour to move around the edge of the paddock to reach the collection point behind the astray sheep. The collection point is at location (15,15), where its location sits on the direction vector from the location of the sheep (20,20) to the location of the centre of the flock (50,50). The characters in this scenario are labelled D for the dog, A for the astray sheep, and F for the flock. The dog needs to communicate its actions every 5 s or when it selects a different behaviour. Below is a series of messages announced by the dog in JSwarm to indicate its actions and what it perceives in the environment. This environment is depicted in **Figure 6**.

Dog.Do.Herd.F % Intent communicated that the dog needs to herd the flock.
Dog.Do.Collecting.A % Intent communicated that the dog is collecting astray sheep A
Dog.Come(x = 15,y = 15).Collecting.CP % Dog is on its way to collection point CP for astray sheep.

The above sentence repeats until Dog reaches the collection point, CP.

Dog.Come(x = 50,y = 50).Collecting.A % Dog is collecting astray sheep in the direction of the flock centre

The above sentence repeats until sheep A joins the flock or the dog drifts away from the collection point. We assume the latter, at which point in time, the dog needs to move towards the new location of the collection point.

Dog.Come(x = 30,y = 30).Collecting.CP % Dog on its way to new collection point for astray sheep
Dog.Come(x = 50,y = 50).Collecting.A % Dog collecting astray sheep in the direction of the flock centre

The above sentence repeats until sheep A joins the flock.

Dog.Do.Driving.F % Dog's intent change to driving the flock
Dog.Come(x = 75,y = 75).Driving.DP % Dog on its way to driving point, DP, for the flock

The above sentence repeats until the dog reaches the driving point.

Dog.Come(x = 100,y = 100).Driving.F % Dog is driving the flock F towards the goal

The above sentence repeats until sheep are at the goal.

Dog.Do.Rest

While in the above example, we focused on the dog communicating its actions, the example could get extended where the dog communicates also its situation awareness, the sheep communicates their actions and situation awareness as individuals, and the AI communicates the sheep flock actions. It is important to notice that in the above example, we did not use the time parameters in the language due to the fact that the simulation for abstract shepherding is normally event-driven rather than clock-driven. We could also decide to represent spatial locations in other formats. The exact representation of the parameters is a flexible user choice.

7 CONCLUSION AND FUTURE WORK

Effective human-AI teaming requires a language that enables bidirectional communication between the humans and the AI agents. While human natural languages could be candidates, we explained that the richness of these languages come with a cost of increased ambiguity. The humans and the swarm need to communicate in an unambiguous manner to reduce confusion and misunderstanding. Consequently, we defined four main requirements in the design choice of a language for human-AI teaming; then we presented a language inspired by the Jingulu language, an Australian Aboriginal language.

The JSwarm language is the first of its kind Human-AI Teaming language that is based on direct mappings from the internal logic, including the equation of motion, of an agent to a human-friendly language. The language is designed to accurately reflect the internal attraction-repulsion equations governing the dynamics of a swarm, including states, intent and behaviours. JSwarm allows humans and swarms to communicate with each other without ambiguity and in a form that could be verified. The language separates semantics from syntax, where the supporting verb acts as a semantic carrier. While the light verb impacts syntax, the supporting verb does not affect the syntax of an expression, thus allowing semantics to be associated and de-associated freely. This latter feature could utilise an ontology, and allows the syntactical-layer of the language to remain intact as it gets applied to different domains, while a replacement of the ontology changes the semantic layer. Moreover, the free word order feature in the language could offer a robust communication setting, where meaning is maintained even if the receiver orders the words differently.

We proposed a simple grammar and representation of sentences in the language, which was intentionally selected such that it mimics the structure of a first-order logical representation, while being semantically-friendly to human comprehension. We concluded the paper with an example to showcase how the language could be used to provide a real-time log for a dog to communicate its actions in a human-friendly language.

For our future work, we will extend the design to connect the JSwarm language as it works on the communication layer with the representations used on the control and reasoner layers. We will also conduct human studies to evaluate the efficacy of JSwarm. It is important that the human usability study to evaluate the effectiveness of JSwarm takes place with a complex scenario with appropriate architectures and implementations of the swarm. The risk of testing the concept in a simple scenario is that the human will find the scenario trivial and the need for explanation unwarranted.

While the JSwarm language is explained using a shepherding example to make the paper accessible to a larger readership, the language is designed to be application-agonistic and could benefit any problem where communication between humans and a large number of AI-enabled agents is required. For example, a swarm of nano–robots combatting cancer cells could offer a perfect illustration where the robots have a very simple logic that needs to be transformed to an explanation to the medical practitioner overseeing the operation of the system. The medical practitioner equally needs a language to command the swarm that is simple to match the internal swarm logic and reduces communication load. In these applications, the sheepdog could be a chemical substance that the swarm of nano–robots react to, which is controlled by an external robot that the medical practitioner needs to command to guide the swarm. Other applications include a swarm of under-water vehicles cleaning the ocean or in the mining industry, a swarm of uncrewed aerial vehicles surveying a large area.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/Supplementary Material, further inquiries can be directed to the corresponding author.

AUTHOR CONTRIBUTIONS

HA conceptualised and wrote the first draft. EP contributed to the linguistic analysis. RH contributed to conceptualising the impact of the work on human-AI teaming. All authors revised and edited the paper.

FUNDING

This work is funded by Grant No N62909-18-1-2140 from Office of Naval Research Global (ONRG). Its contents are solely the responsibility of the authors and do not necessarily represent the official views of the ONRG.

ACKNOWLEDGMENTS

The authors wish to thank Adam Aly for role playing the AI to formulate an example for this paper, and Beth Cardier, Adam Hepworth, and Aya Hussein for providing constructive feedback on the work. We also wish to thank the reviewers for their constructive comments.

REFERENCES

1. Coulmas F. *Sociolinguistics: The Study of Speakers' Choices*. Cambridge, UK: Cambridge University Press (2013).
2. Holmes J, Wilson N. *An Introduction to Sociolinguistics*. Abingdon-on-Thames, Oxfordshire United Kingdom: Taylor & Francis Group (2022).
3. Plun JY, Wilcox CD, Cox KC. *Swarm Language Reference Manual*. Seattle, Washington: Washington University in St. Louis (1992).
4. Vaniya S, Lad B, Bhavsar S. A Survey on Agent Communication Languages. *2011 Int Conf Innovation, Management Serv* (2011) 14: 237–42.
5. Pantelimon G, Tepe K, Carriveau R, Ahmed S. Survey of Multi-Agent Communication Strategies for Information Exchange and mission Control

of Drone Deployments. *J Intell Robot Syst* (2019) 95:779–88. doi:10.1007/s10846-018-0812-x

6. Bocklisch T, Faulkner J, Pawlowski N, Nichol A. *Rasa: Open Source Language Understanding and Dialogue Management*. arXiv preprint (2017).
7. Singh S, Beniwal H. A Survey on Near-Human Conversational Agents. *J King Saud Univ - Computer Inf Sci* (2021) 2021. doi:10.1016/j.jksuci.2021.10.013
8. Almansor EH, Hussain FK. Survey on Intelligent Chatbots: State-Of-The-Art and Future Research Directions. In: *Conference on Complex, Intelligent, and Software Intensive Systems*. Cham, Switzerland: Springer (2019). p. 534–43. doi:10.1007/978-3-030-22354-0_47
9. Su PH, Mrkšić N, Casanueva I, Vulić I. Deep Learning for Conversational Ai. In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Tutorial Abstracts*. Pennsylvania, US: Association for Computational Linguistics (2018). p. 27–32. doi:10.18653/v1/n18-6006
10. Cambier N, Miletitch R, Frémont V, Dorigo M, Ferrante E, Trianni V. Language Evolution in Swarm Robotics: A Perspective. *Front Robot AI* (2020) 7:12. doi:10.3389/frobt.2020.00012
11. Pensalfini R. *A Grammar of Jingulu, an Aboriginal Language of the Northern Territory*. Canberra, Australia: The Australian National University (2003).
12. Abbass HA, Hunjet RA. Smart Shepherding: Towards Transparent Artificial Intelligence Enabled Human-Swarm Teams. In: *Shepherding UxVs for Human-Swarm Teaming: An Artificial Intelligence Approach*. Cham, Switzerland: Springer (2020). p. 1–28. doi:10.1007/978-3-030-60898-9_1
13. Austin P, Bresnan J. Non-configurationality in Australian Aboriginal Languages. *Nat Lang Linguist Theor* (1996) 14:215–68. doi:10.1007/bf00133684
14. Simpson J. *Warlpiri Morpho-Syntax: A Lexicalist Approach*. Berlin, Germany: Springer Science & Business Media Dordrecht (1991).
15. Pensalfini R. The Rise of Case Suffixes as Discourse Markers in Jingulu-A Case Study of Innovation in an Obsolescent Language*. *Aust J Linguistics* (1999) 19:225–40. doi:10.1080/07268609908599582
16. Reynolds CW. Flocks, Herds and Schools: A Distributed Behavioral Model. *SIGGRAPH Comput Graph* (1987) 21:25–34. doi:10.1145/280811.28100810.1145/37402.37406
17. Isobe M, Helbing D, Nagatani T. Experiment, Theory, and Simulation of the Evacuation of a Room without Visibility. *Phys Rev E Stat Nonlin Soft Matter Phys* (2004) 69:066132. doi:10.1103/PhysRevE.69.066132
18. Nalepka P, Kallen RW, Chemero A, Saltzman E, Richardson MJ. Herd Those Sheep: Emergent Multiagent Coordination and Behavioral-Mode Switching. *Psychol Sci* (2017) 28:630–50. doi:10.1177/0956797617692107
19. Paranjape AA, Chung S-J, Kim K, Shim DH. Robotic Herding of a Flock of Birds Using an Unmanned Aerial Vehicle. *IEEE Trans Robot* (2018) 34:901–15. doi:10.1109/TRO.2018.2853610
20. Kakalis NMP, Ventikos Y. Robotic Swarm Concept for Efficient Oil Spill Confrontation. *J Hazard Mater* (2008) 154:880–7. doi:10.1016/j.jhazmat.2007.10.112
21. Cohen D. Cellular Herding: Learning from Swarming Dynamics to Experimentally Control Collective Cell Migration. *APS March Meet Abstr* (2019) 2019:F61.
22. Long NK, Sammut K, Sgarioto D, Garratt M, Abbass HA. A Comprehensive Review of Shepherding as a Bio-Inspired Swarm-Robotics Guidance Approach. *IEEE Trans Emerg Top Comput Intell* (2020) 4:523–37. doi:10.1109/tetci.2020.2992778
23. Tang J, Leu G, Abbass HA. Networking the Boids Is More Robust against Adversarial Learning. *IEEE Trans Netw Sci Eng* (2017) 5:141–55.
24. Lien JM, Rodríguez S, Malric JP, Amato NM. Shepherding Behaviors with Multiple Shepherds. In: Proceedings of the 2005 IEEE International Conference on Robotics and Automation; 18-22 April 2005; Barcelona, Spain. New York, US: IEEE (2005). p. 3402–7.
25. Miki T, Nakamura T. An Effective Simple Shepherding Algorithm Suitable for Implementation to a Multi-Mmobile Robot System. In: First International Conference on Innovative Computing, Information and Control - Volume I (ICICIC'06); 30 August 2006 - 01 September 2006; Beijing, China. New York, US: IEEE (2006). p. 161–5. doi:10.1109/ICICIC.2006.411
26. Strömbom D, Mann RP, Wilson AM, Hailes S, Morton AJ, Sumpter DJT, et al. Solving the Shepherding Problem: Heuristics for Herding Autonomous, Interacting Agents. *J R Soc Interf* (2014) 11:20140719. doi:10.1098/rsif.2014.0719
27. Kalantar S, Zimmer U. A Formation Control Approach to Adaptation of Contour-Shaped Robotic Formations. In: 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems; 09-15 October 2006; Beijing, China. New York, US: IEEE (2006). p. 1490–7. doi:10.1109/IROS.2006.281977
28. Razali S, Meng Q, Yang S-H. Immune-inspired Cooperative Mechanism with Refined Low-Level Behaviors for Multi-Robot Shepherding. *Int J Comp Intel Appl* (2012) 11:1250007. doi:10.1142/s1469026812500071
29. Bat-Erdene B, Mandakh O-E. Shepherding Algorithm of Multi-mobile Robot System. In: 2017 First IEEE International Conference on Robotic Computing (IRC); 10-12 April 2017; Taichung, Taiwan. New York, US: IEEE (2017). p. 358–61. doi:10.1109/IRC.2017.51
30. Masehian E, Royan M. Cooperative Control of a Multi Robot Flocking System for Simultaneous Object Collection and Shepherding. In: *Computational Intelligence*. Cham, Switzerland: Springer (2015). p. 97–114. doi:10.1007/978-3-319-11271-8_7
31. Nalepka P, Kallen RW, Chemero A, Saltzman E, Richardson MJ. Herd Those Sheep: Emergent Multiagent Coordination and Behavioral-Mode Switching. *Psychol Sci* (2017) 28:630–50. doi:10.1177/0956797617692107
32. El-Fiqi H, Campbell B, Elsayed S, Perry A, Singh HK, Hunjet R, et al. The Limits of Reactive Shepherding Approaches for Swarm Guidance. *IEEE Access* (2020) 8:214658–71. doi:10.1109/access.2020.3037325
33. Hussein A, Ghignone L, Nguyen T, Salimi N, Nguyen H, Wang M, et al. Characterization of Indicators for Adaptive Human-Swarm Teaming. *Front Robot AI* (2022) 9:745958. doi:10.3389/frobt.2022.745958
34. Abbass H, Petraki E, Hussein A, McCall F, Elsawah S. A Model of Symbiomemesis: Machine Education and Communication as Pillars for Human-Autonomy Symbiosis. *Phil Trans R Soc A* (2021) 379:20200364. doi:10.1098/rsta.2020.0364
35. Hepworth AJ, Baxter DP, Hussein A, Yaxley KJ, Debie E, Abbass HA. Human-swarm-teaming Transparency and Trust Architecture. *IEEE/CAA J Automatica Sinica* (2020) 8:1281–95.
36. Fernandez-Rojas R, Perry A, Singh H, Campbell B, Elsayed S, Hunjet R, et al. Contextual Awareness in Human-Advanced-Vehicle Systems: A Survey. *IEEE Access* (2019) 7:33304–28. doi:10.1109/access.2019.2902812

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

ORIGINAL RESEARCH
published: 15 July 2022
doi: 10.3389/fphy.2022.944157

Probabilistic Inference and Dynamic Programming: A Unified Approach to Multi-Agent Autonomous Coordination in Complex and Uncertain Environments

Giovanni Di Gennaro[1]*, *Amedeo Buonanno*[2], *Giovanni Fioretti*[1], *Francesco Verolla*[1], *Krishna R. Pattipati*[3] *and Francesco A. N. Palmieri*[1,3]

[1]Dipartimento di Ingegneria, Università degli Studi della Campania "Luigi Vanvitelli", Aversa, Italy, [2]Department of Energy Technologies and Renewable Energy Sources, ENEA, Portici, Italy, [3]Department of Electrical and Computer Engineering, University of Connecticut, Storrs, CT, United States

OPEN ACCESS

Edited by:
William Frere Lawless,
Paine College, United States

Reviewed by:
Hesham Fouad,
United States Naval Research Laboratory, United States
Donald Sofge,
United States Naval Research Laboratory, United States

****Correspondence:***
Giovanni Di Gennaro
giovanni.digennaro@unicampania.it

Specialty section:
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

Received: *14 May 2022*
Accepted: *21 June 2022*
Published: *15 July 2022*

Citation:
Di Gennaro G, Buonanno A, Fioretti G, Verolla F, Pattipati KR and Palmieri FAN (2022) Probabilistic Inference and Dynamic Programming: A Unified Approach to Multi-Agent Autonomous Coordination in Complex and Uncertain Environments. Front. Phys. 10:944157. doi: 10.3389/fphy.2022.944157

We present a unified approach to multi-agent autonomous coordination in complex and uncertain environments, using path planning as a problem context. We start by posing the problem on a probabilistic factor graph, showing how various path planning algorithms can be translated into specific message composition rules. This unified approach provides a very general framework that, in addition to including standard algorithms (such as sum-product, max-product, dynamic programming and mixed Reward/Entropy criteria-based algorithms), expands the design options for smoother or sharper distributions (resulting in a generalized sum/max-product algorithm, a smooth dynamic programming algorithm and a modified versions of the reward/entropy recursions). The main purpose of this contribution is to extend this framework to a multi-agent system, which by its nature defines a totally different context. Indeed, when there are interdependencies among the key elements of a hybrid team (such as goals, changing mission environment, assets and threats/obstacles/constraints), interactive optimization algorithms should provide the tools for producing intelligent courses of action that are congruent with and overcome bounded rationality and cognitive biases inherent in human decision-making. Our work, using path planning as a domain of application, seeks to make progress towards this aim by providing a scientifically rigorous algorithmic framework for proactive agent autonomy.

Keywords: path-planning, dynamic programming, multi-agent, factor graph, probabilistic inference

1 INTRODUCTION

Decision-making problems involve two essential components: the *environment*, which represents the problem, and the *agent*, which determines the solution to the problem by making decisions. The agent interacts with the environment through its decisions, receiving a *reward* that allows it to evaluate the efficacy of actions taken in order to improve future behavior. Therefore, the overall problem consists of a sequence of steps, in each of which the agent must choose an action from the available options. The objective of the agent will be to choose an optimal action sequence that brings the entire system to a trajectory with maximum cumulative reward (established on the basis of the reward obtained at each step). However, when the problem becomes stochastic, the main thing to pay attention to is how to evaluate the various possible rewards based on the intrinsic stochasticity of the environment. The evaluation of the reward on the basis of the probabilistic transition function leads in fact to different reward functions to be optimized. The first part of this work will show how it is possible to manage these different situations through a unified framework, highlighting its potential as a methodological element for the determination of appropriate value functions.

The present work aims to extend this framework to manage the behavior of several interdependent autonomous agents who share a common environment. We will refer to this as a *multi-agent system* (MAS) [1]. The type of approach to a MAS problem strongly depends on how the agents interact with each other and on the final goal they individually set out to achieve. A "fully cooperative" approach arises when the reward function is shared and the goal is to maximize the total sum of the rewards obtained by all the agents. The cooperative MAS can be further subdivided into "aware" and "unaware" depending on the knowledge that an agent has of other agents [2]. Moreover, the cooperative aware MAS can be "strongly coordinated" (the agents strictly follow the coordination protocols), "weakly coordinated" (the agents do not strictly follow the coordination protocols), and "not coordinated." Furthermore, the agents in a cooperative aware and strongly coordinated MAS can be "centralized" (an agent is elected as the leader) or "distributed" (the agents are completely autonomous). Conversely, a "fully competitive" approach ensues when the total sum of the rewards tends to zero, and the agents implicitly compete with each other to individually earn higher cumulative rewards at the cost of other agents.

In various applications, ranging from air-traffic control to robotic warehouse management, there is the problem of centralized planning of the optimal routes. Although *dynamic programming* (DP) [3, 4] provides an optimal solution in the single-agent case, finding the optimal path for a multi-agent system is nevertheless complex, and often requires enormous computational costs. Obviously there are some research efforts that investigate MAS using DP [5, 6], however, they are not directly focused on the solution of a path planning problem, but rather on solving a general cooperative problem. Furthermore, it is worth noting that many research efforts are devoted to the application of *reinforcement learning* (RL) [7, 8] to MAS that constitutes a new research field termed *multi-agent reinforcement learning* (MARL) [9–11]. The main problem with reinforcement learning is the need for a large number of simulations to learn the policy for a given context, and the need to relearn when the environment changes [12]. Indeed, it is essential to understand that the agent (being autonomous but interdependent on others) must consider the actions of other agents in order to improve its own policy. In other words, from the agents' local perspective, the environment becomes non-stationary because its best policy changes as the other agents' policies change [9]. Moreover, as the number of agents increase, the computational complexity becomes prohibitively expensive [11].

Finally, previous works that approach the problem of path planning in a MAS context (both centralized and decentralized) do not consider regions with different rewards, ending up simply generating algorithms whose solution is the minimum path length [13]. Consideration of maps with non-uniform rewards is salient in real world scenarios: think of pedestrians that prefer sidewalks, or bikers who prefer to use bikeroutes, or ships that may use weather information to choose the best paths, etc.

The main reason for focusing on a particular problem of interest lies in the fact that knowledge of it can somehow speed up the calculations. In particular, with regards to path planning, if the goals are known to each agent *a priori* (as we will discuss in this work) the appropriate evaluation of the paths can be obtained using pre-computed value functions. In this case, the optimal paths can be determined without learning the policy directly, but by obtaining it on the basis of the information available from other agents. Through this work, we will show exactly how, using the knowledge of the problem and a *factor graph in reduced normal form* (FGrn) [14, 15], it is possible to find the optimal path in a MAS with minimal computational costs, guaranteeing an optimal solution under certain scheduling constraints. The multi-agent extension of the framework will be achieved by creating a forward flow, which will use the previously computed single-agent backward flow to enable decision making (recalling the classic probabilistic use).

Section 2 presents the Bayesian model and the corresponding factor graph in reduced normal form for the single agent case. This section shows the generality of the factor graph approach by introducing the main equations for the calculation of the value functions related to the various versions of the algorithms from probabilistic inference and operations research. **Section 3** deals with the multi-agent problem, highlighting the algorithmic solution that uses the forward step coupled with the single-agent backward step, while **Section 4** shows some simulation examples. Finally, in **Section 5**, the relevant conclusions are drawn.

2 THE SINGLE-AGENT SCENARIO

When the outcomes generated by the actions are uncertain, because partly under the control of the agent and partly random, the problem can be defined as a *Markov decision process* (MDP) [16, 17]. This discrete-time mathematical tool forms the theoretical basis for the modeling of a general class of sequential decision problems in a single-agent scenario, and consequently the well-known DP, RL, and other classical decision algorithms basically aim to solve an MDP under various assumptions on the evolution of the environment and reward structure.

Mathematically, at any discrete time step t, the agent of a MDP problem is assumed to observe the state $S_t \in \mathcal{S}$ and chooses action $A_t \in \mathcal{A}$. If the sets of states and actions ($\mathcal{S}$ and $\mathcal{A}$) have a finite number of elements, the random process S_t is described as a discrete conditional probability distribution, which can be assumed to be dependent only on the previous state and action

$$p(s_{t+1}|s_t, a_t) = Pr\{S_{t+1} = s_{t+1}|S_t = s_t, A_t = a_t\}$$

for each admissible value of the random variables $s_{t+1}, s_t \in \mathcal{S}$, and $a_t \in \mathcal{A}$. At the next time step, having moved to the state S_{t+1}, the

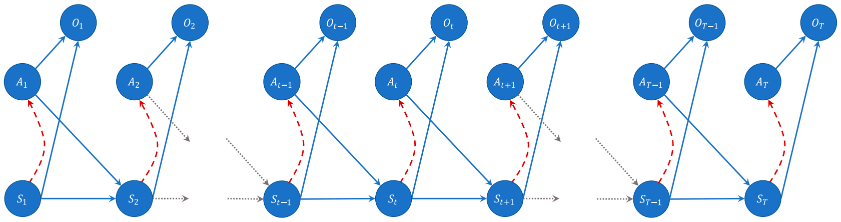


FIGURE 1 | Bayesian graph of the generative model of an MDP, in which the variable O_t represents the optimum for that time step.

agent receives a reward $r(s_t, a_t) \in \mathcal{R} \subset \mathbb{R}$ determined according to the previous state and action, thanks to which it will understand the goodness of the previous action.

A Bayesian representation of the MDP can be obtained by adding a binary random variable $O_t \in \{0, 1\}$ (that denotes if the state-action pair at time step t is optimal or not) to the Markov chain model determined by the sequence of states and actions [18–20].

This addition (**Figure 1**) gives the resulting model the appearance of a *hidden Markov model* (HMM), in which the variable O_t corresponds to the observation. In this way, at each time step, the model emits an "optimality" measure in the form of an indicator function, which leads to the concept of "reward" necessary to solve the problem of learning a policy $\pi(a_t|s_t)$ (highlighted in red in **Figure 1**) that maximizes the expected cumulative reward. Assuming a finite horizon T, [1] the joint probability distribution of the random variables in **Figure 1** can therefore be factored as follows

$$p(s_1, a_1, o_1, \ldots, s_T, a_T, o_T) = p(s_1)p(a_T)p(o_T|s_T, a_T) \times \prod_{t=1}^{T-1} p(s_{t+1}|s_t, a_t)p(a_t)p(o_t|s_t, a_t)$$

where $p(a_t)$ is the prior on the actions at time step t. In other words, the introduction of the binary random variable O_t represents a "trick" used by the stochastic model to be able to condition the behavior of the agent at time step t, so that it is "optimal" from the point of view of the rewards that the agent can get. Specifically, defining with $c(s_t, a_t)$ the general distribution of the random variable O_t, we obtain that when $O_t = 0$, there is no optimality and

$$p(O_t = 0|s_t, a_t) \triangleq c(s_t, a_t) \propto \mathcal{U}(s_t, a_t).$$

where $\mathcal{U}(s_t, a_t)$ is the uniform distribution over states and actions, implying the agent has no preference to any particular state and action of the MDP. Vice-versa optimality with $O_t = 1$ corresponds to

$$p(O_t = 1|s_t, a_t) \triangleq c(s_t, a_t) \propto \exp(r(s_t, a_t))$$

where $r(s_t, a_t)$ is the reward function and the exponential derives from opportunistic reasons that will be clarified shortly. Since what really matters is the optimal solution obtained by conditioning on $O_t = 1$ for every $t = 1, \ldots, T$, we can also omit the sequence$\{O_t\}$ from the factorization, and rewrite the joint distribution of state-action sequence over $[0, T]$ conditioned on optimality as

$$p(s_1, a_1, \ldots, s_T, a_T|\mathrm{O}_{1:T} = \mathbf{1}) = p^*(s_1, a_1, \ldots, s_T, a_T) \times \propto p(s_1)p(a_T)c(s_T, a_T) \prod_{t=1}^{T-1} p(s_{t+1}|s_t, a_t)p(a_t)c(s_t, a_t)$$

It can thus be noted that

$$p^*(s_1, a_1, \ldots, s_T, a_T) \propto \left[p(s_1)p(a_T) \prod_{t=1}^{T-1} p(s_{t+1}|s_t, a_t)p(a_t)\right] \times \exp\left(\sum_{t=1}^{T} r(s_t, a_t)\right)$$

and therefore, through the previous definition of the function $c(s_t, a_t)$ as the exponential of the reward function, the probability of observing a given trajectory also becomes effectively dependent on the total reward that can be accumulated along it.

2.1 The Factor Graph

The probabilistic formulation of the various control/estimation algorithms based on MDP can be conveniently translated into a *factor graph* (FG) [21–23], in which each variable is associated with an arc and the various factors represent its interconnection blocks. In particular, we will see how it is extremely useful to adopt the "reduced normal form" (introduced previously) which allows, through the definition of "shaded" blocks, to map a single variable in a common space; simplifying the message propagation rules through a structure whose functional blocks are all SISO (single-input/single-output). The Bayesian model of interest in **Figure 1** can in fact be easily translated into the FGrn of **Figure 2**, where the *a priori* distributions $p(a_t)$ and $c(s_t, a_t)$ are mapped to the source nodes, and the probabilities of transition $p(s_{t+1}|s_t, a_t)$ are implemented in the $\mathcal{M}$ SISO blocks. Each arc is associated with a "forward" f and a "backward" b message, proportional to the probability distributions and whose composition rules allow an easy propagation of the probability; while, as usual, the diverter (in red) represents the equality constraint

[1] Note that we consider the time horizon T as the last time step in which an action must be performed. The process will stop at instant $T + 1$, where there is no action or reward.

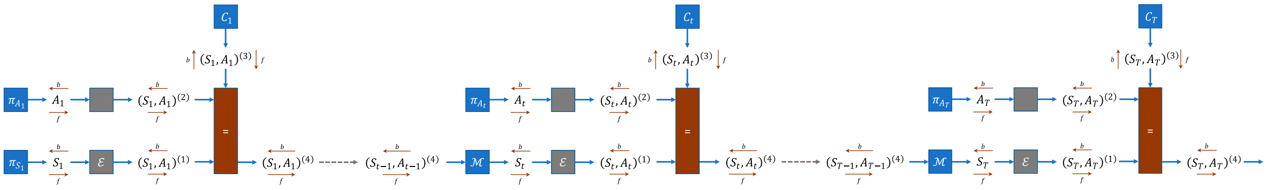


FIGURE 2 | Factor Graph in reduced normal form of an MDP, in which the reward is introduced through the variable C_t.

between the variables that belong to it. Furthermore, using this structure, one has the advantage of easily introducing constraints and *a priori* knowledge into the corresponding messages that propagate through them. [2] In general, all available evidence can in fact be collectively identified through the symbol $K_{1:T}$, so that we can describe the model in the form $p^*(s_1, a_1, \ldots, s_T, a_T|K_{1:T})$ during the process of inference. For the defined graph, we therefore aim to compute mainly the functions

$$p^*(s_t|K_{1:T}) \propto \sum_{\substack{s_j,\ j=1:T, j\neq t \\ a_k,\ k=1:T}} p^*(s_1, a_1, \ldots, s_T, a_T|K_{1:T})$$
$$p^*(s_t, a_t|K_{1:T}) \propto \sum_{s_j, a_j,\ j=1:T, j\neq t} p^*(s_1, a_1, \ldots, s_T, a_T|K_{1:T})$$

which, for example, can be derived from message propagation through the use of the classic *sum-product* rule [22, 24]. Note that these are the only functions needed for our purposes because the optimal policy at time t can be obtained by

$$\pi^*(a_t|s_t) \triangleq p^*(a_t|s_t, K_{t:T}) = \frac{p^*(s_t, a_t|K_{t:T})}{p^*(s_t|K_{t:T})}, \qquad t = 1:T$$

By rigorously applying Bayes' theorem and marginalization, the various messages propagate within the network and contribute to the determination of the posteriors through the simple multiplication of the relative forward and backward messages [14, 25]. In particular, from **Figure 2**, it can be seen that the calculation of the distribution for the policy at time t can generally be rewritten as

$$\pi^*(a_t|s_t) \propto \frac{f_{(S_t,A_t)^{(i)}}(s_t, a_t) b_{(S_t,A_t)^{(i)}}(s_t, a_t)}{f_{S_t}(s_t) b_{S_t}(s_t)}$$
$$= \frac{f_{S_t}(s_t) \mathcal{U}(a_t) b_{(S_t,A_t)^{(i)}}(s_t, a_t)}{f_{S_t}(s_t) b_{S_t}(s_t)} = \frac{b_{(S_t,A_t)^{(i)}}(s_t, a_t)}{b_{S_t}(s_t)}$$

and, therefore, the policy depends solely on the backward flow, [3] since (by conditioning on s_t) all the information coming from the forward direction is irrelevant to calculate it. [4]

Particularly interesting is the passage from the probabilistic space to the logarithmic space, which, within the FGrn, can be obtained through the simple definition of the functions

$$V_{S_t}(s_t) \triangleq \ln b_{S_t}(s_t)$$
$$Q_{(S_t,A_t)^{(i)}}(s_t, a_t) \triangleq \ln b_{(S_t,A_t)^{(i)}}(s_t, a_t), \qquad i = 1, \ldots, 4$$

whose name is deliberately assigned in this way to bring to mind the classic DP-like approaches. [5] Looking at **Figure 2**, the backward propagation flow can then be rewritten considering the passage of messages through the generic transition operators represented by the blocks $\mathcal{M}[\cdot]$ and $\mathcal{E}[\cdot]$ shown as

$$V_{S_t}(s_t) \propto \mathcal{E}\left[Q_{(S_t,A_t)^{(1)}}(s_t, a_t)\right]$$
$$Q_{(S_t,A_t)^{(1)}}(s_t, a_t) \propto \ln p(a_t) + r(s_t, a_t) + \mathcal{M}\left[V_{S_{t+1}}(s_{t+1})\right]$$
$$= R(s_t, a_t) + Q_{(S_t,A_t)^{(4)}}(s_t, a_t)$$

where $R(s_t, a_t) = \ln p(a_t) + r(s_t, a_t)$. Although, in the classic sum-product algorithm, these blocks correspond to a marginalization process, it is still possible to demonstrate that the simple reassignment of different procedures to them allows one to obtain different types of algorithms within the same model [25]. **Supplementary Appendix S1** presents various algorithms that can be used simply by modifying the function within the previous blocks, and which, therefore, show the generality of this framework, while **Table 1** summarizes the related equations by setting $Q(s_t, a_t) = Q_{(S_t,A_t)^{(1)}}(s_t, a_t)$ and $V(s_t) = V_{S_t}(s_t)$. It should also be noted that, for all the algorithms presented in the **Supplementary Appendix**, the definition of the policy can always be described according to the V and Q functions, by setting

$$\pi^*(a_t|s_t) \propto \exp\left(Q_{(S_t,A_t)^{(1)}}(s_t, a_t) - V_{S_t}(s_t)\right)$$

We emphasize the ease with which these algorithms can be evaluated via FGrn, as they can be defined through a simple modification of the base blocks. The pseudocode presented in Algorithm 1 highlights this simplicity, using the generic transition blocks just defined (and illustrated in **Figure 2**) whose function depends on the chosen algorithm.

[2] Think, for example, of *a priori* knowledge about the initial state or even more about the initial action to be performed.

[3] The index i is used in general terms, since for each $i = 1, \ldots, 4$, the value of the product between forward and backward (referring to the different versions of the joint random variable in **Figure 2**) is always identical.

[4] This is consistent with the principle of optimality: given the current state, the remaining decisions must constitute an optimal policy. Consequently, it is not surprising that the backward messages have all the information to compute the optimal policy.

[5] From the definition provided, it is understood that in this case the functions will always assume negative values. However, this is not a limitation because one can always add a constant to make rewards nonnegative.

TABLE 1 | Summarized backup rules in log space.

	Q (s_t, a_t)	**V (s_t)**
Sum product	$R(s_t,a_t) + \ln\sum_{s_{t+1}} e^{\ln p(s_{t+1}\|s_t,a_t)+V(s_{t+1})}$	$\ln\sum_{a_t} e^{Q(s_t,a_t)}$
Max product	$R(s_t,a_t) + \max_{s_{t+1}}(\ln p(s_{t+1}\|s_t,a_t) + V(s_{t+1}))$	$\max_{a_t} Q(s_t,a_t)$
Sum/Max product ($\alpha \geq 1$)	$R(s_t,a_t) + \frac{1}{\alpha}\ln\sum_{s_{t+1}} e^{\alpha(\ln p(s_{t+1}\|s_t,a_t)+V(s_{t+1}))}$	$\frac{1}{\alpha}\ln\sum_{a_t} e^{\alpha Q(s_t,a_t)}$
DP	$R(s_t,a_t) + \sum_{s_{t+1}} p(s_{t+1}\|s_t,a_t)V(s_{t+1})$	$\max_{a_t} Q(s_t,a_t)$
Max-Rew/Ent ($\alpha > 0$)	$R(s_t,a_t) + \sum_{s_{t+1}} p(s_{t+1}\|s_t,a_t)V(s_{t+1})$	$\frac{1}{\alpha}\ln\sum_{a_t} e^{\alpha Q(s_t,a_t)}$
SoftDP ($\beta > 0$)	$R(s_t,a_t) + \sum_{s_{t+1}} p(s_{t+1}\|s_t,a_t)V(s_{t+1})$	$\frac{\sum_{a_t} Q(s_t,a_t)e^{\beta Q(s_t,a_t)}}{\sum_{a'_t} e^{\beta Q(s_t,a'_t)}}$

Algorithm 1. Algorithm for the generic V-function

```
Data: $V_1(s)$, the V-function corresponding to the starting policy (equal to $0\ \forall s \in \mathcal{S}$ if there is
      no starting policy)
Result: $V_0(s)$, the V-function corresponding to the chosen algorithm
$\epsilon \leftarrow$ very small value, typically in the order of $10^{-5}$;
repeat
    $V_0(s) \leftarrow V_1(s)$;
    $Q(s,a) \leftarrow \ln p(a) + r(s,a) + \mathcal{M}[V_0(s)]$;
    $V_1(s) \leftarrow \mathcal{E}[Q(s_t,a_t)]$;
until $V_1(s) - V_0(s) > \epsilon$;
```

3 THE MULTI-AGENT SCENARIO

As stated above, when we have a problem with more than one agent, the computation of the value function collides with the complexity of a changing environment. More specifically, if all agents move around seeking their goals, the availability of states changes continuously and in principle the value functions have to be recalculated for each agent at every time step. Therefore, the problem is no longer manageable as before, and in general it must be completely reformulated to be tractable in many problem contexts.

The theoretical framework to describe a MAS is the *Markov game* (MG) [26, 27], that generalizes the MDP in the presence of multiple agents. Differently from the single agent MDP, in the multiple agent context, the transition probability function and the rewards depend on the joint action $\boldsymbol{A}_t \in \boldsymbol{\mathcal{A}}$, where $\boldsymbol{\mathcal{A}} = \mathcal{A}_1 \times \mathcal{A}_2 \ldots \times \mathcal{A}_n$ with n agents. At each time step t, the ith agent selects an action from its own action space $\mathcal{A}_i$ (simultaneously with other agents) and the system evolves following the transition probability function $P\colon \mathcal{S} \times \boldsymbol{\mathcal{A}} \to [0,1]$ and generates the reward $R_i\colon \mathcal{S} \times \boldsymbol{\mathcal{A}} \times \mathcal{S} \to \mathcal{R} \subset \mathbb{R}$. Consequently, the value function will not only depend on the policy of the single ith agent, but also on all other agents [10, 11]. In other words, considering the general case of an infinite horizon discounted version of the stochastic path planning problem, with a reward function that depends on current state-action pair as well as the next state, the value function for the ith agent will be

$$V^{(i)}_{\pi_i,\boldsymbol{\pi}_{-i}}(s) = \mathbb{E}_{s_{t+1}\sim P,a_t\sim\boldsymbol{\pi}}\left[\sum_{t=0}^{\infty}\gamma^t R_i(s_t,a_t,s_{t+1})|s = s_0\right] \quad (1)$$

where $a_t \in \boldsymbol{A}_t$, $\boldsymbol{\pi} = \{\pi_1, \ldots, \pi_n\}$ is the joint policy, $\boldsymbol{\pi}_{-i} = \{\pi_1, \ldots, \pi_{i-1}, \pi_{i+1}, \pi_n\}$ is the policy of all other agents except the ith and $\gamma \in [0, 1]$ is a discount rate.

The above problem, when the state space and the number of agents grow, becomes very quickly intractable. Therefore, we have to resort to simplifications, such as avoiding the need to consider the global joint spaces, and adopting simplifying distributed strategies that are sufficiently general to be applicable to practical scenarios. We focus here on a path planning problem for multiple agents that act sequentially, are non-competitive, share centralized information and are organized in a hierarchical sequence. Within these constraints, in fact, we will see how it is possible to use the versatility of the FGrn formulation by leveraging just one pre-calculated value function for each goal.

3.1 The Environment

Consider a scenario with n agents moving around a map with a given reward function $r(s_t)$ that depends only on the state. The rewards may represent preferred areas, such as sidewalks for pedestrians, streets for cars, bike routes for bicycles, or related to the traffic/weather conditions for ships and aircraft, etc. Therefore, at each time step, the overall action undertaken by the system is comprised of the n components

$$a_t = \left(a_t^{(1)}, \ldots, a_t^{(n)}\right)$$

where $a_t^{(i)}$ represents the action performed by the ith agent at time step t. The map is a discrete rectangular grid of dimensions $N \times M$ that defines the state space $\mathcal{S}$, in the sense that each free cell of the grid determines a state reachable by the agents. We assume that both the map and the reward function linked to the various states are the same for each agent, but that each agent aims to reach its own goal (some goals may coincide).

The objectives of each agent may represent points of interest in a real map, [6] and the existence of different rewards in particular areas of the map may correspond to preference for movement through these areas. The ultimate goal is to ensure that each agent reaches its target by accumulating the maximum possible reward, despite the presence of other agents. We assume that every action

[6] For example, the targets could be gas stations or ports (in a maritime scenario), whose presence on the map is known regardless of the agents.

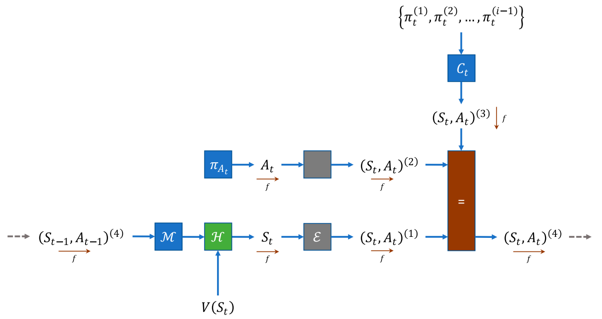


FIGURE 3 | Modified version of FGrn for the Multi-Agent forward step. Each agent will have its own factorial graph, in which information from previous agents (within the scheduling process) modifies the C_t variable. The algorithm block $\mathcal{H}$, which analyzes the value function to block the propagation of superfluous projections is shown in green.

towards an obstacle, the edge of the map or another agent, constrains the agent to remain in the same state (reflection), setting a reward that is never positive (in our formulation the rewards are all negative, except on the goal, where it is null). Furthermore, it is assumed that in the time step following the achievement of the objective, the agent no longer occupies any position on the map (it disappears). This ensures that arriving at the destination does not block the subsequent passage of other agents through that state, which could otherwise make the problem unsolvable.

3.2 The Forward Propagation

This MAS problem is ideally suited for approximate solution using FGrn, performing just some small changes similar to those introduced previously for the various objective functions. In particular, we will show how each agent will be able to perform its own inference on its particular FGrn (taking into account the target and the presence of other agents) simply by establishing an appropriate forward message propagation process. To do this, first, it is assumed that each agent follows a strict scheduling protocol, established in advance, to choose the action to perform. To fix the ideas, the agents are numbered in order of priority from 1 to n. Therefore, the ith agent will be allowed to perform its action at time t only after all the previous agents (from 1 to $i - 1$) have performed their tth step. In this way, similarly to what [12] proposed, the time step t is decomposed into n different sub-steps, relating to the n agents present on the scene. Since each agent's information is assumed to be shared with every other agent, and each agent is assumed to have access to this centralized information, the use of a scheduling protocol provides the next agent with the future policy of the agents who will move first, allowing it to organize its steps in relation to them. The idea is akin to Gauss-Seidel approach to solving linear equations and to Gibbs sampling.

Integrating this information into the FGrn is extremely simple. By looking at **Figure 3**, it is sufficient to make block C_t also dependent on the optimal trajectory at time t that the previous agents have calculated (*calculated but not yet performed!*) for themselves. In other words, at each time step, block C_t provides to each agent i a null value (in probabilistic space) for those states that are supposed to be occupied by the other agents. In this way, the ith agent will be constrained to reach its goal avoiding such states. Focusing on a specific agent i (dropping the index for notational simplicity), *a priori* knowledge on the initial state $S_1 = \hat{s}_1$ can be injected through a delta function [7] in the relative probabilistic forward message

$$f_{S_1}(s_1) = \delta(s_1 - \hat{s}_1)$$

and assuming that $\mathcal{M}$ blocks in FGrn perform the function

$$f_{S_{t+1}}(s_{t+1}) = \max_{s_t, a_t} p(s_{t+1}|s_t, a_t) f_{(S_t, A_t)^{(4)}}(s_t, a_t)$$

with

$$f_{(S_t, A_t)^{(4)}}(s_t, a_t) = f_{(S_t, A_t)^{(1)}}(s_t, a_t) f_{(S_t, A_t)^{(2)}}(s_t, a_t) c(s_t, a_t)$$

the FGrn autonomously modifies its behavior by carrying out a process of pure diffusion which determines the best possible trajectory to reach a given state in a finite number of steps.

Note that this propagation process leads to optimality only if we are interested in evaluating the minimum-time path [28]. [8] Although the reward is accumulated via c (s_t, a_t), the forward process totally ignores it, not being able to consider other non-minimal paths that could accumulate larger rewards. In fact, exhaustively enumerating all the alternatives may become unmanageable, unless we are driven by another process. In other words, this propagation process alone does not guarantee that the first accumulated value with which a goal state is reached, is the best possible. Further exploration, without being aware of the time required to obtain the path of maximum reward, may force us to run the algorithm for a very large number of steps (with increasing computational costs). However, as mentioned above, the reference scenario involves goals that are independent and are known *a priori*. This means that (through any of the algorithms discussed in **Supplementary Appendix S1**) it is possible to calculate the value function in advance for each goal. Note that this offline calculation is independent of the location/presence of the agents in the MAS scenario and therefore could not be used directly to determine the overall action a_t of the system. The following lemma is useful to claim optimality.

LEMMA 1. The value function computed by excluding the agents from the scene represents an upper-bound (in terms of cumulative reward) for a given state.

[7] We refer to the Kronecker Delta $\delta(x)$, which is equal to 1 if $x = 0$ and is zero otherwise.

[8] The very concept of "time" in this case can be slightly misleading. The reward function linked to individual states can in fact represent the time needed to travel in those states (for example due to traffic, or adverse weather conditions). In this case, the number of steps performed by the algorithm does not actually represent the "time" to reach a certain state. We emphasize that in the presence of a reward/cost function, the objective is not to reach a given state in the fewest possible steps, but to obtain the highest/lowest achievable reward/cost.

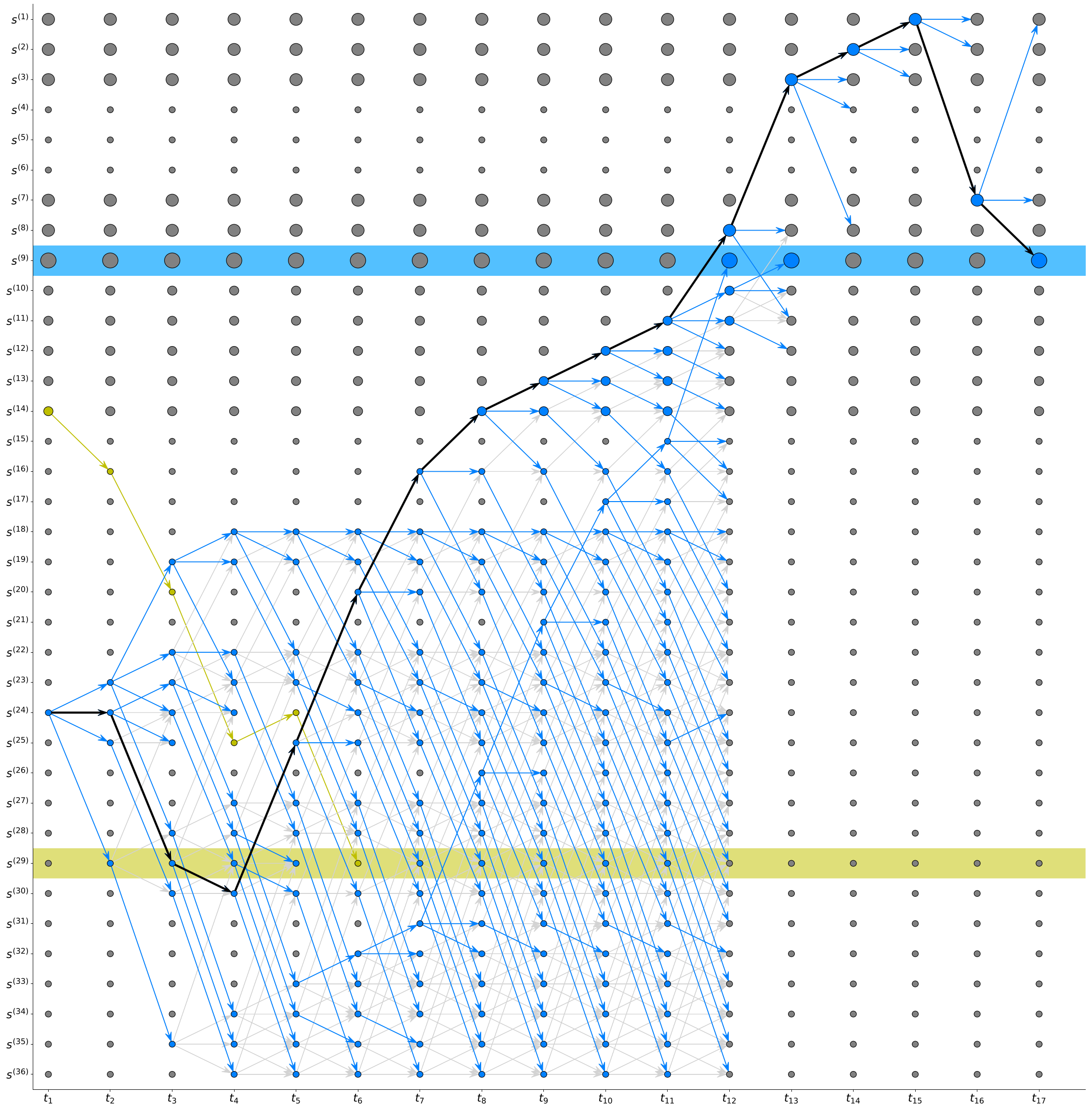


FIGURE 4 | Trellis related to the analysis of the optimal path for the blue agent based on the preliminary presence of the yellow agent. The highest reward map states are represented by larger circles. The blue arrows represent the forward propagated projections, while the light gray ones denote the other discarded possibilities. The optimal path determined by forward propagation within the FGrn is represented in black.

Proof. : The demonstration is trivial as other agents (being moldable as dynamic obstacles) can only reduce the value obtained from the value function for the agent, by hindering a valid passage through their presence and forcing the agent to traverse a sub-optimal path.

Knowing the value function corresponding to the objective of a particular agent, it is therefore possible to limit the $f_{S_{t+1}}(s_{t+1})$ only to the values that actually have the possibility of reaching the goal with an accumulated reward larger than the current value. In other words, at each time step t, after the diffusion arrives on the goal, it is possible to add the value function to the $\ln f_{S_{t+1}}(s_{t+1})$ to compare the active states with the value currently obtained on the goal, eliminating all those paths that could not in any way reach the objective with a higher cumulative reward (since, as mentioned, this represents an upper-bound for every possible state). The only projections left will represent possible steps towards better solutions and will continue to propagate to determine if their dynamics can actually enable the discovery of a better path. To highlight this step, in **Figure 3**, this addition is shown using a $\mathcal{H}$ block (placed separately from the $\mathcal{M}$ block only to facilitate understanding), which takes the pre-computed V-function as input and performs the three algorithmic operations just described (addition, comparison and elimination). The

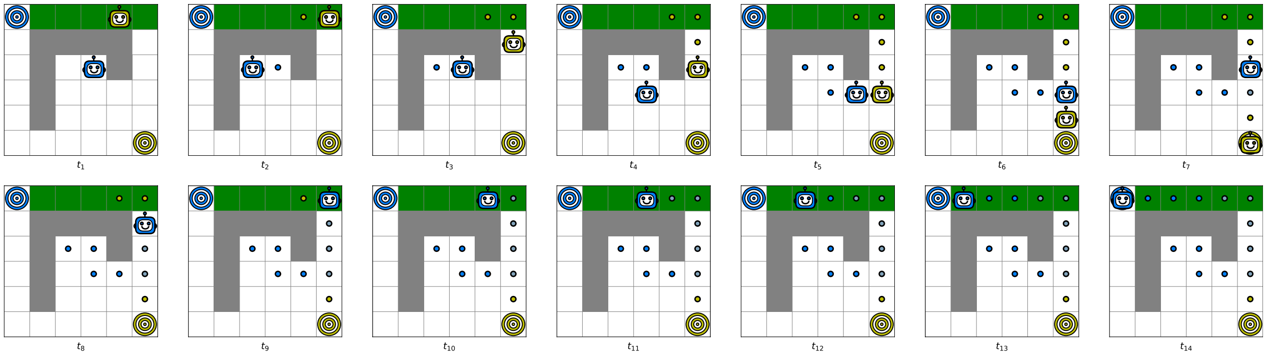


FIGURE 5 | Representation of the movement of two agents on a small map in a deterministic environment. Both agents have only four possible actions {*up, down, left, right*}. The value function is calculated through the DP and both agents have their own goals.

pseudocode of the forward process for the single *i*th agent is shown in Algorithm 2.

Algorithm 2. Algorithm for the forward propagation in a MAS context using the V-functions

```
Data: The positions $T$ of the other $i-1$ agents based on their optimal trajectories
Result: The optimal trajectory for the $i^{\text{th}}$ agent
$t \leftarrow 1$;
$g_v \leftarrow -\infty$;
$s_i \leftarrow$ the position of the agent;
$s_g \leftarrow$ the position of the agent's goal;
$f_{S_1}(s_1) \leftarrow \delta(s_1 - s_i)$;
while $f_{S_t}(s_t) \neq 0$ do
    $f_{A_t}(a_t) \leftarrow p(a_t)$;                    /* the prior on the actions */
    $c(s_t, a_t) \leftarrow \exp(r(s_t, a_t))$;
    if $\exists T_t$ then
        set to 0 all values of $c(s_t, a_t)$ for which $T_t$ is not zero;
    end
    $f_{(S_t,A_t)^{(1)}}(s_t, a_t) \leftarrow f_{S_t}(s_t)\mathcal{U}(a_t)$;
    $f_{(S_t,A_t)^{(2)}}(s_t, a_t) \leftarrow f_{A_t}(a_t)\mathcal{U}(s_t)$;
    $f_{(S_t,A_t)^{(4)}}(s_t, a_t) \leftarrow f_{(S_t,A_t)^{(1)}}(s_t, a_t) f_{(S_t,A_t)^{(2)}}(s_t, a_t) c(s_t, a_t)$;
    $f_{S_{t+1}}(s_{t+1}) \leftarrow \max_{s_t,a_t} p(s_{t+1}|s_t, a_t) f_{(S_t,A_t)^{(4)}}(s_t, a_t)$;
    if $f_{S_{t+1}}(s_g) \neq 0$ then
        $g_v \leftarrow \ln f_{S_{t+1}}(s_g)$;
    end
    for $s \in \mathcal{S}$ do
        if $g_v \geq \ln f_{S_{t+1}}(s) + \max_{s_{t+1} \textit{ reachable from } s} V_{S_{t+1}}(s_{t+1})$ then
            $f_{S_{t+1}}(s) \leftarrow 0$;
        end
    end
    $t \leftarrow t + 1$;
end
Reconstruct the optimal trajectory backwards from the last time step the goal was reached
```

A better understanding of the whole forward propagation process is perhaps achievable by considering the trellis of **Figure 4**, which shows the forward propagation flow for a "blue" agent given the trajectory decided by the "yellow" agent. The trellis shows the various steps on the abscissa (from t_1 to t_{17}) and the accessible states of the map on the ordinate (from $s^{(1)}$ to $s^{(36)}$), highlighting the states related to the two different objectives through rectangles of the respective colors. The blue agent propagates its projections to various time steps taking into account the possible actions, avoiding the states already occupied and considering only the paths with the maximum cumulative reward (the other paths not chosen are graphically represented in light gray).

From the moment a path to the goal is found (t_{12}), the $\mathcal{H}$ block of **Figure 3** performs its tasks by blocking the propagation of projections that have no chance of improving the value of the final cumulative reward (all gray circles reached by an arrow at t_{12} and in subsequent time steps). [9] This also means that if another path is able to reach the goal again, then it will certainly be better than the previous one. In other words, when the control block clears all projections on the map, then the last path that was able to get to the goal is chosen as the preferred trajectory for the agent. In the example of **Figure 4**, the blue agent reaches the goal again at t_{13} and the cumulative reward is higher than the one obtained at t_{12}, but the projections can continue through the state $s^{(3)}$ that allows us to reach the goal at time step t_{17}. Since, at that time step, all the other projections have been blocked, the path (in black) from $s^{(24)}$ at t_1 to $s^{(9)}$ at t_{17} is optimal.

4 SIMULATIONS

If the environment is assumed to be fully deterministic, each agent will have to calculate its optimal trajectory only once and, when all agents have performed the calculation, the movements can be performed simultaneously. In such circumstances, a good scheduling protocol can be obtained by sorting agents according to

$$\max_{s \in N(s_1) \subseteq \mathcal{S} \setminus \hat{\mathcal{S}}} V(s)$$

where $N(s_1)$ is the neighborhood of s_1 given the feasible actions of the agent, and $\hat{S}$ is the set of states relative to the initial positions of all agents. In this way, the agents closest to their respective goals will move independently from the others, arriving first and being irrelevant for the subsequent steps necessary for the other agents. In the various simulations conducted in deterministic environments, this choice has always proved successful, reaching

[9] Note that the $\mathcal{H}$ block actually exists at each step t of the process described, but since the deletion of projections occurs by comparing the sum with the value currently present on the target (and since, at the beginning, this value is considered infinitely negative), the block will practically never delete any projections until the target is achieved for the first time.

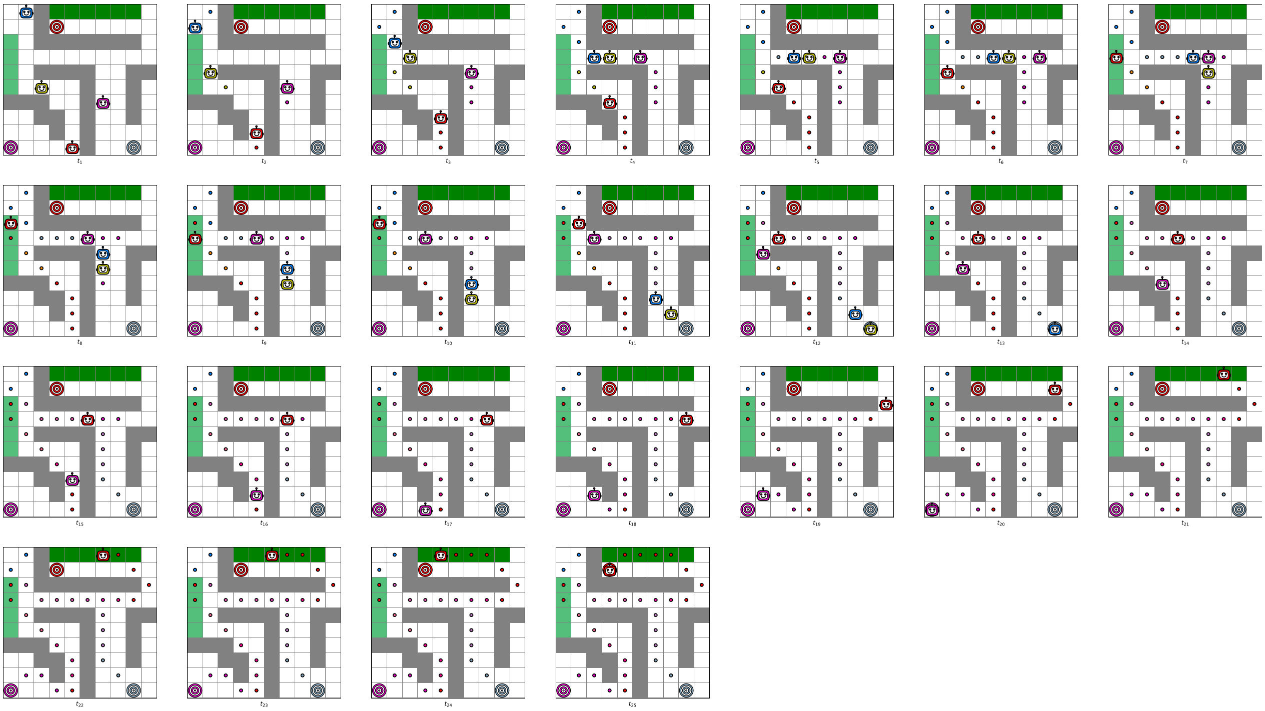


FIGURE 6 | Representation of the movement of four agents in a deterministic environment with eight possible actions {*top-left, up, top-right, left, right, down-left, down, down-right*}. The value function is calculated through the DP and the blue and yellow agents share the same goal while purple and red agents have their own goals.

the maximum total reward for all agents compared to any other possible scheduling sequence. However, the forward procedure guarantees the optimal solution for the particular scheduling sequence chosen. In fact, at each time step, the algorithm considers the maximum for each possible state-action pair, blocking all those paths that (even at their maximum) would never be able to reach the goal. In practice, the upper bound constituted by the value function allows us to avoid considering every possible path, but guarantees us that all the excluded paths are certainly worse. All the paths that the algorithm considers are therefore certainly the best possible, and for this reason the optimal path for the agent in that given scheduling sequence is guaranteed. The overall optimality, in relation to the sum of all the cumulative rewards obtained by each agent, is, however, strongly linked to the chosen scheduling procedure, that can therefore lead to non-optimal solutions, if not appropriately chosen. [10] A simple simulation with two agents is shown in **Figure 5**, where it is assumed that the action space is composed of only four elements $\mathcal{A} = \{up, down, left, right\}$ and that the reward is always equal to −10 except on green states, where it is equal to −1. It can be seen that the blue agent chooses the longest path which, however, represents the one with the higher cumulative reward, due to the presence of a higher reward area. Despite this, from the beginning, the yellow agent blocks the path of the blue one, who is therefore forced to move around until it becomes free (t_5). After this time step, the two agents can reach their respective goals without any interaction. [11] A further example, more complex than the previous one, is shown in **Figure 6**. In this case, the action space is composed of eight elements $\mathcal{A} = \{top\text{-}left, up, top\text{-}right, left, right, down\text{-}left, down, down\text{-}right\}$ with $n = 4$ agents present on the map. It is worth noting how the red agent at t_7, t_8, t_9, t_{10} wanders around in the region with a high reward in order to wait for the purple agent to go through the tunnel and accumulate a higher reward. In the deterministic case, the computational cost of the online procedure is extremely low, as it can be evaluated in $\mathcal{O}(n \log NM)$ in the worst case.

General behavior does not change in the case in which a non-deterministic transition dynamics are assumed, i.e., assuming the agents to be in an environment in which every action does not necessarily lead to the state towards which the action points; providing a certain (lesser) probability of ending up in a different state among those admissible (as if some other action had actually been performed). [12] What changes, however, is the total

[10] The search for an optimal scheduling choice is under consideration and will be published elsewhere.

[11] It should be noted that a different scheduling choice would lead to the yellow agent being blocked by the blue agent, obtaining an overall reward for both agents lower than that obtained.

[12] To make the concept realistic, one can imagine an environment with strong winds or with large waves. In general, this reference scenario aims to perform the control even in the presence of elements that prevent an exact knowledge of the future state following the chosen action.

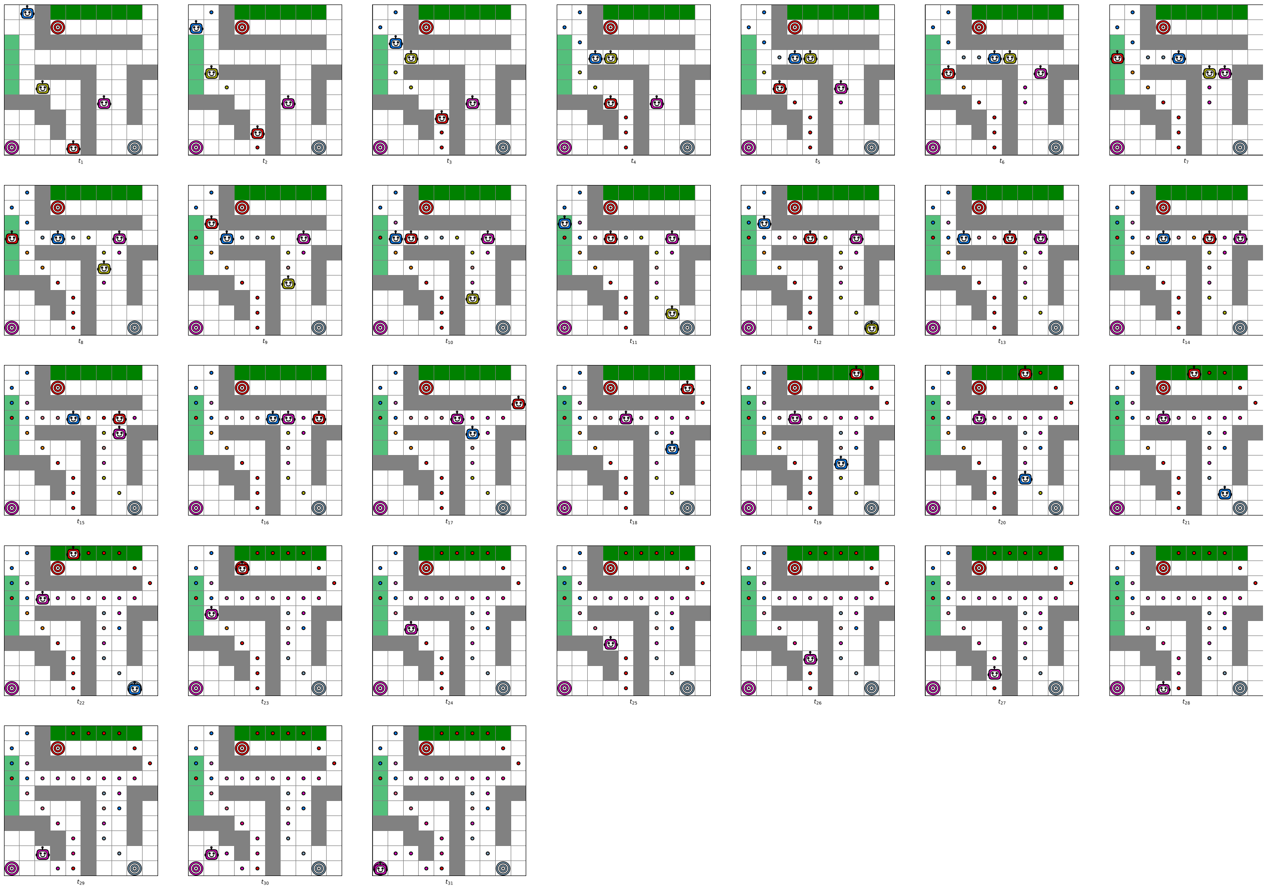


FIGURE 7 | Representation of the movement of four agents in stochastic environment with a 5% chance of error on neighboring actions and eight possible actions {*top-left, up, top-right, left, right, down-left, down, down-right*}.

computational cost. Each time the agent takes a step, in fact, if the step does not fall within the optimal trajectory calculated previously, then it will be forced to recalculate its trajectory again to pass it to other subsequent agents.

It must be considered, however, that each calculation has a low computational cost anyway (definitely much lower than the total recalculation of the value function) and that, at each step, the agents get closer and closer to their goals (making the calculation faster, because it is always less likely to find better alternative routes). These considerations are obviously strictly linked to the uncertainty present in the system. To clarify these observations, in **Figure 7** the same diagram of the deterministic environment of **Figure 6** is presented, with the same four agents positioned within the same map. This time, however, it is assumed to use an action tensor that results in a random error of 5% equally distributed on adjacent actions (i.e., close to the action contemplated). A graphical representation of the action tensor, considering the agent positioned at the center of each grid cell, is shown in **Figure 8**. A comparison with the deterministic case of **Figure 6** allows us to understand the behaviors stemming exclusively from the stochasticity of the environment. For example, it can be observed how the blue agent is pushed in the opposite direction to the action taken (from time step 7–11), but

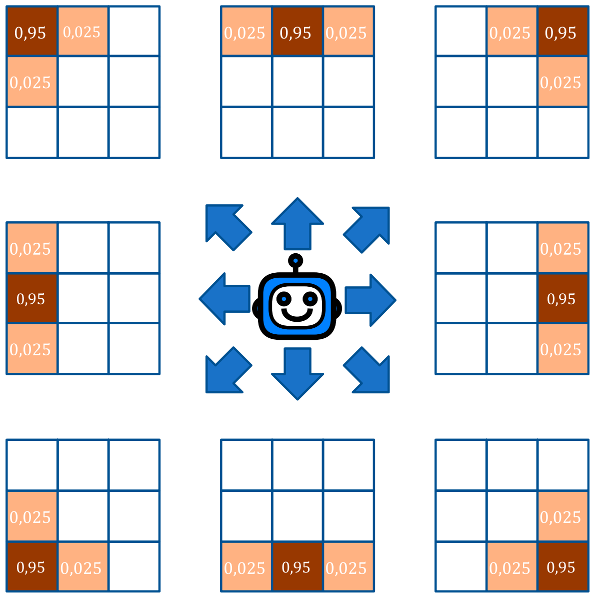


FIGURE 8 | Action probability tensor with an error probability of 5% on neighboring actions.

nevertheless correctly recalculates its trajectory to allow others to take their paths based on the mistakes made. Note that, in the stochastic case, the optimality on the single execution cannot be guaranteed, precisely because of the intrinsic stochasticity of the environment. However, this argument is general and is valid for any algorithm in a stochastic environment. Furthermore, it must be said that if it were possible to regenerate an optimal scheduling sequence at each variation with respect to the previously calculated trajectory, it could be stated that on multiple executions (since the algorithm maximizes the likelihood and since each sub-trajectory would be optimal), the behavior tends asymptotically to the optimum.

5 CONCLUSION

We have shown how it is possible to unify probabilistic inference and dynamic programming within an FGrn through specific message composition rules. The proposed framework allows various classical algorithms (sum-product, max-product, dynamic programming and based on mixed reward/entropy criteria), also by expanding the algorithmic design options (through generalized versions), only by modifying the functions within the individual blocks.

Using a path planning problem context, we have also shown how this framework proves to be decidedly flexible, and how it is possible to use it even in the multi-agent case. Moreover, the forward procedure turns out to be very fast in calculating the optimal trajectory subject to an agent scheduling protocol. The use of the value function as upper bound allows, in fact, to limit the propagation of the projections at the various time steps, accelerating and guaranteeing the achievement of the optimal solution in deterministic cases (again subject to a specified agent scheduling protocol). The proposed simulations have shown how the solution is effective even in a stochastic environment, where the optimal solution is not reachable on a single example due to the intrinsic variability of the environment.

We believe that the work presented here provides a scientifically rigorous algorithmic framework for proactive agent autonomy. The factor graph-based message propagation approach to MAS will enable us to investigate the interdependencies among the key elements of a hybrid team, such as goals, changing mission environment, assets and threats/obstacles/constraints. We believe that the interactive optimization algorithms based on this approach should provide the tools for producing intelligent courses of action that are congruent with and overcome bounded rationality and cognitive biases inherent in human decision-making.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/**Supplementary Material**, further inquiries can be directed to the corresponding author.

AUTHOR CONTRIBUTIONS

GD, conceptualization, methodology, writing-original draft, writing-review and editing, visualization, and software. AB, conceptualization, methodology, writing-original draft, writing-review and editing, and visualization. GF and FV, visualization, software, writing-review and editing. KP and FP, conceptualization, methodology, visualization, supervision, writing-review and editing, and funding acquisition.

FUNDING

This work was supported in part by POR CAMPANIA FESR 2014/2020, ITS for Logistics, awarded to CNIT (Consorzio Nazionale Interuniversitario per le Telecomunicazioni). Research of Pattipati was supported in part by the US Office of Naval Research and US Naval Research Laboratory under Grants #N00014-18-1-1238, #N00014-21-1-2187, #N00173-16-1-G905 and #HPCM034125HQU.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fphy.2022.944157/full#supplementary-material

REFERENCES

1. Weiss G. *Multiagent Systems*. 2nd ed. Cambridge: MIT Press (2016).
2. Farinelli A, Iocchi L, Nardi D. Multirobot Systems: a Classification Focused on Coordination. *IEEE Trans Syst Man Cybern B Cybern* (2004) 34:2015–28. doi:10.1109/tsmcb.2004.832155
3. Bellman RE. *Dynamic Programming*. New York: Dover (2003).
4. Bertsekas DP. Dynamic Programming And Optimal Control *(Athena)* (2017). Cambridge: Athena.
5. Szer D, Charpillet F. Point-Based Dynamic Programming for Dec-Pomdps. *Association for the Advancement of Artificial Intelligence* (2006) 6:1233–8.
6. Bertsekas D. Multiagent Value Iteration Algorithms in Dynamic Programming and Reinforcement Learning. *Results in Control and Optimization* (2020) 1: 1–10. doi:10.1016/j.rico.2020.100003
7. Sutton RS, Barto AG. *Reinforcement Learning: An Introduction*. Cambridge: The MIT Press (2018).
8. Bertsekas DP. Reinforcement Learning And Optimal Control (2019). Cambridge: Athena
9. Busoniu L, Babuska R, De Schutter B. A Comprehensive Survey of Multiagent Reinforcement Learning. *IEEE Trans Syst Man Cybern C* (2008) 38:156–72. doi:10.1109/tsmcc.2007.913919
10. Nowé A, Vrancx P, De Hauwere Y-M. Game Theory and Multi-Agent Reinforcement Learning. In: M Wiering M van Otterlo, editors. *Reinforcement Learning: State-Of-The-Art*. Berlin, Heidelberg: Springer (2012). p. 441–70. doi:10.1007/978-3-642-27645-3_14
11. Yang Y, Wang J. An Overview of Multi-Agent Reinforcement Learning from Game Theoretical Perspective. arXiv (2020). Available at: https://arxiv.org/abs/2011.00583.
12. Bertsekas D. Multiagent Reinforcement Learning: Rollout and Policy Iteration. *Ieee/caa J Autom Sinica* (2021) 8:249–72. doi:10.1109/jas.2021.1003814

13. Lejeune E, Sarkar S. *Survey of the Multi-Agent Pathfinding Solutions* (2021). doi:10.13140/RG.2.2.14030.28486
14. Palmieri FAN. A Comparison of Algorithms for Learning Hidden Variables in Bayesian Factor Graphs in Reduced normal Form. *IEEE Trans Neural Netw Learn Syst.* (2016) 27:2242–55. doi:10.1109/tnnls.2015.2477379
15. Di Gennaro G, Buonanno A, Palmieri FAN. Optimized Realization of Bayesian Networks in Reduced normal Form Using Latent Variable Model. *Soft Comput* (2021) 10:1–12. doi:10.1007/s00500-021-05642-3
16. Bellman R. A Markovian Decision Process. *Indiana Univ Math J* (1957) 6: 679–84. doi:10.1512/iumj.1957.6.56038
17. Puterman ML. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. New York: Wiley (2005).
18. Kappen HJ, Gómez V, Opper M. Optimal Control as a Graphical Model Inference Problem. *Mach Learn* (2012) 87:159–82. doi:10.1007/s10994-012-5278-7
19. Levine S. Reinforcement Learning and Control as Probabilistic Inference: Tutorial and Review. arXiv (2018). Available at: https://arxiv.org/abs/1805.00909
20. O'Donoghue B, Osband I, Ionescu C. Making Sense of Reinforcement Learning and Probabilistic Inference. In: 8th International Conference on Learning Representations (ICLR) (OpenReview.net) (2020).
21. Forney GD. Codes on Graphs: normal Realizations. *IEEE Trans Inform Theor* (2001) 47:520–48. doi:10.1109/18.910573
22. Koller D, Friedman N. *Probabilistic Graphical Models: Principles and Techniques*. Cambridge: The MIT Press (2009).
23. Loeliger H. An Introduction to Factor Graphs. *IEEE Signal Process Mag* (2004) 21:28–41. doi:10.1109/msp.2004.1267047
24. Barber D. *Bayesian Reasoning and Machine Learning*. Cambridge: Cambridge University Press (2012).
25. Palmieri FAN, Pattipati KR, Gennaro GD, Fioretti G, Verolla F, Buonanno A. A Unifying View of Estimation and Control Using Belief Propagation with Application to Path Planning. *IEEE Access* (2022) 10:15193–216. doi:10.1109/access.2022.3148127
26. Shapley LS. Stochastic Games. *Proc Natl Acad Sci* (1953) 39:1095–100. doi:10.1073/pnas.39.10.1953
27. Littman ML. Markov Games as a Framework for Multi-Agent Reinforcement Learning. In: *Machine Learning Proceedings 1994*. San Francisco (CA) (1994). p. 157–63. doi:10.1016/b978-1-55860-335-6.50027-1
28. Palmieri FAN, Pattipati KR, Fioretti G, Gennaro GD, Buonanno A. Path Planning Using Probability Tensor Flows. *IEEE Aerosp Electron Syst Mag* (2021) 36:34–45. doi:10.1109/maes.2020.3032069
29. Loeliger H-A, Dauwels J, Hu J, Korl S, Ping L, Kschischang FR. The Factor Graph Approach to Model-Based Signal Processing. *Proc IEEE* (2007) 95: 1295–322. doi:10.1109/jproc.2007.896497
30. Ziebart BD, Bagnell JA, Dey AK. Modeling Interaction via the Principle of Maximum Causal Entropy. In: Proceedings of the 27th International Conference on International Conference on Machine Learning. Madison, WI, USA: Omnipress (2010). p. 1255–62.

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

TYPE Original Research
PUBLISHED 22 July 2022
DOI 10.3389/fphy.2022.942245

Check for updates

OPEN ACCESS

EDITED BY
William Frere Lawless,
Paine College, United States

REVIEWED BY
Dioneia Motta Monte-Serrat,
University of São Paulo, Brazil
Victoria Hailey,
VHG-The Victoria Hailey Group Corporation, Canada
David Short,
BAE Systems, United Kingdom

*CORRESPONDENCE
Tony Gillespie,
Anthony.gillespie@ucl.ac.uk

SPECIALTY SECTION
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

RECEIVED 12 May 2022
ACCEPTED 27 June 2022
PUBLISHED 22 July 2022

CITATION
Gillespie T (2022), Building trust and responsibility into autonomous human-machine teams.
Front. Phys. 10:942245.
doi: 10.3389/fphy.2022.942245

Building trust and responsibility into autonomous human-machine teams

Tony Gillespie*

Electronic and Electrical Engineering, UCL, London, United Kingdom

Harm can be caused to people and property by any highly-automated system, even with a human user, due to misuse or design; but which human has the legal liability for the consequences of the harm is not clear, or even which laws apply. The position is less clear for an interdependent Autonomous Human Machine Team System (A-HMT-S) which achieves its aim by reallocating tasks and resources between the human Team Leader and the Cyber Physical System (CPS). A-HMT-S are now feasible and may be the only solution for complex problems. However, legal authorities presume that humans are ultimately responsible for the actions of any automated system, including ones using Artificial Intelligence (AI) to replace human judgement. The concept of trust for an A-HMT-S using AI is examined in this paper with three critical questions being posed which must be addressed before an A-HMT-S can be trusted. A hierarchical system architecture is used to answer these questions, combined with a method to limit a node's behaviour, ensuring actions requiring human judgement are referred to the user. The underpinning issues requiring Research and Development (R&D) for A-HMT-S applications are identified and where legal input is required to minimize financial and legal risk for all stakeholders. This work takes a step towards addressing the problems of developing autonomy for interdependent human-machine teams and systems.

KEYWORDS

human machine team, trust, autonomy, legal liability, artificial intelligence, risk, interdependence

Introduction

Achieving Artificial Intelligence's (AI) full potential for any application will require considerable research and engineering effort [1]. New AI-engineering techniques will need to be developed, especially when AI-based systems interact with humans [2]. Technology has evolved to the point where Human Machine Teams (HMTs) can dynamically and automatically reallocate tasks between human and machine team members to optimise workloads and resource usage, an Autonomous Human Machine Team System (A-HMT-S). However, interdependence between team members with very different capabilities raises serious system challenges to ensure the safe, trusted transfer of authority between human and machine.

When the human user of a Cyber-Physical System (CPS) has given it an aim, and its subsequent actions are guided by AI, questions arise about the roles of the human, the AI, and that of the people responsible for its autonomous behaviour. Who was responsible for its actions and any harm caused by those actions? The legal position is evolving, with no clear consensus. Reference [3] covers the current legal position for AI and suggests likely developments.

The use of an A-HMT-S to achieve an aim implies complexity, requiring reasoning to achieve it. Although a team approach may be efficient, there are legal complications when the aim is to take an action, or to provide information for someone or something to take an action that could cause harm. Assignment of responsibly for the consequences of machine-made decisions is becoming an important issue now that CPS such as "autonomous" cars have already caused serious injury to humans. Even in this case, there is divergence between national jurisdictions [4].

Singapore is exploiting its unique geography and legal system to advance the use of Autonomous Vehicles (AVs) through road trials with close interaction between the government, regulators, and industry [5]. They expect to continue this collaboration as technology, public opinion, and law develop.

The United States government also see legal issues arising now and in the future. The US Department Of Transportation's latest autonomous vehicles guidance document [6] states that jurisdictional questions are likely to be raised by Automated-Driving-System (ADS) enabled vehicles which they need to address as a regulatory approach is developed.

The Chinese legal system may also need urgent revision to meet the needs of AVs [7].

The English and Scottish Law Commissions, on behalf of their governments, formally review important societal developments to provide a basis for new legislation. The final report [8] of their AV Project [9] concludes that the problem of assigning legal responsibility and hence liability for harm is unclear and, additionally, that this lack of clarity applies across all autonomous products. Their view is that using autonomy levels to describe a system is legally meaningless; an automated vehicle is either autonomous or it is not, with different laws applying in the two cases. AVs require a new regulatory authority, with responsibility and hence liability lying with the organizations responsible for the supply and maintenance of an automated driving system; in all other cases the driver is responsible. Data must also be recorded, stored, and provided for use in accident enquiries. Their recommendations directly affect all aspects of autonomous system design.

Analogous principles cover lethal autonomous weapon systems [10], so it can be assumed that most, if not all, A-HMT-Ss will provoke similar ones with responsibilities on all participants in the design cycle.

The legal views can be summarized in one system requirement which must be used in deriving more detailed system requirements:

> Responsibility for all decisions and actions of an A-HMT-S must be traceable by an enquiry to an identifiable person, or role-holder, in the organization using or supplying it.

The core problem with meeting this requirement for AI-based actions is their non-deterministic nature and consequent uncertainties in a system's behaviour. Considerable Research and Development (R&D) work will be needed to allow risk management of these legal issues in A-HMT-S lifecycles, as is the case with current safety-related systems. This paper identifies three key questions which are addressed, giving methods for acceptable risk management in meeting the requirement, and identifying the areas for R&D when AI is introduced into an interdependent A-HMT-S.

Assumptions and terminology

An A-HMT-S comprises at least one human and one or more CPS, with continual interaction between them, reallocating tasks as necessary. Only one human can be the Team Leader with responsibility for the actions of the A-HMT-S. Their interaction with the A-HMT-S is through the Human Machine Interface (HMI) which has an important place in an A-HMT-S as emphasised by [11, 12].

It is assumed that any A-HMT-S can cause unacceptable harm to a person or property if its behaviour is not controlled. This gives a requirement for trust which is defined as [13].

> The willingness of a party to be vulnerable to the actions of another party based on the expectation that the other will perform a particular action important to the trustor, irrespective of the ability to monitor or control that other party.

Trustworthiness, the property required to be trusted, is defined as [14]:

> The demonstrable likelihood that the system performs according to designed behavior under any set of conditions as evidenced by characteristics including, but not limited to, safety, security, privacy, reliability and resilience.

Dynamic human machine teaming and trust

The simplest non-adaptive HMT has a human using automated, deterministic subsystems to meet their aims by delegating tasks to single or multiple subsystems. The human issues instructions, updating them based on either their responses, sensor information or a change in aims, i.e. all adaption is by the human. Safety is assured by a combination of testing and mechanical, electrical or software limits. When the subsystem responses are deterministic, human users have a trusted mental model of the system and will accept

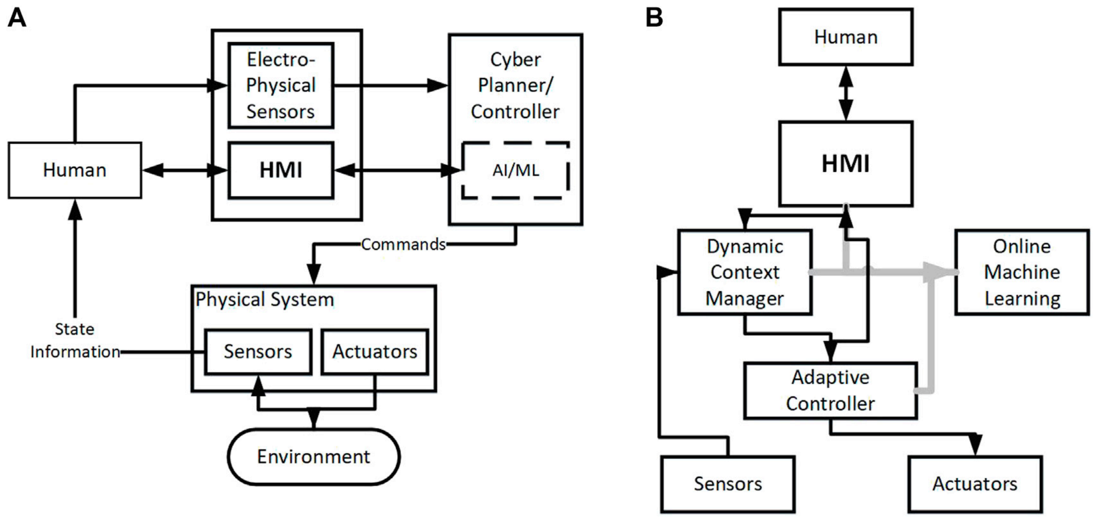


FIGURE 1
(A) An A-HMT-S with AI and ML embedded in the cyber planner/controller (Adapted from Madni & Madni 2018). **(B)** An A-HMT-S with its control elements drawing on results from separate on-line ML systems that are not in the control chain. (Adapted from Madni *et al.* 2018).

responsibility for the consequences of their instructions. It is assumed that the human is authorised to operate the system, indicating a level of trust in them by others, i.e., the HMT is trustworthy.

When resources and tasks need rapid, multiple reassignments, these must be automated and dynamic for optimum performance. “Optimum performance” will be system-dependent but must include ensuring that human workloads allow them to make considered decisions. The HMT is then an interdependent A-HMT-S, responding to external changes by internal reorganizations to meet its aims, with a human Team Leader.

Trust can be seen as a problem of ensuring that the Team Leader knows that the system is reliable, and well tested, with bounds to its action. It then follows that the trustworthiness of each element of the A-HMT-S is known. We take the view here that, in addition to the definition of trustworthy given in the Assumptions and Terminology section, human trust requires that the A-HMT-S responses can be understood and accepted as reasonable even if they are not necessarily the expected ones. The Team Leader is likely to develop trust in the CPS elements if they reliably achieve their aims but report back if there are problems.

The HMI provides the Team Leader with information about team status and task progress. The Team Leader will have a mental model of the A-HMT-S, with varying levels of detail and accuracy of its subsystems and resources, which provides expectations of system behaviour as conditions change. The control problem then becomes one of compatibility between the Team Leader’s expectations and the information presented by the HMI. The HMI is taken to be a control station with a pre-determined range of user controls and displays that change depending on predetermined variables. The variables will include the user’s workload and situational awareness as measured by the HMI, supplemented, if necessary, by other sensors. This implementation is an adaptive HMI as described by Blakeney [15]. Using the information provided by the HMI, the Team Leader decides if new system instructions are needed, checks that the system is trustworthy if changes are necessary, and issues the instructions through the HMI.

The subsystem implementing the instructions by making dynamic system control decisions has a crucial role. This role has been demonstrated for simulated environments using either a cyber planner/controller [11], or splitting it into a dynamic context manager and an adaptive controller [12] as shown in Figure 1. This shows that Machine Learning (ML) can be used in different places to support CPHS performance. However, ML is likely to introduce non-deterministic inputs into the A-HMT-S$'$ control system, giving the potential for instability. This makes it essential to identify the role ML plays in control decisions and which node has the authorisation to initiate the consequent actions. Unless this is known, the Team Leader cannot justify the system’s decisions or be responsible for its consequent actions.

The preceding arguments show three issues when ML plays a role in decisions and actions in an A-HMT-S:

> Issue 1. An adaptive HMI which learns and adapts its outputs, based on its own model of the Team Leader, must still present the essential information for the human to accept responsibility for the actions of the A-HMT-S;
> Issue 2. Automation in the cyber planner/controller means that the Team Leader is not choosing the subsystems for a task at any given time. The introduction of ML into this choice will

lead to the system changing task and resource allocation according to circumstances as judged by a non-deterministic subsystem in the control chain;

Issue 3. ML in subsystems may change their behaviour due to its own understanding of circumstances, not necessarily that of the higher-level systems. The higher levels could request an action from a subsystem whose behaviour has changed and will respond in a manner not expected by a higher level. Both will then try to understand the situation and remedy it but, without close feedback, confusion is highly likely. This is an example of the wicked problem. a well-known one in systems engineering [16].

These issues must be addressed before a human can trust an A-HMT-S and accept responsibility for its actions.

Trust in any AI system depends on many characteristics. Alix et al. [17] give: reliability; robustness; resistance to attack; transparency; predictability; data security; and protection against incorrect use. They then propose that an AI-based system has to implement the three following features:

- Validity *to make sure that the AI-based system must do what it is supposed to do, all what it is supposed to do, and only what is supposed to do. It is crucial to deliver reliable, robust, safe and secure critical systems.*
- Explainability *to make the team leader confident with the AI based system through human-oriented and understandable causal justifications of the AI results. Indeed, the end-team leaders' trust cannot be neglected to adopt AI-based systems.*
- Accountability *in respect of ethical standards and of lawful and fair behaviours.*

These assume that an AI system will act on its decisions without human intervention, implying that the Team Leader is comfortable taking responsibility for all its actions, a very high threshold for trust. The threshold can be lowered if the system makes effective predictions about the consequences of its actions but if there is doubt about the effect of the action, or if it will exceed a predetermined limit, then the action and its justification is referred to the human Team Leader first. This behaviour is analogous to a member of an all-human team referring to the team leader for confirmation of an action or requesting an alternative course of action.

Summarizing, an A-HMT-S must be trustworthy by design and only take actions that are limited and authorised through its organizational structure, with reference to the Team Leader if necessary. The important questions that must be answered before an A-HMT-S can be trusted and used are:

Q1. Can a dynamic A-HMT-S with AI be designed so that the liability for the consequences of every action are clearly assigned to an identifiable human or organisation?

Q2. What guidance can be given to all stakeholders, including regulators, to ensure clear identification of responsibility for actions by the A-HMT-S?

Q3. How will the potentially liable individuals develop sufficient trust to carry out their work?

These questions must be resolved for a new design by setting requirements with possible design solutions. The resolution for an existing system will concern its actual performance and setting limits on its behaviour. An architectural approach is taken as it is a well-known methodology for both new and existing systems The architecture and the views used to describe it must be precise, internally consistent, and describe the system to the level of detail needed to answer the questions.

Architectures for an A-HMT-S

Architecture aim

Every A-HMT-S must have a consistent and coherent structure which can be described by an architecture which drives its design and upgrades by decomposition of high-level requirements into verifiable system and subsystem requirements and behaviours. Every examination of the system will use architecture views to describe the particular aspects required for a specific aim. The views are drawn up and analysed using standard engineering processes to achieve that aim.

The aim in this paper is to demonstrate that a dynamic A-HMT-S with AI, including ML, can be trustworthy; it must answer the questions at the end of Section 3 and meet the top-level requirement given in the *Introduction*. It follows that the architecture must separate decisions from actions and embed clear authorisation of actions before they are taken. It is assumed that the A-HMT-S will have to achieve its goals in environments with varying levels of complexity and associated uncertainties.

The architecture aims should be achieved by:

1. using a model of human cognition and action to describe all subsystems in the architecture;
2. having a clear line of control and action authorisation from the Team leader down to the lowest level subsystem;
3. enabling rapid referral up the control chain if a node does not have the authority to act
4. giving the Team leader visibility of the automated subsystems' options in making decisions if needed; and by
5. providing or establishing clear limits to actions which can be taken by every subsystem in the architecture.

The 4D/RCS architecture

The 4D/RCS Reference Model Architecture for Unmanned Vehicle Systems V 2.0 [18], is used here as it meets the five criteria set out in Section 4.1. It has been demonstrated with human levels of intelligence in its subsystems [19] and for

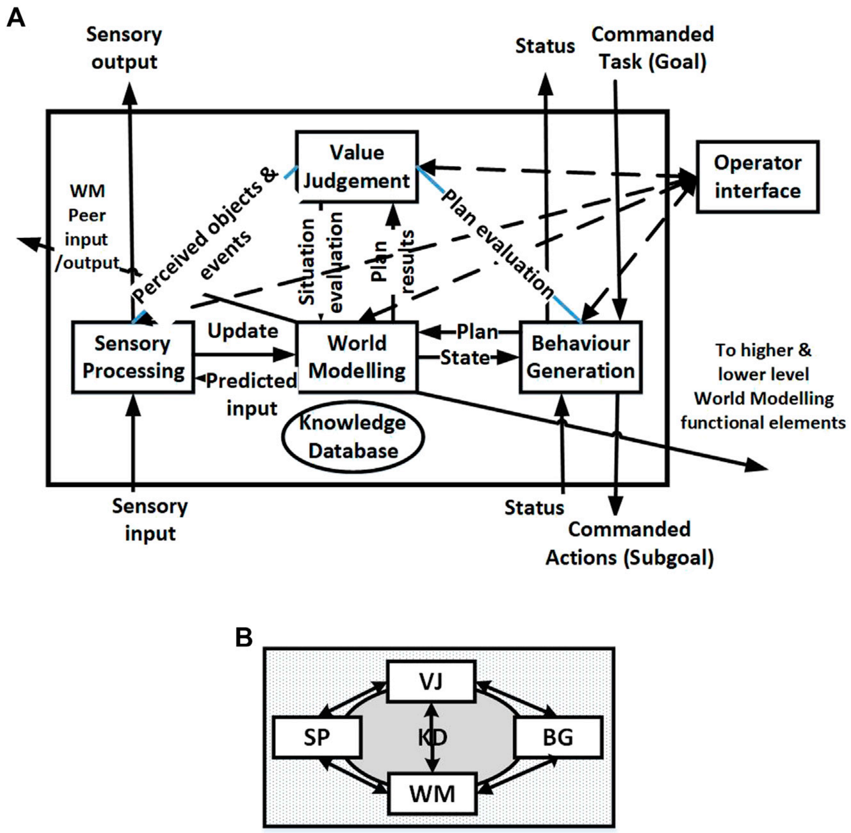


FIGURE 2
(A) A single 4D/RCS node, taken from NSTIR6910. **(B)** A schematic representation of a node used in later figures. Key: Value Judgement (VJ), Behaviour Generator (BG), Sensory Processing (SP), World Modelling (WM), Knowledge Database (KD).

identifying legal responsibilities in autonomous systems [20]. A full description of, and application as a hierarchical control structure for road vehicles is given in Ref. [21]. Other hierarchical architectures could be used provided they clearly identify where decisions are made and where the authorisation of these actions occurs.

The 4D/RCS architecture was devised for military command structures from high command to vehicle actuators. It defines responsibility for actions made by nodes which may be either human, machine or a mixture of the two. A node is defined as an organizational unit of a 4D/RCS system that processes sensory information, computes values, maintains a world model, generates predictions, formulates plans, and executes tasks.

Processes to apply the architecture are described in [22]. Descriptions of its use to identify legal responsibilities for the control of unmanned weapon systems and for autonomous cars is given elsewhere [23–25]. It is applied here to address the problems of trust and responsibility for the human Team leader by consideration of the three questions at the end of previous section.

Figure 2A is from the standard and shows a single node. Figure 2B is a schematic representation of its principle functions used later for simplicity. These functions are:

- the knowledge database which is the common repository for information for all nodes at that level;
- sensory processing which interprets sensor data and reports it to higher levels;
- a dynamic world model at every level with the resolution appropriate to that level. It is continually updated, based on information from the sensory processing function at that level. The distinguishing feature of 4D/RCS is that the world model makes predictions about the consequences of potential actions;
- the value judgement function assesses the predictions from the world model against the node's success criteria and ranks options for action; and

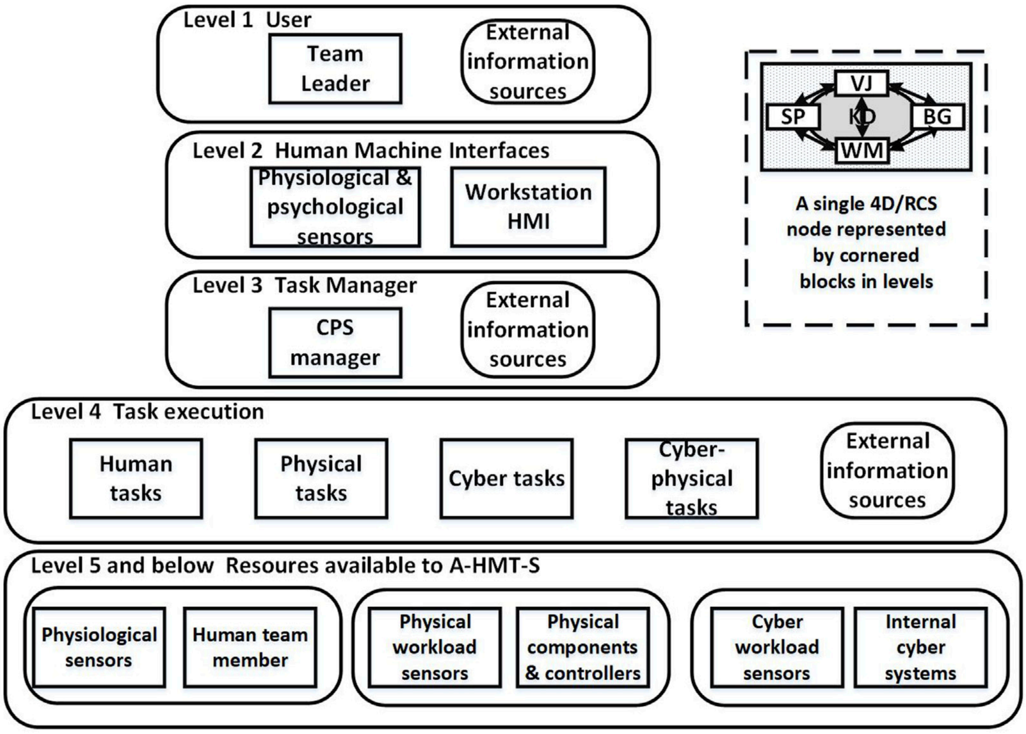

Level 1 User
Team Leader
External information sources
Level 2 Human Machine Interfaces
Physiological & psychological sensors
Workstation HMI
Level 3 Task Manager
CPS manager
External information sources
Level 4 Task execution
Human tasks
Physical tasks
Cyber tasks
Cyber-physical tasks
External information sources
Level 5 and below Resoures available to A-HMT-S
Physiological sensors
Human team member
Physical workload sensors
Physical components & controllers
Cyber workload sensors
Internal cyber systems
VJ
SP
KD
BG
WM
A single 4D/RCS node represented by cornered blocks in levels

FIGURE 3
An interdependent A-HMT-S structured as a 4D/RCS architecture with a 804 human as Team Leader. Acronyms in node are in the key for Figure 2.

- the behaviour generator takes the value judgement's outputs and acts, setting goals and success criteria for lower levels; if there is no safe action, the behaviour generator makes its part of the system execute a fail-safe mode, informing other nodes of its action.

The sensory processing, value judgement and behaviour generator functions form the three-part model of human decision-making and behaviours as described by Rasmussen [26] and the Observe, Orient, Decide, and Act (OODA) loop [27]. This model enables a common representation of both human and automated nodes. A node can have non-deterministic behaviour provided its actions are limited to its level of responsibility. Authority for decisions and responsibility for the consequences of their actions is determined through the commands and responses in the hierarchy of behaviour generators. The concept of authorised power has been introduced recently which sets the limits to a node's freedom of action. It is defined as [28]:

> The range of actions that a node is allowed to implement without referring to a superior node; no other actions being allowed.

This restriction allows hard limits to be set on a node's behaviour. Their sum for all nodes restricts the overall A-HMT-S′ behaviour, giving a basis for specifying trustworthiness in engineering terminology.

4D/RCS applied to an A-HMT-S

Figure 3 gives the broad characteristics of a 4D/RCS architecture for an A-HMT-S, with the user as Team Leader at Level 1 and the plurality of resources needed to complete the system's overall tasks at Level 5. For clarity, individual nodes are shown as blocks, each one representing a 4D/RCS node as shown in the dashed box in the figure. External information sources will be available at many levels, and indicated where appropriate.

Nodes at every level report to only one node in the next higher level, with clear responsibilities and limits to their actions based on their fixed position in the architecture. Sensory processing information is shared across levels in the hierarchy and can be passed up to the highest level. All information is shared between nodes at the same level as they have a common knowledge database.

Common response times, or other characteristics, across a level allows simplification of the data structure and world model at that level. They also enhance detection of differences between the real and expected world at any level, with a rapid escalation of the awareness of a problem. The time divisions for an A-HMT-S are not as straightforward as in the original concept for 4D/RCS. It should be possible to construct a set of timescales for any given application, recognising that there will be a range of timescales for completion of activities at any given level.

ML has been incorporated into 4D/RCS. Initially Aldus *et al.* put ML solely in the world models at different node levels, using it to assimilate data in many formats [29] but later incorporated ML in more node functions in the system [30] during the DARPA Learning Applied to Ground Robots (LAGR) programme; in particular the authors state that:

> The learning in each of the modules is not simply added on to the process that implements the module. It is embedded as part of the module, and operates in accordance with its location in the hierarchy.

An adaptive HMI at Level 2 with the sensors monitoring the Team Leader meets this criterion for an A-HMT-S. Elsewhere ML can be introduced into functions within any node in the hierarchy provided there are clear authorised powers set for each node.

The node hierarchy of 4D/RCS ensures precise specification of every node's individual role and hence its responsibility. It builds in a well-structured control chain that allows tasks to be transferred between nodes provided that this transfer is authorised by the next higher level in the hierarchy. The nodes at any given level are interdependent, but this interdependence is managed at the next higher level. The problem here is that task allocation is dynamic, based on node workload at any given time. Task reallocation can be rapid for the completely automated nodes, but the human nodes must be given enough time and information to make considered decisions. Assessing and quantifying human cognition times in a dynamic system will be a problem requiring a model of the Team Leader.

Inside the architecture

It is necessary to examine each level in Figure 3 in more detail to establish the feasibility of an A-HMT-S and the key problems requiring solutions. An A-HMT-S must assess available options for actions and their consequences by comparing plans with the current "real" world as reported by the sensory processing function. The world model at each level is a key part of this process, with its role brought out in the following sections.

It is likely that the problems highlighted by the analysis here will be common to all architecture frameworks, so they could become potential research topics if not already developed.

Level 1, the human team leader

The Team Leader will have direct access to the functions in the HMI and indirectly to lower-level functions through the behaviour generator chain. Team-Leader visibility of all parts of the system is made available through the HMI sensory processing module.

Level 1 functions are specified to ensure the owner's business priorities are met, monitoring the current team status, predicting future events, and resolving conflicts. Although a team member, the Team Leader's role must be the highest hierarchical level, instructing lower levels. Instructions are given as team goals and success criteria, with priority weightings for the A-HMT-S to interpret. The Team Leader must also trust the CPS to flag up all those problems requiring their attention through the HMI.

It is essential that the Team Leader's workload is manageable so there is time to understand the options considered by lower levels and the issues they cannot resolve. It is assumed that the Team Leader's workload can be monitored at Level 2 supplemented by other sensors if necessary. Potential overloads will be presented to the Team Leader with Level 2's recommendations for their removal. The Team Leader will then decide what new instructions must be issued.

A smaller A-HMT-S may have the Team Leader also carrying out some Level 4 functions in parallel with Level 1 functions. This structure does not fit in an ideal hierarchical architecture and would need detailed attention in system design. Potential solutions might include applying a temporary surrogate chain of command at Levels 1 or 4 whilst the Team Leader concentrates on the higher priority functions, or delaying the Level 4 task and letting the low-level consequences be managed automatically.

Level 2, the HMI

The interaction between the Team Leader and the CPHS will be through the HMI at Level 2. It is put in Figure 3 as a specific function, following Madni & Madni and Madni *et al.* [11, 12] as it plays a key role in any human-machine system. The Team Leader will probably have access to other information sources such as phones, direct visual checks and independent access to the internet.

The HMI's first role is to translates Team-Leader-defined aims or changes into goals for the system with priorities and other necessary information. The information is passed through the behaviour generator chain to Level 3. The Team Leader must have both cognition of the A-HMT-S task status and the detail required to issue effective instructions. Although this is a normal human factors problem, it does not help solve the problem of translating human-language queries or goal changes into team instructions in the machine language used at Level 3.

The HMI's second role is the separation of functions between the Team Leader and the dynamic task manager so that the Team Leader does not become overloaded by involvment in actions which can be handled automatically. Part of this role is to monitor the Team Leader's own workload through indicators such as response times and other indicators of their cognitive and physical state. If the workload is excessive, the HMI must present the Team Leader with options for reducing it. With ML, the A-HMT-S can learn an individual Team Leader's behaviour and overload signatures, but it must recognize individuals and variations in their performance.

The HMI can only direct changes at lower levels, through the behaviour generator chain, if it sees a problem and has the authority to implement a solution. Specific system designs will need to address which actions it can take, and how the reasoning is presented to the Team Leader when action is taken.

The HMI's third role is to ensure that the Team Leader is presented with clear statements of problems which it cannot resolve, backed up with relevant information and options considered for action. It checks that the current and predicted operations are being managed correctly at Level 3, flagging actual and potential problems to Level 1. Problems can be identified by both the HMI and Level 3. The Team Leader may then wish to access further data from Level 3 and add new information into the HMI knowledge database to increase the range of options. It may be necessary for some of this information to be passed to lower levels for more detailed analysis, but kept separate from measured sensor data.

These three roles are fixed, so the HMI is not one of the nodes that can have its tasks changed by the dynamic task manager. However, if it determines that the Team Leader's workload or situation awareness is likely to be outside a safe and efficient level it will inform both the Team Leader and the dynamic task manager. The dynamic task manager can make suggestions to the Team Leader through the HMI but not act on them. The Team Leader can change his or her tasks and workload through the HMI's behaviour generator chain.

The HMI world model requires a model of human capabilities, the human's state and warning signs of overload based on available sensor mechanisms. The model may be supplemented with information about individual Team Leaders if this is permitted. The use of the three-part model of cognition in 4D/RCS will facilitate this interface.

This HMI world model will require all the information from the Level 3 model, and set it in the wider context of external factors acting on the A-HMT-S. The wider factors included at Level 3 will have been filtered for reasons given in the next section; the HMI can use the Level 3 internet access to overwrite its constraints whilst deriving its own options and selections for presentation to the Team Leader. The Team Leader will also have this option through the sensory processing chain for their own mental model.

Level 3, dynamic task manager

Level 3, the dynamic task manager, has only one node, the CPS manager and its external information sources. The external sources may include the internet, but this must be well controlled. The use of external AI engines to search for and select information may not be reliable so, as a minimum, information will need to be tagged with its source and an estimate of reliability. Unquestioning acceptance and use of external search engine results will expose the Team Leader to unacceptable risk as a court may decide later that the information was clearly unsuitable for the A-HMT-S's use.

The CPS manager's role is to provide efficient use of resources at Level 4 and below. It specifies the tasks required to meet the goals and priorities from Level 2, their success criteria and other instructions, then issuing them to Level 4 through the behaviour generator chain. It draws on timely information about task status from Level 4 and allowed external information sources; these form its sensory processing functions. Decisions to assign and reassign resources are taken by its behaviour generator either autonomously or after referral to Level 2 and possibly Level 1. Level 3 is the lowest level at which there is an overview of all tasks.

The Level 3 world model includes: all current tasks and their status; available resources; and their allocation to tasks, both current and future. It will not have all the detailed task information in the Level 4 world model. The Level 3 world model will include the wider activities which do not form part of a task but do affect them. Examples are maintenance and staff holidays.

Comparison of Level 3's sensory processing function output, workload plans and task success criteria will identify potential problem areas for action by Level 2 if it cannot resolve them itself. The system architecture must mandate whether all changes at Level 4 are dictated by Level 3 or if Level 4 nodes are allowed to negotiate due transfer of resources or parts of tasks between themselves at a local level. This transfer could be advantageous as it removes work from Level 3 but could create problems if the Level 3 world model is not aware that these changes have been made. The use of surrogate chains of command may provide a solution to these problems.

Level 4, individual task management

Individual tasks are managed at Level 4 by drawing on the human, physical and cyber resources at Level 5 and below which have been allocated to the task by Level 3. The names for the Level 4 nodes in Figure 3 simply reflect the types of task required, and do not imply a separation of task types based on their required resources. It is unlikely that a human will manage tasks at Level 4, although there may be parts of many tasks which require human resources at Level 5.

The world model and knowledge database common to all Level 4 nodes include resources and their availability for each task as a function of time. Time resolution and resource detail will be lower than that required at Level 5. The world model predicts the effects of changes due to instructions from above or responses from lower levels. Task-related problems will become known at Level 4, giving it the ability to solve many of them. However, each node must have clear limits on its authority to authorize actions with consequences outside its own task. The Level 4 nodes must have the ability to flag problems for attention by higher levels. An example might be when two tasks require the same resource at the same time in the future which Level 3 could resolve by changing the time criteria on one task or by redeploying resources across several tasks.

Level 5, resources

Functions below Level 4 are not considered in any detail here as their structure depends on the specificA-HMT-S, recognising this treatment as a necessary simplification. However the 4D/RCS architecture and the structure of Level 4 do set some constraints on Level 5 nodes and the functions they perform.

The complexity of the nodes at this level will depend on the A-HMT-S under consideration. A node may include all the dedicated resources for one task which are always allocated to that node, the resources being used for other tasks only when that task is not needed. On the other hand, nodes may be subsets of physical or computing resources suitable for a range of tasks; their allocation at any time being under the control of Level 4 task managers. It is unlikely that nodes at this level will be able to negotiate reallocations between them.

Any given Level 5 node may be a complex system in its own right; for example it could be an electro-mechanical system embodying complex adaptive control systems using advanced methods [31]. These systems could easily include AI-based techniques provided their freedom of action is limited by a suitably framed authorised power covering cyber and physical outputs.

There are workload monitors for every resource at Level 5. These may be discrete components such as thermometers for motor drives, or they may be a part of a resource's software. Combinations of individual resource sensors may need to be reconfigured when resources are reallocated to determine the workload being used for current tasks.

The Level 5 world model will be centred on resources and their current and future allocation to tasks on the shortest timescales. It will be based on the structures below Level 5 and their requirements as the tasks evolve. However, it will be visible to higher levels through the sensory processing chain which enables the Team Leader to request information about every resource in the system. The higher levels may consider that changes in resource allocation or task parameters are necessary at Level 5 or below, but they can only make these changes using the behaviour generator chain which will identify the consequences of such requests and then report back.

Decision making process and action authorisation in a node

Each node in Figure 3 fits in the 4D/RCS hierarchy as shown in Figure 4. (For clarity, lower nodes are only shown for the middle node). Every node's aim is to execute its task whilst managing workloads for the resources under its control. It is given tasks and success criteria from its superior node; these are interpreted, and subordinate nodes are given their tasks and success criteria through its behaviour generator. The knowledge database is shared across its level. Every node's actions are constrained by node-specific authorised powers.

Figure 5 shows the information flows inside the node. The four principal node functions are indicated by the shaded areas. For simplicity, it is assumed here that the A-HMT-S is already executing a task and that the new instructions will change its plans. Instructions are aims for the revised task and, if necessary, revised success criteria to assess task completion. The node checks that the task is within its authorised power and then derives one or more workload plans for comparison with the current world.

The current world model covers the timeframe relevant to this level in the hierarchy and is derived from the sensory processing function. Predictions are made for workloads and compared with the available resources to give the N task consequences shown in Figure 5. It is assumed that the node has some freedom in planning its own and its subservient nodes' instructions and that there will be a range of success criteria for different parts of the task. A number of plan options M, which will be less than or equal to N are assessed in the value judgement function and ranked according to criteria set by either the higher node or from its knowledge database. A check is made in the behaviour generator that the node is authorised to implement the chosen plan. If it is, the plan is accepted, if not, another option is chosen. If none are allowed, a fail-safe plan is implemented and the superior node informed. Authorisation of action is still within the node and its own task.

The node's authority will, among other factors, allow it to use resources that are not assigned to other nodes for the period required for an acceptable option. If it does, the change is accepted as a new task, instructions are sent to lower levels as revised success criteria, and the revised plan is incorporated into the knowledge database for that level. The other nodes at that level will compare the revised plan with their plans; should there be a conflict due to their own replanning, then the nodes will cooperate to resolve them with the results passed through the behaviour generator chain to the next higher level. If the problems cannot be resolved, for instance if one node's authorised power will not allow it to act, then the next higher node is informed through the behaviour generator chain. Revised instructions, generated as success criteria, will be created at that level by the same process and the lower nodes will respond accordingly.

The decision-making process described above is generic with differences in the information used at any point in the process at different levels. Table 1 describes the type of information at key points in Figure 5 when applied to Levels 2, 3 and 4 in Figure 3.

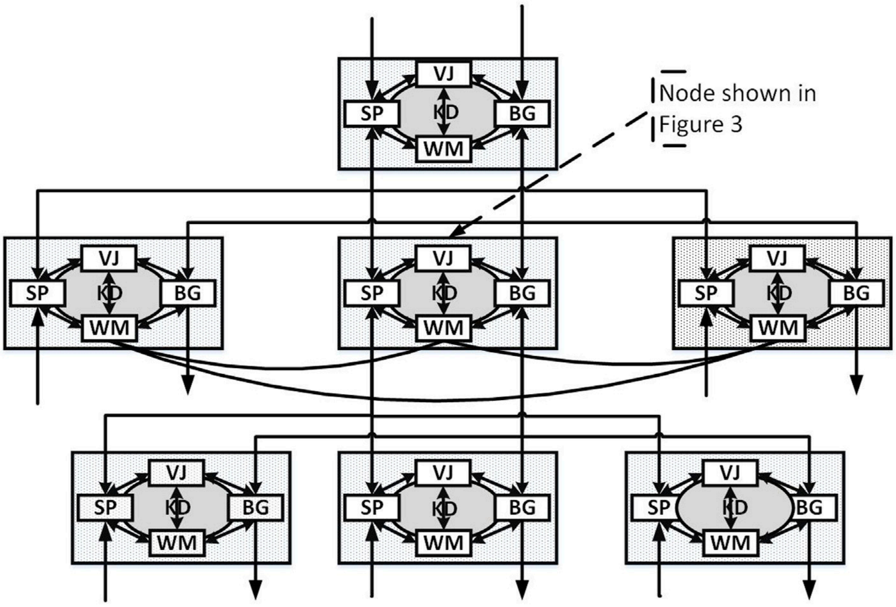


FIGURE 4
Showing the connections between nodes in each position in the hierarchy in Figure 3A 4D/RCS node and connected nodes. Key: Value Judgement (VJ), Behaviour Generator (BG), Sensory Processing (SP), World Modelling (WM), Knowledge Database (KD).

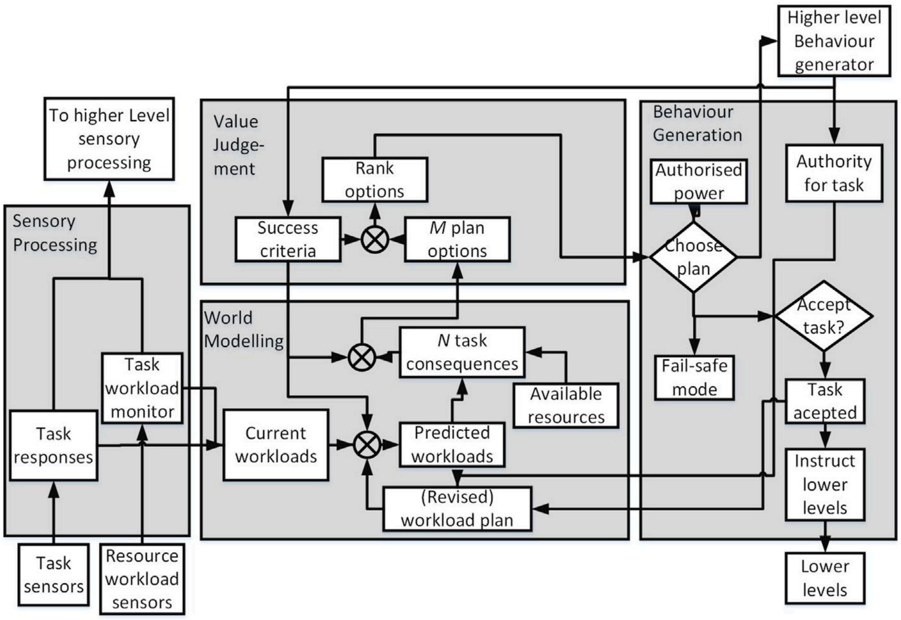


FIGURE 5
Information flows within and between the functions in a node. Each function is a shaded box. Information processing is in the white boxes and comparisons in the circles.

TABLE 1 Description of information used at key points in Figure 5 for different architecture levels.

Term	Level 2	Level 3	Level 4
Inputs from higher level behaviour generator	• A-HMT-S tasking from Team Leader	• New or revised tasks and success criteria	• New or revised task • success criteria • resource allocation
Success criteria	• Business priorities for tasks or groups of tasks	• Match of all proposed options against business criteria	• Cost, time and quality for task
Outputs from Sensory Processing to World Modelling	• Match of future activities and plans against Team Leader's criteria • Human activity and stress level	• Match of task progress and resource allocation against Team Leader's success criteria	• Progress reports on task progress • Workload on resources currently or planned to be used by node
Output from Current Workloads	• Need for extra or fewer resources	• Workload across all resources	• Workload for one node's task
Plan options	• Look for and secure external resources if possible • Present options to Team Leader	• Reassignment of resources across tasks • Slips in progress allowed for lower priority tasks if overall success criteria are met.	• Changes to current resource plans • Slip in task completion deadline
World model horizon	• Current and predicted operations under current plans • External world as it affects current operations	• Resource use across all tasks plus likely new ones • Overall costs	• Detailed task plans with current and predicted progress • Options to reduce costs in individual tasks
Default authorised power	• Limited ability to draw on external resources • Cannot exceed fixed criteria when considering options	• Can reallocate resources across tasks at Level 4 • Can only instruct restricted set of available resources	• Only use previously assigned resources • All systems to follow safety protocols

ML, decisions, and actions in a node

Non-deterministic processes must have strictly limited behaviours when included in a control system requiring any level of safety. AI has been included in a 4D/RCS architecture in [19] and applied to autonomous ground vehicles traversing rough terrain. In these applications AI is used mainly for interpreting sensor data to recognise obstacles in building up a representative world model, although Albus and Barbera [32] propose using AI to adjust parameters in the equations used to decompose goals into tasks and further decomposition.

Human trust has been incorporated into the A-HMT-S by using a dynamic world model that replicates a human mental model and having strict limits on each node's actions. However, although necessary, these are not sufficient. The human may not be able to easily understand why a learning algorithm made a decision, but they can accept it if they think that it is reasonable; i.e., the human perceives the decision as sensible and that fits with their own mental model of the problem. The A-HMT-S must be able to present relevant information about the options considered when choosing an action so that the human can understand and assess the choice.

Restricting every AI-algorithm's operating domain to be within a node limits its effect. The aim of every node is to complete its task by meeting its success criteria, solving problems within the limits of the authority it has in the architecture. The learning system will decide its best option for solving the node's problem by using information available at that level in the heirarchy. This solution can then be considered as one option of the N options generated in the world model. It will then be assessed with those not generated by the learning algorithm in the value judgement module, and the result passed to the behaviour generator. Whichever option is chosen, the behaviour generator checks if consequent actions are within its authority, ensuring that the Team Leader and any regulatory authority know that actions arising from the learning algorithm cannot exceed predetermined safe limits.

The sensory processing chain can pass information directly from any level to all higher levels. The Team Leader can interrogate the data and other information used by any node in its decision making, giving the potential safeguard of human-initiated enquiries. However it will be impossible for the Team Leader to query every decision before its consequent action, so triggers should be included in the value judgement and behaviour generator modules to initiate a Team Leader enquiry into the decisions leading to defined types of critical action before their execution.

Examples of learning systems which can improve efficiency in the A-HMT-S include:

- tasking workload monitors to pick up warning signs of overloads due to variations and tolerances on workload data, timing *etc.*;
- adding to task consequences based on historic data;
- assessing changes in the business environment that may alter task success criteria;
- ranking of options, but rank order may need confirmation by the next higher level before acting;
- using AI to identify potential problems at the level below the node it is in; and
- monitoring external conditions to give early warnings e.g. an approaching snowstorm probably changing delivery times for materials.

Some nodes at Level 4 may be people acting in accordance with their task and management instructions. They will not be fully autonomous as their authorised power will be set by their organization's management processes. Their workload sensors may be more subjective than for other parts of the A-HMT-S and so will need explicit inclusion in the sensory processing chain. They will almost certainly use network support tools for their work, and their use may provide a suitable mechanism to monitor their workloads.

Example architecture applications

The architecture presented in Figures 3–5 is generic in nature. It needs to be applied to some sample scenarios both to check their practical validity and to identify more precisely the topics that warrant further R&D effort. One vignette is taken from each of the three classes of HMT used at a recent conference on human-machine teaming [33].

Recommendations to a human for their immediate action

Automated identification of targets to a pilot who is about to release a weapon is a well-known A-HMT-S problem and the subject of international debate [10]. The 4D/RCS architecture has been applied to it in Chapter 12 of [20] with architectures similar to Figure 3 shown in Figures 12.8 and 12.9 in that reference and Figure 5 similar to the one shown in Figure 13.4. The architecture took an incremental evolution of current systems by replacing human nodes with automated ones whilst maintaining the necessary response speed for human assessment of action and the consequent changes in military tactics and rules of engagement.

[25] shows how legal responsibilities for the driver and vehicle can be derived for autonomous vehicles at all autonomy levels.

The consequent changes to responsibilities in the design chain for military and civilian products are discussed in Ref. [34]. It is shown there that a hierarchical architecture is essential for the design of an autonomous system so that safety-related decisions can be identified with the legal responsibility for the system's actions assigned to individual organizations and role-holders. The principal issues are link-integrity to ensure continuous control of the weapon, and reliable identification of both targets, non-targets, and the civilian objects which should not be attacked. Similar issues will apply for vehicles.

Carebots

We take the case of a robot caring for an elderly person in their own home which has one floor. The carebot is leased from a health care provider who are responsible for its maintenance and updates. Figure 6 gives the broad characteristics of a carebot HMT architecture equivalent to Figure 3.

The Team Leader is the elderly person giving instructions to the CPS part of the team. Mutual trust and interdependence is critical. The CPS can provide facilities or resources such as medication but cannot force the person to take them as this legally is assault; similarly, the elderly person may be critically dependent on the CPS for provision of medication and their regular supply. The person will have normal interaction with other people and resources using the non-carebot resources that they are capable of using; these may be restricted but could be extensive for a mentally agile but physically infirm person.

The HMI at Level 2 will be safety-related as a minimum standard if it provides calls to emergency services on behalf of the Team Leader. This places high demands at Level 2, making an adaptive HMI essential with a sophisticated model of the Team Leader and voice recognition for a range of human emotions. The adaptive HMI will be very different from that assumed in earlier sections with considerable scope for AI-based development here. There is only one human to model, and scope to incorporate intelligent analysis of physiological sensors looking for precursors of serious medical conditions. Actions will be requested from the Team Leader and passed, as necessary, to Level 3 to alert necessary medical or social services or relatives. This may raise the software standard to safety-critical with

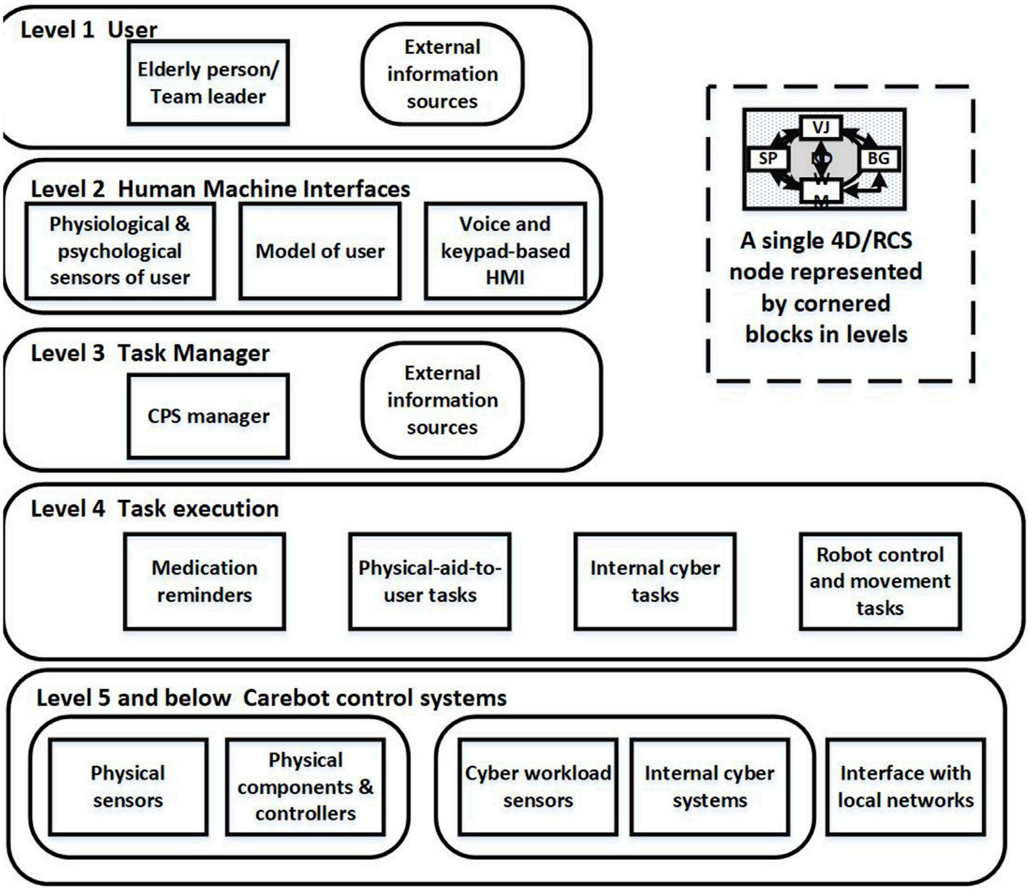

Level 1 User
Elderly person/ Team leader
External information sources
Level 2 Human Machine Interfaces
Physiological & psychological sensors of user
Model of user
Voice and keypad-based HMI
Level 3 Task Manager
CPS manager
External information sources
Level 4 Task execution
Medication reminders
Physical-aid-to-user tasks
Internal cyber tasks
Robot control and movement tasks
Level 5 and below Carebot control systems
Physical sensors
Physical components & controllers
Cyber workload sensors
Internal cyber systems
Interface with local networks
VJ
SP
KD
BG
WM
A single 4D/RCS node represented by cornered blocks in levels

FIGURE 6
Carebot as an A-HMT-S with the elderly person as the Team Leader.

associated standard and regulatory requirements, a very high standard for an evolving model of an individual.

The dynamic task manager at Level 3 will perform approximately the same as those described for Level 3 in the Inside the Architecture section, but there will almost certainly be mandated external interfaces for medical and emergency services. Medical records, medication and related data will need to be in the knowledge database; their location at Level 2 or 3 or a split between these levels will be a design decision, as will the method of updating them. The task to alert external organizations must decide the type of aid sought and be able to communicate with them effectively. The decision will be based on a comparison of the person's current status compared with their expected status, the level and type of difference, and the confidentiality of information in its database. There will probably be a need for a medical professional to talk to or visit the Team Leader so arrangements may need to be made for this. This interface represents a large R&D challenge.

The carebot will need to continuously monitor the Team Leader's well-being through signatures such as movement and heart beat as well as external environmental conditions such as a sudden cold snap or thunderstorm which may necessitate precautionary measures in the house or changes to the Team Leader's diet, for example, by offering more hot drinks.

The CPS aspects of controlling, maintaining, and upgrading the functions and resources at Level 5 will be similar to any other A-HMT-S system. The main difference will be the notifications and revised instructions given to the Team Leader in a way that they are understand, possibly with prior warning and a familiarisation session before installation, based on the Team Leader's specific needs.

A system which operates alone for long periods then reforms as an A-HMT-S

An example of this type of system is a robotic planetary explorer that is visited periodically by humans who rely on it for support while they are on a planet. Levels 4 and 5 will be similar to most robotic applications, but the higher levels will have major

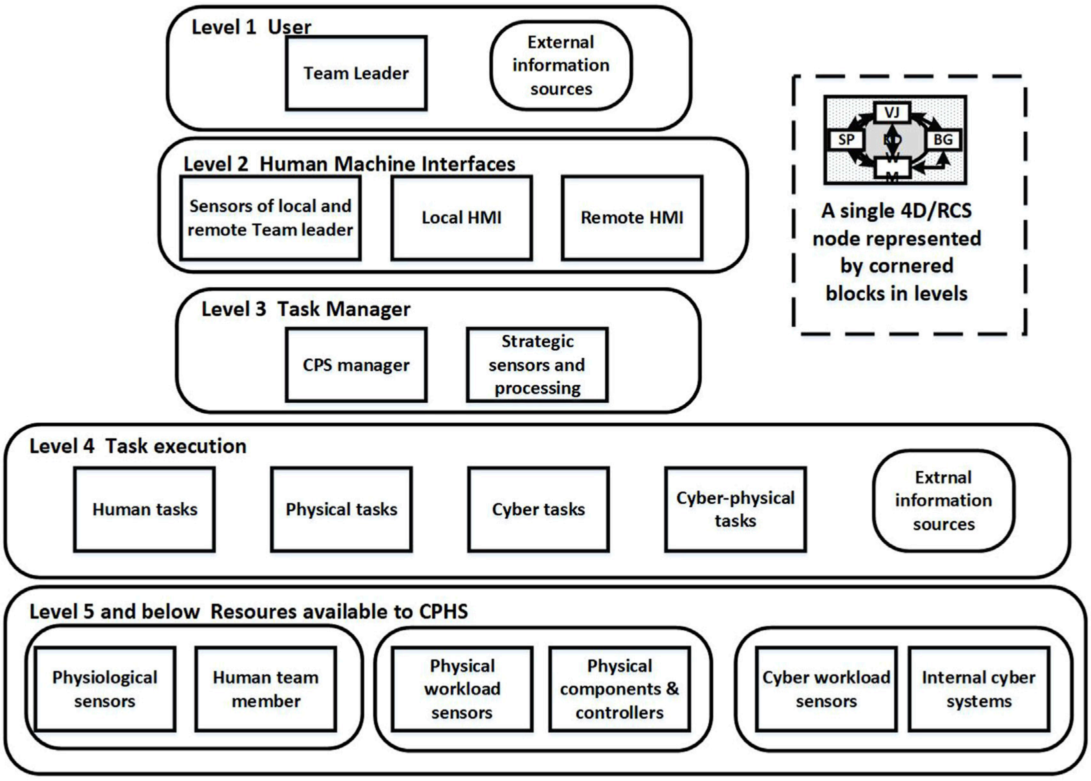

Level 1 User
Team Leader
External information sources
Level 2 Human Machine Interfaces
Sensors of local and remote Team leader
Local HMI
Remote HMI
Level 3 Task Manager
CPS manager
Strategic sensors and processing
Level 4 Task execution
Human tasks
Physical tasks
Cyber tasks
Cyber-physical tasks
Extrnal information sources
Level 5 and below Resoures available to CPHS
Physiological sensors
Human team member
Physical workload sensors
Physical components & controllers
Cyber workload sensors
Internal cyber systems
VJ
SP
KD
BG
WM
A single 4D/RCS node represented by cornered blocks in levels

FIGURE 7
A planetary rover as an A-HMT-S showing the two Team Leaders.

architectural problems. There will be two types of human interaction: remote monitoring, with instructions and updates sent from Earth or a relay satellite, and local interactions by visiting astronauts using an embedded HMI. These are shown in Figure 7.

Level 1 is shown with one Team Leader as it assumed that there will be protocols to prevent a remote person sending instructions when the rover has a local Team Leader. The Level 2 sensors will be for data transfer using remote links, checking for errors and missing messages, and will operate at all times. It is assumed that the astronauts' state of health will be monitored by other means such as their personal life-support equipment, reducing the HMI requirements considerably from the general case for an A-HMT-S.

Extensive fault detection systems will be necessary due to long periods without human attention in a hazardous environment with, for example, high radiation levels increasing the chance of semiconductor failure in the narrow-track high-frequency processors needed for advanced AI systems. Contingency reconfiguration of functions and tasks will need to be chosen based on probably incomplete diagnosis of apparently random failures and clear symptomatic information passed to the Team Leader, another area for R&D.

The strategic sensors at Level 3 will monitor local planetary conditions and provide assurance that software updates are not only received and installed, but will also run the required performance tests, sending the results back to Earth and to the local astronauts before and after their arrival for checking. This is to ensure the vital mutual trust between Team Leader and machine when restarting an interdependent relationship. The information will be held in the Level 3 database and its world model compared with the Levels 3 and 4 sensor processing outputs. The CPS manager will play a similar role to that in all the other A-HMT-S.

Discussion

Trust for an A-HMT-S

Three important questions were posed in at the end of the third section:

> Q1. Can a dynamic A-HMT-S with AI be designed so that the liability for the consequences of every action are clearly assigned to an identifiable human or organisation?

Q2. What guidance can be given to all stakeholders, including regulators, to ensure clear identification of responsibility for actions by the A-HMT-S?
Q3. How will the potentially liable individuals develop sufficient trust to carry out their work?

Questions 1 and 2 can be answered by the use of a hierarchical architecture. It can be used to identify important and critical issues for each stakeholder based on the following points:

- The architecture can and must give clear separation of human nodes and automated ones in the hierarchy. This separation ensures that liabilities can be clearly assigned to the Team Leader or the organization responsible for the design or upkeep f the automated node. The architecture in this paper has taken Level 1 to be exclusively human, interacting through an adaptive HMI at Level 2. Other humans will be at Level 4, participating as Team Leaders of tasks, again with separation by levels within the tasks. These Level 4 humans receive their tasking from Level 3 and are monitored as part of the overall A-HMT-S.
- The architecture should separate decisions from actions with an assessment of the reliability of the decision. The Level 3 dynamic task manager is automated and should refer uncertain actions to the Team Leader through the Level 2 HMI when problems cannot be resolved within its authority level. Action is based on the choice of an option from those arising from the comparison of the world model with physical reality, a difficult task for a complex environment. The decision to refer to a higher level is critical as the false alarm rate must be low in order to maintain trust. This decision will require intelligent AI analysis based on mainly uncertain data.
- A bounded system, such as a distribution network or airport where tasks and progress can be readily quantified, will make the comparison of the real world and its world model easier than with subjective information. Additionally, the range of actions and their authorisation node can be defined uniquely. Developing such an A-HMT-S with humans at several levels would give opportunities for R&D progress in developing Level 3 CPS techniques for both complex (Level 3) and simpler (Level 4 or 5) scenarios.
- The use of predictive models in 4D/RCS and the information flow model used here ensures that the consequences of an action are assessed and authorised before it happens. Auditable authorisation of actions by the system enables consequent identification of responsibility for the consequences of every action. The choice between automated or human authorisation becomes a part of the design process as it is recognised that ultimately any human authorisation of an action must be legal and follow local and national requirements.
- The use of authorised power as part of the behaviour generator in every node ensures that no unauthorised actions can be carried out without reference to a higher node and ultimately the human Team Leader. This does place the onus for safety on the person who specifies what must be raised to the next architectural level. However, when the specification is for a function within one node with defined authority, its implementation becomes a tractable problem which can be addressed by CPS designers. They will also require clear directions about local changes, regulations, and processes for system upgrades. Every A-HMT-S will be designed, or tailored, for specific applications so explicit considerations of authority levels and the allowed options for action at each node should give answers to questions 1 and 2 above.

The third question should be answered by the following points

- Limiting the behaviour of every node by setting and applying limits to actions based on a comparison of the real world and predicted consequences of a range of actions leads to it being trustworthy for defined conditions. Defining the conditions becomes a design and procedural issue which can be addressed by current engineering processes.
- Careful specification of the adaptive HMI so that it presents clear information about problems, whilst allowing the Team Leader to see the options and consequences that the lower-level nodes considered. This transparency should allow trust to develop. If it does not, the Team Leader can alter the authorised power of specific nodes so that actions that appear untrustworthy will be highlighted for further human action.

It is possible to set up a trustworthy A-HMT-S that satisfies the three critical questions and has little or no AI in it for specific applications. In these cases the A-HMT-S would have limited flexibility because most of its decisions would be made using deterministic processes with well-understood uncertainties. It could be argued that these are not teams but are adaptive control systems that change their behaviour in defined ways, triggered by pre-determined thresholds. AI is needed to achieve flexibility, autonomy and interpretation of uncertain inputs.

Trust-specific R&D

It was noted earlier in this paper that the authors of [30] found that learning processes must be embedded in nodes and not across them. That work was for one specific system and mainly concerned the sensory processing chain. A more general approach is to consider a node's functions in detail. Figure 5 gives more detail than the NSTIR standard, allowing an examination of the processes to identify which will benefit from AI and the type of R&D work that is needed. The following sections highlight the important areas for A-HMT-Ss without giving a review of current research, which is beyond the scope of this paper. There will be uncertainties at all points in the architecture, arising from many sources, making AI-based solutions attractive, but they must not detract from the CPS's trustworthiness or the HMT becomes untrustworthy. This places large requirements on all AI processes in the A-HMT-S and hence on the R&D work for every application.

Sensory processing chain

Levels 2 to 4 are mainly concerned with creating task instructions from human-defined aims, workload issues in the A-HMT-S, and selecting problems which require human cognition and authority to resolve. Level 3 has restricted access to outside networks so the access limits can be tailored to the ability of the CPS manager to interpret the information.

Level 5 and below will have the application-specific sensors for the outside world such as imagers and collision warning systems. Many of these already have some AI and some will be safety-critical.

At Level 4, the sensory processor outputs from Level 5 are interpreted by the task manager as progress on the tasks required for the A-HMT-S to achieve its aim. An accurate interpretation will be impossible if the world models at Levels 4 and 5 are incompatible or in conflict. Comparison will be difficult as they have different levels of detail with different time horizons, so checking their consistency may be a better approach, carried out in the "current workloads" process in the world modelling function in Figure 5.

World models, world modelling and value judgement

All nodes at a given level have a common world model in their knowledge database. The world modelling function uses it for multiple comparisons and predictions. The results are assessed by a node against its success criteria, ready for decisions and action. The success criteria may include non-interference with higher priority tasks. The world models will need regular and intermittent updates for two reasons: real-time changes in the environment; and the detection of incompatibilities between world models at different architectural levels. All world models must be under configuration control, with a process for updates and the knock-on effects in other nodes and levels. Authorisation of a change to a model can only come from the next higher level as that has an overview of all the lower level's nodes and, with AI, will develop a model of each node's behaviour.

World models at all levels must be consistent, even though they have different time horizons. The model at any level must include the available resources, their current allocation into the future, and the authority vested in lower nodes to change an allocation. Figure 5 illustrates that each node will create its own set of $M \leq N$ predicted world models based on its interpretation of its workload plan and its success criteria. The N predictions could be based on multiple simulations representing the uncertainty range in the world model at that level. Alternatively, it may be straightforward to introduce AI into nodes performing well-bounded tasks and then to generate one preferred option. Each option will affect resource usage and the environment at different times due to their interdependencies, so each option's affects must be assessed before implementation of any action.

Behaviour generator

The behaviour generator function in each node has limits on its actions set by the system design. These may be temporarily changed by the next higher level if that level's predictions allow it. When the task is within a node's authority, a chosen option is created and offered to the behaviour generator by the value judgement function. This choice includes the plans and tasks for lower levels.

The final check before action is taken is to compare the chosen option with the node's authorised power. This includes not only what the node can do, but also what it cannot do. Prohibitions may come from higher levels, including higher priority levels of other tasks on available resources, and effects on the wider world. At the highest A-HMT-S levels (Level 3 and above) this will include the societal issues such as interpretation of laws and regulations. An example for the carebot is a lower level offering of an approved medication, the Team Leader refusing, which is their legal right, and Level 2 issuing instructions to re-offer in 5 min; several refusals could trigger an alert as an external human medical judgement would be needed, the fail-safe mode. The behaviour generator could include comparison tools developed using AI techniques, utilizing the power they bring to the assessment function. However, they must be thoroughly tested to ensure they do not evolve after installation to ensure that they have deterministic behaviour.

The check against a node's authorised power is effectively asking if the consequences of choosing the offered option are reasonable. If they are, the option is chosen and action taken. If not, and no other option is acceptable, the task is rejected, the higher node's behaviour generator informed and the higher-level node must reconsider its options. If no choice is acceptable to the higher node then the problem is escalated, eventually to the Team Leader for human assessment. This guards against the build-up of errors or large uncertainties producing an unexpected and unreasonable action which must requires human assessment. The Team Leader has access to information at all levels in the 4D/RCS architecture, enabling them to make a more-targeted assessment of the problem and potential solutions than the unaided CPS can make.

The definition of reasonable is crucial as it is a societal and legal term, not an engineering one. At lower levels limits can be set by their design as clear technical bounds can be set for most tasks, based on avoiding interference with other higher-priority tasks and preventing physical harm. At higher levels the limits become softer making an AI-based approach attractive, but this jeopardises its role as a safeguard because of the non-deterministic nature of AI. The interpretation of the softer issues and translating them into the engineering terminology of deterministic limits will require iterations between lawyers, social scientists and engineers. It is possible that eventually robust ML algorithms, their training data, and their automated reasoning approaches may develop to the stage of meeting legal challenges but this is unlikely at the current state of technology.

Conclusion

The use of a hierarchical architecture improves the effectiveness of A-HMT-S design and development. The analysis presented here gives approaches to solving problems in R&D for autonomy in interdependent A-HMT-Ss in three ways:

i) Specific ML tools can be introduced into a task where it will produce clear benefits as part of the world model at that node's level, yet all consequences of its decisions will still be bounded by that node's authorised power;
ii) ML can be introduced into all parts of a node, *except in the behaviour generator function*. This design decision will ensure that actions cannot happen based on unexpected decisions without authorization by the human Team Leader; and
iii) Introducing ML into the node's value judgement function highlights the often-subjective nature of assessing the value of tasks when setting priorities. Recognising the associated risks before introducing ML in this function should explicitly raise, and help resolve, the complex questions in these applications.

An underlying problem with the use of AI is that of uncertainties in the interpretation of input information for comparison with world models which are themselves incomplete or inaccurate in some respects. Solving this problem is Research Objective 2-2, *AI Uncertainty Resolution* in the 2022 NAS report [1] for the general case: the approach presented here allows the uncertainties to be identified and their effects limited for specific cases. The offering of the alternatives considered by the system to the human goes some way to addressing Research Objective 5-5, *Explainability and Trust*.

AI will always generate a solution, so there must be a safeguard against unreasonable action, as interpreted by society or an accident inquiry. Setting limits using authorised power, and their use for deterministic testing of reasonable behaviour in every node provides a potential safeguard, although it does create its own design problems. However, locating authorized power in the behaviour generator function of every node bounds the problems, and provides a clear context for the essential cross-disciplinary and societal agreements before an A-HMT-S can be considered trustworthy.

Decomposition of A-HMT-S requirements using a hierarchical architecture into requirements for nodes comprising functions, with limited authority to act, allows targeted introduction of AI into the areas where it will bring maximum benefit, and will also identify the R&D needs before its safe introduction. This goes some way to meeting the 2022 NAS Report's Research Objective 10-1, *Human-AI Team Design and Testing Methods* and Research Objective 10-2, *Human-AI Team Requirements*.

Data availability statement

The original contributions presented in the study are included in the article/Supplementary Material, further inquiries can be directed to the corresponding author.

Author contributions

The author is sole contributor to this work except where referenced in the text.

Acknowledgments

The author is grateful for discussions with and comments from Bill Lawless, Sarah Spurgeon and attendees at the AAAI Spring Symposium 2022, *Putting AI in the Critical Loop: Assured Trust and Autonomy in Human-Machine Teams*.

Conflict of interest

The author declares that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

1. National Academies of Sciences. *Engineering, and medicine. Human-AI teaming: State-of-the-Art and research needs*. Washington, DC: The National Academies Press (2021).

2. Llinas J. Motivations for and initiatives on AI engineering. In: WF Lawless, J Llinas, DA Sofge, R Mittu, editors. *Engineering artificially intelligent systems*. Springer (2021). p. 1–18.

3. Barfield W, Pagallo U. *Advanced introduction to law and artificial intelligence*. Cheltenham UK, Northampton MA USA: Edward Elgar Publishing (2020).

4. Baker S, Theissen CM, Vakil B. Connected and autonomous vehicles: A cross-jurisdictional comparison of regulatory developments. *J Robotics, Artif Intelligence L* (2020) 3(No. 4):249–73. https://heinonline.org/HOL/LandingPage?handle=hein.journals/rail3 div=38 id= page=.

5. Tan SY, Taeihagh A. Adaptive governance of autonomous vehicles: Accelerating the adoption of disruptive technologies in Singapore. *Government Inf Q* (2021) 38(2):101546. doi:10.1016/j.giq.2020.101546

6. USDOT. Autonomous vehicle guidance AV 3.0, "preparing for the future of transportation" (2022). Available at: https://www.transportation.gov/av/3 (Accessed June 16, 2022).

7. Sun Y. Construction of legal system for autonomous vehicles. In: In4th International Conference on Culture, Education and Economic Development of Modern Society (ICCESE 2020), 19. Atlantis Press (2020). p. 598–602.

8. Law Commission and Scottish Law Commission (2022). Automated vehicles: Joint report, (25 January 2022) HC 1068 SG/2022/15.

9. English and Scottish Law Commissions. Automated vehicle Project (2022). Available at: https://www.lawcom.gov.uk/project/automated-vehicles/ (Accessed March 31, 2022).

10. United Nations. Annex III to the final report on the 2019 meeting of the high contracting parties to the convention on prohibitions or restrictions on the use of certain conventional weapons which may Be deemed to Be excessively injurious or to have indiscriminate effects (2019). Available at: CCW/MSP/2019/9-E-CCW/MSP/2019/9-Desktop(undocs.org) (Accessed March 18th, 2021).

11. Madni AM, Madni CC. Architectural framework for exploring adaptive human-machine teaming options in simulated dynamic environments. *Systems* (2018) 6(4):44. doi:10.3390/systems6040044

12. Madni AM, Sievers M, Madni CC. Adaptive cyber-physical-human systems: Exploiting cognitive modeling and machine learning in the control loop. *Insight* (2018) 21(3):87–93. doi:10.1002/inst.12216

13. Mayer RC, Davis JH, Schoorman FD. An integrative model of organizational trust. *Acad Manage Rev* (1995) 20(3):709–34. doi:10.5465/amr.1995.9508080335

14. Griffor E, Wollman DA, Greer C. NIST special publication 1500-201; framework for cyber-physical systems. *Working Group Rep* (2017) 2. doi:10.6028/NIST.SP.1500-202

15. Blakeney RA. *A systematic review of current adaptive human-machine interface research (doctoral dissertation)*. Embry-Riddle Aeronautical University (2020).

16. SEBoK Editorial Board. The guide to the systems engineering body of knowledge (SEBoK), v. 2.5. In: RJ Cloutier, editor. Hoboken, NJ: The Trustees of the Stevens Institute of Technology (2021). BKCASE is managed and maintained by the Stevens Institute of Technology Systems Engineering Research Center, the International Council on Systems Engineering, and the Institute of Electrical and Electronics Engineers Systems Council Available at: www.sebokwiki.org (Accessed December 7, 2021).

17. Alix C, Lafond D, Mattioli J, De Heer J, Chattington M, Robic PO. Empowering adaptive human autonomy collaboration with artificial intelligence. In: 2021 16th International Conference of System of Systems Engineering (SoSE). IEEE (2021). p. 126–31.

18. Albus J, Huang HM, Lacaze A, Schneier M, Juberts M, Scott H, et al. 4D/RCS: A reference model architecture for unmanned vehicle systems version 2.0. In: NIST Interagency/Internal Report (NISTIR). Gaithersburg, MD: National Institute of Standards and Technology (200). [online]. doi:10.6028/NIST.IR.6910

19. Albus JS, Barbera AJ. Rcs: A cognitive architecture for intelligent multi-agent systems. *Annu Rev Control* (2005) 29(1):87–99. doi:10.1016/j.arcontrol.2004.12.003

20. Gillespie T. Good practice for the development of autonomous weapons: Ensuring the art of the acceptable, not the art of the possible. *RUSI J* (2021) 165(5-6):58–67. doi:10.1080/03071847.2020.1865112

21. Albus J, Barbera T, Schlenoff C. *'RCS: An intelligent agent architecture'. Intelligent agent architectures: Combining the strengths of software engineering and cognitive systems*. Palo Alto, California: AAAI Press. no. WS-04-07 in AAAI Workshop Reports 2004. Available at: https://web.archive.org/web/20110324054054/http://www.danford.net/boyd/essence.htm. (Accessed July 8, 2022).

22. Madhavan R, Messina ER, Albus JS. *Intelligent vehicle systems: A 4D/RCS approach*. New York, NY, USA (2007). Available at https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=823578 (Accessed April 24, 2022).

23. Gillespie T. *Systems engineering for ethical autonomous systems*. London: SciTech, Institution of Engineering and Technology (2019). p. 512.

24. Serban AC, Poll E, Visser J. A standard driven software architecture for fully autonomous vehicles. In: IEEE International Conference on Software Architecture Companion. ICSA-C (2018). p. 120–7.

25. Gillespie T, Hailes S. Assignment of legal responsibilities for decisions by autonomous cars using system architectures. *IEEE Trans Technol Soc* (2020) 1(3):148–60. doi:10.1109/tts.2020.3014395

26. Rasmussen J. Skills, rules, and knowledge; signals, signs, and symbols, and other distinctions in human performance models. *IEEE Trans Syst Man Cybern* (1983) 13(3):257–66. doi:10.1109/tsmc.1983.6313160

27. Boyd JR. The essence of winning and losing. *Unpublished lecture notes* (1996) 12(23):123–5. Slides also available at http://pogoarchives.org/m/dni/john_boyd_compendium/essence_of_winning_losing.pdf (Accessed April 22, 2022).

28. Gillespie T. *Systems engineering for autonomous systems*. Stevenage: SciTech, Institution of Engineering and Technology (2019). p. 292.

29. Shneier M. Learning in a hierarchical control system: 4D/RCS in the DARPA LAGR program. *J Field Robot* (2006) 23(11-12):975–1003. doi:10.1002/rob.20162

30. Chang T. Integrating learning into a hierarchical vehicle control system. *Integr Comput Aided Eng* (2007) 14(2):121–39. doi:10.3233/ica-2007-14202

31. Ding Z. *Nonlinear and adaptive control systems*. London, United Kingdom: Stevenage: The Institution of Engineering and Technology (2013). p. 287.

32. Aldus J, Barbera A. 4D/RCS reference model architecture for unmanned ground vehicles chapter 1 in R Madhavan, ER Messina, JS Albus. In: *intelligent vehicle systems: A 4D/RCS approach*. New York, NY, USA (2007). Available at: https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=823578 (Accessed April 24, 2022).

33. Linas J, Gillespie T, Fouse S, Lawless W, Mittu R, Sofge D, et al. *AAAI spring Symposium 2022 proceedings. Putting ai in the critical loop: Assured trust and autonomy in human-machine teams*. Elsevier (2022). in preparation.

34. Gillespie T. Risk reduction for autonomous systems. In: WF Lawless, J Llinas, DA Sofge, R Mittu, editors. *Engineering artificially intelligent systems*. Springer (2021). p. 174–91.

ORIGINAL RESEARCH
published: 09 September 2022
doi: 10.3389/fphy.2022.946095

Grounding Human Machine Interdependence Through Dependence and Trust Networks: Basic Elements for Extended Sociality

Rino Falcone and Cristiano Castelfranchi*

Institute of Cognitive Sciences and Technologies, National Research Council of Italy, Rome, Italy

In this paper, we investigate the primitives of collaboration, useful also for conflicting and neutral interactions, in a world populated by both artificial and human agents. We analyze in particular the dependence network of a set of agents. And we enrich the connections of this network with the beliefs that agents have regarding the trustworthiness of their interlocutors. Thanks to a structural theory of what kind of beliefs are involved, it is possible not only to answer important questions about the power of agents in a network, but also to understand the dynamical aspects of relational capital. In practice, we are able to define the basic elements of an extended sociality (including human and artificial agents). In future research, we will address autonomy.

Keywords: dependence network, trust, autonomy, agent architecture, power

OPEN ACCESS

Edited by:
William Frere Lawless,
Paine College, United States

Reviewed by:
Giancarlo Fortino,
University of Calabria, Italy
Luis Antunes,
University of Lisbon, Portugal

****Correspondence:***
Rino Falcone
rino.falcone@istc.cnr.it

Specialty section:
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

Received: *17 May 2022*
Accepted: *23 June 2022*
Published: *09 September 2022*

Citation:
Falcone R and Castelfranchi C (2022) Grounding Human Machine Interdependence Through Dependence and Trust Networks: Basic Elements for Extended Sociality. Front. Phys. 10:946095. doi: 10.3389/fphy.2022.946095

1 INTRODUCTION

In this paper we develop an analysis that aims to identify the basic elements of social interaction. In particular, we are interested in investigating the primitives of collaboration in a world populated by both artificial and human agents.

Social networks are studied extensively in the social sciences both from a theoretical and empirical point of view [1–3] and investigated in their various facets and uses. These studies have shown how relevant the structure of these networks is for their active or passive use by different phenomena (from the transmission of information to that of diseases, etc.). These networks can provide us with interesting characteristics of the collective and social phenomena they represent. For example, the paper [4] shows how the collaboration networks of scientists in biology and medicine "seem to constitute a "small world" in which the average distance between scientists via a line of intermediate collaborators varies logarithmically with the size of the relevant community" and "it is conjectured that this smallness is a crucial feature of a functional scientific community". Other studies on social networks have tried to characterize subsets by properties and criteria for their definition: for example, the concept of "community" [5].

The primitives of these networks in which we are interested, which are essential both for collaborative behaviors and for neutral or conflicting interactions, serve to determine what we call an "extended sociality", i.e. extended to artificial agents as well as human agents. For this to be possible it is necessary that the artificial agents are endowed, as well as humans, with a capacity that refers to a "theory of mind" [6] in order to call into question not so much and not only the objective data of reality but also the prediction on the cognitive processing of other agents (in more simple words: is relevant also the ability to acquire knowledge about other agents' beliefs and desires).

In this sense, a criticism must be raised against the theory of organization which has not sufficiently reflected on the relevance of beliefs in relational and social capital [7–11]: the thing that transforms a relationship into a capital is not simply the structure of the network objectively considered (who is connected with whom and how much directly, with the consequent potential benefits of the interlocutors) but also the level of trust [12, 13] that characterizes the links in the network (who trusts who and how much). Since trust is based on beliefs–including also the believed dependence (who needs whom)—it should be clear that relational capital is a form of capital, which can be manipulated by manipulating beliefs.

Thanks to a structural theory of what kind of beliefs are involved it is possible not only to answer important questions about agents' power in network but also to understand the dynamical aspects of relational capital. In particular, it is possible to evaluate how the differences in beliefs (between trustor and trustee) relating to dependence between agents allow to pursue behaviors, both strategic and reactive, with respect to the goals that the different interlocutors want to achieve.

2 AGENTS AND POWERS

2.1 Agent's Definition

Let us consider the theory of intelligent agents and multi-agent systems as the reference field of our analysis. In particular, the BDI model of the rational agent [14–17]. In the following we will present our theory in a semi-formal way. The goal is to develop a conceptual and relational apparatus capable of providing, beyond the strictly formal aspects, a rational, convincing and well-defined perspective that can be understood and translated appropriately in a computational modality.

We define an *agent* through its characteristics: a repertoire of actions, a set of mental attitudes (goals, beliefs, intentions, etc.), an architecture of the agent (i.e., the way of relating its characteristics with its operation). In particular, let a set of agents[1]:

$$\boldsymbol{AGT}{=}_{def}\left\{\boldsymbol{Ag}_{1,}\boldsymbol{Ag}_{2,}\ldots\boldsymbol{Ag}_{n}\right\}. \tag{1}$$

We can associate to each agent $Ag_i \in Agt$:

$$\boldsymbol{BEL}_{Ag_i}{=}_{def}\left\{\boldsymbol{B}_1^{Ag_i},\boldsymbol{B}_{2,}^{Ag_i}\ldots\boldsymbol{B}_m^{Ag_i}\right\} \tag{2}$$

(a set of beliefs representing what the agent believes to be true in the world);

$$\boldsymbol{GOAL}_{Ag_i}{=}_{def}\left\{\boldsymbol{g}_1^{Ag_i},\boldsymbol{g}_2^{Ag_i},\ldots\boldsymbol{g}_k^{Ag_i}\right\} \tag{3}$$

(a set of goals representing states of the world that the agent wishes to obtain; that is, states of the world that the agent wants to be true);

$$\boldsymbol{AZ}_{Ag_i}{=}_{def}\left\{\boldsymbol{\alpha}_1^{Ag_i},\boldsymbol{\alpha}_2^{Ag_i},\ldots\boldsymbol{\alpha}_v^{Ag_i}\right\} \tag{4}$$

(a set of actions representing the *elementary actions* that Ag_i is able to perform and that affect the real world; in general, with each action are associated preconditions - states of the world that guarantee its feasibility - and results, that is, states of the world resulting from its performance);

$$\boldsymbol{\Pi}_{Ag_i}{=}_{def}\left\{\boldsymbol{p}_1^{Ag_i},\boldsymbol{p}_2^{Ag_i},\ldots\boldsymbol{p}_u^{Ag_i}\right\} \tag{5}$$

(the Ag_i's plan library: a set of rules/prescriptions for aggregating agent actions); and

$$\boldsymbol{R}_{Ag_i}{=}_{def}\left\{\boldsymbol{r}_1^{Ag_i},\boldsymbol{r}_2^{Ag_i},\ldots\boldsymbol{r}_w^{Ag_i}\right\} \tag{6}$$

(a set of resources representing available tool or capacity to the agent, consisting of a material reserve).

Of course, the same *belief*, *goal*, *action*, *plan* or *resource* can belong to different agents (i.e., shared), unless we introduce intrinsic limits to these notions[2]. For example, for the goals we can say that g_k could be owned by Ag_i or by Ag_j and we would have: g_k^{Agi} or g_k^{Agj}.

We can say that an agent is able to obtain on its own behalf (at a certain time, t, in a certain environmental context, c[3]) its own goal, g_x^{Agi}, if it possesses the mental and practical attitudes to achieve that goal. In this case we can say that it has the power to achieve the goal, g_x^{Agi} applying the plan, p_x^{Agi}, (which can also coincide with a single elementary action).

In general, as usual [12, 13], we define a task τ, that is a couple

$$\boldsymbol{\tau}{=}_{def}\left(\boldsymbol{\alpha},\boldsymbol{g}\right). \tag{7}$$

in practice, we combine the goal g with the action α, necessary to obtain g, which may or may not be defined (in fact, indicating the achievement of a state of the world always implies also the application of some action).

2.2 Agent's Powers

Given the above agent's definition, we introduce the operator $\boldsymbol{Pow}(\boldsymbol{Ag_x},\boldsymbol{\tau},\boldsymbol{c},\boldsymbol{t})$ to indicate the power of $\boldsymbol{Ag_x}$ to achieve goal g through action α, in a certain context c at a certain time t. This power may or may not exist. In positive case, we will have:

[1] We introduce the symbol A $=_{def}$B to indicate that the symbol A is by definition associated with the expression B.

[2] For a more complete and detailed discussion of actions and plans (on their preconditions and results; on how the contexts may affect their effects; on their explicit or implicit conflicts, etc.), please refer to [18, 19].

[3] The context c defines the boundary conditions that can influence the other parameters of the indicated relationship. Different contexts can determine different outcomes of the *actions*, affect the agent's *beliefs* and even the agent's *goals* (for example, determine new ones or change their order of priority). To give a trivial example: being in different meteorological conditions or with a different force of gravity, so to speak, could strongly affect the results of the agent's actions, and/or have an effect on the agent's beliefs and/or on its own goals (changing their mutual priority or eliminating some and introducing new ones). In general, standard conditions are considered, i.e. default conditions that represent the usual situation in which agents operate: and the parameters (actions, beliefs, goals, etc.) to which we refer are generally referred to these standard values.

$$\boldsymbol{Pow}\left(\boldsymbol{Ag_x}, \boldsymbol{\tau}, \boldsymbol{c}, \boldsymbol{t}\right) = \boldsymbol{true} \tag{8}$$

that means that Ag_x has the ability (physical and cognitive) and the internal and/or external resources to achieve (or maintain) the state of the world corresponding with the goal g through the (elementary or complex) action (α or p) in the context c at the time t. We can similarly define an operator (*lack of power*: *LoPow*) in case it does not have this power:

$$\boldsymbol{LoPow}\left(\boldsymbol{Ag_x},\ \boldsymbol{\tau}, \boldsymbol{c}, \boldsymbol{t}\right) =_{\boldsymbol{def}} \neg \boldsymbol{Pow}\left(\boldsymbol{Ag_x},\ \boldsymbol{\tau}, \boldsymbol{c}, \boldsymbol{t}\right) \tag{9}$$

As we have just seen, we define the power of an agent with respect to a τ task, that is, with respect to the couple (action, state of the world). In this way we take into account, on the one hand, the fact that in many cases this couple is inseparable, i.e., the achievement of a certain state of the world is consequent (and expected) to be bound to the execution of a certain specific action (α) and to the possession of the resources ($r_1,..,r_n$) necessary for its execution. On the other hand, in this way we also take into consideration the case in which it is possible to predict the achievement of that state in the world with an action not necessarily defined *a priori* (therefore, in this case the action α in the τ pair would turn out to be undefined *a priori*). In the second case it would be possible to assign that power to the agent if it is able to obtain the indicated state of the world (g) regardless of the foreseeable (or expected) action to be applied (for example, it may be able to take different alternative actions to do this).

In any case, $Pow(Ag_x, \tau, c, t)$ implies that the goal (g) is *potentially active* for the Ag_x. It is always in relation to a goal (g) that an Ag_x has some "Power of/on".

It is important to emphasize that arguing that Ag_x has the power to perform a certain task τ means attributing to that agent the possession of certain characteristics and the consequent possibility of exercising certain specific actions. This leads to the indication of a high probability of success but not necessarily to the certainty of the desired result. In this regard we introduce a *Degree of Ability* (*DoA*), i.e. a number (included between 0 and 1) which expresses - given the characteristics possessed by the agent, the state of the world to be achieved and the context in which this takes place - the probability of successfully realization of the task.

So, we can generally say that if Ag_x has the power $Pow(Ag_x, \tau, c, t)$, then its *degree of ability* (*DoA*) exceeds a certain threshold (for example σ) considered of adequate value to ensure (on a theoretical rather than an experimental basis) the success of the task: in practice, if *DoA* >σ than the probability of success is high; so:

$$\left(\boldsymbol{Pow}\left(\boldsymbol{Ag_x}, \boldsymbol{\tau}, \boldsymbol{c}, \boldsymbol{t}\right) = \boldsymbol{true}\right) \rightarrow \boldsymbol{DoA}\left(\boldsymbol{Ag_x}, \boldsymbol{\tau}, \boldsymbol{c}, \boldsymbol{t}\right) > \boldsymbol{\sigma} \tag{10}$$

Where $\boldsymbol{A} \rightarrow \boldsymbol{B}$ means A implies B; and σ has a high value in the range (0,1).

In words: if Ag_x has the power to achieve the goal g then the agent's degree of ability (*DoA*) is above a defined threshold.

Similarly, we can define the absence of power in the realization of the task τ, by introducing a lower threshold (*?*), for which:

$$\left(\boldsymbol{LoPow}\left(\boldsymbol{Ag_x}, \boldsymbol{\tau}, \boldsymbol{c}, \boldsymbol{t}\right) = \boldsymbol{true}\right) \rightarrow \boldsymbol{DoA}\left(\boldsymbol{Ag_x}, \boldsymbol{\tau}, \boldsymbol{c}, \boldsymbol{t}\right) < \boldsymbol{\zeta} \tag{11}$$

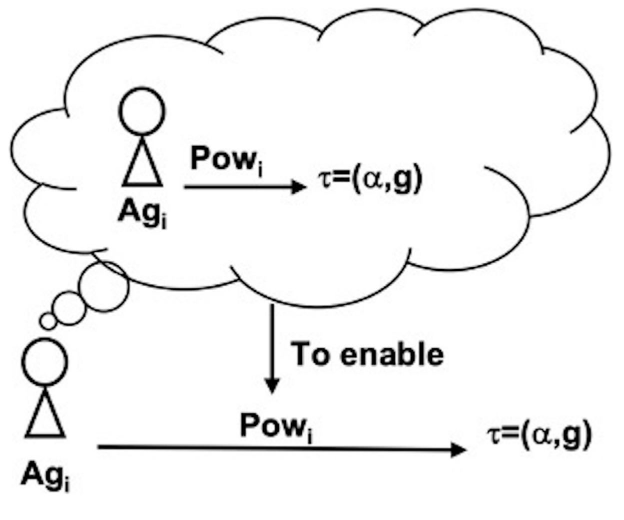


FIGURE 1 | *Ag_i* to really have the power to accomplish the task τ, it must believe that it possesses that power. This belief actually enables the real power of it to act.

In the cases in which $\zeta < DoA(Ag_x, \tau, c, t) < \sigma$ we are uncertain about Ag_x's power to accomplish the task τ .

We will see later the need to introduce probability thresholds.

3 SOCIAL DEPENDENCE

3.1 From Personal Powers to Social Dependence

Sociality presupposes a "common world", hence "interference": the action of one agent can favor (positive interference) or hamper/compromise the goals of another agent (negative interference). Since agents have limited personal powers, and compete for achieving their goals, they need social powers (that is, to have the availability of some of the powers collected from other agents). They also compete for resources (both material and social) and for having the power necessary for their goals.

3.2 Objective Dependence

Let us introduce the relevant concept of objective dependence [20–22]. Given $Ag_i, Ag_j \in AGT$; a set of tasks $\mathrm{T}=_{\boldsymbol{def}}\{\tau_1, \tau_2, \ldots \tau_l\}$; a set of contexts $\Gamma=_{\boldsymbol{def}}\{c_1, c_2, \ldots c_n\}$; and defined t_x the specific time interval x, we can define:

$$\begin{aligned}\boldsymbol{ObjDep}\left(\boldsymbol{Ag_i},\ \boldsymbol{Ag_j},\ \boldsymbol{\tau_k}, \boldsymbol{c_k},\ \boldsymbol{t_k}\right) =_{\boldsymbol{def}} \boldsymbol{LoPow}\left(\boldsymbol{Ag_i},\ \boldsymbol{\tau_k}, \boldsymbol{c_k},\ \boldsymbol{t_k}\right) \\ \cap \boldsymbol{Pow}\left(\boldsymbol{Ag_j},\ \boldsymbol{\tau_k}, \boldsymbol{c_k},\ \boldsymbol{t_k}\right)\end{aligned} \tag{12}$$

where $\tau_k \in \mathrm{T}$, $c_k \in \Gamma$; and the time interval is t_k.

It is the combination of a lack of Power (*LoPow*) of one agent (Ag_i), relative to one of its own tasks/goal (τ_k); and the corresponding Power (*Pow*) of another agent (Ag_j), under certain specific contextual (c_k) and temporal (t_k) conditions. It is the result of some interference between the two agents. It is "objective" in the sense that it holds independently of the involved agents' awareness/beliefs and wants.

In words: an agent Ag_i has an *Objective Dependence Relationship with respect to a task* τ_k with agent Ag_j if for realizing τ_k, regardless of its awareness, are necessary actions, plans and/or resources that are owned by Ag_j and not owned (or not available, or less convenient to use) by Ag_i.

More in general, Ag_i has an *Objective Dependence Relationship* with Ag_j if for achieving at least one of its tasks τ_k, with $g_k \in GOAL_{Agi}$, are necessary actions, plans and/or resources that are owned by Ag_j and not owned (or not available or less convenient to use) by Ag_i.

3.3 Awareness as Acquisition or Loss of Powers

Given that to decide to pursue a goal, a cognitive agent must believe/assume (at least with some degree of certainty) that it has that power (*sense of competence*, *self-confidence*, *know-how* and *expertise/skills*), then it does not really have that power if it does not know it has that power (**Figure 1**). Thus, the meta-cognition of agents' internal powers and the awareness of their external resources empower them (enable them to make their "power" usable).

This awareness allows an agent to use this power also for other agents in the networks of dependence: social power (who could depend on it: power relations over others, relational capital, exchanges, collaborations, etc.).

Acquiring power and therefore autonomy (on that dimension) and power over other agents can therefore simply be due to the awareness of this power and not necessarily to the acquisition of external resources or skills and competences (learning): in fact, it is a *cognitive power*.

3.4 Types of Objective Dependence

A very relevant distinction is the case of a *two-way dependence* between agents (*bilateral dependence*). There are two possible kinds of bilateral dependence (to simplify, we make the task coincide with the goal: $\tau_k = g_k$):

- *Reciprocal Dependence*, in which Ag_i depends on Ag_j as for its goal g_1^{Agi}, and Ag_j depends on Ag_i as for its own goal g_2^{Agj} (with $g_1 \neq g_2$). They need each other's action, but for two different personal goals. This is the basis of a pervasive and fundamental form of human (and possibly artificial) interaction: *Social Exchange*. In this kind of interaction Ag_i performs an action useful-for/required by Ag_j for $\mathrm{g}_2^{\mathrm{Agj}}$, to obtain an action by Ag_j useful for its personal goal g_1^{Agi}. Ag_i and Ag_j are not co-interested in the fulfillment of the goal of the other.

- *Mutual Dependence*, in which Ag_i depends on Ag_j as for its goal g_k^{Agi}, and Ag_j depends on Ag_i as for the same goal g_k^{Agi} (both have the goal g_k). They have a common goal, and they depend on each other as for this shared goal. When this situation is known by Ag_i and Ag_j, it becomes the basis of *true cooperation*. Ag_i and Ag_j are co-interested in the success of the goal of the other (instrumental to g_k). Ag_i helps Ag_j to pursue her own goal, and vice versa. In this condition to defeat is not rational; it is self-defeating.

In the case in which an agent Ag_i depends on more than one other agent, it is possible to identify several typical objective dependence patterns. Just to name a few relevant examples, very interesting are the *OR-Dependence*, a disjunctive composition of dependence relations, and the *AND-dependence*, a conjunction of dependence relations.

In the first pattern (*OR-Dependence*) the agent Ag_i can potentially achieve its goal through the action of *just one* of the agents with which it is in that relationship. In the second pattern (*AND-dependence*) the agent Ag_i can potentially achieve its goal through the action of *all* the agents with which it is in that relationship (Ag_i needs all the other agents in that relationship).

The Dependence Network *determines* and *predicts* partnerships and coalitions formation, competition, cooperation, exchange, functional structure in organizations, rational and effective communication, and *negotiation power*. Dependence networks are very dynamic and *unpredictable*. In fact, they change by changing an individual goal; by changing individual resources or skills; by the exit or entrance of a new agent (open world); by acquaintance and awareness (see later); by indirect power acquisition.

3.5 Objective and Subjective Dependence

Objective Dependence constitutes the basis of all social interaction, the reason for society; it motivates cooperation in its different kinds. But objective dependence relationships that are the basis of adaptive social interactions, are not enough for predicting them. *Subjective dependence* is needed (that is, the dependence relationships that the agents know or at least believe).

We introduce the $SubjDep_{Ag_i}(Ag_i, Ag_j, \tau_k, c, t)$ that represents the Ag_i's point of view with respect its dependence relationships (for simplicity we neglect time and context). Formally:

$$\boldsymbol{SubjDep_{Ag_i}\left(Ag_i,\ Ag_j,\ \tau_k\right) =_{def} Bel_{Ag_i}\left(ObjDep\left(Ag_i,\ Ag_j,\ \tau_k\right)\right)}$$
$$\boldsymbol{Bel_{Ag_i}\left(ObjDep\left(Ag_i,\ Ag_j,\ \tau_k\right)\right) =_{def} Bel_{Ag_i}\left(LoPow\left(Ag_i,\ \tau_k\right)\right) \wedge Bel_{Ag_i}\left(Pow\left(Ag_j,\ \tau_k\right)\right)} \qquad (13)$$

where $Ag_i, Ag_j \in AGT$ and $Bel_{Ag_i}(\tau_k = (\alpha_k, g_k))$ and $Bel_{Ag_i}((\alpha_k \in AZ_{Ag_j}) \cap (\alpha_k \notin AZ_{Ag_i}) \cap (g_k \in GOAL_{Ag_i}))$. That is, the relationship of dependence as we have introduced it in an objective way becomes aware of the single agent when it becomes its own belief.

When we introduce the concept of subjective view of dependence relationships, as we have just done with the *SubjDep*, we are considering what our agent believes and represents about its own dependence on others. Vice versa, it should also be analyzed what our agent believes about the dependence of other agents in the network (how it represents the dependencies of other agents). We can therefore formally introduce the formula for each Ag_i in potential relationship with other agents of the AGT set:

$$\boldsymbol{Bel_{Ag_i}\left(SubjDep_{Ag_j}\left(Ag_j, Ag_i, \tau_k\right)\right) =_{def} Bel_{Ag_i}\left(Bel_{Ag_j}\left(LoPow\left(Ag_j,\ \tau_k\right)\right) \wedge Bel_{Ag_j}\left(Pow\left(Ag_i,\ \tau_k\right)\right)\right)} \qquad (14)$$

where $Ag_i, Ag_j \in AGT$ and $Bel_{Ag_i}(Bel_{Ag_j}(\tau_k = (\alpha_k, g_k)))$ with $Bel_{Ag_i}(Bel_{Ag_j}((\alpha_k \in AZ_{Ag_i}) \wedge (\alpha_k \notin AZ_{Ag_j}) \wedge (g_k \in GOAL_{Ag_j})))$. So resuming we can say:

1) The *objective dependence* says who needs who for what in each society (although perhaps ignoring this). This dependence has already the power of establishing certain asymmetric relationships in a potential market, and it determines the actual success or failure of the reliance and transaction.
2) The *subjective (believed) dependence*, says who is believed to be needed by who. This dependence is what potentially determines relationships in a real market and settles on the *negotiation power* (see §3); but it might be illusory and wrong, and one might rely upon unable agents, while even being autonomously able to do as needed.

If the world knowledge would be perfect for all the agents, the above-described objective dependence would be a *common belief* (a belief possessed by all agents) about the real state of the world: there would be no distinction between objective and subjective dependence.

In fact, however, the important relationship is the network of dependence believed by each agent. In other words, we cannot only associate to each agent a set of goals, actions, plans and resources, but we must evaluate these sets as believed by each agent (*the subjective point of view*), also considering that they would be partial, different each of others, sometime wrong, with different degrees and values, and so on. In more practical terms, each agent will have a different (subjective) representation of the dependence network and of its positioning: it is from this subjective view of the world that the actions and decisions of the agents will be guided.

So, we introduce the $Bel_{Ag_i}(GOAL_{Ag_z})$ that means the Goal set of Ag_z believed by Ag_i. The same for $Bel_{Ag_i}(AZ_{Ag_z})$, $Bel_{Ag_i}(\Pi_{Ag_z})$, $Bel_{Ag_i}(R_{Ag_z})$, and also for $Bel_{Ag_i}(BEL_{Ag_z})$. In practice, the dependence relationships should be re-modulated based on the agents' subjective interpretation.

In a first approximation each agent should correctly believe the sets it has, while it could mismatch the sets of other agents[4]. In formulas:

$$\boldsymbol{Bel_{Ag_i}\left(GOAL_{Ag_i}\right) = GOAL_{Ag_i}} \tag{15}$$

$$\boldsymbol{Bel_{Ag_i}\left(AZ_{Ag_i}\right) = AZ_{Ag_i}} \tag{16}$$

$$\boldsymbol{Bel_{Ag_i}\left(\Pi_{Ag_i}\right) = \Pi_{Ag_i}} \tag{17}$$

$$\boldsymbol{Bel_{Ag_i}\left(R_{Ag_i}\right) = R_{Ag_z}} \tag{18}$$

$$\boldsymbol{Bel_{Ag_i}\left(BEL_{Ag_i}\right) = BEL_{Ag_i}} \tag{19}$$

$$(\forall \boldsymbol{Ag_i} \in \boldsymbol{AGT}).$$

We define $Dependence - Network(AGT, t, c)$ the set of dependence relationships (both subjective and objective) among the agents included in AGT set (also in this case we neglect time and context):

$$\begin{gathered}\boldsymbol{Dependence - Network(AGT) =_{def}} \\ \boldsymbol{\left(ObjDep\left(Ag_i, Ag_j, \tau_k\right)\right.} \\ \boldsymbol{\bigcup_{i=1}^{n} SubjDep_{Ag_i}\left(Ag_i, Ag_j, \tau_k\right)} \\ \boldsymbol{\bigcup_{i=1}^{n} \bigcup_{j=1}^{m} Bel_{Ag_i}\left(SubjDep_{Ag_j}\left(Ag_j, Ag_i, \tau_k\right)\right)} \\ \forall\left(\boldsymbol{Ag_i, Ag_j}\right) \in \boldsymbol{AGT}\end{gathered} \tag{20}$$

For each couple (Ag_i, Ag_j) in $ObjDep(Ag_i, Ag_j, \tau_k)$ *with* $\tau_k =_{\boldsymbol{def}} (\alpha_k, g_k)$ *we have:* $(g_k \in GOAL_{Ag_i}) \wedge (\alpha_k \in AZ_{Ag_j})$.

For each couple (Ag_i, Ag_j) in $SubjDep_{Ag_i}(Ag_i, Ag_j, \tau_k)$, *with* $Bel_{Ag_i}(\tau_k =_{\boldsymbol{def}} (\alpha_k, g_k))$ *we have* : $Bel_{Ag_i}(g_k \in GOAL_{Ag_i}) \wedge Bel_{Ag_i}(\alpha_k \in AZ_{Ag_j})$.

For each couple (Ag_i, Ag_j) in $Bel_{Ag_i}(SubjDep_{Ag_j}(Ag_j, Ag_i, \tau_k))$, *with* $Bel_{Ag_i}(Bel_{Ag_j}(\tau_k =_{\boldsymbol{def}} (\alpha_k, g_k)))$, *we have*: $Bel_{Ag_i}(Bel_{Ag_j}(g_k \in GOAL_{Ag_j}) \wedge Bel_{Ag_j}(\alpha_k \in AZ_{Ag_i}))$.

The three relational levels indicated (*objective*, *subjective* and *subjective dependence believed by others*) in the network of dependence defined above, determine the basic relationships to initiate even minimally informed negotiation processes. The only level always present is the objective one (even if the fact that the agents are aware of it is decisive). The others may or may not be present (and their presence or absence determines different behaviors in the achievement of the goals by the various agents and consequent successes or failures).

3.6 Relevant Relationships within a Dependence Network

The dependence network (**Formula 20**) collecting all the indicated relationships represents a complex articulation of objective situations and subjective points of view of the various agents that are part of it, with respect to the reciprocal powers to obtain tasks. However, it is interesting to investigate the situations of greatest interest within the defined network. Let's see some of them below.

3.6.1 Comparison Between Agent's Point of View and Reality

A first consideration concerns the *coincidence or otherwise of the subjective points of view of the agents with respect to reality* (objective dependence).

That is, given two agents, $(Ag_i, Ag_j) \in AGT$, the subjective dependence of Ag_i with respect to Ag_j for the task τ may or may not coincide with the objective dependence. So, remembering that:

$SubjDep_{Ag_i}(Ag_i, Ag_j, \tau) =_{\boldsymbol{def}} Bel_{Ag_i}(ObjDep(Ag_i, Ag_j, \tau))$ and calling $ObjDep_{i,j,\tau} =_{\boldsymbol{def}} ObjDep(Ag_i, Ag_j, \tau)$, we can have:

[4] Our beliefs can be considered with true/false values or included in a range (0,1). In this second case it will be relevant to consider a threshold value beyond which the belief will be considered valid even if not completely certain.

[5] Of course it can also happen that an agent does not have a good perception of its own characteristics/beliefs/goals/etc..

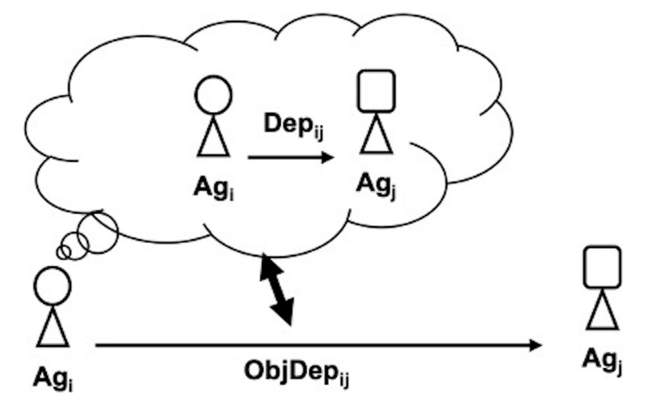


FIGURE 2 | Dependence of Ag$_i$ by Ag$_j$ on the task τ. Comparison on how it is believed by Ag$_i$ and objective reality.

$$\boldsymbol{Bel_{Ag_i}\left(ObjDep_{i,j,\tau}\right) = ObjDep_{i,j,\tau}} \tag{21}$$

the subjective dependence believed by Ag_i with respect to Ag_j coincides with reality, that is, it is objective; or

$$\boldsymbol{Bel_{Ag_i}\left(ObjDep_{i,j,\tau}\right) \neq ObjDep_{i,j,\tau}} \tag{22}$$

the subjective dependence believed by Ag_i with respect to Ag_j does not coincide with reality, that is, it is not objective.[6]

By defining A↔B as the *comparison*[7] between the expressions A and B, the two cases above described (**formulas 21**, **22**) are the result of the following comparison (see **Figure 2**):

$$\boldsymbol{Bel_{Ag_i}\left(ObjDep\left(Ag_i, Ag_j, \tau\right)\right) \leftrightarrow ObjDep\left(Ag_i, Ag_j, \tau\right)} \tag{23}$$
$$\boldsymbol{\left(Ag_i, Ag_j\right) \in AGT}$$

3.6.2 Comparison Among Points of View of Different Agents

What Ag_i believes about Ag_j's potential subjective dependencies (from various agents in the network, including Ag_k third-party agents, and on various tasks in T) may or may not coincide with the subjective dependencies actually believed by Ag_j, where $(Ag_i, Ag_j, Ag_k) \in AGT$.

And vice versa, what Ag_j believes about Ag_i's subjective dependence (on the various agents in the network, including Ag_k third-party agents, and on various tasks in T) may or may not coincide with the subjective dependence of Ag_i (and the various Ag_k third-party agents); furthermore, one can compare these subjective beliefs and dependencies with objective dependence and verify or not the coincidence. This is divided into the following interesting combinations.

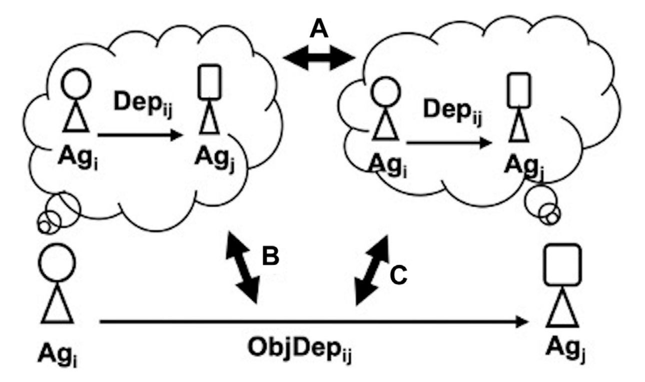


FIGURE 3 | Dependence of Ag$_i$ from Ag$_j$ on the task τ. **(A)** comparison on how it is believed by Ag$_i$ and by Ag$_j$; **(B)** comparison on how it is believed by Ag$_i$ and objective reality; **(C)** comparison on how it is believed by Ag$_j$ and objective reality.

Comparison between what Ag_i believes about the dependence of Ag_i by Ag_j $(Bel_{Ag_i}(ObjDep_{i,j,\tau}))$ and what Ag_j believes about the same dependence $(Bel_{Ag_j}(ObjDep_{i,j,\tau}))$: So, Ag_i and Ag_j can believe the same thing $(Bel_{Ag_i}(ObjDep_{i,j,\tau}) = Bel_{Ag_j}(ObjDep_{i,j,\tau}))$, or not $(Bel_{Ag_i}(ObjDep_{i,j,\tau}) \neq Bel_{Ag_j}(ObjDep_{i,j,\tau}))$.

In the first case $(Bel_{Ag_i}(ObjDep_{i,j,\tau}) = Bel_{Ag_j}(ObjDep_{i,j,\tau}))$, this situation may coincide with the reality $(Bel_{Ag_i}(ObjDep_{i,j,\tau}) = Bel_{Ag_j}(ObjDep_{i,j,\tau}) = ObjDep_{i,j,\tau})$, or not $(Bel_{Ag_i}(ObjDep_{i,j,\tau}) = Bel_{Ag_j}(ObjDep_{i,j,\tau}) \neq ObjDep_{i,j,\tau})$.

In the second case, $(Bel_{Ag_i}(ObjDep_{i,j,\tau}) \neq Bel_{Ag_j}(ObjDep_{i,j,\tau}))$, the point of view of Ag_i may coincide with reality $(Bel_{Ag_i}(ObjDep_{i,j,\tau}) = ObjDep_{i,j,\tau})$ and therefore does not correspond to the real the Ag_j's point of view; or Ag_j's point of view coincides with reality $(Bel_{Ag_j}(ObjDep_{i,j,\tau}) = ObjDep_{i,j,\tau})$, and therefore Ag_i's point of view does not correspond to reality; or finally neither of the two points of view (of Ag_i and Ag_j) coincide with reality: $Bel_{Ag_i}(ObjDep_{i,j,\tau}) \neq ObjDep_{i,j,\tau}$ and at the same time $Bel_{Ag_j}(ObjDep_{i,j,\tau}) \neq ObjDep_{i,j,\tau}$.

That is, the comparisons are in this case expressed by (see **Figure 3**):

$$\boldsymbol{\left(Bel_{Ag_i}\left(ObjDep\left(Ag_i, Ag_j, \tau\right)\right) \leftrightarrow Bel_{Ag_j}\left(ObjDep\left(Ag_i, Ag_j, \tau\right)\right)\right) \wedge}$$
$$\boldsymbol{\left(Bel_{Ag_i}\left(ObjDep\left(Ag_i, Ag_j, \tau\right)\right) \leftrightarrow ObjDep\left(Ag_i, Ag_j, \tau\right)\right) \wedge}$$
$$\boldsymbol{\left(Bel_{Ag_j}\left(ObjDep\left(Ag_i, Ag_j, \tau\right)\right) \leftrightarrow ObjDep\left(Ag_i, Ag_j, \tau\right)\right)} \tag{24}$$
$$\left(Ag_i, Ag_j\right) \in AGT$$

Another case is the comparison between Ag_j's subjective dependence on Ag_i for a task τ'∈ T $(Bel_{Ag_j}(ObjDep_{j,i,\tau'}))$ and what Ag_i believes about this dependence $(Bel_{Ag_i}(ObjDep_{j,i,\tau'}))$: in this case it is Ag_j who thinks it depends on Ag_i. We therefore want to compare this subjective dependence with what the agent to whom it is addressed (i.e. the agent Ag_i) believes on its content: $(Bel_{Ag_i}(ObjDep_{j,i,\tau'}))$. Also in this case there can be coincidence $(Bel_{Ag_j}(ObjDep_{j,i,\tau'}) = Bel_{Ag_i}(ObjDep_{j,i,\tau'}))$ or not $(Bel_{Ag_j}(ObjDep_{j,i,\tau'}) \neq Bel_{Ag_i}(ObjDep_{j,i,\tau'}))$.

[6] The fact of being aware of one's own goals is of absolute importance for an agent as it determines its subjective dependence which, as we will see, is the basis of its behavior.

[7] As we have defined the dependence, this non-coincidence may depend on different factors: wrong attribution of one's own powers or the powers of the other agent.

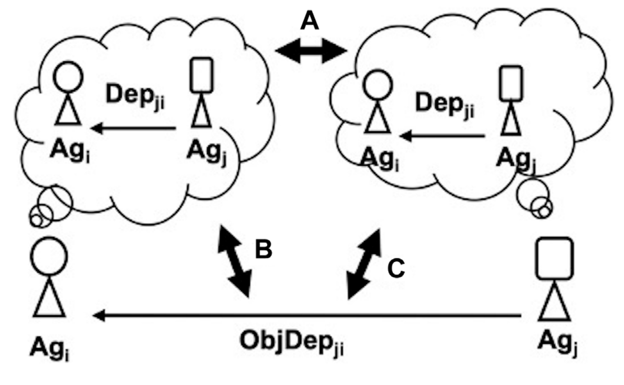


FIGURE 4 | : Dependence of Ag$_j$ from Ag$_i$ on the task τ'. **(A)** comparison on how it is believed by Ag$_i$ and by Ag$_j$; **(B)** comparison on how it is believed by Ag$_i$ and objective reality; **(C)** comparison on how it is believed by Ag$_j$ and objective reality.

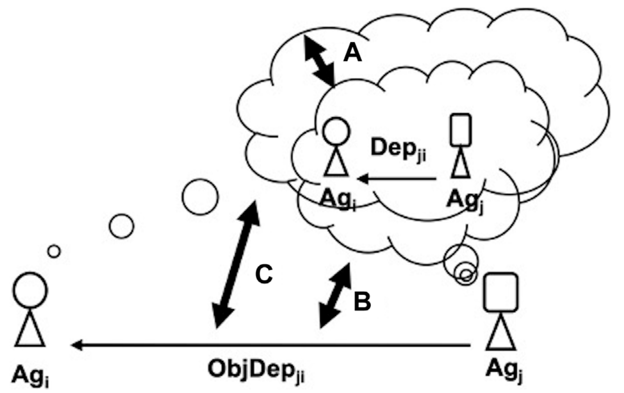


FIGURE 5 | Dependence of Ag$_j$ from Ag$_i$ on the task τ'. **(A)** comparison on how it is believed by Ag$_j$ and how Ag$_i$ believes it is believed by Ag$_j$; **(B)** comparison on how it is believed by Ag$_j$ and objective reality; **(C)** comparison on how Ag$_i$ believes it is believed by Ag$_j$ and objective reality.

For both of these situations we can further compare these two cases with objective reality.

In the first case, $(Bel_{Ag_j}(ObjDep_{j,i,\tau'}) = Bel_{Ag_i}(ObjDep_{j,i,\tau'}))$ we can have coincidence with $ObjDep_{j,i,\tau'}$: $(Bel_{Ag_j}(ObjDep_{j,i,\tau'}) = Bel_{Ag_i}(ObjDep_{j,i,\tau'}) = ObjDep_{j,i,\tau'})$, that not: $(Bel_{Ag_j}(ObjDep_{j,i,\tau'}) = Bel_{Ag_i}(ObjDep_{j,i,\tau'}) \neq ObjDep_{j,i,\tau'})$.

In the second case, $(Bel_{Ag_j}(ObjDep_{j,i,\tau'}) \neq Bel_{Ag_i}(ObjDep_{j,i,\tau'}))$ can coincide with reality the point of view of Ag_i $(Bel_{Ag_i}(ObjDep_{j,i,\tau'}) = ObjDep_{j,i,\tau'})$ and therefore does not correspond to the real Ag_j's point of view $(Bel_{Ag_j}(ObjDep_{j,i,\tau'}) \neq ObjDep_{j,i,\tau'})$; or the point of view of Ag_j $(Bel_{Ag_j}(ObjDep_{j,i,\tau'}) = ObjDep_{j,i,\tau'})$ coincides with reality and therefore does not correspond to the real the Ag_i's point of view $(Bel_{Ag_j}(ObjDep_{j,i,\tau'}) = ObjDep_{j,i,\tau'})$; or finally neither of the two points of view (of Ag_i and Ag_j) coincide with the real: $Bel_{Ag_i}(ObjDep_{j,i,\tau'}) \neq ObjDep_{j,i,\tau'}$ and at the same time $Bel_{Ag_j}(ObjDep_{j,i,\tau'}) \neq ObjDep_{j,i,\tau'}$.

That is, the comparisons are in this case expressed by (see **Figure 4**):

$$\begin{gathered}
\left(\mathbf{Bel}_{\mathbf{Ag}_i}\left(\mathbf{ObjDep}\left(\mathbf{Ag}_j, \mathbf{Ag}_i, \boldsymbol{\tau}'\right)\right) \leftrightarrow \mathbf{Bel}_{\mathbf{Ag}_j}\left(\mathbf{ObjDep}\left(\mathbf{Ag}_j, \mathbf{Ag}_i, \boldsymbol{\tau}'\right)\right)\right) \wedge \\
\left(\mathbf{Bel}_{\mathbf{Ag}_i}\left(\mathbf{ObjDep}\left(\mathbf{Ag}_j, \mathbf{Ag}_i, \boldsymbol{\tau}'\right)\right) \leftrightarrow \mathbf{ObjDep}\left(\mathbf{Ag}_j, \mathbf{Ag}_i, \boldsymbol{\tau}'\right)\right) \wedge \\
\left(\mathbf{Bel}_{\mathbf{Ag}_j}\left(\mathbf{ObjDep}\left(\mathbf{Ag}_j, \mathbf{Ag}_i, \boldsymbol{\tau}'\right)\right) \leftrightarrow \mathbf{ObjDep}\left(\mathbf{Ag}_j, \mathbf{Ag}_i, \boldsymbol{\tau}'\right)\right) \quad (25) \\
\left(Ag_i, Ag_j\right) \in AGT
\end{gathered}$$

3.6.3 Comparison Among Agents' Points of View on Others' Points of View and Reality

Another interesting situation is the comparison between what Ag_i believes of Ag_j's subjective dependence on itself: $Bel_{Ag_i}(Bel_{Ag_j}(ObjDep_{j,i,\tau'})$ with Ag_j's belief of this dependence: $Bel_{Ag_j}(ObjDep_{j,i,\tau'})$. Also in this case we have: Ag_j can believe that it depends on Ag_i and at the same time Ag_i believe the same thing $Bel_{Ag_j}(ObjDep_{j,i,\tau'}) = Bel_{Ag_i}(Bel_{Ag_j}(ObjDep_{j,i,\tau'}))$, i.e. Ag_i believes that Ag_j believes that it depends on Ag_i) or not $Bel_{Ag_j}(ObjDep_{j,i,\tau'}) \neq Bel_{Ag_i}(Bel_{Ag_j}(ObjDep_{j,i,\tau'}))$.

In the first case $(Bel_{Ag_j}(ObjDep_{j,i,\tau'}) = Bel_{Ag_i}(Bel_{Ag_j}(ObjDep_{j,i,\tau'})))$, the situation can coincide with reality $(Bel_{Ag_j}(ObjDep_{j,i,\tau'}) = Bel_{Ag_i}(Bel_{Ag_j}(ObjDep_{j,i,\tau'})) = ObjDep_{j,i,\tau'})$, or not $(Bel_{Ag_j}(ObjDep_{j,i,\tau'}) = Bel_{Ag_i}(Bel_{Ag_j}(ObjDep_{j,i,\tau'})) \neq ObjDep_{j,i,\tau'})$.

In the second case $(Bel_{Ag_j}(ObjDep_{j,i,\tau'}) \neq Bel_{Ag_i}(Bel_{Ag_j}(ObjDep_{j,i,\tau'})))$, the point of view of Ag_j can coincide with reality $(Bel_{Ag_j}(ObjDep_{j,i,\tau'}) = ObjDep_{j,i,\tau'})$ and therefore Ag_i's point of view does not correspond to the real; or Ag_i's view point coincides with reality $(Bel_{Ag_i}(Bel_{Ag_j}(ObjDep_{j,i,\tau'}))) = ObjDep_{j,i,\tau'})$, and therefore Ag_j's point of view does not correspond to the real; or finally, neither of the two points of view (of Ag_i and Ag_j) coincide with reality: $(Bel_{Ag_j}(ObjDep_{j,i,\tau'}) \neq ObjDep_{j,i,\tau'}$ and at the same time $(Bel_{Ag_i}(Bel_{Ag_j}(ObjDep_{j,i,\tau'}))) \neq ObjDep_{j,i,\tau'}$.

That is, the comparisons are in this case expressed by (see **Figure 5**):

$$\begin{gathered}
\left(\mathbf{Bel}_{\mathbf{Ag}_i}\left(\mathbf{Bel}_{\mathbf{Ag}_j}\left(\mathbf{ObjDep}\left(\mathbf{Ag}_j, \mathbf{Ag}_i, \boldsymbol{\tau}'\right)\right) \leftrightarrow \mathbf{Bel}_{\mathbf{Ag}_j}\left(\mathbf{ObjDep}\left(\mathbf{Ag}_j, \mathbf{Ag}_i, \boldsymbol{\tau}'\right)\right)\right)\right) \wedge \\
\left(\mathbf{Bel}_{\mathbf{Ag}_i}\left(\mathbf{Bel}_{\mathbf{Ag}_j}\left(\mathbf{ObjDep}\left(\mathbf{Ag}_j, \mathbf{Ag}_i, \boldsymbol{\tau}'\right)\right) \leftrightarrow \mathbf{ObjDep}\left(\mathbf{Ag}_j, \mathbf{Ag}_i, \boldsymbol{\tau}'\right)\right)\right) \wedge \\
\left(\mathbf{Bel}_{\mathbf{Ag}_j}\left(\mathbf{ObjDep}\left(\mathbf{Ag}_j, \mathbf{Ag}_i, \boldsymbol{\tau}'\right)\right) \leftrightarrow \mathbf{ObjDep}\left(\mathbf{Ag}_j, \mathbf{Ag}_i, \boldsymbol{\tau}'\right)\right) \quad (26) \\
\left(Ag_i, Ag_j\right) \in AGT
\end{gathered}$$

In the same but reversed situation, is interesting the comparison between Ag_i's subjective dependence on Ag_j $(Bel_{Ag_i}(ObjDep_{i,j,\tau}))$ and what Ag_j believes about this subjective belief of Ag_i $(Bel_{Ag_j}(Bel_{Ag_i}(ObjDep_{i,j,\tau})))$: Ag_i may believe that it depends on Ag_j for the task τ and at the same time Ag_j believe that Ag_i believes this thing $(Bel_{Ag_i}(ObjDep_{i,j,\tau})= Bel_{Ag_j}(Bel_{Ag_i}(ObjDep_{i,j,\tau})))$ or not $(Bel_{Ag_i}(ObjDep_{i,j,\tau})\neq Bel_{Ag_j}(Bel_{Ag_i}(ObjDep_{i,j,\tau})))$.

In the first case, this situation may coincide with reality $(Bel_{Ag_i}(ObjDep_{i,j,\tau}) = Bel_{Ag_j}(Bel_{Ag_i}(ObjDep_{i,j,\tau})) = ObjDep_{i,j,\tau})$, or not $(Bel_{Ag_i}(ObjDep_{i,j,\tau})= Bel_{Ag_j}(Bel_{Ag_i}(ObjDep_{i,j,\tau})) \neq ObjDep_{i,j,\tau})$.

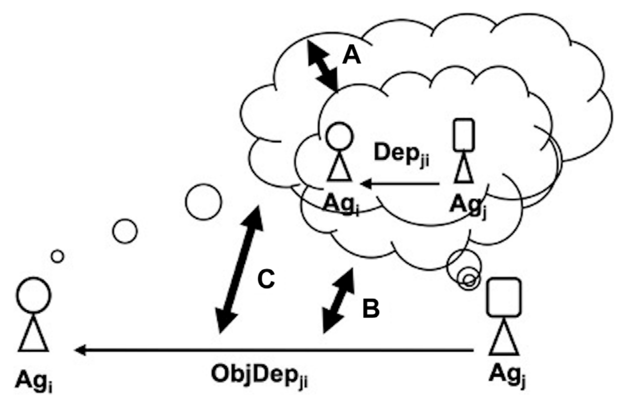


FIGURE 6 | Dependence of Ag$_i$ from Ag$_j$ on the task τ. **(A)** comparison on how it is believed by Ag$_i$ and how Ag$_j$ believes it is believed by Ag$_i$; **(B)** comparison on how it is believed by Ag$_i$ and the objective reality; **(C)** comparison on how Ag$_j$ believes it is believed by Ag$_i$ and objective reality.

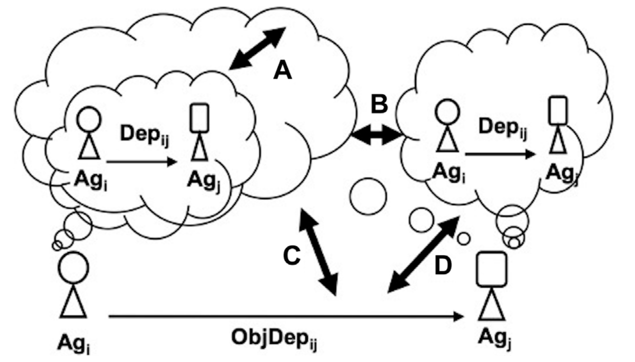


FIGURE 7 | Dependence of Ag$_i$ from Ag$_j$ on the task τ. **(A)** comparison on how it is believed by Ag$_i$ and how Ag$_j$ believes it is believed by Ag$_i$; **(B)** comparison on how it is believed by Ag$_j$ and the objective reality; **(C)** comparison on how Ag$_j$ believes it is believed by Ag$_i$ and how it is believed by Ag$_j$; **(D)** comparison on how Ag$_j$ believes it is believed by Ag$_i$ and the objective reality.

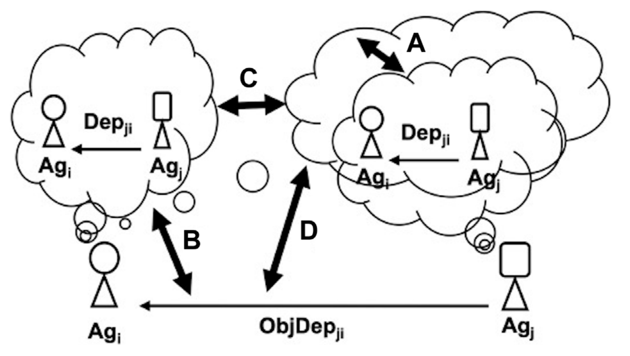


FIGURE 8 | Dependence of Ag$_j$ from Ag$_i$ on the task τ'. **(A)** comparison on how it is believed by Ag$_j$ and how Ag$_i$ believes it is believed by Ag$_j$; **(B)** comparison on how it is believed by Ag$_i$ and objective reality; **(C)** comparison on how Ag$_i$ believes it is believed by Ag$_j$ and how it is believed by Ag$_i$; **(D)** comparison on how Ag$_i$ believes it is believed by Ag$_j$ and objective reality.

In the second case, the point of view of Ag_i may coincide with reality ($Bel_{Ag_i}(ObjDep_{i,j,\tau})= ObjDep_{i,j,\tau}$) and therefore does not correspond to the real Ag_j's point of view; or Ag_j's point of view coincides with reality ($Bel_{Ag_j}(Bel_{Ag_i}(ObjDep_{i,j,\tau}) = ObjDep_{i,j,\tau}$), and therefore Ag_i's point of view does not correspond to reality; or finally neither of the two points of view (of Ag_i and Ag_j) coincides with reality: $Bel_{Ag_i}(ObjDep_{i,j,\tau})\neq ObjDep_{i,j,\tau}$ and at the same time $Bel_{Ag_j}(Bel_{Ag_i}(ObjDep_{i,j,\tau}) \neq ObjDep_{i,j,\tau}$.

That is, the comparisons are in this case expressed by (see **Figure 6**):

$$\begin{gathered}
\left(Bel_{Ag_j}(Bel_{Ag_i}\left(ObjDep\left(Ag_i, Ag_j, \tau\right)\right) \leftrightarrow Bel_{Ag_i}\left(ObjDep\left(Ag_i, Ag_j, \tau\right)\right)\right) \wedge \\
\left(Bel_{Ag_j}\left(Bel_{Ag_i}\left(ObjDep\left(Ag_i, Ag_j, \tau\right)\right) \leftrightarrow ObjDep\left(Ag_i, Ag_j, \tau\right)\right) \wedge \\
\left(Bel_{Ag_i}\left(ObjDep\left(Ag_i, Ag_j, \tau\right)\right) \leftrightarrow ObjDep\left(Ag_i, Ag_j, \tau\right)\right) \qquad (27) \\
\left(Ag_i, Ag_j\right) \in AGT
\end{gathered}$$

3.6.4 More Complex Comparisons

In this case we consider the comparison between the subjective dependence of Ag_i on Ag_j ($Bel_{Ag_i}(ObjDep_{i,j,\tau})$) and what Ag_j believes of this subjective belief of Ag_i ($Bel_{Ag_j}(Bel_{Ag_i}(ObjDep_{i,j,\tau}))$) also in relation to what Ag_j believes directly of this dependence ($Bel_{Ag_j}(ObjDep_{i,j,\tau})$): Ag_j may believe that its belief on $ObjDep_{i,j,\tau}$ coincides, or not, with Ag_i's belief on the same dependence ($ObjDep_{i,j,\tau}$), that is: $Bel_{Ag_j}(ObjDep_{i,j,\tau}) = Bel_{Ag_j}(Bel_{Ag_i}(ObjDep_{i,j,\tau}))$ or not: $Bel_{Ag_j}(ObjDep_{i,j,\tau}) \neq Bel_{Ag_j}(Bel_{Ag_i}(ObjDep_{i,j,\tau}))$.

In both cases the comparison with the real situation is also of interest (see **Figure 7**):

$$\begin{gathered}
\left(Bel_{Ag_j}(Bel_{Ag_i}\left(ObjDep\left(Ag_i, Ag_j, \tau\right)\right) \leftrightarrow Bel_{Ag_i}\left(ObjDep\left(Ag_i, Ag_j, \tau\right)\right)\right) \wedge \\
\left(Bel_{Ag_j}(Bel_{Ag_i}\left(ObjDep\left(Ag_i, Ag_j, \tau\right)\right) \leftrightarrow Bel_{Ag_j}\left(ObjDep\left(Ag_i, Ag_j, \tau\right)\right)\right) \wedge \\
\left(Bel_{Ag_j}\left(Bel_i\left(ObjDep\left(Ag_i, Ag_j, \tau\right)\right) \leftrightarrow ObjDep\left(Ag_i, Ag_j, \tau\right)\right) \wedge \\
\left(Bel_{Ag_j}\left(ObjDep\left(Ag_i, Ag_j, \tau\right)\right) \leftrightarrow ObjDep\left(Ag_i, Ag_j, \tau\right)\right) \qquad (28) \\
\left(Ag_i, Ag_j\right) \in AGT
\end{gathered}$$

This relational schema can be analyzed by considering Ag_j's point of view. It can compare what both Ag_i and Ag_j itself believe of the dependency relationship ($ObjDep_{i,j,\tau}$). The link with what really corresponds to the possible dependence of the two beliefs (of Ag_i and Ag_j on $ObjDep_{i,j,\tau}$) allows us to highlight many interesting specific cases.

We will see later how the use of the various relationships in the dependency network produces accumulations of "dependency capital" (truthful and/or false) and the phenomena that can result from them.

Finally, we consider the comparison between the subjective dependence of Ag_j from Ag_i ($Bel_{Ag_j}(ObjDep_{j,i,\tau'})$) and what Ag_i

believes of this subjective belief of Ag_j (Bel_{Ag_i} (Bel_{Ag_j} ($ObjDep_{j,i,\tau'}$))) also in relation to what Ag_i directly believes of this dependence ($Bel_{Ag_i}(ObjDep_{j,i,\tau'})$): Ag_i may believe that its belief on $ObjDep_{j,i,\tau'}$ coincides, or not, with Ag_j's belief on the same dependence, namely: $Bel_{Ag_i}(ObjDep_{j,i,\tau'}\ ) = Bel_{Ag_i}$ ($Bel_{Ag_j}(ObjDep_{j,i,\tau'})$) or not $Bel_{Ag_i}(ObjDep_{j,i,\tau'}) \neq Bel_{Ag_i}$ (Bel_{Ag_j} ($ObjDep_{j,i,\tau'}$)).

In both cases, the comparisons with the reality are also of interest (see **Figure 8**):

$$\left(\boldsymbol{Bel}_{\boldsymbol{Ag}_i}\left(\boldsymbol{Bel}_{\boldsymbol{Ag}_j}\left(\boldsymbol{ObjDep}\left(\boldsymbol{Ag}_j,\boldsymbol{Ag}_i,\boldsymbol{\tau}\right)\right) \leftrightarrow \boldsymbol{Bel}_{\boldsymbol{Ag}_j}\left(\boldsymbol{ObjDep}\left(\boldsymbol{Ag}_j,\boldsymbol{Ag}_i,\boldsymbol{\tau}\right)\right)\right)\right) \wedge$$
$$\left(\boldsymbol{Bel}_{\boldsymbol{Ag}_i}\left(\boldsymbol{Bel}_{\boldsymbol{Ag}_j}\left(\boldsymbol{ObjDep}\left(\boldsymbol{Ag}_j,\boldsymbol{Ag}_i,\boldsymbol{\tau}\right)\right) \leftrightarrow \boldsymbol{Bel}_{\boldsymbol{Ag}_i}\left(\boldsymbol{ObjDep}\left(\boldsymbol{Ag}_j,\boldsymbol{Ag}_i,\boldsymbol{\tau}\right)\right)\right)\right) \wedge$$
$$\left(\boldsymbol{Bel}_{\boldsymbol{Ag}_i}\left(\boldsymbol{Bel}_{\boldsymbol{Ag}_j}\left(\boldsymbol{ObjDep}\left(\boldsymbol{Ag}_j,\boldsymbol{Ag}_i,\boldsymbol{\tau}\right)\right)\right) \leftrightarrow \boldsymbol{ObjDep}\left(\boldsymbol{Ag}_j,\boldsymbol{Ag}_i,\boldsymbol{\tau}\right)\right) \wedge$$
$$\left(\boldsymbol{Bel}_{\boldsymbol{Ag}_i}\left(\boldsymbol{ObjDep}\left(\boldsymbol{Ag}_j,\boldsymbol{Ag}_i,\boldsymbol{\tau}\right)\right) \leftrightarrow \boldsymbol{ObjDep}\left(\boldsymbol{Ag}_j,\boldsymbol{Ag}_i,\boldsymbol{\tau}\right)\right) \tag{29}$$
$$\left(Ag_i, Ag_j\right) \in AGT$$

3.6.5 Reasoning on the Dependence Network

As can be understood from the very general analyses, just shown the cross-dependence relationships between them can determine different ratios, degrees and dimensions. In this sense we must consider that what we have defined as the "power to accomplish a certain task" can refer to different actions (AZ), resources (R) and contexts (Γ), producing complex and interesting situations.

Not only that, but we also associate the "power of" ($Pow(Ag_x, \tau)$) with a degree of ability ($DoA(Ag_x, \tau)$) above a certain threshold (σ). But precisely for this reason it is possible to believe that there are different degrees of skill of the interlocutor when it is considered to have the "power of". Let's see the cases of greatest interest.

Agents may have beliefs about their dependence on other agents in the network, whether or not they match objective reality. This can happen in two main ways:

- In the first, looking at (**formula 24**) we can say that there is some task τ for which Ag_i does not believe it is dependent on some Ag_j agent and at the same time there is instead (precisely for that task from that agent) an objective dependency relationship. In formulas:

$$\left(\boldsymbol{Bel}_{\boldsymbol{Ag}_i}\left(\boldsymbol{ObjDep}\left(\boldsymbol{Ag}_i,\boldsymbol{Ag}_j,\boldsymbol{\tau}\right)\right) = \boldsymbol{false}\right) \wedge \left(\boldsymbol{ObjDep}\left(\boldsymbol{Ag}_i,\boldsymbol{Ag}_j,\boldsymbol{\tau}\right) = \boldsymbol{true}\right) \tag{30}$$

Evaluating how that belief can be denied, given that $ObjDep(Ag_i, Ag_j, \tau) = LoPow(Ag_i, \tau) \wedge Pow(Ag_j, \tau)$ not believing that dependence can mean denying one or both of the functions that define it, namely:

i) Thinking of having a power that it does not have ($Bel_{Ag_i}(Pow(Ag_i, \tau))$) while objectively it is $LoPow(Ag_i, \tau)$;
ii) Thinking that Ag_j does not have that required power ($Bel_{Ag_i}(LoPow(Ag_j, \tau))$) while objectively ($Pow(Ag_j, \tau)$);
iii) Believing both above as opposed to objective reality.

- In the second case, we can say that there is some task τ for which Ag_i believes it is dependent on some Ag_j agent and at the same time there is no objective dependency relationship (precisely for that task from that agent). In formulas:

$$\left(\boldsymbol{Bel}_{\boldsymbol{Ag}_i}\left(\boldsymbol{ObjDep}\left(\boldsymbol{Ag}_i,\boldsymbol{Ag}_j,\boldsymbol{\tau}\right)\right) = \boldsymbol{true}\right) \wedge \left(\boldsymbol{ObjDep}\left(\boldsymbol{Ag}_i,\boldsymbol{Ag}_j,\boldsymbol{\tau}\right) = \boldsymbol{false}\right) \tag{31}$$

believing this dependence may mean confirming one or both hypotheses that are denied in reality, namely:

i) Thinking (on the part of Ag_i) that it does not have a power ($Bel_{Ag_i}(LoPow(Ag_i, \tau))$) while objectively and potentially it is ($Pow(Ag_i, \tau)$), that is, it has that power[8];
ii) Thinking (on Ag_i's part) that Ag_j has that required power ($Bel_{Ag_i}(Pow(Ag_j, \tau))$ while objectively it is ($LoPow(Ag_j, \tau)$);
iii) Believing both above as opposed to objective reality.

Going deeper, we can say that the meaning concerning the belief of having or not having the "power" to carry out a certain task, τ must be carefully analyzed. With $\tau = (\alpha, g)$. In fact, given the definition of τ, we can say that the Ag_i agent has the power to realize τ if:

$$-\boldsymbol{Bel}_{\boldsymbol{Ag}_i}\left(\boldsymbol{\tau} = \left(\boldsymbol{\alpha}, \boldsymbol{g}\right)\right) \tag{32}$$

that is, Ag_i believes that the application of the action α (and the possession of the resources for its execution) produces the state of the world g (with a high probability of success, let's say above a rather high threshold).

$$-\boldsymbol{Bel}_{\boldsymbol{Ag}_i}\left(\boldsymbol{\alpha} \in \boldsymbol{AZ}_{\boldsymbol{Ag}_i}\right) \tag{33}$$

that is, Ag_i believes it has the action α in its repertoire. And:

$$-\boldsymbol{Bel}_{\boldsymbol{Ag}_i}\left(\boldsymbol{g} \in \boldsymbol{GOAL}_{\boldsymbol{Ag}_i}\right) \tag{34}$$

that is, in addition to having the power to obtain the task τ, the Ag_i agent should also have the state of the world g among the *active goals* it wants to achieve (we said previously that having the power implies the presence of the goal in potential form). We established (for simplicity) that an agent knows the goals/needs/duties that it possesses, while it may not know the goals of the other agents.

Given the conditions indicated above, there are cases of ignorance with respect to actually existing dependencies or of evaluations of false dependencies. As we have seen above, the beliefs of the agent Ag_i must also be compared with those of the agent with whom the interaction is being analyzed (Ag_j). So back to the belief:

$$\left(\boldsymbol{Bel}_{\boldsymbol{Ag}_i}\left(\boldsymbol{ObjDep}\left(\boldsymbol{Ag}_i,\boldsymbol{Ag}_j,\boldsymbol{\tau}\right)\right) = \boldsymbol{true}\right) \vee \left(\boldsymbol{Bel}_{\boldsymbol{Ag}_i}\left(\boldsymbol{ObjDep}\left(\boldsymbol{Ag}_i,\boldsymbol{Ag}_j,\boldsymbol{\tau}\right)\right) = \boldsymbol{false}\right) \tag{35}$$

[8] The comparison operator (↔) allows to relate the two compared expressions (A and B in this case) to check whether they are equal or not and, in the second case, what are the possible factors that determine the difference.

putting it from the point of view of Ag_j we analogously have:

$$\left(\boldsymbol{Bel}_{Ag_j}\left(\boldsymbol{ObjDep}\left(\boldsymbol{Ag}_i, \boldsymbol{Ag}_j, \boldsymbol{\tau}\right)\right) = \boldsymbol{true}\right) \vee \left(\boldsymbol{Bel}_{Ag_j}\left(\boldsymbol{ObjDep}\left(\boldsymbol{Ag}_i, \boldsymbol{Ag}_j, \boldsymbol{\tau}\right)\right) = \boldsymbol{false}\right) \tag{36}$$

The divergence or convergence of the beliefs of the two agents (Ag_i, Ag_j) on the dependence of Ag_i with respect to Ag_j can be completely insignificant. What matters for the pursuit of the task and for its eventual success is what Ag_i believes and whether what it believes is also true in reality $ObjDep(Ag_i, Ag_j, \tau)$.

Another interesting analysis concerns the inconsistent fallacious beliefs of agents on dependence on them, of other agents in the network, with respect to objective reality.

That is, Ag_i may believe that Ag_j is dependent from it or not. And this may or may not coincide with reality. There are four possible combinations:

$$\left(\boldsymbol{Bel}_{Ag_i}\left(\boldsymbol{ObjDep}\left(\boldsymbol{Ag}_j, \boldsymbol{Ag}_i, \boldsymbol{\tau}\right)\right) = \boldsymbol{true}\right) \wedge \left(\boldsymbol{ObjDep}\left(\boldsymbol{Ag}_j, \boldsymbol{Ag}_i, \boldsymbol{\tau}\right) = \boldsymbol{true}\right) \tag{37}$$

$$\left(\boldsymbol{Bel}_{Ag_i}\left(\boldsymbol{ObjDep}\left(\boldsymbol{Ag}_j, \boldsymbol{Ag}_i, \boldsymbol{\tau}\right)\right) = \boldsymbol{true}\right) \wedge \left(\boldsymbol{ObjDep}\left(\boldsymbol{Ag}_j, \boldsymbol{Ag}_i, \boldsymbol{\tau}\right) = \boldsymbol{false}\right) \tag{38}$$

$$\left(\boldsymbol{Bel}_{Ag_i}\left(\boldsymbol{ObjDep}\left(\boldsymbol{Ag}_j, \boldsymbol{Ag}_i, \boldsymbol{\tau}\right)\right) = \boldsymbol{false}\right) \wedge \left(\boldsymbol{ObjDep}\left(\boldsymbol{Ag}_j, \boldsymbol{Ag}_i, \boldsymbol{\tau}\right) = \boldsymbol{true}\right) \tag{39}$$

$$\left(\boldsymbol{Bel}_{Ag_i}\left(\boldsymbol{ObjDep}\left(\boldsymbol{Ag}_j, \boldsymbol{Ag}_i, \boldsymbol{\tau}\right)\right) = \boldsymbol{false}\right) \wedge \left(\boldsymbol{ObjDep}\left(\boldsymbol{Ag}_j, \boldsymbol{Ag}_i, \boldsymbol{\tau}\right) = \boldsymbol{false}\right) \tag{40}$$

As we have seen the belief of dependence implies attribution of powers and lack of powers (and the denial of dependence belief in turn determines similar and inverted attributions). Compared to the previous case, in this case being possible not to necessarily know about the goals of the interlocutor, it is also possible to misunderstand on these goals: for example, considering that $g \in GOAL_{Ag_j}$(or $g \notin GOAL_{Ag_i}$) while instead it is the opposite. In this way, introducing an attribution error.

An interesting thing is that there are cases where one can believe that another agent has no power to achieve a task due not to its inability to perform an action (or lack of resources for that execution) but from the fact that the task's goal is not included among its goals.

4 DEPENDENCE AND NEGOTIATION POWER

Given a *Dependence Network* (*DN*, see ***formula 20***) and an agent in this Network ($Ag_i \in AGT$), if the Ag_i has to achieve the task $\tau_s^{Ag_i}$, from here on τ_s, we can consider as its interlocutors the *m* agents included in the set *Potential Solvers (PS),* in practice the ones that have the power for achieving τ_s:

$$\boldsymbol{PS}\left(\boldsymbol{Ag}_i, \boldsymbol{\tau}_s\right) =_{def} \bigcup_{v=1}^{m} \boldsymbol{Ag}_v \in \boldsymbol{AGT} | \left(\boldsymbol{Pow}\left(\boldsymbol{Ag}_v, \boldsymbol{\tau}_s\right) = \boldsymbol{true}\right) \tag{41}$$

The same Ag_i (if it has the appropriate skills) could be included among these agents.

We define *Objective Potential for Negotiation* of $Ag_i \in AGT$ about an its own task τ_s - and call it $OPN\,(Ag_i, \tau_s)$- the following function:

$$\boldsymbol{OPN}\left(\boldsymbol{Ag}_i, \boldsymbol{\tau}_s\right) =_{def} \sum\nolimits_{Ag_l \in PS\left(Ag_i, \tau_s\right)} \frac{\boldsymbol{ObjDep}\left(\boldsymbol{Ag}_l, \boldsymbol{Ag}_i, \boldsymbol{\tau}_k\right) * \boldsymbol{DoA}\left(\boldsymbol{Ag}_l, \boldsymbol{\tau}_s\right) * \boldsymbol{DoA}\left(\boldsymbol{Ag}_i, \boldsymbol{\tau}_k\right)}{\boldsymbol{1} + \boldsymbol{p}_{sl}} \tag{42}$$

So, the agents Ag_l are all included in $PS(Ag_i, \tau_s)$ and they are dependent by Ag_i about one of their own task ($\tau_k^{Ag_l}$, from here on τ_k). Remind that if $ObjDep\,(Ag_l, Ag_i, \tau_k)$ is true, it is also true $Pow(Ag_i, \tau_k)$. So Ag_i and Ag_l can balance the negotiating potential. We establish by convention that $ObjDep$ (Ag_l, Ag_i, τ_k) is equal one if it is true and 0 if it is false. In addition, the negotiation potential OPN is measured on the respective abilities of Ag_i and Ag_l to realize their respective tasks: $DoA\,(Ag_l, \tau_s)$ and $DoA\,(Ag_i, \tau_k)$.

In words, m represents the number of agents (Ag_l) who can carry out the task τ_s and at the same time have tasks to perform that are potentially achievable by the agent Ag_i. This dependence relation should be either *reciprocal* (the tasks under negotiation are $\tau_k^{Ag_l}$ and $\tau_s^{Ag_i}$) or *mutual* (the tasks under negotiation are $\tau_s^{Ag_l}$ and $\tau_s^{Ag_i}$): more specifically, there should be an action, plan, or resource owned by Ag_i that is necessary for Ag_l to obtain $\tau_k^{Ag_l}$ (possibly coincident with $\tau_s^{Ag_l}$) and at the same time there should be an action, plan, or resource owned by Ag_l that is necessary for Ag_i to obtain $\tau_s^{Ag_i}$ (possibly coincident with $\tau_s^{Ag_l}$).

p_{sl} is the number of agents in AGT who need from Ag_l of a different task (τ_q) in competition with the request by Ag_i (in the same context and at the same time, and being able to offer it help on an Ag_l's task in return). We are considering that these parallel requests cause a reduction in availability, as our agent Ag_l has to contribute to multiple requests (p_{sl} + 1) at the same time.

We can therefore say that every other agent in Ag_i's network of dependence (either reciprocal or mutual) contributes to $OPN\,(Ag_i, \tau_s)$ with a value between $(DoA\,(Ag_l, \tau_s)^* DoA\,(Ag_i, \tau_k))$ and $(DoA\,(Ag_l, \tau_s)^* DoA\,(Ag_i, \tau_k))/(1 + p_{sl})$. We have therefore, to simplify, considered that the contribution to the negotiation potential is the same for each agent in reciprocal or mutual dependence with our agent Ag_i (with the same number of other p_{sl} contenders).

If we indicate with *PSD* all the agents included in *PS* with objective dependence equal to 1, so:

$$\boldsymbol{PSD}\left(\boldsymbol{Ag}_i, \boldsymbol{\tau}_s\right) =_{def} \bigcup_{v=1}^{m} \boldsymbol{Ag}_v \in \boldsymbol{AGT} | \left(\boldsymbol{Pow}\left(\boldsymbol{Ag}_v, \boldsymbol{\tau}_s\right) = \boldsymbol{true}\right) \wedge \boldsymbol{ObjDep}\left(\boldsymbol{Ag}_v, \boldsymbol{Ag}_i, \boldsymbol{\tau}_k\right) = \boldsymbol{1}\Big) \tag{43}$$

we can say that:

$$0 < \boldsymbol{OPN}\left(\boldsymbol{Ag}_i, \boldsymbol{\tau}_s\right) \leq \boldsymbol{Card}\left(\boldsymbol{PSD}\right) \tag{44}$$

In **Figure 9** we represent the objective dependence of Ag_i: considering the areas of spaces A, B and C proportional to the number of agents they represent, we can say that: A represents the set of agents (Ag_v) who depend from Ag_i for some their task $\tau_k^{Ag_v}$, from here on τ_k, B represents the set of agents from which Ag_i

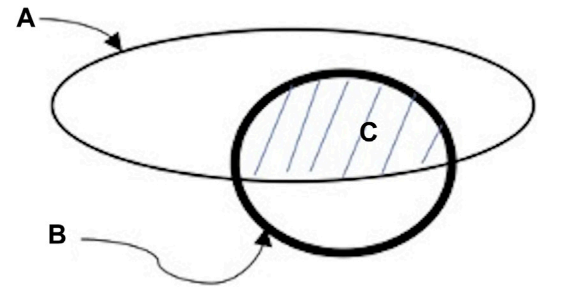


FIGURE 9 | Area A is proportional to the number of agents dependent by Ag_i per τ_k; Area B is proportional to the number of agents on which Ag_i depends per τ_s; Area C is the intersection of **(A,B)**

depends for achieving the task τ_s ($B = PS(Ag_i, \tau_s)$ and at the same time it represents all the Ag_v agents who are able to achieve the goal g_s through some α_s action). The intersection between A and B (dashed part C) is the subset of $PS(Ag_i, \tau_s)$ with whom Ag_i could potentially negotiate for achieving τ_s ($C = PSD(Ag_i, \tau_s)$). The greater the overlap the greater the negotiation power of Ag_i in that context.

However, as we have seen above, the negotiation power of Ag_i also depends on the possible alternatives (p_{sl}) that its potential partners (Ag_v) have: the few alternatives to Ag_i they have, the greater its negotiation power (see below)[9]. Not only that, the power of negotiation should also take into account the abilities of the agents in carrying out their respective tasks ($DoA(Ag_l, \tau_s)^* DoA(Ag_i, \tau_k)$).

The one just described is the *objective potential* for negotiating agents. But, as we have seen in the previous paragraphs, the operational role of dependence is established by being aware of (or at least by believing) such dependence on the part of the agents.

We now want to consider the set of agents with whom Ag_i can negotiate to get its own task (τ_s). This set, called *Real set of Agents for Negotiation* (*RAN*), includes all the agents that believe to be able to achieve that task (τ_s) and at the same time believe to be dependent by Ag_i about one's own task (τ_k). At the same time, Ag_i must also be aware of Ag_v's potential:

$$\begin{aligned}\boldsymbol{RAN(Ag_i, \tau_s)} &\boldsymbol{=_{def} \bigcup_{v=1}^{m} (Ag_v \in AGT) \big| Bel_{Ag_v}(Pow(Ag_v, \tau_s)} \\ &\boldsymbol{= true) \wedge Bel_{Ag_v}(ObjDep(Ag_v, Ag_i, \tau_k)} \\ &\boldsymbol{= 1) \wedge Bel_{Ag_i}(Pow(Ag_v, \tau_s)} \\ &\boldsymbol{= true) \wedge Bel_{Ag_i}(ObjDep(Ag_v, Ag_i, \tau_k) = 1)} \end{aligned} \tag{45}$$

We also define the *Real Objective Potential for Negotiation* (*ROPN*) of $Ag_i \in AGT$ about an its own task τ_s the following function:

$$\boldsymbol{ROPN(Ag_i, \tau_s) =_{def} \sum_{Ag_l \in RAN(Ag_i, \tau_s)} \frac{ObjDep(Ag_l, Ag_i, \tau_k) * DoA(Ag_l, \tau_s)^* DoA(Ag_i, \tau_k)}{1 + p_{sl}}} \tag{46}$$

As can be seen also *ROPN*, like *OPN*, depends on the objective dependence of the selected agents. In this case, however, the selection is based on the beliefs of the two interacting agents. We have:

$$0 < \boldsymbol{ROPN(Ag_i, \tau_s) \leq Card(RAN)} \tag{47}$$

We have made reference above to the believed (by Ag_i and Ag_v) dependence relations (not necessarily true in the world). This is sufficient to define $RAN(Ag_i, \tau_s)$ and, therefore, $ROPN(Ag_i, \tau_s)$ which determine the actions of Ag_i and Ag_v in the negotiation[10].

Analogously, we can interpret **Figure 9** as the set of believed relationships by the agents.

In case Ag_i has to carry out the task τ_s, and does not have the power to do it by itself, it can be useful to evaluate the *list of agents* given by the set $RAN(Ag_i, \tau_s)$[11] and who have negotiating power with Ag_i, ordered by quantity of available commitment: that is, Ag_i, on the basis of its beliefs will be able to order the potential interlocutors of the negotiation in direct order with respect to the ability values attributed to Ag_l (by Ag_i) for the accomplishment of the task ($DoA(Ag_l, \tau_s)$), and in reverse order to the number of parallel competitors, see $ROPN(Ag_i, \tau_s)$. Obviously, other criteria can be added for selecting the agent to choose. For example:

- based on the reciprocity task to be performed: the most relevant, the most pleasing, the cheapest, the simplest, and so on.
- based on the agent with whom it is preferred to enter into a relationship: usefulness, friendship, etc.
- based on the trustworthiness of the other agent with respect to the task delegated to it.

This last point leads us to the next paragraph.

5 THE TRUST ROLE IN DEPENDENCE NETWORKS

Let us introduce into the dependence network the trust relationships. In fact, although it is important to consider dependence relationship between agents in a society, there will be not exchange in the market if there is not trust to enforce these connections. Considering the analogy with **Figure 9**, now we will have a representation as given in **Figure 10** (where we introduced

[9] if it was aware of it.

[10] Obviously, this is a possible hypothesis, linked to a particular model of agent and of interaction between agents. We could also foresee different agency hypotheses.

[11] Of course, the success or failure of these negotiations will also depend on how true the beliefs of the various agents are.

[12] We assume, for simplicity, that if Ag_i has the beliefs $Bel_{Ag_i}(Pow(Ag_v, \tau_s^{Ag_i}) = true) \cap Bel_{Ag_i}(ObjDep(Ag_v, Ag_i, \tau_k^{Ag_v}) = 1)$ then it believes that those same beliefs are also held by Ag_v.

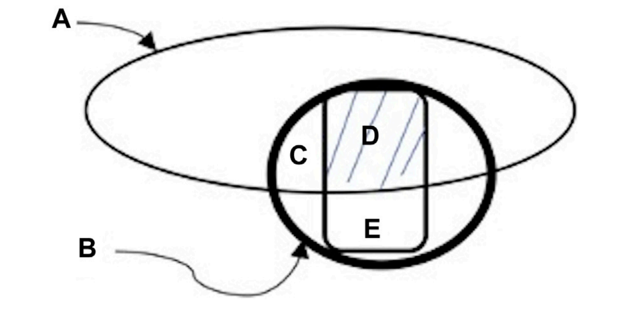


FIGURE 10 | The rectangle introduced with respect to **Figure 9** represents the trustworthy agents with respect to τ_s.

the rectangle that represents the trustworthy agents with respect to the task τ_s).

The potential agents for negotiation are the ones in the dashed *part D*: they are trustworthy on the task τ_s for which Ag$_i$ depends on them, and they are themselves dependent on Ag$_i$ on another their task.

While *part E* includes agents who are trustworthy by Ag_i on the task τ_s for which Ag_i depends on them but they are not dependent by Ag_i on their own tasks. For *part B* and *C* are true the old definitions in **Figure 9**.

Therefore, not only the decision to trust presupposes a belief of being dependent but notice that a dependence belief implies on the other side a piece of trust. In fact, to believe to be dependent means: $Bel_{Ag_i}(LoPow(Ag_i, \tau_s) = true)$ *and* $Bel_{Ag_i}(Pow(Ag_v, \tau_s) = true)$. With $\tau_s = (\alpha_s, g_s)$. In basic beliefs:

- ($B_1^{Ag_i}$) to believe (by Ag_i) not to be able to perform action α_s and, therefore, not to be able to achieve goal g_s; and
- ($B_2^{Ag_i}$) to believe (by Ag_i) that Ag_v is able and in condition to achieve g_s, through the performance of the α_s action.

Notice that $B_2^{Ag_i}$ is precisely one component of trust concept in our analysis [12, 13]: the positive evaluation of Ag_v as competent, able, skilled, and so on. However, the other fundamental component of trust as evaluation is lacking, its reliability/trustworthiness: Ag_v really intends to do, is persistent, is loyal, is benevolent, etc. Thus, Ag_v will really do what Ag_i needs.

So, starting from the objective dependence of the agents, we must include the motivational aspects. In particular, we have a new set of interesting agents, called *Potential Trustworthy Solvers* (*PTS*):

$$\boldsymbol{PTS(Ag_i, \tau_s) =_{def} \bigcup_{v=1}^{m} Ag_v \in AGT \big| (Pow(Ag_v, \tau_s) = true) \wedge (Mot(Ag_v, \tau_s) = true)} \qquad (48)$$

Where $Mot(Ag_v, \tau_s^{Ag_i})$ means that the Ag_v agent is motivated to carry out the $\tau_s^{Ag_i}$ task. Recall that in the case of skills (evaluated through the *Pow* function) reference was made to the degree of ability (*DoA*). Also, in the case of motivations (*Mot*) we must consider that an agent can be considered to have successful motivations if its degree of motivation/willingness (*DoW*) is above a given threshold (ξ).

$$\boldsymbol{(Mot(Ag_v, \tau_s) = true) \rightarrow DoW(Ag_v, \tau_s) > \eta} \qquad (49)$$

where η has a high value in the range (0,1).

For Ag_v to be successful in the $\tau_s^{Ag_i}$ task, it is therefore necessary that both conditions are met:

$$\boldsymbol{(DoA(Ag_v, \tau_s)) > \sigma) \wedge)DoW(Ag_v, \tau_s) > \eta} \qquad (50)$$

We must now move from the objective value of PTS to what Ag_i believes about it (*Potential Trustworthy Solvers* (*PTS*) believed by Ag_i):

$$\boldsymbol{Bel_{Ag_i}(PTS(Ag_i, \tau_s)) =_{def} \bigcup_{v=1}^{m} Ag_v \in AGT \big| Bel_{Ag_i}\big(Pow(Ag_v, \tau_s) = true) \wedge Bel_{Ag_i})Mot(Ag_v, \tau_s) = true\big)} \qquad (51)$$

In fact, $Bel_{Ag_i}(PTS(Ag_i, \tau_s))$ returns the list of agents who are believed by Ag_i to be trustworthy for the specified task (i.e. as capable as they are willing).

One of the main reasons why Ag_v is motivated (i.e., $DoW(Ag_v, \tau_s) > \eta$) is given by its dependence on Ag_i with respect to a task of the Ag_v itself ($\tau_k^{Ag_v}$) and thus the possibility of successful negotiation between agents.

So, an interesting case is when:

$$\boldsymbol{Mot(Ag_v, \tau_s) =_{def} Bel_{Ag_v}(ObjDep(Ag_v, Ag_i, \tau_k) = true) \wedge Bel_{Ag_v}(Mot(Ag_i, \tau_k) = true)} \qquad (52)$$

That is, Ag_v's motivation to carry out the task τ_s for the Ag_i ($DoW(Ag_v, \tau_s) > \ \eta$) is linked to the fact that Ag_v believes it depends on Ag_i with respect to the task τ_k and similarly believes that Ag_i is capable and motivated to accomplish that task.

We have therefore defined the belief conditions of the two agents (Ag_i, Ag_v) in interaction so that they can negotiate and start a collaboration in which each one can achieve its own goal. These conditions show the need to be in the presence not only of bilateral dependence of Ag_i and Ag_v but also of their bilateral trust.

5.1 The Point of View of the Trustee: Towards Trust Capital

Let us, now, explicitly recall what are the cognitive ingredients of trust and reformulate them from the point of view of the trusted agent [23]. In order to do this, it is necessary to limit the set of trusted entities. It has in fact been argued that trust is a mental attitude, a decision and a behavior that only a cognitive agent endowed with both goals and beliefs can have, make and perform. But it has been underlined, also, that the entities that is trusted is not necessarily a cognitive agent. When a cognitive agent trusts another cognitive agent, we talk about *social trust*. As we have seen, the set of actions, plans and resources owned/available by an agent can be useful for achieving a set of tasks ($\tau_1, \ldots, \tau_r$).

We take now the point of view of the trustee agent in the dependence network: so, we present a *cognitive theory of trust as a capital*, which is, in our view, a good starting point to include this concept in the issue of negotiation power. That is to say what really matters are not the skills and intentions declared by the owner, but those actually believed by the other agents. In other words, it is on the *trustworthiness perceived* by other agents that our agent's real negotiating power is based.

We call *Objective Trust Capital* (*OTC*) of $Ag_i \in AGT$ about a generic task τ_s the function:

$$\boldsymbol{OTC}(\boldsymbol{Ag_i}, \boldsymbol{\tau_s}) =_{def} \sum\nolimits_{Ag_v \in AGT} \boldsymbol{Bel}_{Ag_v}(\boldsymbol{DoA}(\boldsymbol{Ag_i}, \boldsymbol{\tau_s})^{*}\boldsymbol{DoW}(\boldsymbol{Ag_i}, \boldsymbol{\tau_s})) \tag{53}$$

With

$$0 \le \boldsymbol{OTC}(\boldsymbol{Ag_i}, \boldsymbol{\tau_s}) \le \boldsymbol{Card}(\boldsymbol{AGT})^{13} \tag{54}$$

We can therefore determine on the basis of (*OTC*) the set of agents in the Ag_i's DN that potentially consider the Ag_i reliable for the task τ_s. If we call *Potential Objective Trustors* (*POT*) this set we can write:

$$\boldsymbol{POT}(\boldsymbol{Ag_i}, \boldsymbol{\tau_s}) =_{def} \bigcup_{v=1}^{m} \boldsymbol{Ag_v} \in \boldsymbol{AGT} \big| \boldsymbol{Bel}_{Ag_v}(\boldsymbol{DoA}(\boldsymbol{Ag_i}, \boldsymbol{\tau_s}) > \boldsymbol{\sigma}) \wedge \boldsymbol{Bel}_{Ag_v}(\boldsymbol{DoW}(\boldsymbol{Ag_i}, \boldsymbol{\tau_s}) > \boldsymbol{\eta}) \tag{55}$$

We are talking about "generic task" as the g_S goal is not necessarily included in $GOAL_{Ag_i}$ but indicates a task for which Ag_i could be considered trustworthy in its implementation. In other words, Ag_i would be able to carry out that task by having the possibility of mobilizing (i.e. possessing) its skills, competences and intentionality suitable for the task itself.

As showed in [13] we call Degree of Trust of the Agent Ag_v on the agent Ag_i about the task τ_s:

$$\boldsymbol{DoT}(\boldsymbol{Ag_v}, \boldsymbol{Ag_i}, \boldsymbol{\tau_s}) =_{def} \boldsymbol{Bel}_{Ag_v}(\boldsymbol{DoA}(\boldsymbol{Ag_i}, \boldsymbol{\tau_s}) * \boldsymbol{DoW}(\boldsymbol{Ag_i}, \boldsymbol{\tau_s})) \tag{56}$$

We call the *Subjective Trust Capital* (*STC*) of $Ag_i \in AGT$ about a generic task τ_s the function:

$$\boldsymbol{STC}(\boldsymbol{Ag_i}, \boldsymbol{\tau_s}) =_{def} \sum\nolimits_{Ag_v \in AGT} \boldsymbol{Bel}_{Ag_i}(\boldsymbol{Bel}_{Ag_v}(\boldsymbol{DoA}(\boldsymbol{Ag_i}, \boldsymbol{\tau_s})^{*}\boldsymbol{DoW}(\boldsymbol{Ag_i}, \boldsymbol{\tau_s}))) \tag{57}$$

In words, the cumulated trust capital of an agent Ag_i with respect a task τ_s, is the sum (on all the agents in the Ag_i's network dependence) of the corresponding potential abilities and willingness believed about Ag_i on the task τ_s, by each dependent agent. The subjectivity consists in the fact that both the network dependence and the believed potential abilities and willingness are believed by (the point of view of) the agent Ag_i.

We can therefore determine on the basis of (*STC*) the set of agents in the Ag_i's DN which Ag_i believes may be potential trustors of Ag_i itself for the task τ_s. If we call *Potential Believed Trustors* (*PBT*) this set we can write:

$$\begin{aligned} \boldsymbol{PBT}(\boldsymbol{Ag_i}, \boldsymbol{\tau_s}) =_{def} & \bigcup_{v=1}^{m} \boldsymbol{Ag_v} \in \boldsymbol{AGT} \,| \\ & |\boldsymbol{Bel}_{Ag_i}(\boldsymbol{Bel}_{Ag_v}(\boldsymbol{DoA}(\boldsymbol{Ag_i}, \boldsymbol{\tau_s}) > \boldsymbol{\sigma})) \\ & \wedge \boldsymbol{Bel}_{Ag_i}(\boldsymbol{Bel}_{Ag_v}(\boldsymbol{DoW}(\boldsymbol{Ag_i}, \boldsymbol{\tau_s}) > \boldsymbol{\eta}\big) \end{aligned} \tag{58}$$

We can call *Believed Degree of Trust* (*BDoT*)of the Agent Ag_v on the agent Ag_i as believed by the agent Ag_i, about the task τ_s:

$$\boldsymbol{BDoT}(\boldsymbol{Ag_v}, \boldsymbol{Ag_i}, \boldsymbol{\tau_s}) =_{def} \boldsymbol{Bel}_{Ag_i}(\boldsymbol{Bel}_{Ag_v}(\boldsymbol{DoA}(\boldsymbol{Ag_i}, \boldsymbol{\tau_s})^{*}\boldsymbol{DoW}(\boldsymbol{Ag_i}, \boldsymbol{\tau_s}))) \tag{59}$$

At the same way we can also call the *Self-Trust* (*ST*) of the agent Ag_i about the task τ_s. We can write:

$$\boldsymbol{ST}(\boldsymbol{Ag_i}, \boldsymbol{\tau_s}) =_{def} \boldsymbol{Bel}_{Ag_i}(\boldsymbol{DoA}(\boldsymbol{Ag_i}, \boldsymbol{\tau_s}) * \boldsymbol{DoW}(\boldsymbol{Ag_i}, \boldsymbol{\tau_s})) \tag{60}$$

From the comparison between $OTC(Ag_i, \tau_s)$, $STC(Ag_i, \tau_s)$, $DoT(Ag_v, Ag_i, \tau_s)$ and $ST(Ag_i, \tau_s)$ a set of interesting actions and decision could be taken from the agents (we will see in the next paragraphs).

6 DYNAMICS OF RELATIONAL CAPITAL

An important consideration we have to do is that a dependence network is mainly based on the set of actions, plans and resources owned by the agents and necessary for achieving the agents' goals (we considered a set of tasks each agent is able to achieve). The dependence network is then closely related to the dynamics of these sets (actions, plans, resources, goals), from their modification over time. In particular, the dynamics of the agents' goals, from their variations (from the emergency of new ones, from the disappearance of old ones, from the increasing request of a subset of them, and so on). On this basis changes the role and relevance of each agent in the dependence network, changes in fact the trust capital of the agents.

For what concerns the dynamical aspects of this kind of capital, it is possible to make hypotheses on how it can increase or how it can be wasted, depending on how each of basic beliefs involved in trust are manipulated. In the following, let us consider what kind of strategies can be performed by Ag_i to enforce the other agents' dependence beliefs and their beliefs about Ag_i's competence/motivation.

6.1 Reducing Ag_l's Power

Ag_i can make the other agent (Ag_l) dependent on it by making the other lacking some resource or skill (or at least inducing the other to believe so).

We can say that there is at least one action (α^{Ag_i}) in Ag_i's action library which, if carried out by Ag_i, allows Ag_l to believe that it is no longer able to obtain τ_s on its own (whether the belief is true or false is not important). In practice:

$$\boldsymbol{Do}(\boldsymbol{Ag_i}, \boldsymbol{\alpha}^{Ag_i}) \rightarrow \big(\boldsymbol{Bel}_{Ag_l}(\boldsymbol{LoPow}(\boldsymbol{Ag_l}, \boldsymbol{\tau_s}) = \boldsymbol{true})\big) \tag{61}$$

Where $A \rightarrow B$ means that A implies B. And at the same time:

$$\boldsymbol{Bel}_{Ag_l}\big(\boldsymbol{Pow}(\boldsymbol{Ag_i}, \boldsymbol{\tau_s}) = \boldsymbol{true}) \wedge \boldsymbol{Bel}_{Ag_l}(\boldsymbol{Mot}(\boldsymbol{Ag_i}, \boldsymbol{\tau_s}) = \boldsymbol{true}\big) \tag{62}$$

So:

$$\begin{aligned} & \boldsymbol{Do}(\boldsymbol{Ag_i}, \boldsymbol{\alpha}^{Ag_i}) \wedge \boldsymbol{Bel}_{Ag_l}\big(\boldsymbol{Pow}(\boldsymbol{Ag_i}, \boldsymbol{\tau_s}) \\ & \quad = \boldsymbol{true}) \wedge \boldsymbol{Bel}_{Ag_l}(\boldsymbol{Mot}(\boldsymbol{Ag_i}, \boldsymbol{\tau_s}) \\ & \quad = \boldsymbol{true}\big) \rightarrow \boldsymbol{Bel}_{Ag_l}(\boldsymbol{ObjDep}(\boldsymbol{Ag_l}, \boldsymbol{Ag_i}, \boldsymbol{\tau_s}) = \boldsymbol{true}) \end{aligned} \tag{63}$$

[13] Being both $DoA(Ag_i, \tau_s)$ and $DoW(Ag_i, \tau_s)$ included in the interval (0,1).

6.2 Inducing Goals in Ag_l

Ag_i can make Ag_l dependent on it by activating or inducing in Ag_l a new goal (need, desire) on which Ag_l is not autonomous (or believes so): effectively introducing a new bond of dependence.

We can say that there is at least one action ($\boldsymbol{\alpha^{Ag_i}}$) in Ag_i's action library which, if carried out by Ag_i, generates (directly or indirectly) a goal (g_k, up to that moment not present) of Ag_l for which Ag_l itself believes to be dependent on Ag_i (whether the belief is true or false is not important). In practice:

$$\boldsymbol{Do\left(Ag_i, \alpha^{Ag_i}\right) \rightarrow \left(g_k \in Goal_{Ag_l}\right)} \tag{64}$$

And at the same time is true:

$$\boldsymbol{Bel_{Ag_l}\left(ObjDep\left(Ag_l, Ag_i, \tau_k\right) = true\right)} \tag{65}$$

6.3 Reducing Other Agents' Competition

Ag_i could work for reducing the believed (by Ag_l) value of ability/motivation of each of the possible competitors of Ag_i (in number of p_{kl}) on that specific task τ_k.

We can say that there are actions (α^{Ag_i}) of Ag_i that make Ag_l believe to be less dependent on other Ag_i's competitors (on the task τ_s) as they (Ag_z) are less capable or motivated:

$$\boldsymbol{Do\left(Ag_i, \alpha^{Ag_i}\right) \rightarrow Bel_{Ag_l}\left(LoPow\left(Ag_z, \tau_s\right)\right.} \\ \boldsymbol{\left.= true\right) \vee Bel_{Ag_l}\left(Mot\left(Ag_z, \tau_s\right) = false\right)} \tag{66}$$

In practice, the application of the action α^{Ag_i} allows to reduce the number of agents potentially able to negotiate with Ag_l (*RAN*, **formula 45**) *and therefore its ROPN(Ag_l, τ_k) value* (**formula 46**). *Similarly, by influencing* the motivations of other agents (Ag_z) the action α^{Ag_i} can affect the number of trustees with whom Ag_l negotiates (*PTS*(Ag_l, τ_k)) (**formula 48**) and therefore *PBT*(Ag_l, τ_k) (**formula 58**).

In the two cases just indicated (§6.1 and §6.2) the effects on the beliefs of Ag_l could derive not from the action of Ag_i but from other causes produced in the world (by third-party agents, by Ag_l or by environmental changes).

6.4 Increasing its Own Features

Competition with other agents can also be reduced by inducing Ag_l to believe that Ag_i is more capable and motivated. We can say that there are actions (α^{Ag_i}) of Ag_i that make Ag_l believe that Ag_i's degree of ability and of motivation have increased.

$$\boldsymbol{Do\left(Ag_i, \alpha^{Ag_i}\right) \Rightarrow DoT\left(Ag_l, Ag_i, \tau_s, t_1\right) > DoT\left(Ag_l, Ag_i, \tau_s, t_0\right)} \tag{67}$$

where t_1 is the time interval in which the action was carried out while t_0 is the interval time prior to its realization. Remembering that

$$\boldsymbol{DoT\left(Ag_l, Ag_i, \tau_s,\ t\right) =_{def} Bel_{Ag_l}\left(DoA\left(Ag_i, \tau_s, t\right) * DoW\left(Ag_i, \tau_s, t\right)\right)} \tag{68}$$

6.5 Signaling its Own Presence and Qualities

Since dependence beliefs is strictly related with the possibility of the others to see the agent in the network and to know its ability in performing useful tasks, the goal of the agent who wants to improve its own relational capital will be to signaling its presence, its skills, and its trustworthiness on those tasks [24–26]. While to show its presence it might have to shift its position (either physically or figuratively like, for instance, changing its field), to communicate its skills and its trustworthiness it might have to hold and show something that can be used as a signal (such as certificate, social status etc.). This implies, in its plan of actions, several and necessary sub-goals to make a signal. These sub-goals are costly to be reached and the cost the agent has to pay to reach them can be taken has the evidence for the signals to be credible (of course without considering cheating in building signals). It is important to underline that using these signals often implies the participation of a third subject in the process of building trust as a capital: a third part which must be trusted. We would say the more the third part is trusted in the society, the more expensive will be for the agent to acquire signals to show, and the more these signals will work in increasing the agent's relational capital.

Obviously also Ag_i's previous performances are 'signals' of trustworthiness. And this information is also provided by the circulating reputation of Ag_i [27].

6.6 Strategic Behavior of the Trustee

As we have seen previously there are different points of view for assessing trustworthiness and trust capital of a specific agent (Ag_i) with respect to a specific task (τ_s). In particular:

- its *Real Trustworthiness (RT)*, that which is actually and objectively assessable regardless of what is believed by the same agent (Ag_i) and by the other agents in its world:

$$\boldsymbol{RT\left(Ag_i, \tau_s\right) =_{def} DoA\left(Ag_i, \tau_s\right) * DoW\left(Ag_i, \tau_s\right)} \tag{69}$$

- its own perceived trustworthiness, that is what we have called the *Self-Trust* (*ST*):

$$\boldsymbol{ST\left(Ag_i, \tau_s\right) =_{def} Bel_{Ag_i}\left(DoA\left(Ag_i, \tau_s\right) * DoW\left(Ag_i, \tau_s\right)\right)} \tag{70}$$

- there is, therefore, the *Objective Trust Capital* (*OTC*) of Ag_i, i.e. the accumulation of trust that Ag_i can boast of what other agents in its world objectively believe:

$$\boldsymbol{OTC\left(Ag_i, \tau_s\right) =_{def} \sum\nolimits_{Ag_v \in AGT} Bel_{Ag_v}\left(DoA\left(Ag_i, \tau_s\right) * DoW\left(Ag_i, \tau_s\right)\right)} \tag{71}$$

to which corresponds the set of agents (POT) who are potential trustors of Ag_i:

$$\boldsymbol{POT\left(Ag_i, \tau_s\right) =_{def} \bigcup_{v=1}^{m} Ag_v \in AGT \middle| Bel_{Ag_v}\left(DoA\left(Ag_i, \tau_s\right) > \sigma\right) \wedge Bel_{Ag_v}\left(DoW\left(Ag_i, \tau_s\right) > \eta\right)} \tag{72}$$

- And finally, there is the *Subjective Trust Capital* (*STC*) of Ag_i, i.e. the accumulation of trust that Ag_i believes it can boast with respect to other agents in its world, that is, based on its own beliefs with respect to how other agents deem it trustworthy:

$$\boldsymbol{STC\left(Ag_i, \tau_s\right) =_{def} \sum\nolimits_{Ag_v \in AGT} Bel_{Ag_i}\left(Bel_{Ag_v}\left(DoA\left(Ag_i, \tau_s\right)^* DoW\left(Ag_i, \tau_s\right)\right)\right)} \tag{73}$$

to which corresponds the set of agents (PBT) who are believed by Ag_i to be potential trustors of Ag_i:

$$\begin{aligned} &\boldsymbol{PBT}(\boldsymbol{Ag}_i, \boldsymbol{\tau}_s) =_{def} \bigcup_{v=1}^{m} \boldsymbol{Ag}_v \in \boldsymbol{AGT} | \\ &|\boldsymbol{Bel}_{Ag_i}(\boldsymbol{Bel}_{Ag_v}(\boldsymbol{DoA}(\boldsymbol{Ag}_i, \boldsymbol{\tau}_s) > \boldsymbol{\sigma})) \\ &\wedge \boldsymbol{Bel}_{Ag_i}(\boldsymbol{Bel}_{Ag_v}(\boldsymbol{DoW}(\boldsymbol{Ag}_i, \boldsymbol{\tau}_s) > \boldsymbol{\eta})) \end{aligned} \tag{74}$$

In fact, there is often a difference between how the others actually trust an agent and what the agent believes about (difference between *OTC/POT* and *STC/PBT*); but also between these and the level of trustworthiness that agent perceives in itself (difference between *OTC/POT* and *ST or* difference between *STC/PBT* and *ST*).

The subjective aspects of trust are fundamental in the process of managing this capital, since it can be possible that the capital is there but the agent does not know to have it (or vice versa).

At the base of the possible discrepancy in subjective valuation of trustworthiness there is the perception of how much an agent feels trustworthy in a given task (*ST*) and the valuation that agent does of how much the others trust it for that task (*STC/PBT*). In addition, *this perception can change and become closer to the objective level while the task is performed* (*ST* relationship with both *RT* and *OTC/POT*): the agent can either find out of being more or less trustworthy than what it believed or realize that the others' perception was wrong (either positively or negatively). All these factors must take into account and studied together with the different component of trust, in order to build hypotheses on strategic actions the agent will perform to cope with its own relational capital. Then, we must consider what can be implied by these discrepancies in terms of strategic actions: how they can be individuated and valued? How will the trusted agent react when aware of them? it can either try to acquire competences to reduce the gap between others' valuation and its own one, or exploiting the existence of this discrepancy, taking advantage economically of the reputation aver its capability and counting on the others' scarce ability of monitoring and testing its real skills and/or motivations. In practice, it is on this basis of comparison between reality and subjective beliefs that the most varied behavioral strategies of agents develop. In the attempt to use the dependence network in which they are immersed at best. Dependence network that represents the most effective way to realize the goals they want to achieve.

7 CONCLUSION

With the expansion of the capabilities of intelligent autonomous systems and their pervasiveness in the real world, there is a growing need to equip these systems with autonomy and collaborative properties of an adequate level for intelligent interaction with humans. In fact, the complexity of the levels of interaction and the risks of inappropriate or even harmful interference are growing. A theoretical approach on the basic primitives of social interaction and the articulated outcomes that can derive from it is therefore fundamental.

This paper tries to define some basic elements of dependence relationships, enriched through attitudes of trust, in a network of cognitive agents (regardless of their human or artificial nature).

We have shown how, on the basis of the powers attributable to the various agents, objective relationships of dependence emerge between them. At the same time, we have seen how what really matters is the dependence believed by social agents, thus highlighting the need to consider *cognition* as a decisive element for highly adaptive systems to social interactions.

The articulation of the possibilities of confrontation within the network of dependence between the different interpretations that can arise from them, in a spirit of collaboration or at least of avoidance of conflicts, highlights the need for a clear ontology of social interaction.

By introducing, in the spirit of emulation of truly operational autonomies [28], also the dimension of intentionality and priority choice on this basis, the attitude of trust is particularly relevant, both from the point of view of those who must to choose a partner to trust with a task, as well as from the point of view of those who offer their availability to solve the task. In this sense we have introduced concepts such as relational capital and trust capital.

The future developments of this work will go on the one hand in the direction of further theoretical investigations: on the basis of the model introduced we will define with precision the various and articulated forms of autonomy that derive from it; we will tackle the problem of the "degree of dependence" that derives from many and varied dimensions such as: the value of the goal to be achieved; the number of available and reliable alternative agents that can be contacted; the degree of ability/reliability required for the task to be delegated; and so on.

In parallel, we will try to develop a simulative computational model for trusted dependency networks that we have introduced, with the ambition of having feedback on the basic conceptual scheme and at the same time trying to verify its operability in a concrete way.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author.

AUTHOR CONTRIBUTIONS

RF and CC have equally contributed to the theoretical model; RF developed most of the formalization.

REFERENCES

1. Wasserman S, Faust K. *Social Network Analysis*. Cambridge): Cambridge Univ.Press (1994).
2. Watts DJ. *Small Worlds*. Princeton, NJ): Princeton Univ. Press (1999).
3. Guare J. *Six Degrees of Separation*. New York: Vintage (1990).
4. Newman MEJ. The Structure of Scientific Collaboration Networks. *Proc Natl Acad Sci U.S.A* (2001) 98:404–9. doi:10.1073/pnas.98.2.404
5. Radicchi F, Castellano C, Cecconi F, Loreto V, Parisi D. Defining and Identifying Communities in Networks. *Proc Natl Acad Sci* (2004). doi:10.1073/pnas.0400054101
6. Nichols S, Stich S. *Mindreading*. Oxford: Oxford University Press (2003).
7. Granovetter MS. The Strength of Weak Ties. *Am J Sociol* (1973) 78:1360–80. doi:10.1086/225469
8. Putnam RD. *Making Democracy Work. Civic Traditions in Modern Italy*. Princeton NJ: Princeton University Press (1993).
9. Putnam RD. *Bowling Alone. The Collapse and Revival of American Community*. New York: Simon & Schuster (2000).
10. Coleman JS. Social Capital in the Creation of Human Capital. *Am J Sociol* (1988) 94:S95–S120. doi:10.1086/228943
11. Bourdieu P. Forms of Capital. In: JC Richards, editor. *Handbook of Theory and Research for the Sociology of Education*. New York: Greenwood Press (1983).
12. Castelfranchi C, Falcone R. Principles of Trust for MAS: Cognitive Anatomy, Social Importance, and Quantification. In: *Proceedings of the International Conference of Multi-Agent Systems (ICMAS'98)*. Paris: July (1998). p. 72–9.
13. Castelfranchi C, Falcone R. *Trust Theory: A Socio-Cognitive and Computational Model*. John Wiley & Sons (2010).
14. Bratman M. *Intentions, Plans and Practical Reason*. Cambridge, Massachusetts: Harvard U. Press (1987).
15. Cohen P, Levesque H. Intention Is Choice with Commitment. *Artif Intelligence* (1990)(42). doi:10.1016/0004-3702(90)90055-5
16. Rao A, Georgeff M. *Modelling Rational Agents within a Bdi-Architecture* (1991). Availabl at: http://citeseer.ist.psu.edu/122564.html.
17. Wooldridge M. *An Introduction to Multi-Agent Systems*. Wiley and Sons (2002).
18. Pollack ME. Intentions in Communication. In: J Morgan ME Pollack, editors. Plans As Complex Mental Attitudes *in Cohen*. USA: MIT press (1990). p. 77–103.
19. Bratman ME, Israel DJ, Pollack ME. *Plans and Resource-Bounded Practical Reasoning*. Hoboken, New Jersey: Computational Intelligence (1988).
20. Sichman J, Conte R, Castelfranchi C, Demazeau Y. A Social Reasoning Mechanism Based on Dependence Networks. In: *Proceedings of the 11th ECAI* (1994).
21. Castelfranchi C, Conte R. The Dynamics of Dependence Networks and Power Relations in Open Multi-Agent Systems. In: *Proc. COOP'96 – Second International Conference on the Design of Cooperative Systems, Juan-Les-Pins, France, June, 12-14*. Valbonne, France: INRIA Sophia-Antipolis (1996). p. 125–37.
22. Falcone R, Pezzulo G, Castelfranchi C, Calvi G. Contract Nets for Evaluating Agent Trustworthiness. *Spec Issue "Trusting Agents Trusting Electron Societies" Lecture Notes Artif Intelligence* (2005) 3577:43–58. doi:10.1007/11532095_3
23. Castelfranchi C, Falcone R, Marzo F. Trust as Relational Capital: Its Importance, Evaluation, and Dynamics. In: *Proceedings of the Ninth International Workshop on "Trust in Agent Societies"*. Hokkaido (Japan): AAMAS 2006 Conference (2006).
24. Spece M. Job Market Signaling. *Q J Econ* (1973) 87:296–332.
25. Bird RB, Alden SE. Signaling Theory, Strategic Interaction, and Symbolic Capital. *Curr Antropology* (2005) 46–2. doi:10.1086/427115
26. Schelling T. *The Strategy of Conflict*. Cambridge: Harvard University Press (1960).
27. Conte R, Paolucci M. Reputation in Artificial Societies. In: *Social Beliefs for Social Order*. Amsterdam (NL): Kluwer (2002). doi:10.1007/978-1-4615-1159-5
28. Castelfranchi C, Falcone R. From Automaticity to Autonomy: The Frontier of Artificial Agents. In: H Hexmoor, C Castelfranchi, R Falcone, editors. *Agent Autonomy*. Amsterdam (NL): Kluwer Publisher (2003). p. 103–36. doi:10.1007/978-1-4419-9198-0_6

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

TYPE Original Research
PUBLISHED 06 October 2022
DOI 10.3389/fphy.2022.941824

Check for updates

OPEN ACCESS

EDITED BY
William Frere Lawless,
Paine College, United States

REVIEWED BY
Chris Arledge,
Johns Hopkins University, United States
Laurent Mary Chaudron,
Aix Marseille Université, France

*CORRESPONDENCE
Ryan Phillip Quandt,
ryan.quandt@cgu.edu

SPECIALTY SECTION
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

RECEIVED 11 May 2022
ACCEPTED 20 September 2022
PUBLISHED 06 October 2022

CITATION
Quandt RP (2022), AI in society: A theory. *Front. Phys.* 10:941824. doi: 10.3389/fphy.2022.941824

AI in society: A theory

Ryan Phillip Quandt*

Economics, Claremont Graduate University, Claremont, CA, United States

Human-machine teams or systems are integral parts of society and will likely become more so. Unsettled are the effects of these changes, their mechanism(s), and how to measure them. In this article, I propose a central concept for understanding human-machine interaction: convergent cause. That is, Agent 1's response to the object is caused by the object and Agent 2's response, while Agent 2 responds to Agent 1's response and the object. To the extent a human-machine team acts, AI converges with a human. One benefit of this concept is that it allows degrees, and so avoids the question of Strong or Weak AI. To defend my proposal, I repurpose Donald Davidson's triangulation as a model for human-machine teams and systems.

KEYWORDS

triangulation, intersubjectivity, object constitution, social reality, human-machine interaction, action, language

1 Introduction

An automated vacuum zig zags across the floor. On less guarded days, it is easy to think the vacuum is "looking" for dirt and is satisfied as it crackles over some. Some of its crossings look random and inefficient, yet as it maneuvers around chair legs, cautiously passes under curtains, tracks walls, and detects streaks of dirt, its motions enforce the impression that it "looks" for dirt. Colloquial explanations of its behavior use words like maneuver, caution, tracking, detection, words that exemplify propositional attitude reporting sentences, or sentences that concern cognitive relations [1]. Standard examples: "Jack likes Jill," "Jack wants Jill to fetch a pale of water," and "Jack accidentally broke his crown." Describing the vacuum's behavior with such sentences suggests the vacuum has a mental life devoted to cleaning floors. Whether the vacuum is intelligent is less relevant here than our tendency to describe behavior in terms of propositional attitudes. This tendency is my first premise.

Still, there are good reasons to think the vacuum lacks propositional attitudes, like intending to pick up dirt, and these reasons weaken our tendency to think about the vacuum as intentional. Resistance to taking our colloquial way of describing machine behavior seriously qualifies my first premise. The vacuum must believe it is picking up dirt (or failing to) to intend as much—at least, an observer like you or I must infer a belief from its behavior. To intend is to believe, and *vice versa*. I cannot intend to pick up a cup unless I believe there is a cup nearby, that I can reach it, that extending my hand just so, applying pressure, and retracting my arm will pick it up, et cetera. Another reason to deny the vacuum propositional attitudes is that doing so fails the substitution test [14, pg. 97]. Suppose the vacuum was designed to sense, then report, what it inhales. If the vacuum reports "dirt," does it also intend to pick up soil? Crumbs? fur? Arguably, no. A vacuum does not distinguish them, nor would more sensitive sensors and precise reports do so. Substitution and synonymy test whether the vacuum has a concept. So although we may describe machine behavior with sentences that report propositional attitudes, the tendency may be weaker or stronger depending on how sophisticated the machine behaves. In the weakest of cases, when machines look unintelligent, our language conveys its own limit. There seems no adequate way to describe events that come between mindless events, on one hand, and thought or action, on the other [14, Essay 9].

An automated vacuum is one appliance within "smart" or "helpful" homes. Others are air purifiers, cameras, thermostats, lights, door bells, displays, garage doors, and apps to control them all. Many are voice, motion, or light activated, too. Control and security are not their sole ends; the house is becoming a human-machine system. If our tendency to ascribe propositional attitudes to machines has degrees, their integration into the home (clothes, cars, business) strengthens this tendency.[1] Google's Rishi Chandra, Vice President of Product and General Manager at Google Nest, said in 2019 that we are transitioning from mobile computing (having a computer on one's phone, for example) to ambient computing, or "having an always accessible computer right at your fingertips, that understands you, that can do things on your behalf to help you in different ways" [2].[2] One AI system will manage various devices and be sensitive to a user's needs, habits, and desires so that an evolving intelligence forms the environment independent of the person's actions, yet responsive to their own attitudes and patterns.[3] Within such a system, the automated vacuum will be deployed when and where the floor is dirty. Ambient homes (and advances in machine learning, generally) motivate my first premise: colloquial descriptions of machine behavior (will) shape how we perceive their behavior.

Forecasting, prediction, and prophesy are notoriously hard and uncertain. Measuring the effects of AI in society has three associated challenges: 1) incorporating the social, contextual, or purposive nature of action (coordinated or not), 2) conceptualizing a trajectory of development that incorporates human agents,[4] and 3) allowing artificial intelligence to differ from expectations in productive ways. These challenges hang together insofar as machines emerge in society as social agents. They operate among others in rapidly changing and unexpected ways. Hence problems of brittleness [3] and perception [, 60, 2, 4]. And, regardless if AI has intelligence proper, the sophistication of these machines are often treated as if they were intelligent, and so behavior adjusts likewise. This may explain human decision biasing in which AI system recommendations lead humans into error [61, 38, 27, 7] as well as loss of situational awareness among humans and performance degradation [53, 45, 20, 62, 55, 11]. Theory accounts for these challenges and the proposal here outlines how certain limits of AI and problematic effects on human behavior are related.[5]

The question, 'Do humans change when living in a 'smart' home?' requires a theoretical model with empirical studies.[6] A theory informs how we interpret a study's results, design experiments, select methods, credit some results while discounting others. Theory fixes what to look for, expect, and conclude. The theory proposed in this paper is triangulation, which expresses a trajectory as well as interaction. In mathematics, triangulation is a way of discovering a point's distance from a baseline by measuring another point systematically related to it. Put within social relationships, triangulation describes how someone conceives an object relative to another person (the baseline relation), who also interacts with the same object. One person correlates their response to an object according to the concurrent response of someone else and, as a result, their responses converge on an object from their mutual correlations. When persons intend their response relative to perceiving another person's intended response (and the observed person does likewise), their responses causally converge—the basic concept of triangulation. This theory clarifies the dynamic of, and requirements for, human-machine teams and systems. While an argument for triangulation follows, an argument which motivates its use,[7] the theory stands or falls from empirical study.

2 Triangulation

Humans tend to talk about machine behavior as if it was intended, and so think of it as such. Acting as if machines were intentional and acting with machines differ, however, since joint action requires aligned intent at minimum. Two or more agents

[1] Their integration also enables increased autonomy of the human-machine system, though I put this aside for future work.

[2] Think, too, of Weiser's "ubiquitous computing," in which computer chips permeate one's environment and body [37]. Also see Kaku's prediction of the next hundred years for AI [[38], Ch. 2].

[3] And so these systems will be autonomous since they will perform tasks without continuous human input [39] and possess intelligence-based capacities, that is, responding to situations that were not anticipated in the design [40] and function as a proxy for human decisions [41]. AI also approximates human activities like the "ability to reason, discover meaning, generalize, or learn from past experiences" [42]. This understanding of intelligence adds specificity to McCarthy's claim that artificial intelligence is goal-directed activity, though it is important to note that his definition is intentionally open-ended [43].

[4] This paper assumes AI has reasoning-like processes and these will likely become more sophisticated and sensitive. This second challenge involves placing evolving capacities among persons.

[5] And so this paper joins those responding to Wiener's earlier call for philosophy in light of rapid technological progress [44].

[6] On the importance of theory for empirical analysis, see [45].

[7] To be clear, the argument is incomplete since my purpose here is to defend triangulation's plausibility for use in research. More rigorous argumentation, however, is needed.

act for the same end in coordination. There is a give and take of deliberation, reasons and counters, adaption to unforeseen circumstances, problem solving. How much AI contributes to these daily processes measure its integration into society. First, I will set out the requirements for joint action, which names a threshold and degrees.[8] In doing so, I skirt debate over Strong or Weak AI.[9] Triangulation clarifies the extent to which machines can jointly act with humans by meeting certain requirements, although some requirements may be barred in principle.

Across essays, the philosopher, Donald Davidson, proposed triangulation as an analogy, a model, and an argument.[10] Commentators disagree on what the argument is or whether one name stands for two arguments. Myers and Verheggen note, "…there is no such thing as 'the' triangulation argument explicitly laid out in Davidson's writings" [17], so the argument below is not strictly his. Adding to the ambiguity are the various conclusions Davidson inferred from triangulation: language is social (or there is no language only one person understands) [14, Essay 8], communication requires the concept of an intersubjective world [14, Essay 7], language is required for thoughts [14, Essay 7], and that stimuli become an object when two people recognize one another reacting to that stimuli in similar ways [14, Essay 8]. More commitments are at stake under the heading, "triangulation," than I will defend—broader views on thought, language, action, subjectivity, and objectivity—since Davidson's system threads through triangulation. Yet he never polished a formal argument. By repurposing it for human-machine teams and systems, I underscore its empirical bearing (abstract as it is). This move, if prompting select interpretations of experiment, is my main contribution.

Triangulation models how interaction shapes an intersubjective reality that is never given once and for all. Ideally, the model has empirical purchase (explanatory and predictive) and is falsifiable. Arguments couple then with testing. Davidson's remarks on decision theory generalize: tests only partially support theory insofar as tests depend on how the theory is applied [14, pgs. 125-126]. Experiment design, in other words, assumes theoretical commitments. Before testing a theory, we expect an argument for why the theory nears truth.

2.1 The argument

The threshold from stimulus to object, conditioned reflexes to thought and action, marks the difference between one agent acting as if an object had agency to acting with another agent. Triangulation defines this threshold. When machines obtain agency, and so pass the threshold, they enter society. Theorizing intelligent (in the sense of mental) interaction also explains and predicts how humans will respond to AI systems in teams. Convergent causality sets the trajectory and critical point, and includes requirements for human-machine action, how activity changes with machines, and how objects change as well. Again, convergent causality is how two beings simultaneously correlate their responses to the same object in light of one another. Triangulation, then, fixes the irreducible elements from which causal convergence occurs. The stakes are set.

Some definitions are in order. An object is something taken as such and as existing independently of the one so taking. A language is an abstract object composed of a finite list of expressions, rules for combining them, and interpretations of these expressions according to how they are combined [14, pg. 107]. An action is something done with a belief and an intent. These definitions are meant as weak, ordinary senses of "object," "language," and "action" to get us going. More precision comes in the argument for triangulation since these concepts draw from each other.

Mental content will be synonymous with conceptual or intentional content here [34, pg. 12], and so triangulation concerns requirements for concepts or intent. Other prevailing notions of the mental, such as non-conceptual [18], representational [19], phenomenal [20], and intuitional content [21], are left out.[11] Propositional attitudes (id est, mental content) have three properties, which are described below and assumed. Contestable, though plausible.[12] Each depends on a close parallel between thought and the meaning of sentences [14, pg. 57], and so may be dubiously assumed in an argument that language is sufficient for thought. Still, there are reasons for accepting them.

First, propositional attitudes can be expressed using sentences that are true or false. So when Archidamus exclaims, "I think there is not in the world either malice or matter to alter it," speaking of Sicilia and Bohemia's alliance, his sentence is true or false.[13] Davidson argues that meaning is truth-conditional by recycling Tarski's theory of truth [22] as a theory

[8] My concern is not AI-mediated forms of communication platforms, such as social media. A main difference is that users are largely unaware of how machine-learning algorithms respond to and anticipate their choices for information. This is not human-machine interaction as I conceive it here, which requires transparency and mutual responsiveness.

[9] And so sidestep Searle's famous Chinese Room thought experiment, which argues that strong AI is impossible [46]. For a later reflection of his, see [47].

[10] Davidson has been criticized for obscuring its status. He invokes model and argument in "The Emergence of Thought" [14, Essay 9; pgs. 128-134].

[11] Davidson never defended his view on mental content, though acknowledging other options [48].

[12] Following Myers and Verheggen [34, pgs. 12-15], I begin with propositional attitudes. I do not begin with the first property, the holism of the mental, since I find it the most questionable.

[13] From Shakespeare's *A Winter's Tale*, I.1.

of meaning, which leaves truth undefined [23]. For every sentence, the theory generates a T-sentence: "'*s*" is true if and only if *p*,' or '*s* means *p*.' That is, "'Archidamus thinks *x*" if and only if Archidamus thinks *x*,' and so the sentence is dequoted, such that any speaker of the language used by the T-sentence would know the original sentence's truth conditions. The theory works if it successfully sets criteria for understanding a language, describes what a speaker intuitively knows about their language, and can be used to interpret their utterances [14, pg. 132].

Second, sentences about someone's propositional attitudes are opaque semantically because their meaning depends on belief and intent. So "Archidamus thinks *x*" is true or false relative to Archidamus' beliefs. He may change his mind so a sentence once true become false. Or a hearer misinterpret a joke as an avowal. For a third person who did not hear Archidamus' utterance but a report about his beliefs, the best they can do to verify is ask Archidamus. Meaning cannot be reduced to extension, which spans behavior, gestures, acts, or objects [24]. A speaker must intend their words to be taken as such by a hearer and the hearer rightly pick up on that intention [[25], Essays 5 and 6]. Attitudes, like intent, belief, and desire, inform an utterance's meaning.

The third, and last, trait of propositional attitudes is mental holism, which, in Davidson's words, means "the interdependence of various aspects of mentality" [14, pg. 124]. Intent clings to belief, belief to intent, and to parse one from the other distorts both. An intention cannot be understood without beliefs, and beliefs are mute without intentions that express them. This is not to say that all beliefs are public, but that all we have to go on for understanding another person's beliefs must be.[14] More, a single attitude requires mastery of many concepts, just as possessing one concept assumes many. Consider what must be in place to misapply a concept. Besides a concept in question, other concepts pick out a spectrum of relevance for what rightly or wrongly falls under the concept. Invoked by the concept, 'dirt,' are cleanliness, a distinction between indoors and outdoors, an entryway and a bedroom, work boots and high heels, soil, sand, and so on, with each assuming their own concepts. This is why discriminating between fur, crumbs, or hair differs from mastering the concept, as noted in my opening example.

With traits of propositional attitudes in place, the argument can be put within two thought experiments.[15] The first argues extension is limited by indeterminacy, whereas the second expands indeterminacy to words themselves. Triangulation hones in on the requirements for successful communication despite.

Indeterminacy, "inscrutability of reference," or "ontological relativity" were introduced by W. V. O. Quine [26]. His claim, a step toward mental holism, is that a word cannot be fixed to one object. Speakers cannot divulge word meaning from ostention alone; hearers understand the utterance and act within a purposive context, that is, by ascribing intention and beliefs. The richer this purposive context (more precise concepts shared by persons), the more likely communication succeeds since agents can navigate situations of high uncertainty (such as meeting strangers). Quine argues for indeterminacy with a thought experiment called radical translation.

Imagine this scenario [40, pgs. 28-30]. A field linguist meets a speaker from an unknown land, who speaks a language unlike any she knows. The linguist has only query and ostension at her disposal. As she gestures at objects to elicit a response, a rabbit jumps out of a bush and runs between them. The unknown speaker looks down, gestures at the rabbit, and exclaims, "Gavagai." The linguist jots down the words, "gavagai" and "rabbit." Another rabbit appears shortly after. The linguist gestures and prompts, "Gavagai?" and the man nods. Once the linguist has done the same with other speakers of the same language, she can be confident in her translation. Even so, indeterminacy surfaces. 'Gavagai,' that is, can mean "rabbit," "undetached rabbit part," "rabbit stage," 'the unique appearance of the rabbit's left foot while running less than 20 miles per hour,' and so on, and no number of queries settles things.

In the proclivity of the native to say "gavagai" and English-speakers, "rabbit," that is, their speech dispositions, Quine argues for persisting indeterminacy. A more complex syntactic apparatus enables the linguist to pick out rabbits, their parts, and stages within the other's tongue, but that apparatus is relative to an entire catalogue of phrase pairings (what Quine calls a translation manual). Catalogue in hand, indeterminacy seems to disappear, but only seems. Whole catalogues can be compiled for every speech disposition of the language consistently, yet these catalogues rival one another by offering inconsistent interpretations of a given utterance [[27], pg. 73]. Their internal coherence and explanatory power cannot rule out rivals. Put again, one language cannot perfectly and uniquely map onto the words, phrases, references, or meanings of another. By a backdoor of indeterminacy, we come to triangulation. Davidson calls triangulation before language primitive.

Convergent cause is the basis of interaction in triangulation. Davidson glosses, "Each creature learns to correlate the reactions of other creatures with changes or objects in the world to which it also reacts" [14, pg. 128]. Responses to environs or objects are tailored to others' responses. In Quine's scenario, the field linguist supposes the rabbit prompted the speaker to say "gavagai." Organisms discriminate a like cause apart from language in primitive triangulation as conditioned responses to stimuli. When one deer hears a predator and runs, other deer run, too, even if they did not hear the predator. These responses are learned, much like Pavlov's salivating dogs.

[14] This does not entail that meaning is extensional. Davidson explains, "Propositional attitudes can be discovered by an observer who witnesses nothing but behavior without the attitudes being in any way reducible to behavior" [14, pg. 100].

[15] Ludwig alludes to the same [[49], pg. 81].

Learned discrimination is part of mental life, but it does not pass the threshold of conceptual, or intentional, content, which requires beliefs as well, so the stimuli are not conceptualized as such either.

Discerning a threat or food source differs from applying a concept since the latter assumes the possibility of misapplying. Except from our thoughtful vantage, a creature's behavior apart from language does not evince defeasible beliefs. Deer may return to a meadow after a predator does not appear, but their return does not suggest the notion of a false alarm. That said, this claim is not to deny the possibility that deer have a rich mental life. They may even have their own language, and so have concepts and beliefs. There is no way for us to know without more precise ways of communicating. Their triangulation is primitive to us. Describing the deer's behavior as a false alarm projects our concepts, but recourse to our own propositional attitudes does not justify inferring concepts about them. Again, behavior alone does not sufficiently evince one or the other. It is indeterminate, as is the object. The stimulus that caused the coordinated responses does not reify into an object as such because there is no criteria for right or wrong responses.

By contrast, a solitaire, someone who never observes someone else, does not have discernible thoughts either.[16] Davidson's remarks suggest a solitaire has a poorer mental life than mute creatures who triangulate. Lacking shared stimuli, responses are conditioned to a narrow sequence of stimulus and response. There is little "distance" between the solitaire and the stimulus because there is no one else to observe responding to the same. Once another creature enters the scene, the response separates from the stimulus since it is one's own rather than the other's response. There is a perception of the stimulus and the perception of the other responding to the stimulus, and so an added dimension of correlation. In this way, the solitaire differs from primitive triangulators.

The scenario of primitive triangulation names a requirement for shared stimuli. Like uttering "gavagai," the stimulus harbors indeterminacy. The linguist banks on the dramatic moment when the rabbit bounds out of the bush. Maybe the speaker responds to the event otherwise than the linguist expects (and so calls for a hunt or invokes a god). Without words, responses to stimuli lack a mechanism for specifying what causes the response. Davidson mentions two ambiguities [14, pgs. 129-130]: first, those features of the total cause that are relevant to the response; second, whether the stimulus is proximal or distal. The former explains how creatures correlate responses. One creature must be able to recognize in another creature's response what that other creature is responding to. And the second ambiguity concerns the stimulus itself. Is it the rabbit itself, a rabbit part, the suddenness of the event, or its wider social significance (such as a good or bad omen). Until these ambiguities are overcome, creatures do not identify a cause from mutual responses to stimuli since evidence lacks that the creatures are responding to the same thing [34, pg. 17]. A cause proper must be socially identified, public and precise. In sum, the stimulus and correlated responses are underdetermined until creatures evoke language.

Met, the requirements for successful (linguistic) communication identify a cause, and so surmount the aforementioned ambiguities. Stimulus becomes concept, assuming a plethora of other concepts. Davidson specifies the requirements with an idealized model [17, Essay 7], which does not present what happens in the mind or self-aware expectations. People talk without applying an internal dictionary and grammar. The model below serves a distinct purpose: it concerns communication, whereas triangulation depicts how thought and language are mutually social. Still, these requirements inform the baseline of the triangle (the interaction of agents), which, in turn, enables agents to identify and respond to the same cause.

Say a speaker has a theory for how to speak so that a hearer will rightly hear her and the hearer has a theory for how to make sense of the speaker's words. Each theory splits into a priory theory, or ways of interpreting an utterance before the uttering, and a passing theory, which form during the occasion of utterance (how the words are voiced and heard in the moment). Prior theories consist in knowledge of grammar, idioms, definitions, past uses. A hearer anticipates a speaker and the speaker a hearer according to prior theories. Passing theories are how *this* hearer interprets *this* speaker's utterances, and how the speaker voices them. If a speaker slips, saying, "Our watch, sir, have indeed comprehended two auspicious persons," the hearer may rightly understand 'comprehended' as 'apprehended' and "auspicious" as "suspicious." If so, passing theories converge without loss. Maybe the mistake was never made before nor again so that past uses do not prepare us for one-off utterances. Less dramatic examples bare this out as hearers make sense of utterances never heard before. Their prior theories do not align. So successful communication only requires that a hearer pick up a speaker's intent—that is, passing theories converge.

The intent behind an utterance becomes more precise as grunts and gestures become proper names and predicates, truth functional connectives ("and," "or," "not," "if . . . then"), and quantification ("some," "all," "this"). Better specification of intent conveys the same cause for the utterance and obtains a threshold to move from primitive triangulation to its mature form [14, pg. 130]. Complex as language is, though, the indeterminacy our earlier linguist faces confronts neighbors.[17] Robust prior theories do not secure interpretations of utterances. Similar words or phrases may be used differently across persons, in endless reams of contexts, or with various forces (asserting, exclaiming, asking,

[16] There may never be an actual solitaire. The idea of a solitaire is hypothetical and meant to draw out commitments.

[17] This is his thesis of radical interpretation [13, Essay 9].

joking). Still, hearers often hear rightly, which narrows on linguistic competence. The formal apparatus is not enough to communicate, though required.

For passing theories to converge, speaker and hearer must (largely) share the same world.[18] Davidson gets at this when he claims that speaker and hearer must agree on most things to disagree [23]. The point of contention assumes other concepts are shared. Widespread agreement also facilitates communication. That is, the intersubjective world of objects and concepts enables creatures to overcome indeterminacy. Hearers make sense of speakers by taking cues from how they interact with the world. Note the circular reasoning, which may be virtuous or vicious. Thought requires two creatures to interact since the cause of thought must be a certain shared stimulus. The creatures correlate responses, but that correlation is not action proper until one creature recognizes the other's intent.[19] A formal apparatus with signs does not suffice due to persisting indeterminacy (and the nature of linguistic competence [17, Essay 7]). Objects and concepts, the stuff of mental life, allows creatures to express their intention. Thus, convergent cause relies on the social bearing of language and thought simultaneously.

Setting out a few commitments, then: the content of a belief comes from its cause, its causes are the object and the correlated responses of at least two persons, and one person perceives the intent of another in their simultaneous and mutual response to the object. Triangulation moves from primitive to robust with language since the two ambiguities from before can be resolved. An agent specifies relevance within the total cause through linguistic precision. In doubt, a hearer queries for more information to know what a speaker means. The same can be said for whether the cause is proximal or distal. Degrees matter since indeterminacy threatens fluent agents, yet these ambiguities are more or less resolved as they act in an intersubjective world. From shared objects, concepts, and beliefs, a hearer can navigate malapropisms and other novel, idiosyncratic utterances.

Triangulation (by which I mean robust, or linguistic, triangulation from here on) explains how thought is objective. Truth or falsity is independent of the thinker [14, pg. 129]. To apply a concept, someone must have the notion of misapplying. In this way someone thinks "this" rather than "that," oaks not elms. These distinctions come from diverging responses to the same cause, as when I stand beside an arborist and exclaim, "What a beautiful elm!" and she replies, "That is an oak." Triangulators correlate responses to specify the same cause by getting it right and, sometimes, wrong. Without frustrated or vague attempts that are corrected by others, triangulation would not rise beyond discerning stimuli. Such interactions engender external criteria for right and wrong uses of concepts.

Besides explaining how thought is objective, convergent cause objectifies the cause and so enables a hearer to interpret a speaker since the hearer can refer the speaker's utterance to its cause. But Davidson grants the notion of convergent cause—the crux notion—remains unclear and uncertain [[28], pg. 85]. Hence his initial use of triangulation as an analogy. Triangulation depicts requirements that approximate it, yet empirical analysis may clarify and test convergence. So my recommendation of the recycled theory has two ends: 1) to guide studies on human-machine interaction and 2) to illumine the theory itself. Triangulation picks out sufficient conditions for thought and language, but, here, convergence has not been shown as necessary for thought and language. More on this shortly. We posit that, absent another agent, a shared cause, or language, there is no thought or action, if action is understood as doing something intentionally or for reasons [29]. But, with them, agents have everything they need.

This section began with traits of propositional attitudes, exponed primitive triangulation and two persisting ambiguities, and how linguistic triangulation resolves those ambiguities through convergence. An upshot is that first, second, and third person lose primacy to the irreducible relation between two agents and a mutual object [34, ft [30]] [31]. Action expresses a robust correlation of responses to the same. In the triangle, focus shifts off a given entity to an interaction according to an object, loosely defined. Convergence is the pith and marrow. Primitive and linguistic triangulation mark a threshold in which conditioned reflexes to stimuli refine into thoughts with convergence of simultaneous responses. Let me address some objections to better position triangulation with respect to human-machine interaction.

2.2 Objections

Triangulation has critics. Recall that Davidson never shaped one argument for its defense. Most readers of him, according to Verheggen and Myers [34, pg. 11], find two arguments: one concluding that triangulation fixes meanings; another that triangulation is required for the concept of objectivity. Critics pick apart each in turn. The notion of convergent cause, by contrast, offers a central concept for objectivity and meaning, grounding one argument. Above, I sketched such an argument to shift presumption in favor of triangulation as a theory for human-machine teaming. More argument will be needed to resolve objections than provided here, but my aims are modest. The theory of triangulation merits testing.

[18] And so radical interpretation theorizes the requirement for a "common ontology" to share meaning in a multi-agent system [50]. At the same time, an implication of radical interpretation is that, assuming a largely common ontology, two agents can recognize and navigate discrepancies in their use of words.

[19] Ascribing propositional attitudes happens within the time and place of speech [16, Essay 5].

Verheggen and Myers note four lines of critique, but two bear on my recycling [34, Ch.1, Section 3]. The first states that perceiving objects fixes meaning, not language. Burge champions this objection, appealing to perceptual psychology, and uses empirical evidence to support the claim that perception picks out and specifies objects by observing a creature's behavior to a stimulus alongside one's own.[20] Due to the nature of perception, in other words, primitive triangulation suffices. Discernment and detection identify the cause of the other's behavior with which one correlates one's response.

This first objection entails that sensing enables joint action rather than language. Before responding, these objections merit a brief foray into their consequences for development. At stake are how we allocate resources, what to expect from our successes, and how to understand our failures. Burge's view puts perceptual mechanism at the center of human-machine teams. AI recognizes an object, an agent, and an agent's reaction to that object, and as perceptual limits are overcome, AI will enter society as contributing agents. Language is a helpful appendage, streamlines certain activities, and encourages trust. And how humans perceive machines changes their own behavior, linguistic competencies aside. Convergence, on Burge's view, results from perceiving the same and coordinating.

For a response, here is a low-hanging fruit: we are concerned with propositional content, Burge with perceptual content. If that is all, better to prefer perception to triangulation since the latter demands more than the former. Triangulation requires perceptual sophistication and then some: linguistic competency, teleological behavior, and, ultimately, intelligence. One reason for adopting triangulation is that perception is not enough for joint action. This motive is bolstered by a recent publication of the National Academies of Sciences, Engineering, and Medicine, which finds that more than perception is required for coordination [32].

Perception, however acute, cannot proffer objectivity because its content cannot be true or false. The requirement for a truth value is an external standard, which, in turn, requires the notion of misapplied concepts. While perception responds to a stimulus, Burge must add that perceiving the stimulus causes a belief, mental content that is either true or false to the perceiver. Again, there is no satisfying evidence that mute perception contains propositional content. Burge is right that perception (in the wide sense of interacting with objects and agents *via* the senses) is required for mental content. Ambiguities of scope and depth frustrate the identification of a cause that one creature simultaneously responds to in light of another creature's response (who also responds to the first creature's response to the object). Perceived content couples with predicates when involving belief and intent, but predicates are expressed *via* language. Only then do we have information that resolves ambiguity, underdeterminacy, and indeterminacy, however defeasibly. The content of our (human) perception is always more than sheer perception.[21]

A second objection deserves pause. Scholars criticize triangulation as a circular account of language and thought. This is either a bug or a feature. Given circularity, triangulation is vicious (the charge), uninformative, which can be decided by experimentation, or beneficial as a consistent non-reductive account of language, thought, and action. The circularity surfaces in the move from a primitive triangle to a robust, linguistic one. If there is language, there is thought, but language requires thought. Objectivity, too, can replace either "language" or "thought" in the prior sentence. One assumes the others.

A vicious circle means that at least one of the triangle's three points reduces to another, and so triangulation distorts the relation. The theory puts undue burden on human-machine action. More damning still, convergence collapses. An agent no longer acts by responding to an object in view of another agent's response. On triangulation, the task of picking up a cup differs from refilling a mug with coffee, a bottle with water, and emptying a cup of grease. Triangulation explains how closed contexts, such as programming for a specific task, differ from open contexts with uncertainty (and so theorizes brittleness). If wrong, human-machine teams may enter open contexts gradually by programming machines to identify select tasks from a catalogue of closed contexts according to a set of rules. Such task-based development is severely limited if triangulation is right.

Proof for triangulation depends on 1) showing that no element reduces to another, 2) closing off alternative theories, and 3) offering a convincing account of how and why the elements hang together. I return to 2) in a moment. On 1) and 3), Davidson grants that triangulation stems from conviction in humanity's sociability.[22] This conviction is either empirical or *a priori* depending on the status of mental capacity. Empirical, if one takes facts about speaking and thinking as natural facts about how we speak and think. *A priori*, if triangulation presents what the concepts of speaking, thinking, and acting mean [24]. Triangulation is theoretical in either case such that empirical testing is at best indirect. Experiments assume theory. An experiment that seems to justify or falsify the theory can be explained away. But how well triangulation makes sense of successes and failures, not to mention spawn development and illuminate tests, favors the theory. A social theory of thought, language, and action would benefit AI research. That

[20] See [, 3, 7, 51].

[21] A point eloquently argued by McDowell [53].

[22] Which is not to say that triangulation is immune from argument.

said, if designs based on reductive theories widely and repeatedly succeed, triangulation may rightly be discarded.

Alternative theories have attracted support in AI (and so we come to 2) above). Major contenders come from Language of Thought and Computational Theory of Mind [42, 33]. A full defense of triangulation must engage with these theories. My modest aim has been met if the theory seems plausible, attractive, and beneficial. Triangulation names sufficient conditions for thought, language, and action, and so articulates a threshold for human-machine teams to act jointly. More, the distinction between primitive and robust triangulation expresses the grey area before AI comes to its own rich mental life, yet is treated as such by humans. This natural default leads to application.

3 AI in the triangle

If triangulation as a non-reductive account in the end depends on conviction, that is, one is convicted over what language and thought are as natural facts, then empirical tests shoulder or dampen the conviction. Davidson uses triangulation as an example [14, pg. 105] and analogue [14, Essay 9] for describing a set of conceptual claims. Experiments put flesh on these claims and their underlying conviction.[23] Applying triangulation may also gain a better understanding of convergence, and so clarify the argument. But I step between planes, if you will, by "applying" triangulation: from conceptual argument to empirical theory and analysis. This move can be opposed by someone who agrees with triangulation as a set of claims yet objects to its refashioning as empirical theory. Or by someone who objects to the refashioning below but accepts another.

My main research question, recall, concerns how humans will act with machines, especially in teams. The irreducibly social element of triangulation means that communication is bound up in joint action and thought. More, how humans describe events, objects, and persons contribute to how they think of them, and humans lack a vocabulary to describe the murky area between thoughtless objects or coincident events, on one hand, and intention-filled ones, on the other hand. Yet AI, as neither an inert object, nor fully rational agent, falls somewhere in this blur. Triangulation, I now argue, helps us conceptualize this situation and the trajectory for human-machine interaction.

The theory looks like a three dimensional triangle (see Figure 1 below). The back plane depicts primitive triangulation, where two languageless creatures discern conditioned by past experiences. The front plane represents robust triangulation. As the baseline interaction becomes more complex, it is as if the triangle slides forward. This motion is represented by the lines joining the points. Robust triangulation is a solid line because it marks the threshold for thought, language, and action.

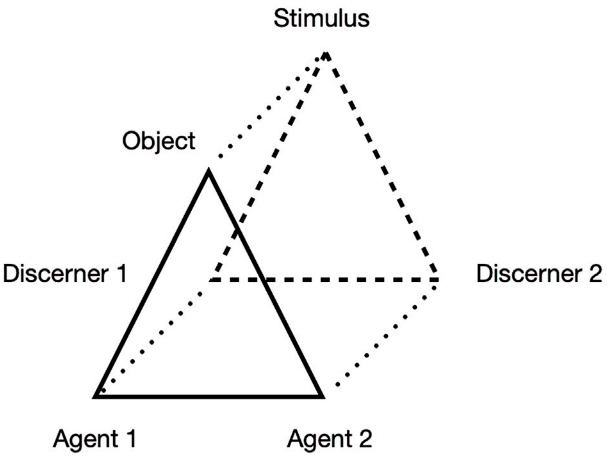


FIGURE 1
Depiction of triangulation. Front surface presents robust triangulation, while back surface presents primitive triangulation. The motion forward, as baseline between creatures becomes more rich, presented by lines connecting points. A respondent has conditioned reflexes to stimuli, whereas an agent conceptualizes that which prompts their response, given another agent doing likewise. While this distinction is a matter of degrees, the requirements for robust triangulation define a threshold. For autonomous human-machine teams, AI systems need to respond to objects in light of their human partner's response, and *vice versa*.

The triangle helps us answer two main questions: What is relevant for placing AI between discerner and agent? And what is required to move toward agency? These questions are closely tied since agency depends on how one is perceived by others. Not only must Agent 1 correlate their responses with Agent 2, Agent 2 must correlate theirs with Agent 1.[24] One way to think of AI considers how it operates, its hardware, and programming. Since few know or engage with machines according to hardware, we can put materials aside. AI as a social actor does not depend on silicon rather than graphene. That leaves us with how a machine operates and its programming. The latter is indecisive for our questions.

[23] Davidson writes that "his version of externalism depends on what I think to be our actual practice" [54].

[24] Yet the situation is more complex: there are often three, four, five, or more agents, temporal lapses between various agents' responses, agents responding to different objects at staggered times, and a history of past interactions with other agents to the same object or the same agents with different objects. These factors are likewise implicated in convergence.

3.1 Programs

A language is a recursively axiomatized system: a set of finite rules joined to a finite vocabulary that produce an indefinite number of expressions. Programming languages are formal since they operate by explicit rules.[25] Computation is a formal system that lacks insight or ingenuity, and so is closed, and has explicit, inviolable rules [[35], pg. 17]. Order of input from a fixed set of rules outputs predictably. The function does not assign meaning to the variables since the input results in a set output apart from an interpretation of the input. That is, the operation is "blind." A calculator computes 1 + 1 irrespective of whether the numbers represent tangible objects or not, though the algorithms of machine learning dwarf basic arithmetic.

Computation has a few properties, such as requiring a sequential, definite, and finite sequence of steps.[26] The output forms according to rules and protocols so that the result can be traced back through the program. A program can also be operated by anyone with the same result (it is worrisome if an analysis cannot be replicated). Also, a concrete, external symbolic system makes up the language [37, pgs. 25-27]. On its own terms, there is no indeterminacy in the program until the variables inputted and outputted occur within a purposive or intentional context–that is, until a machine acts among humans.

Put again, syntax lacks semantics until the output makes sense to others, expresses an intent, and endorses some beliefs as opposed to others. Davidson points out, for this reason, that exhaustive knowledge of how a machine works does not entail an interpretation of how the machine acts in the world. While software and hardware limit and sculpt behavior, design and function do not fix meaning (assuming machines can generate meaningful expressions and acts). As a result, computational language's definition and properties cannot make sense of how machines enter society. The program does not surface in the triangle. Limits and possibilities may be set, but these bounds do not give content to their realizations. How AI operates does.

Nor can material capacities or constraints bar AI from entering society in principle (at least, a conclusive argument has yet to appear). And even if Strong AI is impossible, machines may discern and come close to agency. More, humans may take machines as agents with whom to decide and act. Relevant evidence will come from human and machine behavior as their responses to stimuli converge.

3.2 Convergence

Machines in the triangle respond to an agent and an object (or event) concurrently. A solitaire, as opposed to a triangulator, lives in the world responding to stimuli apart from another agent with whom to correlate. One reason to think AI systems operate as a solitaire is that they respond to an agent or object, not an object in light of an agent's response to the same object. Humans may help machines correlate through teams since conditioned responses to either agent or object alone do not rise to convergence. Through teams, machines may become more sensitive to context. Supposing machines are not solitaires does not mean they triangulate. A human-machine team does not guarantee triangulation if the machine's response is not correlated from the human's response. A threshold must be passed cruxing on convergence.[27] And even if machines triangulate, it does not follow that humans triangulate with them. Humans may treat them as objects regardless. As promised, triangulation enlightens the grey area before machines have agency proper.

In a team, machines are more than solitaires if less than agents because humans interact with them toward an end. Art objects are an analogue. Artefacts of writing, for example, deviate from the original triangle with a lapse in time from the original inscription to the reading, and the settings differ [17, pg. 161]. A reader is blind to the writer's facial expressions, gestures, breathing, pace, and posture when the words were written. Instead, the writer uses textual cues to let the reader know what they mean. Through inscriptions, a successful author brings a reader into a shared conceptual space akin (not the same as) a shared world evoked by the triangle. The analogy misleads, however, if someone takes a machine's output to express the programmer's intent, as if the machine mediates an interaction between the human teammate and the programmer. A programming language cannot give meaning to the output since the programmer is no better off in interpreting a machine's behavior. As AI advances, machines will more frequently act unexpectedly.

The key insight from art is that certain objects gain meaning from how they elicit a response from a reader or viewer. While a written statement refers to something beyond the page, sculptures do not (except for monuments). A sculpture does not prompt the thought that the piece resembles a person qua art, but mimics the experience of meeting them [17, pg. 162]. They elicit a response through stone. But AI is also unlike sculpture insofar as it moves, recognizes, responds, makes noise, completes tasks. So machines may not make meaning *per se* until they obtain agency, yet elicit meaning from persons. My claim is that

[25] 'Formal language' has various meanings [55]. I adopt computable language.

[26] This holds in the case of parallel processing and an indefinite loop.

[27] More on this point shortly.

machines in teams act as more than solitaires because their behavior elicits and gains meaning from human responses, which machines respond to in turn for the sake of an end. Besides behaving as a programmer intends, machines facilitate human partnerships or not. And the success of teams depends on this facilitation. Again similar to art, success depends on the elicited response (among other conditions).

To the extent someone presumes a machine's intent from their elicited response, the machine's behavior converges toward human action. The presumption measures how far behavior converges, at least from the standpoint of a human agent. Risking redundancy, complete convergence means that 1) machines behave so as to respond to the agent as the agent responds to the object and 2) for the agent to perceive this response alongside their response to the object and act accordingly. Correlation is a step toward joint action and collaboration. But as long as the machine's behavior seems to express a specifiable intent (since intention cannot be hardwired), the machine elicits a response from humans that may adjust their behavior, beliefs, or the end for which they act. The response will be stronger and more precise as machine behavior becomes more precise, familiar, reliable, and consistent across time. Linguistic capacities, appearance, and conventions (even fabricated ones) will cultivate the response effected by machines. The human default to presume an intent is how we make sense of someone else's behavior: we will presume an intent until interaction suggests otherwise. Although an elicited response stands in for an intended act, there is a degree of potential convergence since 2) is met.

So measuring an elicited response is a test of convergence, but, as I argue in the next section, this effect is hard to isolate. The theoretical reasons for testing convergence have been stated. Humans lack the concepts or language that fill in the degrees from inanimate things to animate ones,[28] or living things from thinking ones. For this reason, humans default to presume an intent for behavior. That is, we make sense of activity by acting as if said activity expresses an intent until the presumption no longer makes sense. Depending on the strength of the default, humans presume an intent from AI and, given certain conditions of machine behavior, the presumption has more or less precision and effect. Triangulation exposes broader, contextual requirements for convergence since well-designed AI systems mesh with human routine, expectations, conversation, and so on. Insofar as systems succeed, humans will presume an intent behind machine behavior and act accordingly.

3.3 Social robot

Davidson's criticisms of the Turing Test frame the requirements for convergence, which define the threshold of, and trajectory toward, agency and autonomy. Triangulation severely qualifies the results of narrow experiments with a subject and a machine performing a task or interacting in a lab. First, let me describe the classic test. Turing argued that the question whether computers think can be answered by examining how humans understand them [36]. In his test, a participant sat at a screen and could type questions into the consul. Another person sat at another, hidden consul, an automated system operated another consul, and both attempted to convince the questioner that they are human and the other is the computer. The questioner only sees their answers on the screen. At the end of a short period, the participant would be asked which of the two was human and which the computer. Turing's test focuses on how someone interacts with a machine instead of asking about its isolated nature.[29] If thought is social, this interaction determines the nature of AI's operation—whether a machine has a mental life, agency, and autonomy.

Triangulation helps us spot limits with Turing's Test. Linguistic output on a screen leaves ambiguous whether the words were intended, manufactured, and elicit presumed intent from the questioner. A person cannot tell whether the answerer is thinking apart from deciding what the answerer thinks. Words cannot distinguish a person typing a response of their own or typing a prewritten response intended by someone else, which means intention cannot be recognized by the output. Evidence for a semantics of properly formed expressions consists in the following: 1) words refer to objects in the world, 2) predicates are true of things in the world, and 3) to specify the cause of uttering the words is to know the words' truth conditions [16, pg. 83]. These are conditions for ascribing propositional attitudes, for a hearer to think a speaker means something by their words. Davidson believes Turing subtracts vital evidence. A questioner before a screen cannot see how the answerer relates in a setting so that the questioner has less reasons for presuming the answerer's mental life and insufficient evidence for testing it.

How AI is housed, positioned in social situations, and navigates them reveals the extent humans believe the systems think. Humans likewise respond when teamed with machines from a presumed intent that is not frustrated from divergence (or frustrated attempts to correlate responses).[30] Using triangulation

[28] For an overview of the shades between inanimate and animate things that challenge its clean distinction, see [56].

[29] For appraisals of Turing's Test, see [8, 4, 9, 5, 57].

[30] As argued by [62], there is a decision of one agent to communicate as well, which means that full triangulation may be blocked if one agent decides not to communicate.

as a guide, Davidson states three conditions for something to think:

- Understood by a human interpreter;
- Resembles humans in certain ways;
- Possesses the appropriate history of observing causal interactions that prompt select utterances [16, pg. 86]

Turing held the first condition, though impoverishing how humans understand another, and excluded the second and third. A machine's behavior must make sense: humans recognize an intent that is consistent with apparent beliefs—both expressed in behavior, linguistic and otherwise—and the design of the machine facilitate such recognition. If a robot has a random tick, say, it 'comes off' as defective and hinders interaction. The last condition is hard to quantify, Davidson grants, since it brings out the holism of the mental. Using sentences goes beyond information since it draws from causal relations people have experienced. The conceptual map forms and evolves organically, or through a history of learned and correlated responses.

Controlled experiments enable us to isolate effects, yet risk removing needed assumptions of the variable of interest. So Davidson argues for Turing's Test. This paper began with claims represented by triangulation: mainly, that convergent cause is required for thought and action, which in turn requires language. This concept names the social nature of action. Objections bring out how theoretical commitments lead us to anticipate the role of machines, design experiments, and interpret successes or failures. Then we exponed the theory for application with a foray into the arts to argue that elicited responses from presumed intent should be the variable of interest as AI continues to develop, which presents a trajectory alongside the conditions for thought. Whether human-machine teams succeed depends on how machines elicit responses over time and how humans correlate their own responses as a result. This interaction allows flexibility for machines to behave in surprising ways without 'breaking' the interaction. Humans only need to be able to correlate their responses. Experiments can be designed that respect the aforementioned three conditions for thought since the conditions also name the setting in which humans interact among themselves as thinking animals. How well the theory makes sense of past experiments, prompts illuminating new ones, and upholds results from isolating elicited responses from humans marks the theory's success or failure.

Data availability statement

The original contributions presented in the study are included in the article/Supplementary material, further inquiries can be directed to the corresponding author.

Author contributions

The author confirms being the sole contributor of this work and has approved it for publication.

Acknowledgments

Thanks are due to the reviewers, Chris Arledge and Laurent Mary Chaudron, for reading and commenting on an early draft of this paper. Their feedback led to significant improvements, though all remaining errors are my own. I am also thankful to the editor's invitation to contribute to this special edition.

Conflict of interest

The author declares that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

1. Nelson M. Propositional attitude reports. In: EN Zalta, editor. The stanford Encyclopedia of philosophy. *Spring*. Metaphysics Research Lab, Stanford University (2022).

2. Phelan D. *Google exec on the future of nest: "No one asked for the smart home"* (2019). url: Available at: https://www.forbes.com/sites/davidphelan/2019/07/20/google-exec-no-one-asked-for-the-smart-home/?sh=3cb8f0bf3f3d July, 2019) (Accessed September 27, 2022).

3. Woods DD. The risks of autonomy: Doyle's catch. *J Cogn Eng Decis Making* (2016) 102:131–3. doi:10.1177/1555343416653562

4. Akhtar N, Mian A. Threat of adversarial attacks on deep learning in computer vision: A survey. *IEEE Access* (2018) 6:14410–30. doi:10.1109/access.2018.2807385

5. Alcorn MA, Li Q, Gong Z, Wang C, Mai L, Ku WS, Nguyen A. Strike (with) A pose: Neural networks are easily fooled by strange poses of familiar objects. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition IEEE/CVF*. Verantst (2019). p. 4845–54.

6. Yadav A, Patel A, Shah M. A comprehensive review on resolving ambiguities in natural language processing. *AI Open* (2021) 2:85–92. doi:10.1016/j.aiopen.2021.05.001

7. Endsley MR, Jones DG. *Designing for situation awareness: An approach to human-centered design*. 2nd. London: Taylor & Francis (2012).

8. Layton C, Smith PJ, McCoy CE. Design of a cooperative problem-solving system for en-route flight planning: An empirical evaluation. *Hum Factors* (1994) 361:94–119. doi:10.1177/001872089403600106

9. Olson WA, Sarter NB. Supporting informed consent in human machine collaboration: The role of conflict type, time pressure, and display design. In: *Proceedings of the human factors and ergonomics society annual meeting bd. 43 human factors and ergonomics society*. Veranst (1999). p. 189–93.

10. Yeh M, Wickens C, Seagull F. J.. Target cueing in visual search: The effects of conformity and display location on the allocation of visual attention. *Hum Factors* (1999) 41:27–32.

11. Moray N *Monitoring Behavior and Supervisory Control*, 2. New York: John Wiley & Sons (1986).

12. Wiener EL, Curry RE. Flight deck automation: Promises and problems. *Ergonomics* (1980) 2310:995–1011. doi:10.1080/00140138008924809

13. Young LRA. On adaptive manual control. *Ergonomics* (1969) 12:635–74. doi:10.1080/00140136908931083

14. Endsley MR, Kiris EO. The out-of-the-loop performance problem and level of control in automation. *Hum Factors* (1995) 372:381–94. doi:10.1518/001872095779064555

15. Sebok A, Wickens CD. Implementing lumberjacks and black swans into model-based tools to support human-automation interaction. *Hum Factors* (2017) 59:189–203. doi:10.1177/0018720816665201

16. Wickens CD. *The tradeoff of design for routine and unexpected PerformanceDaytona beach*. Daytona Beach, FL: Implications of Situation Awareness Embry-Riddle Aeronautical University Press (1995). p. 57–64.

17. Myers RH, Verheggen C. *Donald Davidson's triangulation argument: A philosophical inquiry*. Oxfordshire: Routledge (2016).

18. Evans G, McDowell J. *The varieties of reference*. Oxford: Oxford University Press (1982).

19. Burge T. *Origins of objectivity*. Oxford: Clarendon Press (2010).

20. Kriegel U. Phenomenal content. *Erkenntnis* (2002) 57:175–98. doi:10.1023/a:1020901206350

21. McDowell J. Avoiding the myth of the given. In: J Lindgaard, editor. *John McDowell: Experience, norm and nature*. Oxford: Blackwell Publishing, Ltd. (2008). p. 1–14.

22. Tarski A. The concept of truth in formalized languages. In: JH Woodger, editor. *Logic, semantics, metamathematics: Papers from 1923 to 1938*. Oxford: Clarendon Press (1956). p. 8.

23. Davidson D. *Inquiries into truth and interpretation*. Oxford: Clarendon Press (2001).

24. Davidson D. Comments on karlovy vary papers. In: P Kotatko, editor. *Pagin, peter (hrsg.) ; segal, gabriel (hrsg.):* Interpreting Davidson. Stanford: CSLI Publications (2001).

25. Grice P. *Studies in the way of words*. Cambridge and London: Harvard University Press (1989).

26. Quine WVO. Three indeterminacies. In: D Fèllesdal DB Quine, editors. *Confessions of a confirmed extentionalist: And other essays*. Harvard University Press (2008). p. 368386.

27. Quine WVO. *Word and object*. Cambridge, England: M.I.T. Press (1960).

28. Davidson D. *Problems of rationality*. Oxford: Clarendon Press (2004).

29. Davidson D. *Essays on actions and events*. Oxford: Oxford University Press (1980).

30. Davidson D. *Truth, language, and history*. Oxford: Clarendon Press (2005).

31. Stoutland F. Critical notice. *Int J Philos Stud* (2006) 141:579–96. doi:10.1080/09672550601003454

32. Endsley MR. *Human-AI teaming: State-of-the-Art and research needs*. Washington, DC: The National Academies Press (2022).

33. Fodor JA *The Language of Thought*, 2. Cambridge, Massachusetts: Harvard University Press (1980).

34. Schneider S. *the language of thought: A new philosophical direction*. Cambridge: MIT Press (2011).

35. Novaes CD. *Formal languages in logic: A philosophical and cognitive analysis*. Cambridge University Press (2012).

36. Turing AM. I.—computing machinery and intelligence. *Mind* (1950) 433–60. doi:10.1093/mind/lix.236.433

37. Weiser M. The computer for the 21st century (1991). URL Available at: https://www.scientificamerican.com/article/the-computer-for-the-21st-century/ (Accessed September 27, 2022).

38. Kaku M. *Physics of the future: How science will shape human destiny and our daily lives by the year 2100*. New York and London: Doubleday (2011).

39. Groover M. *Automation, production systems, and computer-integrated manufacturing*. 5th. New York: Pearson (2020).

40. USAF. *Air force research laboratory autonomy science and technology strategy/United States air force*. Wright-Patterson Air Force Base (2013). Forschungsbericht. URL Available at: https://web.archive.org/web/20170125102447/http://www.defenseinnovationmarketplace.mil/resources/AFRL_Autonomy_Strategy_DistroA.pdf.

41. USAF. *Autonomous horizons: The way forward/office of the U.S. Air force chief scientist*. Washington, DC: Forschungsbericht (2015).

42. Copeland BJ. *Artificial intelligence* (2021). URL Available at: https://www.britannica.com/technology/artificial-intelligence. (Accessed September 27, 2022)

43. McCarthy J. *What is artificial intelligence?* (2007). URL Available at: http://jmc.stanford.edu/articles/whatisai/whatisai.pdf (Zugriffsdatum: 11/24/2007.

44. Wiener EL. Cockpit automation: In need of a philosophy. In: *Fourth aerospace behavioral engineering technology conference proceedings SAE*. Veranst (1985).

45. Wolpin KI. *Limits of inference without theory*. Cambridge: MIT Press (2013). (Tjalling C. Koopmans Memorial Lectures).

46. Searle J. Minds, brains, and programs. *Behav Brain Sci* (1980) 3:417–24. doi:10.1017/s0140525x00005756

47. Searle J. Twenty-one years in the Chinese Room. In: J Preston, editor. *Views into the Chinese Room: New essays on Searle and artificial intelligence*. Oxford: Clarendon Press (2002). p. 51–69.

48. Davidson D. Responses to barry stroud, john McDowell, and tyler Burge. *Philos Phenomenol Res* (2003) 67:691–9. doi:10.1111/j.1933-1592.2003.tb00317.x

49. Ludwig K. Triangulation triangulated. In: MC Amoretti G Preyer, editors. *Triangulation: From an epistemological point of view*. Berlin: De Gruyter (2013). p. 69–95.

50. Williams AB. Learning to share meaning in a multi-agent system. *Autonomous Agents Multi-Agent Syst* (2004) 82:165–93. doi:10.1023/b:agnt.0000011160.45980.4b

51. Burge T. Social anti-individualism, objective reference. *Philos Phenomenol Res* (2003) 67:682–90. doi:10.1111/j.1933-1592.2003.tb00316.x

52. Bridges J. Davidson's transcendental externalism. *Philos Phenomenol Res* (2006) 732:290–315. doi:10.1111/j.1933-1592.2006.tb00619.x

53. McDowell J. *Mind and world*. Cambridge and London: Harvard University Press (1994).

54. Davidson D. *Subjective, intersubjective, objective*. Oxford: Clarendon Press (2001).

55. Novaes CD. The Different Ways in which Logic is (said to be) Formal. *Hist Philos Logic* (2011) 32:303–32. doi:10.1080/01445340.2011.555505

56. Margulis L, Sagan D. *What is Life?* New York: Simon & Schuster (1995).

57. Siegelmann HT. Computation beyond the turing limit. *Science* (1995) 268:545–8. doi:10.1126/science.268.5210.545

58. Bringsjord S, Bello P, Ferrucci D. Creativity, the turing test, and the (better) lovelace test. *Minds and Machines* (2001) 11:3–27. doi:10.1023/a:1011206622741

59. Cohen PR. If not Turing's test, then what? *AI Mag* (2006) 26:4.

60. Bringsjord S. The symbol grounding problem .remains unsolved. *J Exp Theor Artif Intell* (2015) 27:63–72. doi:10.1080/0952813x.2014.940139

61. Clark M, Atkinson DJ. (Is there) A future for lying machines? In: *Proceedings of the 2013 deception and counter-deception symposium* (2013).

62. Xuan P, Lesser V, Zilberstein S. Communication decisions in multi-agent cooperation: Model and experiments. In: *Proceedings of the fifth international conference on autonomous agents (AGENTS '01)*. New York (2001). p. 616–23.

frontiers | Frontiers in Physics

TYPE Mini Review
PUBLISHED 11 October 2022
DOI 10.3389/fphy.2022.880706

Check for updates

OPEN ACCESS

EDITED BY
Víctor M. Eguíluz,
Spanish National Research Council (CSIC), Spain

REVIEWED BY
Linjie Liu,
Northwest A&F University, China
Mendeli Henning Vainstein,
Federal University of Rio Grande do Sul, Brazil
Paolo Grigolini,
University of North Texas, United States

*CORRESPONDENCE
Fred Petry,
frednavy3@gmail.com

†These authors have contributed equally to this work and share first authorship

SPECIALTY SECTION
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

RECEIVED 21 February 2022
ACCEPTED 18 August 2022
PUBLISHED 11 October 2022

CITATION
Dennis S, Petry F and Sofge D (2022), Game theory approaches for autonomy. *Front. Phys.* 10:880706. doi: 10.3389/fphy.2022.880706

Game theory approaches for autonomy

Steven Dennis[1†], Fred Petry[1*†] and Donald Sofge[2]

[1]Naval Research Laboratory, Stennis Space Center, MS, United States, [2]Naval Research Laboratory, Washington, DC, United States

Game theory offers techniques for applying autonomy in the field. In this mini-review, we define autonomy, and briefly overview game theory with a focus on Nash and Stackleberg equilibria and Social dilemma. We provide a discussion of successful projects using game theory approaches applied to several autonomous systems.

KEYWORDS
game theory, robotics, autonomous vehicles, nash equilibrium, self-driving cars

1 Introduction: Autonomy and game theory

Autonomous systems are designed with tools to respond to situations that were not anticipated during design; e.g., decisions; self-directed behavior; human proxies [12]. Autonomous systems likely follow rules like their human counterparts (e.g., laws, commanders' intents, etc.). This short review paper is intended to illustrate how game theory can be effectively used in representative autonomous systems.

Game theory is the study of the ways in which interacting choices of agents produce outcomes with respect to the preferences (or utilities*)* of those agents, where the outcomes in question might have been intended by none of the agents [21].

A game represents situations in which at least one agent or player acts to maximize its utility through anticipating the responses to its actions by one or more other agents. The game provides a model of interactive situations among rational players. The key to game theory is that one player's payoff relies on the strategy used by the other player. The structure of a game includes players and their preferences, the strategies available, and outcomes of the strategies [32].

In the interaction of rational agents [3], non-cooperative game theory is an approach often utilized to obtain intended objectives. The strategic game is the most used non-cooperative game. For this game, only the strategies and outcomes available from a combination of choices incorporated.

The strategy of an agent specifies the procedure based on how a player chooses their actions. A solution concept is a well-specified set of rules used to predict how a game will develop. For example, a Nash equilibrium is a solution concept and when agents have no incentive to deviate from their selected actions, the game is in Nash equilibrium [23]. When agents or players opt for what they view as the most appropriate action to oppose their opponent's actions, it is termed a Nash equilibrium.

The strategic (or normal form) game is typically represented by a matrix which shows the players, strategies, and payoffs (Table 1). It can be represented by a function that

TABLE 1 Prisoners' game matrix.

Player 1	**Confess**	**Silent**
Player 2		
Confess	−1, −1	−5, 0
Silent	0, −5	−2, −2

associates a payoff for each player with every possible combination of actions. For example, for two players and a game matrix: one player chooses the row and the other chooses the column. As determined by the number of columns and rows, there are two strategies determined for each agent/player. The payoffs are provided in the intersections. The row/column intersections contain the payoffs as a pair of values. The first value is the payoff for a row player and the second is payoff for a column player. When each agent or player in the game performs simultaneous actions or is at least ignorant of another player's actions, the game is in normal form.

For example, in the prisoner's dilemma [18], each prisoner can either "confess" or be "silent". If exactly one prisoner confesses, their sentence is less and the other prisoner has a longer sentence. However, if they both confess, they both have shortened sentences. Hence we see that *confess* is strictly dominated by *silent*. This can be seen by comparing in Table 1, the first numbers in each column, in this case 0 > −1 and −2 > −5. This comparison shows that no matter what the column player chooses, the row player does better by choosing *silent*. Also when for every row the second payoff is examined, we see the same options, the values compared are the same: 0 > −1; −2 > −5. This shows that no matter what choice row player does, column is better by choosing *silent*. This demonstrates that the unique Nash equilibrium of this game is (*silent, silent*).

2 Robotic applications

Several projects have considered the utilization of game theory for applications in robotics. First we can examine a cooperative situation in which robots agree on strategies that may involve sacrifices by all to have a lower overall cost but still achieve their goals. However, every robot must take into account that the other robots are also trying to resolve their goals independently, which is termed a non-cooperative situation. An equilibrium solution occurs when, by taking account the possibilities of other robots performing operations in conflict with its goals, the robot selects its actions.

One example of this in the context of a complicated set of corridors is how to provide for autonomous coordination of two robots. The robots have independent goal locations and initial locations. The conflicts arise when robots need to occupy the same corridors at the same point in time while traversing their optimal paths. Game theory provides a solution suitable for both robots. However, the same choices for an individual robot may be less than optimal [19].

A multi-robot searching task can be modeled as a multiplayer cooperative nonzero-sum game. The robotic players choose their strategies simultaneously at the beginning of the game. Although the overall process of searching is dynamic, it can be treated as a sequence of static games at each discrete point in time. The players must resolve a non-zero sum static game for every discrete interval. This process follows if, with conditioned probability, that observations by the other team are available.

Specifically a game-theory based strategic searching approach has been developed for cooperation of a multi-robot system performing a searching task. To consider the interactions between robots, dynamic programming estimated the utility function, based on using the *a priori* probability map, travel costs, and the other robots' current state. Based on this utility function, a utility matrix was developed for an N-robot non-zero-sum game, where both pure Nash and mixed-strategy equilibria were applied to guide the robots to their decisions [22].

A distributed decision-making approach to the problem of control effort allocation to robotic team members in a warehouse has been designed [27]. In this approach, coordination of the robotic team in completing a task in an efficient manner was the objective. A controller design methodology was developed which allowed the robot team to work together based on game theoretic learning algorithms using fictitious play and extended Kalman filters. In particular, each robot of the team predicts the other robots' planning actions while making decisions to maximize its own expected reward that is dependent on the reward for joint completion of the task. The algorithm was successfully tested on collaborations for material handling and for patrolling robots in warehouses.

In [8], a game theory-based negotiation is utilized for allocating functions and tasks among multiple robots. After the initial task allocation, a new approach employing utility functions was developed to choose the negotiation robots and construct the negotiation set. All the robots have various tasks and the problem is assigning jobs to them minimizing costs and without conflicts. There are m robots and n tasks and x_{ij} indicates if the job J_i is allocated to robot R_j. Then the objective is to minimize overall cost

$$\underset{j=1}{\overset{m}{Min}}\left(\varphi_j\left(\sum_{i=1}^{n} w_{ij}x_{ij}\right)\right) \text{ where } \sum_{i=1}^{n} w_{ij}x_{ij} \le C_{j\cdot j-\cdot j}$$

where C_j is max cost allowed for R_j and w_{ij} is max cost for job J_i assigned to R_j. φ_j is a design objective function.

3 Autonomous/self-driving cars

There has been extensive use of game theory to control self-driving cars. For cars, decisions are constantly interacting between drivers and roadways, composing a game-theoretic problem. Accurately planning through road interactions is a safety-critical challenge in autonomous driving [13]. To deal with the mutual influence between autonomous vehicles and humans with computational feasibility, a game structure provided the authors a hierarchical framework. Their design accounted for the complex interactions with lane changing, road intersections and roundabout decisions.

The road decisions of autonomous vehicles interact with the decisions of other drivers/vehicles. Decisions include passing another car, road merging or accident avoidance. This mutual dependence, best captured by dynamic game theory, creates a strong coupling between the vehicle's planning and its predictions of other drivers' behavior. The basic approach in [13] considers one human driver, H, and one autonomous vehicle, A. The dynamics of their joint state is x^{t} and $\mathrm{x}^{\mathrm{t+1}}$ is the expression of its evolution

$$\mathrm{x}^{t+1} = \mathrm{f}\left(\mathrm{x}^{t}, \mathrm{u}_{A}^{t}, \mathrm{u}_{H}^{t}\right)$$

where u_{A}^{t}, u_{H}^{t} are the driving actions of the human and autonomous vehicles.

The system must maximize an objective that depends on the evolution of the vehicles over a finite time. The reward function R_{A} captures specifications of the vehicle's behavior such as fuel consumption, safety, etc. It is the cumulative return for t = 0:N,

$$\mathrm{Max}\ [\mathrm{R}_{\mathrm{A}}\ (\mathrm{x}^{0:N},\ \mathrm{u}_{A}^{0:N},\ \mathrm{u}_{H}^{0:N}) = \sum_{t=0}^{N} \mathrm{r}_{a}\ (\mathrm{x}^{t}, \mathrm{u}_{A}^{t}, \mathrm{u}_{H}^{t}\ )]$$

3.1 Autonomous vehicle lane changes

Another project considers a particular urban traffic scenario in which an autonomous vehicle needs to determine the level of cooperation of the vehicles in the adjacent lane in order to change a lane [26]. Smirnov's team developed a game theory-based decision-making model for lane changing in congested urban intersections. As input, driving parameters were related to vehicles in an intersection before a car stopped completely. For game players to enhance and protect their independent interests, strategies must consider mutual awareness of the situation and the predicted outcomes. The authors reported that non-cooperative dynamic games were the most effectively used for lane-changing

Differential games were used to design a fully automated lane-changing and car following control system [35]. Decisions computed the vehicles under control minimized costs for several undesirable/unexpected situations. Evaluations of the discrete and continuous control variables for lane-changes and accelerations were simulated. To provide optimal lane changing decisions and speed-ups, they used both cooperative and non-cooperative controllers.

A mandatory lane-changing decision-making model [2] was designed based on game theory for a two-player nonzero-sum non-cooperative game under incomplete information. Using the Harsanyi transformation [16], they transformed the model into a game that contained imperfect information to cover traditional and connected environments given complete and incomplete information inputs. They restructured the game with incomplete information to an imperfect information game

3.2 Intersection problems

A decision-making model based on a dynamic non-cooperative game was also used to investigate lane changing in an urban scenario of a congested intersection [29]. The game's results can be predicted if each vehicle maximizes its payoff in the interaction. For this approach the context proposed was management of traffic with red lights at two-lane road intersections.

In [15] the authors developed an approach to mimic human behavior. In their project, various styles of driving operation were assessed using utility functions involving safety of driving, comfort of riding and efficiency of total travel routing. They used non-cooperative games for Stackleberg and Nash equilibria [25]. They concluded that the algorithms developed performed the proper decisions under different driving situations. They also tested two scenarios to change lanes, i.e., merging and overtaking, to evaluate the feasibility and effectiveness of the proposed decision-making framework for different human behaviors. Their experimental evaluations showed that decision-making for autonomous vehicles similar to observed human behaviors can be achieved using both game theory approaches. In situations modeling ordinary styles of driving, the Stackleberg equilibrium game compared to Nash equilibria reduced the cost value by 20 percent.

A model of cooperative behavior strategy in conflict situations between autonomous vehicles in roundabouts has used game theory [4]. Roundabout intersections promote a more efficient and continuous flow of traffic. Roundabout entries move traffic through an intersection quickly and with less congestion for intersections. They can be managed more effectively using cooperative decisions by autonomous vehicles. This approach leads to shorter waiting times and more efficient traffic control while following all traffic regulations.

For roundabouts, well defined rules of the road dictate how autonomous vehicles should interact in traffic [17]. A game strategy based on the prisoner's dilemma has been used [4] for such roadways. The entry problem for roundabouts has been solved using non-zero sum games to yield shorter waiting intervals for each individual car.

3.3 Open road autonomy

For autonomous vehicles in uncomplicated environments with few interactions, mapping and planning is well developed [28]. However, the unresolved problems are the complexities of human interactions on the open road. Still, this approach presents a model for negotiation between an autonomous vehicle and another vehicle at an unsigned intersection or (equivalently) with a pedestrian at an unsigned road-crossing (i.e., jaywalking), using discrete sequential game [14]. In this model if only car location indicates intent, a non-zero collision probability provides optimal behavior for both vehicles. They also concluded that to reduce probabilities of collisions, alternative forms of control and signal usage should be considered for autonomous vehicles.

4 Aerial and underwater autonomous vehicles

A game theoretic real time planning approach for an autonomous vehicle (e.g., an aerial drone) for competitive races against several opponents over a race course while accounting for opponents' decisions has been developed [34]. It uses an iterative best-response scheme with a sensitivity term to find approximate Nash equilibria in the space of multiple robot trajectories. The sensitivity term develops Nash equilibria that provide an advantage to individual robots. Through extensive multi-player racing simulations, where the planner exhibits rich behaviors like blocking, overtaking, nudging or threatening, it demonstrated behaviors similar to human racers.

Modeling the interactions of agents that are risk sensitive is important to allow more real-world and efficient agent behavior. During interactions, the extent to which agents exhibit risky maneuvers is not solely determined by their risk tolerance; it also depends on the risk-sensitivity of their opponents. Agent interactions involving risk were modeled in a game-theoretical framework [20]. By being aware of the underlying risks during interactions, this approach leads to safer behaviors by being at a farther distance from other agents [33]. Anticipating feedback in game-theoretic interactions leverages other agent's risk-awareness to plan for safe and time-efficient trajectories.

An important related issue that can arise is that use of "best responses" may have a potential downside. That is in some situations, cooperation can involve possibly violating certain ethical rules [10] and has engendered discussions recently about self-driving cars and autonomous weapons. This can also be considered from the point of view whether user stress has an effect. It has been shown that even with the increased cognitive load such as during stressful situations, individuals are generally honest [24] and this aspect can be significant in game theory modeling of human and autonomous systems interactions. It is important that such issues be considered in system designs as there will be ever more various autonomous systems and their human interactions involved in the future.

4.1 Applications of unmanned aerial vehicles

Autonomous unmanned aerial vehicles can be tasked to search an unknown/uncertain environment, and neutralize targets perceived as threats. This problem can be formulated by issues that a uav faces when it detects multiple such targets and needs to decide which target to neutralize, given the uncertainty over the decisions of its opponents [5], in a game theoretic framework. Bardhan and colleagues use a correlated equilibrium concept based decentralized game theoretic solution that requires local information of the uavs.

Another game was designed [31] for a swarm (group) of autonomous uavs, where each uav is tasked with collecting information from an area of interest. In this setting, a mission needs to maximize the amount of information collected by uavs. This is formulated by dividing the region of interest into discrete cells, each having potential information value. Each selfish uav (i.e., player) makes the simplest decision for itself by selecting a path among available choices (i.e., strategies) it will fly**.** So each player or uav behaves selfishly by choosing the best choice of available paths. Game payoffs are determined using information fusion for aggregating information from the multiple uavs operating at multiple locations. Efficiency of a mission is the ratio of an optimal output to a pure strategy Nash equilibrium for the corresponding game.

Stackelberg games can obtain flight routes for uavs operating in areas with malicious opponents using gps spoofing attacks to divert uavs from their chosen flight paths [11]. In a Stackelberg game between a uav acting as the game leader and a gps spoofer, the leader chooses a group of uavs to protect, after which the spoofer opponent determines its actions by observing the choice of the leader. Strategies during this game reflect abilities of each uav group to estimate its location using positions of its nearby uavs, allowing it to succeed to gain a destination despite ongoing gps spoofing attacks.

4.2 Autonomous underwater vehicles

Autonomous underwater vehicles multi-vehicle coordination and cooperation has been formulated with game theory. Very simple games have been used [7] to stably steer an auv formation in its position underwater that is the best compromise between target destination of each vehicle and preservation of communication capabilities among all of the vehicles due to limits on underwater communications.

A specific type of security game is a Stackelberg Security Game [30]. A key concept in this type of security game is a leader-follower framework for strategies for underwater auv patrols. In the real world, it can be assumed that any security pattern can be exploited by attackers beforehand through reconnaissance. Thus, security patrols must have a certain degree of randomness while maintaining their efficiency. The leader will commit to an optimal policy and the follower will find an optimal policy after observing the leader's actions. The leader's policy, x, a probability distribution over the leader's pure strategies where x_i is the percentage of times strategy i was used in the policy. Then q and q_j are the follower's optimal policy and strategy j's percentage in response to the leader's strategy. R_{ij} and C_{ij} are the reward matrices of the leader and follower respectively when the leader commits to strategy i, and the follower strategy j [9]. The leader will then solve the following Mixed Integer Quadratic Problem:

$$\underset{x,q,a}{Max}\left[\sum_{i\in X}\left(\sum_{j\in Q} C_{ij}X_i q_j\right)\right]$$

$$0\le\left(a-\sum_{i\in X} C_{ij}\, x_i\right)\le\left(1-q_j\right)M\,\forall j\in Q$$

X and Q are index sets of leader's and follower's strategies, M is a large positive number and a $\in A$ is the follower's maximum reward.

5 Conclusion

We have provided a mini-review illustrative but not exhaustive of successful autonomy applications of game theory based on Nash, Stackleberg and social dilemmas. There are recent research developments in game theory that can enhance such applications. Mean-field (MF) game theory [6] is a model created to deal with an environment where several participants interact smoothly. Standard game theories are used to deal with how two participants interact with each other. MF however describes one participant deals with a group of others. Due to the complexity of interactions between participants, the original theory was nonapplicable to large groups but using mean-field game theory, situations involving large groups can be solved quickly and easily. Another new approach is evolutionary game theory which focuses on evolutionary dynamics that are frequency dependent [1, 36]. The fitness payoff for a particular phenotype depends on population composition. Classical game theory focuses largely on the properties of the equilibria of games. A central feature of EGT is a focus on dynamics of strategies and their composition in a population rather than on properties of equilibria.

Author contributions

All authors listed have made a substantial, direct, and intellectual contribution to the work and approved it for publication.

Acknowledgments

The authors would like to thank the Naval Research Laboratory's Base Program for sponsoring this research.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

1. Alexander J. The stanford encyclopedia of philosophy. In: EN Zalta, editor. *Evolutionary game theory* (2021). (Summer 2021 Edition). Available at: https://plato.stanford.edu/archives/sum2021/entries/game-evolutionary/.

2. Ali Y, Zheng Z, Haque MM, Wang M. A game theory-based approach for modelling mandatory lane-changing behaviour in a connected environment: A game theory approach. *Transportation Res C: Emerging Tech* (2019) 106:220–42. doi:10.1016/j.trc.2019.07.011

3. Amadae SM. Rational choice theory. Encyclopedia Britannica (2021). Available at: https//www.britannica.com/topic/rational-choice-theory/.

4. Banjanovic-Mehmedovic L, Halilovic E, Bosankic I, Kantardzic M, Kasapovic S. Autonomous vehicle-to-vehicle (V2V) decision making in roundabout using game theory. *Int J Adv Comput Sci Appl* (2016) 7:292–8. doi:10.14569/ijacsa.2016.070840

5. Bardhan R, Bera T, Sundaram S. A decentralized game theoretic approach for team formation and task assignment by autonomous unmanned aerial vehicles. In: *International conference on unmanned aircraft systems (ICUAS)* (2017). p. 432–7.

6. Cardaliaguet P, Porretta A. An introduction to mean field game theory. In: P Cardaliaguet A Porretta, editors. *Mean field games. Lecture notes in mathematics*. Springer Pub (2020). p. 2281.

7. Cococcioni M, Fiaschi L, Lermusiaux PFJ. Game theory for unmanned vehicles path planning in the marine domain: State of the art and new possibilities. *J Mar Sci Eng* (2021).

8. Cui R, Guo J, Gao B. Game theory-based negotiation for multiple robots task allocation. *Robotica* (2013) 31:923–34. doi:10.1017/s0263574713000192

9. Dennis S, Petry F, Sofge D. Game theory framework for agent decision-making in communication constrained environments.*Unmanned Syst Tech XXIII* (2021). 117580P. doi:10.1117/12.2585922

10. Du Y, Ma W, Sun Q, Sai L. Collaborative settings increase dishonesty. *Front Psychol* (2021) 12:650032. retrieved 4/11/2022 from. doi:10.3389/fpsyg.2021.650032

11. Eldosouky AR, Ferdowsi A, Saad W. Drones in distress: A game-theoretic countermeasure for protecting UAVs against GPS spoofing. *IEEE Internet Things J* (2019) 7:2840–54. doi:10.1109/jiot.2019.2963337

12. Endsley MR. *Human-AI teaming: State-of-the-Art and research needs. The national academies of sciences-engineering-medicine*. Washington, DC: National Academies Press (2021). Retrieved 12/27/2021 from. Available at: https://www.nap.edu/catalog/26355/human-ai-teaming-state-of-the-art-and-research-needs.

13. Fisac J, Bronstein E, Stefansson E, Sadigh D, Sastry S, Dragan A. Hierarchical game-theoretic planning for autonomous vehicles. In: *2019 international conference on robotics and automation, ICRA* (2019). p. 9590–6.

14. Fox CW, Camara F, Markkula G, Romano RA, Madigan R, Merat N. When should the chicken cross the road? Game theory for autonomous vehicle-human interactions. In: *Proceedings of the 4th international conference on vehicle technology and intelligent transport systems* (2018). p. 429–31.

15. Hang P, Lv C, Xing Y, Huang C, Hu Z. Human-like decision making for autonomous driving: A noncooperative game theoretic approach. *IEEE Trans Intell Transp Syst* (2020) 22:2076–87. doi:10.1109/tits.2020.3036984

16. Hu H, Stuart H. An epistemic analysis of the Harsanyi transformation. *Int J Game Theor* (2002) 30:517–25. doi:10.1007/s001820200095

17. Isebrands H, Hallmark S. Statistical analysis and development of crash prediction model for roundabouts on high-speed rural roadways. *Transportation Res Rec* (2012) 2389:3–13. doi:10.3141/2312-01

18. Kuhn S. *The stanford Encyclopedia of philosophy* (winter 2019 edition). In: E Zalta, editor. *Prisoner's dilemma* (2019). Available at: https://plato.stanford.edu/archives/win2019/entries/prisoner-dilemma/.

19. LaValle S, Hutchinson S. Game theory as a unifying structure for a variety of robot tasks. In: *Proc 1993 int symp on intelligent control* (1993). p. 429–34.

20. Lawless WF, Sofge DA. *Risk determination versus risk perception: From misperceived drone attacks, hate speech and military nuclear wastes to human-machine autonomy*. Stanford University (2022). Presented at AAAI-Spring 2022. Available at: https://www.aaai.org/Symposia/Spring/sss22symposia.php#ss09.

21. Maschler M, Solan E, Zamir S. *Game theory*. Cambridge University Press (2013).

22. Meng Y. Multi-robot searching using game-theory based approach. *Int J Adv Robotic Syst* (2008) 5(4):44–350. doi:10.5772/6232

23. Nash J. Non-cooperative games. *Ann Math* (1951) 54(2):286–95. doi:10.2307/1969529

24. Reis M, Pfister R, Foerster A. Cognitive load promotes honesty. *Psychol Res* (2022). doi:10.1007/s00426-022-01686-8

25. Simaan M, Cruz J. On the Stackelberg strategy in nonzero-sum games. *J Optim Theor Appl* (1973) 11(5):533–55. doi:10.1007/bf00935665

26. Smirnov N, Liu Y, Validi A, Morales-Alvarez W, Olaverri-Monreal C. A game theory-based approach for modeling autonomous vehicle behavior in congested, urban lane-changing scenarios. *Sensors* (2021) 21:1523–32. doi:10.3390/s21041523

27. Smyrnakis M, Veres S. Coordination of control in robot teams using game-theoretic learning. *IFAC Proc Volumes* (2007) 47(3):1194–202. doi:10.3182/20140824-6-za-1003.02504

28. Taeihagh A, Lim H. Governing autonomous vehicles: Emerging responses for safety, liability, privacy, cybersecurity, and industry risks. *Transport Rev* (2019) 39(1):103–28. doi:10.1080/01441647.2018.1494640

29. Talebpour A, Mahmassani HS, Hamdar SH. Modeling lane-changing behavior in a connected environment: A game theory approach. *Transportation Res Proced* (2015) 7:420–40. doi:10.1016/j.trpro.2015.06.022

30. Tambe M. *Security and game theory: Algorithms, deployed systems, lessons learned*. Cambridge University Press (2012).

31. Thakoor O, Garg J, Nagi R. Multiagent UAV routing: A game theory analysis with tight price of anarchy bounds. *IEEE Trans Autom Sci Eng* (2019) 17:100–16. doi:10.1109/tase.2019.2902360

32. von Stengel B. *Game theory basics*. Cambridge University Press (2022).

33. Wang M, Mehr N, Gaidon A, Schwager M. Game theoretic planning for risk-aware interactive agents. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems Las Vegas* (2020). p. 6998–7005.

34. Wang Z, Spica R, Schwager M. Game theoretic motion planning for multi-robot racing. *Distributed Autonomous Robotic Syst* (2020) 9:225–38. doi:10.1007/978-3-030-05816-6_16

35. Wang M, Hoogendoorn SP, Daamen W, van Arem B, Happee R. Game theoretic approach for predictive lane-changing and car-following control. *Transportation Res Part C: Emerging Tech* (2015) 58:73–92. doi:10.1016/j.trc.2015.07.009

36. Weibull J. *Evolutionary game theory*. Cambridge MA: MIT Presss (1997).

frontiers | Frontiers in Physics

TYPE Original Research
PUBLISHED 12 October 2022
DOI 10.3389/fphy.2022.944086

OPEN ACCESS

EDITED BY
William Frere Lawless,
Paine College, United States

REVIEWED BY
Eugene Santos,
Dartmouth College, United States
Jean-Sébastien Sottet,
Luxembourg Institute of Science and Technology (LIST), Luxembourg

*CORRESPONDENCE
Nicolas Hili,
nicolas.hili@univ-grenoble-alpes.fr

SPECIALTY SECTION
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

RECEIVED 14 May 2022
ACCEPTED 22 September 2022
PUBLISHED 12 October 2022

CITATION
Castellanos-Paez S, Hili N, Albore A and Pérez-Sanagustín M (2022), BOARD-AI: A goal-aware modeling interface for systems engineering, combining machine learning and plan recognition. *Front. Phys.* 10:944086.
doi: 10.3389/fphy.2022.944086

BOARD-AI: A goal-aware modeling interface for systems engineering, combining machine learning and plan recognition

Sandra Castellanos-Paez[1], Nicolas Hili[1]*, Alexandre Albore[2] and Mar Pérez-Sanagustín[3]

[1]Université Grenoble Alpes, CNRS, LIG, Grenoble, France, [2]Onera DTIS, Toulouse, France, [3]IRIT, Université de Toulouse, CNRS, INP, UT3, Toulouse, France

Paper and pens remain the most commonly used tools by systems engineers to capture system models. They improve productivity and foster collaboration and creativity as the users do not need to conform to formal notations commonly present in Computer-Aided Systems Engineering (CASE) tools for system modeling. However, digitizing models sketched on a whiteboard into CASE tools remains a difficult and error-prone activity that requires the knowledge of tool experts. Over the past decade, switching from symbolic reasoning to machine learning has been the natural choice in many domains to improve the performance of software applications. The field of natural sketching and online recognition is no exception to the rule and most of the existing sketch recognizers rely on pre-trained sets of symbols to increase the confidence in the outcome of the recognizers. However, that performance improvement comes at the cost of trust. The lack of trust directly stems from the lack of explainability of the outcomes of the neural networks, which hinders its acceptance by systems engineering teams. A solution shall not only combine the performance and robustness but shall also earn unreserved support and trust from human users. While most of the works in the literature tip the scale in favor of performance, there is a need to better include studies on human perception into the equation to restore balance. This study presents an approach and a Human-machine interface for natural sketching that allows engineers to capture system models using interactive whiteboards. The approach combines techniques from symbolic AI and machine learning to improve performance while not compromising explainability. The key concept of the approach is to use a trained neural network to separate, upstream from the global recognition process, handwritten text from geometrical symbols, and to use the suitable technique (OCR or automated planning) to recognize text and symbols individually. Key advantages of the approach are that it does not resort to any other interaction modalities (e.g., virtual keyboards) to annotate model elements with textual properties and that the explainability of the outcomes of the modeling assistant is preserved. A user experiment validates the usability of the interface.

KEYWORDS

machine learning, plan recognition, artificial intelligence, systems engineering, sketch recognition, model-driven development

1 Introduction

Despite its proven modeling value in many engineering domains, Computer-Aided Systems Engineering (CASE) tools have only had moderate acceptance by system engineers and architects to assist them in their day-to-day tasks [1]. The complexity of creating, editing, and annotating models of system engineering takes its root from different sources: unsuitable representations, outdated interfaces, laborious modifications, and difficult collaborations [2].

As a result, especially in early development phases, systems architects tend to favor more traditional tools, such as whiteboards, paper, and pencils over CASE tools to quickly and easily sketch a problem and its solution. Among the benefits of sticking to traditional tools, whiteboards foster collaboration and creativity as the users do not need to strictly conform to formal notations.

A common pitfall for using traditional tools, however, is that human users are required to reproduce any sketched solutions inside of formal tools when it comes to formalizing the models. Modern post-WIMP[1] interfaces (e.g., electronic whiteboards) could help to automate this task by allowing users working on a digital representation of the model, one that can be directly exported, to be modified *via* modeling tools. Bridging the informality of the working sketches captured on interactive whiteboards with formal notations and representations has the potential to lower the barrier of acceptance of CASE tools by industry [3, 4]. This acceptance can be obtained by automatically or semi-automatically translating informal sketches into their corresponding formal elements using a specific and conventional notation.

Natural sketch recognition aims at bridging the gap between free-form modeling and formal representations using dedicated graphical notations. A significant body of related work can be found in the literature, spanning offline and online recognition [5–13]. Offline recognition allows users to capture sketches using pens and paper and to further digitizes them, while online recognition relies on interactive digital displays such as electronic whiteboards for user inputs [6]. With advances in modern post-WIMP interfaces, recent pieces of work tend to favor online sketch recognition systems over offline ones. Yet, providing a robust online sketch recognition is still a hot research topic and is not well-settled in systems engineering.

1.1 An early model recognition assistant

In our previous work [14, 15], we suggest the use of symbolic Artificial Intelligence (AI) techniques to aid systems engineers to design models in a freehand way using large multi-touch screens. The heart of this approach lies in the use of goal recognition techniques to translate user's sketches into model elements. More precisely, a modeling assistant identifies the most probable model elements intended to be drawn by a user from an initial sketch, even when partial. The outcome of the assistant is a list of suggestions ordered by the probability that a complete model element corresponds to the user's intent. This probability is based on the "distance" between the partial sketch and any possible model elements that can be drawn from that partial sketch measured in terms of the number of steps that would remain to finish drawing the model element completely.

The main benefit of relying on symbolic AI rather than on Machine Learning (ML) is explainability. "Explainability" is the property of a system that provides an output that makes understandable to the human user the reasons of an algorithm's choice. This is a condition needed by any process-directed tool that allows users to evaluate the criteria behind a choice to use the tool more efficiently [16]. Not only the modeling assistant provides the user with a list of suggestions, but it also details the remaining steps to draw the suggested model elements completely. Our preliminary evaluation suggests that recognizing complex shapes (e.g., an operational actor made of four straight lines and one circle) using AI methods is suitable for online incremental recognition. In the present study, we make the following contributions:

1) We refine a part of the approach that no longer relies on goal recognition alone, but rather on the combination of symbolic AI and ML techniques. Handwritten text and geometrical shapes composing model sketches require two distinct recognition processing and, thus, must be decoupled prior to the recognition process to occur. We train a Neural Network (NN) to distinguish text from geometrical shapes such that handwritten text can be recognized using traditional Optical Character Recognition (OCR) engines while geometrical shapes are determined by our initial goal recognition algorithm to identify the model elements. This approach allows us to enhance our recognition process to identify model elements annotated with text without resorting to virtual keyboards or voice recognition. We present the extended approach, describe a training platform we developed to train the NN, and summarize the results of the training process.

[1] Windows, mouse, and pointer interfaces.

2) We reformulate the representation of the sketching environment used by our shape recognition engine in order to improve the speed and accuracy of the goal recognition process, and to make it more tolerant to drawing imprecision. This new representation, expressed in the PDDL language proper to automated planners, is lighter than the one previously used, hence speeding up the recognition process. On the one hand, the simplification of the planning representation comes at the cost of more ambiguity when recognizing model elements from (very) partial drawings. On the other hand, it led to better results in real case scenarios, as completing a sketch adds new constraints, thus removing most of the ambiguity. In addition to these improvements, we replace the previous search algorithm by the anytime algorithm used in LAMA [17]. It results in a generation of plans which is faster, and that in time provides plans with increasing quality.
3) The refined approach and enhanced recognition algorithm resulted in a modeling environment called BOARD-AI. It consists in an electronic whiteboard interface coupled with both shape and text recognition software, and an automated planning algorithm that provides completion suggestions for the sketch drawn by the users.
4) Finally, this study presents an early evaluation of the human-machine interactions and of the usability of BOARD-AI on two groups of users. These users employed the modeling environment to design a system engineering system before answering to a questionnaire that we then evaluated. This study provides a first assessment of the validity of our approach, and of the trust that users are willing to place in an artificial intelligence-based recommendation system. This evaluation protocol eventually provided us with useful indications on future improvements of the modeling framework.

1.2 Paper structure

The rest of the study is structured as follows: Section 2 presents background concepts; Section 3 presents the approach and describes the implementation of BOARD-AI; Section 4 describes the user evaluation of BOARD-AI and the conclusions we drew; Section 5 presents related work; and Section 6 concludes.

2 Background

The term sketch has a very broad definition. It includes a variety of freehand drawings made by amateurs or professionals, such as doodles, clip-arts, caricatures [13], but also drawings used in various engineering domains, e.g., electrical circuits. Natural sketching is becoming more and more popular due to the increased use of post-WIMP interfaces, including interactive whiteboards, interactive walls, large multi-touch screens, interactive pen displays, and personal tablets [9]. The emergence of AI-enabled sketching tools such as Google Autodraw also largely contributed to the digital revolution of natural sketching from a very broad range of applications and use cases.

The present study targets sketches used to capture models of systems in a more natural way using traditional software and systems engineering languages. Sketching tools offer an alternative approach to traditional CASE tools to rapidly design a model where users have more freedom and flexibility as they are not required to learn how to use complex tools to create the models desired [7]. However, sketching tools dedicated to modeling differ from more general-purpose sketching tools on two fundamental aspects.

First, sketches vary in terms of representation, size, and style, and general-purpose sketching tools have to handle a large number of individual sketches. As such, recent work tends to favor classification techniques and complex deep learning models to handle such variation [13], for it relies on large sets of sketches to train the recognizers. As opposed, diagrams in computer science and systems engineering rely on relatively stable and simple representations composed of simple geometrical shapes (rectangles, circles, *etc.*).

Second, text is omnipresent in traditional diagrams in computer science or systems engineering to label model elements. However, text and shapes do not rely on the same training models to be efficiently recognized. The duality of recognizing text and symbols individually has been explored in the literature, e.g., in [11] and takes its source from traditional text/non-text separation techniques in offline document analysis [18]. While some work moved the problem aside by providing users with alternative text editing capabilities (e.g. [19]), others (e.g. [11, 20], provide seamless modeling capabilities, for it relies on segmentation techniques to separate text and non-text before starting to recognize shapes and text individually. In our previous work [14, 15], we relied on alternative editing capabilities, including virtual keyboard and voice recognition to annotate model elements with text. Our present work adheres to this second approach as it seems more natural for users to only rely on a single modality to draw models.

2.1 Classification

One trend to recognize modeling elements is to rely on AI tools and algorithms, more specifically, on ML techniques based on NNs. This family of approaches typically involves two phases [21]. During the training phase, algorithms are trained to recognize elements based on pre-existing libraries. During the recognition phase, these algorithms can identify elements with a certain degree of confidence.

NNs can learn and generalize from training data; they are particularly fault and noise tolerant [22]. Thus, they are often used in several domains for the classification of input data into categories [23–25]. An essential element of NNs is the neuron. A neuron is an information processing unit taking several inputs and producing one output [26]. Each input has its own weight; the neuron calculates the sum of the weighted inputs plus a bias term (this term represents how easily the neuron fires). Afterwards, this sum is passed trough an activation function to obtain the output. The role of an activation function is to introduce non-linearity into the output of a neuron. It also determines whether and how much a neuron should be activated, and its choice may improve or reduce the neuron's performance. Neurons connect to each other to form NNs; a neuron's output then provides the input to another neuron.

Most NNs are organized into multiple layers, and each layer has a specific number of neurons. We can distinguish three types of layers [27]: the input layer that provides data from the world to the network (in our case, the extracted features from the user's drawing); hidden layers that compute and transfer information from that input layer to the output layer; and the output layer that corresponds to the output prediction of the network. Multiple hidden layers can be stacked along each other. A network is called fully connected if every node in each layer is connected to every adjacent node in the adjacent forward layer.

NNs are widely used to classify data when a labeled dataset to train it is provided (supervised learning). A NN classifier tries to approximate a function that maps all of the elements in a space (the elements to classify) to the elements in another space (the categories of these elements). The network, by adjusting weights and biases, tries to approximate this function as best as possible, and then maps the elements to be classified to their respective categories.

In this work, we implement these concepts to develop an NN classifier able to find a function that maps collected data from user's drawings to one of two categories, either text or geometrical shape. To do so, we represent the training data as a label dataset consisting of a set of features (e.g. the number of sharp corners, a bounding box ratio, etc. See Section 3.3) and a target (the corresponding category, i.e., text or geometrical shape).

2.2 Automated deterministic planning

AI planning [28] has been used to perform activity recognition in the context of a system managed by human operators whose currently pursued operational goal has yet to be determined [29]. Several goal recognition [30] fields of application have surfaced, including "operator modeling" to improve the efficiency of man-machine systems. Early applications of the approach failed because of the complexity of plans, the issues due to evaluating actions that did not fit any plan, or the issues from interleaving planning and execution. Moreover, the work in goal recognition has historically proceeded independently from the planning community, using handcrafted libraries rather than planners [31].

An automated planning task can be represented as a directed graph model, where the nodes correspond to the different situations (or states) in which a system can be, and the edges represent actions that drive the system from one situation to a new one. Solving a planning problem consists in finding a sequence of actions $\langle a_1, \ldots, a_n \rangle$, also called plan π, that drive the system from an initial state to a desired goal, or a final situation. The length $|\pi|$ corresponds to the number of actions in the plan π: the length of a plan is commonly considered as a preference criterion to evaluate it.

To achieve automated deterministic planning, we adopt the STRIPS formalism [32]. In STRIPS, a factored representation represents states *via* a set of Boolean variables, interpreted as a conjunction, and such that each state s is a complete assignment of state variables. A planning problem is then defined as a 4–tuple $\langle \mathcal{F}, \mathcal{A}, \mathcal{I}, \mathcal{G} \rangle$, where $\mathcal{F}$ is the set of state variables (assuming Boolean values), $\mathcal{A}$ is the set of operators (or actions), and $\mathcal{I}, \mathcal{G} \subseteq \mathcal{F}^2$ are two sets of variables describing the initial state and the goal state(s), respectively. An action $a \in \mathcal{A}$ is defined as the 3–tuple $\langle$ pre(a), add(a), del(a) $\rangle$, where pre(a) is the set of preconditions of a, add(a) and del(a) are the sets of post-conditions of a, respectively defining the set of propositions added and deleted from the state. The pre-conditions determine in which state an action can be applied, while post-conditions specify the changes to variable assignments made by applying the action in a state. In other words, an action a is applicable in state s iff pre(a) $\subseteq s$, where the application of a in s is defined by the transition function $T(s, a) = (s/\text{del}(a)) \cup \text{add}(a)$.

In order to solve these planning problems, we adopt in this study an anytime approach. The planning algorithm first runs a search in the graph, aimed at finding a solution as quickly as possible. Once a plan is found, it searches for progressively better solutions by running a series of more expensive searches (in terms of computation time). The cost of the best known solution is used for pruning the subsequent searches. The final result is a set of solution, obtained at increasing time intervals but with (hopefully) a better solution quality.

Anytime algorithms are usually based on Weighted A*: an heuristic search algorithm that uses a weight to scale the heuristic value of each node of the graph about to be visited [33]. The underlying idea is to continue the search after the first obtained solution, possibly adjusting search parameters like the weight or pruning bound, and thus progressively find better solutions [34–36].

In our previous work [15], we describe an approach based on AI automated planning to recognize complex sketches

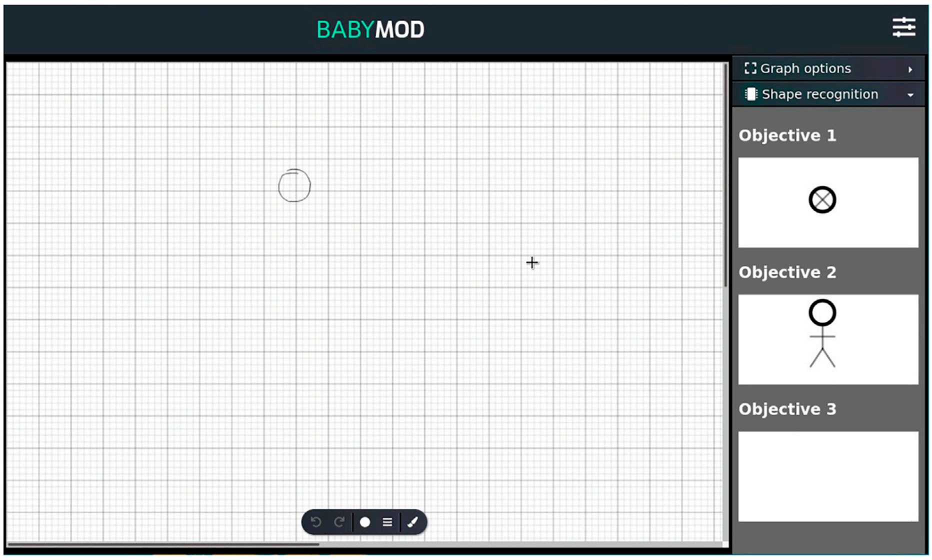


FIGURE 1
A preliminary implementation of the modeling assistant [15]. The central area is an HTML5 Canvas where the user can sketch model elements. The right sidebar shows suggestions to complete the sketch.

representing model elements and to guide users in their completion. For our preliminary sketching tool (see [15]), we adopted a strategy that uses the planning framework for goal recognition to perform the converse task of automated planning [37], i.e., recognizing the most probable goal given an initial state and a plan. Here, the task is to identify the shapes yet to be drawn and their placement to create a meaningful system-engineering sketch from an incomplete drawing. We use a goal library to describe the possible solutions of a plan. Sketching is therefore represented as a planning problem, where the initial state corresponds to a partial drawing from the user and the goals represent the different model elements the planner is able to recognize. The actions represent the different operations a user or a hypothetical drawing-agent would perform to complete an initial sketch.

3 Materials and methods

Figure 1 is an overview of a preliminary implementation of the modeling assistant. As soon as the user starts to draw some shapes on the screen, the modeling assistant is able to propose suggestions in terms of systems engineering sketched elements. Suggestions are ordered based on the length of the plan π calculated by the planer. Explainability is a central concern in our implementation: suggestions of the final shape form—calculated by the modeling assistant—is in direct correlation with how much of the complete model element sketch is left to draw.

One limitation of our initial implementation is that it does not support handwritten text annotating model elements. Automated planning is usually less fault-and-noise tolerant than other techniques such as NNs. It is therefore not suitable for text recognition, traditionally done through OCR. For our preliminary sketching tool, we then relied on two other interaction modalities. A user can annotate a model element through a virtual keyboard, or through voice recognition. This implies that the model element is first recognized before it can be annotated.

In this study, we extend our initial approach with handwritten text annotation support. The key concept is to use a trained neural network upstream from the global recognition process to separate handwritten text annotation from geometrical shapes, and to use the suitable technique (OCR or automated planning) to recognize text and shapes individually.

3.1 Approach definition

An overview of the extended approach is illustrated in Figure 2. The sketching work starts with a user's drawing. That drawing can be partial (i.e., it only represents a part of an element to be recognized) or complete (it completely represents the element to recognize). Compared to our previous approach, the drawing can be annotated with handwritten text. We assume in our approach that text and geometrical shapes belong to two different classes of problems

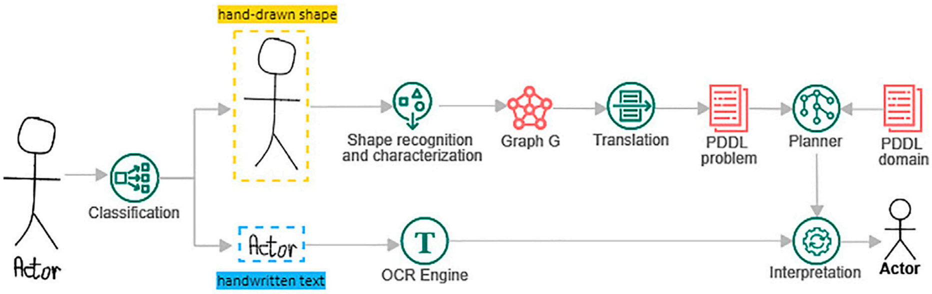


FIGURE 2
Overview of the recognition process.

requiring different techniques. Therefore, our process splits into two sub-processes to recognize text and geometrical shapes, respectively. Text recognition is performed *via* classical components-off-the-shelf OCR algorithms, while geometrical shape recognition is handled by our goal recognition algorithm described in [15]. Our approach then reconciles the outputs of both sub-processes once they terminate to reconstruct the global model element annotated with textual properties.

3.1.1 Classification

We developed a data efficient neural network classifier to distinguish the different elements composing the user's drawing into two categories: text and geometrical shapes. To address the lack of data, we opted for a feature-based neural network instead of an image-based NN since the former are lightweight and easy to train. First, we analyzed several inputs to identify features of detected geometrical shapes and text to be used in classifying a user's drawing. We then proposed a set of features to be computed from the data acquired using a training interface (see Section 3.3.1), and to use them in the classification. The list of the features is detailed in Section 3.3.

We then provided the NN classifier with a training data set of the appropriate network behavior in the form of labeled data. The training dataset consisted of a set of features (the extracted features from each category of the user's drawing) and a target (the corresponding category, i.e., text or geometrical shape). We used supervised learning where the network, provided with inputs, adjusted its weights and biases to move its outputs as close as possible to the targets.

3.1.2 Shape recognition and characterization

We use goal recognition to identify final geometrical shapes. This involves two steps. The first step consists in recognizing and characterizing primitive shapes with a simple shape detection algorithm. As described in [15], the approach only focuses on two primitive shapes: ellipses and straight lines. A polyline consists of a series of connected straight line segments. When a polyline is recognized, it is not characterized as a whole, but instead, each segment composing it is characterized individually and independently of the others. Circles are also recognized as a specific case of ellipses where the two foci are on the same spot (the center).

The rationale behind the algorithm's minimalist recognition strategy is twofold: first, this strategy reduces the time required to perform this step, hence, speeding up the complete recognition process. Second, most of the graphical elements employed in standard modeling languages and used by systems engineers can be simply expressed in terms of the two primitive shapes. This strategy considerably reduces the complexity of our goal recognition algorithm as its does not have to deal with multiple alternative ways of drawing the same model element. It allows the algorithm to deal with different drawing habits the same way. For example, a user can draw a rectangle in a single-line drawing (without lifting the pen from the surface of the screen) to represent the outer frame of a UML class, while a second user can draw four straight lines connected to their end-points.

Mathematically speaking, we consider the set S = (*ellipse*, *straight_line*) as the set of primitive shapes recognized by the algorithm. S is a minimal functionally complete set (by analogy with mathematical logic) as all other geometrical shapes can be expressed in terms of the two constituents of S.

Once primitive shapes are recognized, specific characteristics are extracted from them. Ellipses are tagged as being circles or not. A straight line is characterized by its four possible orientations O such that O = (*horizontal*, *vertical*, *diagonal_left*, *diagonal_right*). To compute straight line orientations, the raw angle between the two end points of a line is 'smoothed' to its closest remarkable $\frac{i\pi}{4}$-angle. Finer-grain fractions can be chosen for smoothing angles, but they would be less tolerant to drawing imperfections. For example, to sketch an operational actor, left and right legs could be characterized as 45° or 60° straight lines depending on the user's talent for drawing.

After the recognition process, we compute the position of every primitive shape relatively to their connection the other shapes composing the same element. We distinguish if a connection intersects a shape in the middle or at its end-

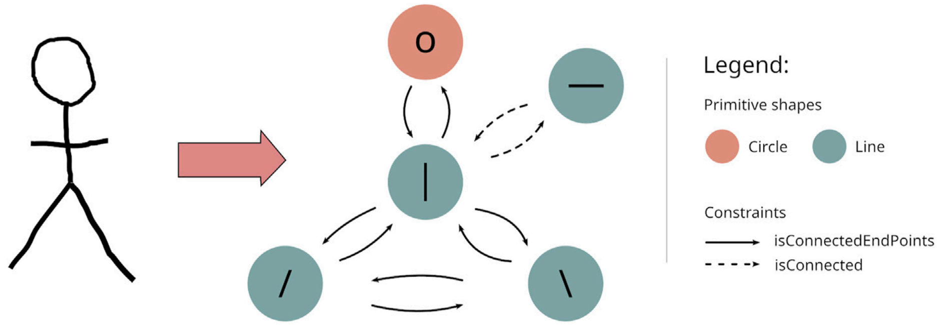


FIGURE 3
An actor being drawn (left side) and the resulting graph *G* (right side). Each edge is a relation between two primitive shapes connected at their end points or intersecting. The orange node represents the head of the actor. Green nodes are straight lines whose orientations are indicated by the node labels.

points. The set of possible intersections O is such that $O = (isConnectedEndPoints, isConnected)$.

The relation *isConnected* is bijective. It occurs when two primitive shapes are intersecting at the center:

$$\forall s_1, s_2 \mid s_2 \in isConnected(s_1) \Leftrightarrow s_1 \in isConnected(s_2) \tag{1}$$

The relation *isConnectedEndPoint* implies that the two elements are connected:

$$\forall s_1, s_2 \mid s_2 \in isConnectedEndPoint(s_1) \Rightarrow s_1 \in isConnected(s_2) \tag{2}$$

and is bijective as well:

$$\forall s_1, s_2 \mid s_2 \in isConnectedEndPoint(s_1) \Leftrightarrow s_1 \in isConnectedEndPoint(s_2) \tag{3}$$

The output of the first step is a directed graph $G = (V, E, l_v, l_e)$ where the set of vertices V corresponds to the set of primitive shapes composing a sketch, and the set of edges E corresponds to the connecting constraint between the vertices (see Figure 3). We apply two labeling functions. The vertex labeling function, l_v: $V \rightarrow \mathbb{L}^2$, decorates each vertex with a label denoting the nature (ellipse or straight line) and the distinctive feature (orientation for straight lines, nature for circles or not for ellipses) of the primitive shape corresponding to the vertex. The edge labeling function, l_e: $E \rightarrow \mathbb{L}$, decorates each edge with the corresponding connecting relation (*isConnected* or *isConnectedEndPoints*) that binds each pair of primitive shapes.

Listing 1: Predicates of the PDDL domain definition.

```
(define (domain BOARD)
(:requirements :strips :typing :equality)
(:types block shape)
(:predicates (isConnected ?x ?y - block)
    (isConnectedEndPoints ?x ?y - block)
    (onboard ?x - block)
    (removed ?x - block)
    (hasShape ?x - block ?z - shape)
    (hasOrientation ?x - block ?z - orientation) )
```

3.1.3 Translation into PDDL

The second step of the shape recognition sub-process consists in translating the graph obtained during the previous step into the Planning Domain Definition Language (PDDL) format [38]. PDDL is an attempt to formalise a standard to describe AI planning problems that is shared by various components-off-the-shelf AI planners [28, 39]. We use PDDL to formalize our drawing problem, describe the initial sketch, and describe the list of all the model elements deemed possible. This list constitutes the goal library, i.e., the set of the possible goals that our framework will consider when doing goal recognition. The PDDL domain definition contains the formalization of the drawing problem. Listings 1 and 2 are excerpts of the domain definition. It describes predicates that accept variables of two different types: blocks (made of polylines) and shapes (ellipses or straight lines). The coding of the predicates follows the relations given earlier.

Listing 2: A PDDL example of connecting a block to another one. Here, effects make use of conditions encoded with “when”: Conditions are like preconditions, but if they do not hold in a state, the actions are still executed, but the conditional effect that does not hold will simply not be applied in the state.

```
(:action connectEndPoint
   :parameters (?x ?y - block)
   :precondition (and
      (not (= ?x ?y))
      (onboard ?x)
      (onboard ?y) )
   :effect (and
      (isConnectedEndPoints ?x ?y)
      (isConnectedEndPoints ?y ?x)
      (when (isConnected ?x ?y)
        (and
          (not (isConnected ?x ?y))
          (not (isConnected ?y ?x)) ))
    ) )
```

The domain definition also contains actions that can be performed by a user, or, more symbolically, by a drawing agent, to complete a sketch. We define *connect* actions to complete an existing sketch with new shapes. For example, Listing 2 shows the definition of the *connectEndPoint* action to connect two polylines. We also define actions in our plan to support quick fixes [15]. A *remove* action consists in removing from a graph *G* a node and the edges that connect that node to other nodes of the graph. A *remove* action is interpreted as the primitive shape (represented by that node) has been incorrectly drawn and should not be considered as part of the sketch. *Update* actions consist in modifying a node or an edge of a graph *G*. An example of a node update action is changing an eclipse into a circle, or changing a straight line's orientation (*change-orientation*).

Every action also implements collateral effects on the shapes composing a sketch, according to the different relations formalized in Section 3.1.2. For instance, if the block *A* is connected to block *B*, then *B* will also be connected to *A* (Listing 2). We decided to encode collateral effects in the effects of the actions rather than using *axioms*[2] [40] as it was the case in a previous work [15]. The reason lies in the computing overhead that axioms yield, and in the scarcely availability of automated planners implementing them. Thus, the PDDL encoding of the sketching problem presented here is an evolution of a former implementation that was modeling the relative position between blocks in order to express their position on the board (e.g. `left-of, right-of` etc.). We decided to model the connections only between elements, as they are sufficient, along with their shape, to define the model elements used in the sketches. This simplification was dictated by the performances of the previous version of the model. The relative position between blocks needed to be adjusted for all the drawn blocks (this was done by using axioms) after any action. The current model is more compact, and produces shorter plans for certain models, also because of choice of having the property of bijectivity for the connections, as indicated in Eqs 1, 3. Of course the degrees of liberty implied by this modeling yield some ambiguity in the representations, but 1) some ambiguity would be present for any modeling choice, 2) in the context of software diagrams (e.g., UML), these modeling choices allow to represent all the sketches without ambiguity.

The formalization of the drawing problem and the goal library are generic and reused across different executions of the recognition process. Only the formalization of the initial sketch in PDDL is specific and is automatically generated from the graph obtained in the previous step. Listing 3 shows an example of a PDDL problem carrying the information of the initial sketch and a possible goal to reach. The initial state (lines 7–25) is generated automatically on the basis of the sketch currently drawn. The goal describes the positioning constraints required to build an operational actor.

Listing 3: An example of a PDDL problem generated from our approach. The mapping between the variables of the initial state and the variables of the goal is automatically carried out by the translation process.

```
 1 (define (problem BOARD-actor)
 2 (:domain BOARD)
 3   (:objects
 4       c1 v1 h1 dr1 dl1 h2 v3 h3 v4 h4 v2 - block
 5       line circle - shape
 6       none vertical horizontal diagonal_left diagonal_right - orientation)
 7 (:INIT
 8   (onboard v1) (onboard h2) (onboard v3) (onboard h3)
 9   (onboard v4) (onboard h4) (onboard v2) (onboard h1)
10   (isConnectedEndPoints v1 h2) (isConnectedEndPoints h2 v1)
11   (isConnectedEndPoints v1 h3) (isConnectedEndPoints h3 v1)
12   (isConnectedEndPoints h2 v3) (isConnectedEndPoints v3 h2)
13   (isConnectedEndPoints v3 h3) (isConnectedEndPoints h3 v3)
14   (isConnectedEndPoints v4 h4) (isConnectedEndPoints h4 v4)
15   (isConnectedEndPoints v4 h1) (isConnectedEndPoints h1 v4)
16   (isConnectedEndPoints h4 v2) (isConnectedEndPoints v2 h4)
17   (isConnectedEndPoints v2 h1) (isConnectedEndPoints h1 v2)
18   (hasShape v1 line) (hasOrientation v1 vertical)
19   (hasShape h2 line) (hasOrientation h2 horizontal)
20   (hasShape v3 line) (hasOrientation v3 vertical)
21   (hasShape h3 line) (hasOrientation h3 horizontal)
22   (hasShape v4 line) (hasOrientation v4 vertical)
23   (hasShape h4 line) (hasOrientation h4 horizontal)
24   (hasShape v2 line) (hasOrientation v2 vertical)
25   (hasShape h1 line) (hasOrientation h1 horizontal) )
26 (:goal  (AND
27     (onboard c1) (onboard v1) (onboard h1) (onboard dr1) (onboard dl1)
28     (hasShape c1 circle) (hasShape v1 line) (hasOrientation v1 vertical)
29     (hasShape h1 line) (hasOrientation h1 horizontal) (hasShape dr1 line)
30     (hasOrientation dr1 diagonal_right) (hasOrientation dl1 diagonal_left)
31     (isConnected v1 h1) (isConnected h1 v1) (hasShape dl1 line)
32     (isConnectedEndPoints c1 v1) (isConnectedEndPoints v1 c1)
33     (isConnectedEndPoints v1 dr1) (isConnectedEndPoints dr1 v1)
34     (isConnectedEndPoints v1 dl1) (isConnectedEndPoints dl1 v1)
35     (isConnectedEndPoints dl1 dr1) (isConnectedEndPoints dr1 dl1)
36     (not (onboard h2)) (not (onboard v3)) (not (onboard h3))
37     (not (onboard v4)) (not (onboard h4)) (not (onboard v2))) ) )
```

Based on the three inputs, we run the planner for the sketch being drawn by the user. To do it, we used the Fast Downward planning system [41], running the version of LAMA Planner [17] that participated in IPC 2011, and that has been integrated into Fast Downward's code. The anytime algorithm used in LAMA, to the contrary of the previous algorithms discussed above, does not continue the weighted A* search once it finds a solution. Instead, it start a new weighted A*-based search from the initial state. The planner then outputs several plans, with increasing quality (measured in the number of actions in the plan). Each plan is an ordered list of possible matches between the sketch being drawn by the user and the goals denoting the different model elements that could be recognized. The set of possible matches is ordered based on the degree of confidence of the match regarding the element currently drawn. The degree depends on the *distance* (in the plan) between the current sketch and the possible goal, i.e., the number of steps that would remain to finish drawing the element completely.

3.2 Implementation

Figure 4 pictures the BOARD-AI main interface. The interface is developed using Web technologies (HTML, CSS, and JavaScript) so it can be used remotely and it can be run on any interactive pen display devices, ranging from tablets to large screens equipped with stylus. We adopted a minimalist style where the entirety of the screen can be used to draw a model so that users can fully focus on the sketching activity without being disturbed by an overloaded interface. The main area is an HTML5 Canvas for drawing model elements. A toolbar provides some useful features, such as undo/redo, page management, different edition modes (drawing, erasing, and

[2] Axioms are specific actions applicable to a state, but they do not contribute to the evaluation of the distance between the current state and a goal.

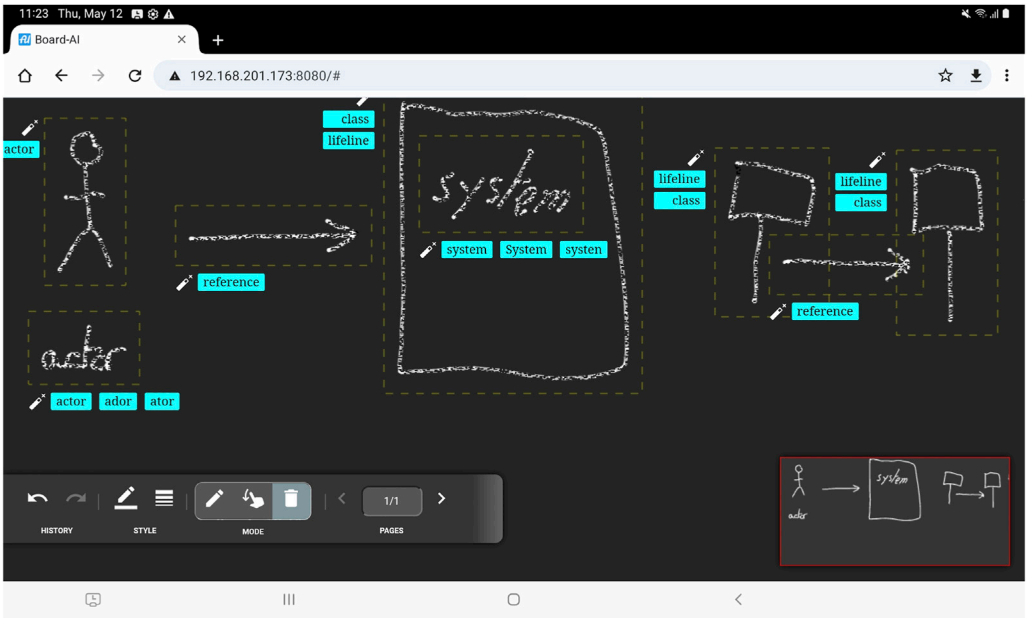


FIGURE 4
Screenshot of the BOARD-AI interface running on a Samsung Galaxy S6 Lite. Its minimalist interface features a wide area to sketch model elements, a collapsible toolbar, and an outline. Blue bubbles along with drawn sketches provide hints about the model elements to recognize.

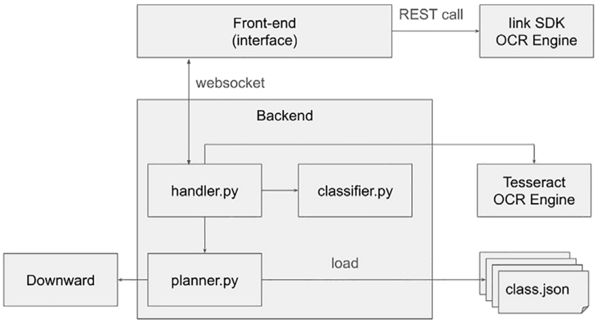


FIGURE 5
Overview of the BOARD-AI architecture.

selection), and two options to customize the thickness and the color of the drawing. The toolbar can be collapsed to maximize the drawing area. Besides these features, a chalk effect is applied to replicate the writing on a blackboard.

Two distinct interaction modalities are used to interact with the screen. Drawing is only possible using the stylus while touch gestures enable the user to navigate across the interface. Zooming in and out is achieved by pinching the screen. Single finger panning can also be used to navigate within the Canvas when it is scaled up. When the interface is scaled up, an outline at the bottom right of the screen shows the visible area.

When the user starts drawing, the recognition process is performed. Under the hood, the recognition engine consecutively identifies and characterizes the primitive shapes drawn by the user, invokes the classifier, and performs text or shape recognition according to the output of the classifier. Visual hints are given in the shape of bubbles accompanying the elements to recognize. Clicking on a visual hint results in converting the partial drawing into the suggested model element or text. If the partial drawing is updated by the user, the suggestions are updated as well.

Multiple shapes and/or textual annotations can be recognized at the same time. Hints to geometrical shapes are provided by the Fast Downward planning system and are updated according to the anytime algorithm used. A limit of the three best suggestions is kept and displayed along with the partial drawing. Recognizing textual annotation is done using third-party OCR engines. Our implementation currently supports both MyScript Interactive Ink SDK (iink SDK) and Tesseract OCR engines. When using iink SDK, multiple suggestions are provided and displayed to the user. A metric distance is then applied to reconcile model elements with textual annotation. As an example illustrated in Figure 4, textual annotations will be respectively converted into class or actor names.

Figure 5 describes the architecture of BOARD-AI. It follows a client-server architecture where the back-end (in Python) is responsible for calling the different services for shape recognition. It consists of three main modules. The *handler* module makes the link with the interface. It starts a Web-socket server to communicate with the front-end. Shape recognition being incremental, the web-socket communication is used to update the interface every-time a more optimized plan is found. A *Classifier* module built on top of *numpy* distinguishes geometrical shapes from textual annotations. The *planner* module is responsible for translating sketched elements into PDDL as discussed in Section 3.1.3. Models elements to be recognized are structured as JSON objects (see Listing. 4 as an example) and stored in separate files. When the *planner* module starts, it first loads the different files and then translates each JSON object into PDDL problem templates as shown in Listing 3[3].

Listing 4: An example of the definition of the goal in JSON for recognizing a UML class.

```
{
  "name": "class",
  "shapes": [
    {
      "name": "h1",
      "type": "line",
      "direction": "horizontal"
    },
    {
      "name": "h2",
      "type": "line",
      "direction": "horizontal"
    },
    {
      "name": "h3",
      "type": "line",
      "direction": "horizontal"
    },
    {
      "name": "v1",
      "type": "line",
      "direction": "vertical"
    },
    {
      "name": "v2",
      "type": "line",
      "direction": "vertical"
    }
  ],
  "areConnectedEndPoints": [
    {"s1": "h1", "s2": "v1"},
    {"s1": "h1", "s2": "v2"},
    {"s1": "v1", "s2": "h2"},
    {"s1": "v2", "s2": "h2"}
  ],
  "areConnected": [
    {"s1": "h3", "s2": "v1"},
    {"s1": "h3", "s2": "v2"}
  ]
}
```

Upon requesting a recognition of a sketched elements, the following actions are executed. First, the *classifier* module starts by classifying the sketched element as text or shape elements. If a sketched element is classified as text, text recognition is performed by an OCR engine. Depending on the selected OCR engine used (being Tesseract or iink SDK), text recognition is performed by the back-end or the front-end. However, if the sketched element is classified as a geometrical shape, shape recognition is performed by the *planner* module. This module translates the sketched elements into PDDL and invokes multiple instances of Fast Downward as parallel sub-processes. As soon as a plan is generated by one sub-process, it is broadcast to the front-end interface through the *handler* module.

Running multiple instances of Fast Downward in parallel speeds up the recognition process as suggestions of shapes are provided in the interface as soon as they arrive. In our tests, recognizing multiple model elements (mixing geometrical shapes and text annotations) as the one pictured in Figure 4 takes between 500 milliseconds and 1 s. However, it is computationally heavy and requires a proper server architecture to reduce the computational time.

3.3 Model training

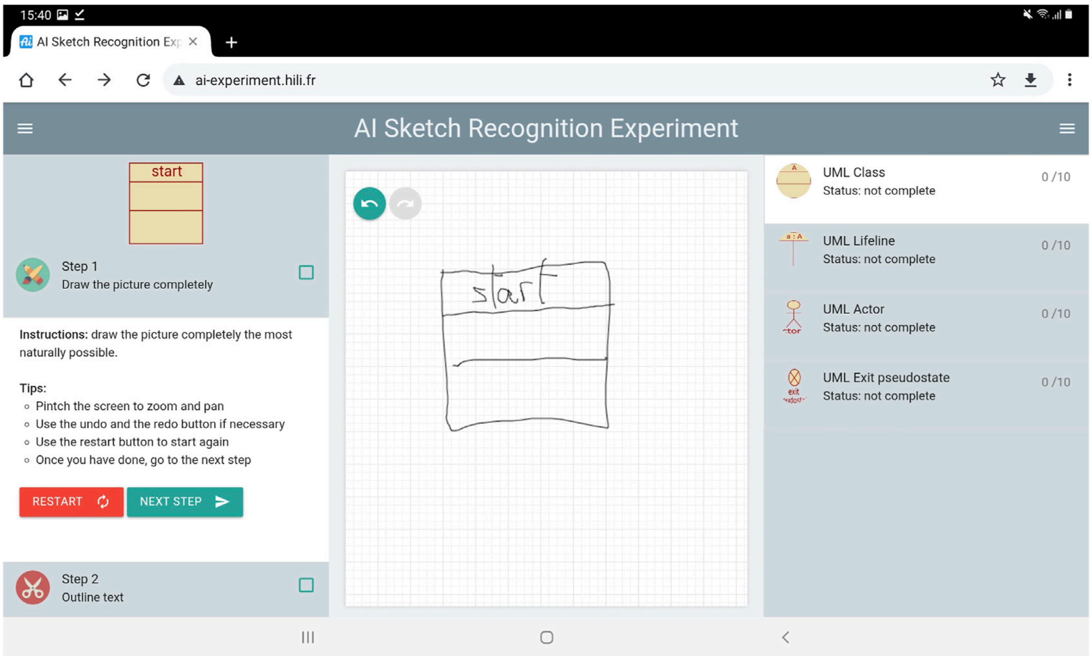
This section describes the training of the NN classifier we used. We chose to set up a NN based on specific features rather than on exploiting images as the first solution is more cost-effective and requires less data than the second one. The section first details the training interface we developed, then it discusses the data acquisition and the features that were extracted to correctly train the classifier. Finally, it presents the validation of the training.

3.3.1 Training interface

Figure 6 is an overview of the training interface. It has been used to train the NN to learn how to differentiate geometrical shapes and text. The interface shares several similarities with the interface of BOARD-AI. It was developed using the same technologies (HTML, CSS, and JavaScript) and relies on the two same interaction modalities. We chose to separate the two interaction modalities so as not to lead to a bias during the training. Finger drawing may result in a bad training of the NN. Besides, styli seem to be the most natural way of drawing onto a screen and provide a nicer user experience for text annotation than finger drawing. During the experiment, we observed that the participants naturally use one hand to hold the stylus while using the second one to navigate within the HTML5 Canvas.

The training interface contains three areas. The main area is an HTML5 Canvas for drawing model elements, as it is done in the interface of BOARD-AI. Both sidebars contain indications for the user to understand how to use the training interface. The interface has been used as an experimental platform to collect users' drawing data so as to train neural networks to predict which parts of a drawing relate to text and which parts relate to

[3] The templates are later filled with the missing part describing the initial state of the problem.



AI Sketch Recognition Experiment

Step 1 □
Draw the picture completely

Instructions: draw the picture completely the most naturally possible.

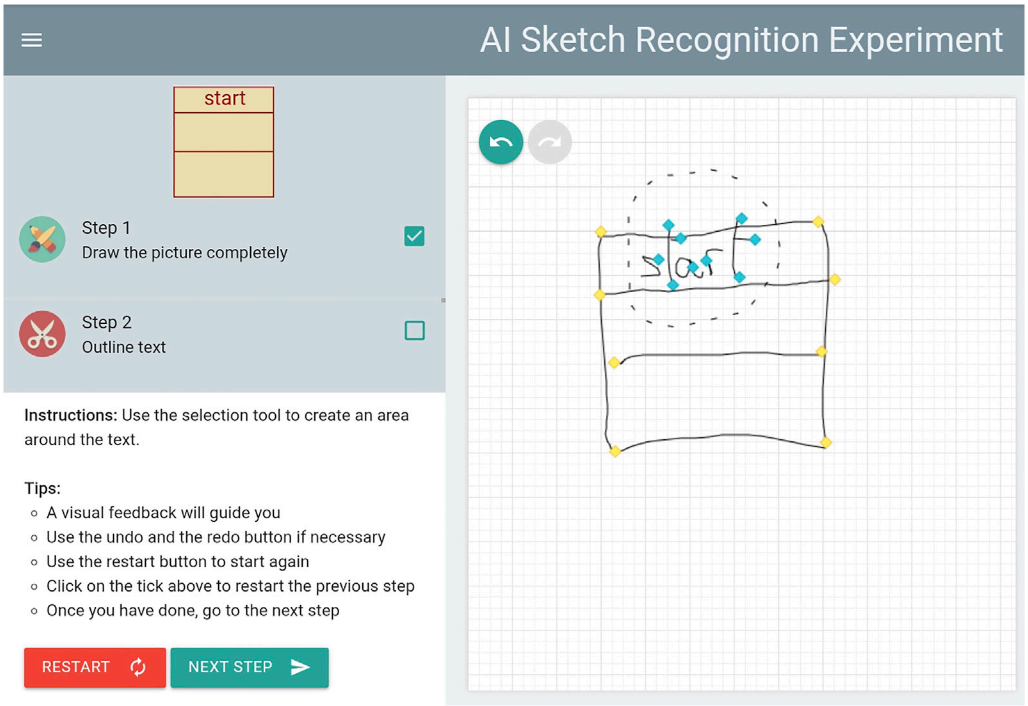
Tips:

- Pintch the screen to zoom and pan
- Use the undo and the redo button if necessary
- Use the restart button to start again
- Once you have done, go to the next step

Step 2 □
Outline text

UML Class Status: not complete	0 /10
UML Lifeline Status: not complete	0 /10
UML Actor Status: not complete	0 /10
UML Exit pseudostate Status: not complete	0 /10

FIGURE 6
Overview of the training interface. It is used to train the neural network. The left sidenav shows indications to guide participants through the training. The right sidenav shows the progress of the participant. The main central area accepts pointer events to draw and touch events to navigate within the HTML5 Canvas.


AI Sketch Recognition Experiment

Step 1 ☑
Draw the picture completely

Step 2 □
Outline text

Instructions: Use the selection tool to create an area around the text.

Tips:

- A visual feedback will guide you
- Use the undo and the redo button if necessary
- Use the restart button to start again
- Click on the tick above to restart the previous step
- Once you have done, go to the next step

FIGURE 7
Upon completing a sketch, participants are asked to select (using a Lasso Tool) the text only. Sharp corners are shown in yellow. They turn blue once they have been selected.

geometrical shapes. Using the interface, users are invited to perform two successive steps: *drawing* and *text selection*.

Drawing

During the first step, users are invited to draw graphical model elements as they appear on the top left corner of the interface. Each drawing is stored as a set of paths. A path starts when the user touches the screen with the pen and ends when the pen is lifted from the surface of the screen. A path is a collection of points acquired by the device during this timeframe. A point corresponds to a pixel drawn on the screen. It contains a xy-coordinate on the screen. The frequency rate of events fired to detect that a point is drawn on the screen is device-specific and depends on the hardware implementation of the tablet [42]. In addition to its xy-coordinate on the screen, a point also contains various metadata collected from the stylus. Specifically, we collected the pressure applied on the stylus, the plane angles *tiltX* and *tiltY* between the stylus and the surface, and the timestamp when the pixel is drawn on the screen. We decided to keep several pieces of metadata to find which ones are relevant to efficiently train the neural network, but also for being able to precisely and accurately replicate the experiments offline, and to understand the user's habits in terms of drawings. This understanding will lead to new experiments in the future.

Text selection

Once the user completes his/her drawing, a line recognition algorithm detects sharp corners. Sharp corners are particular points of a path denoting marked directional changes. This step is used by our algorithm to transform a path into multiple straight lines. During the second step, users are invited to select sharp corners belonging to any textual part of the drawing using a *Lasso* tool (see Figure 7). This second step is important for the classification process to distinguish text from geometrical shapes.

3.3.2 Data acquisition

We used the interface to collect data to train the NN. Each participant was asked to draw ten times, four graphical elements composed of one of four chosen elements coming from UML (class, actor, lifeline, and pseudo-state) and a randomly chosen English word. In practice, this represents, for each participant, a total of eighty training samples for the network (forty for each class, text or shape).

We took different measures so as not to introduce any bias in the experiment. All experiments were performed on the same device, a Samsung Galaxy S6 Lite tablet equipped with an active pen stylus. Finger drawing was disabled. Random words were chosen from a dataset of the most commonly used English words to add variability in the training process. We developed a flip mode for left-handed users where both sidebars are flipped. This mode has been developed so that the drawing directives always appear on the opposite side of the hand holding the stylus and are not covered by the hand. As a final measure, the four types of model elements were presented to the participants in a random order to prevent "muscle memory".

3.3.3 Feature design

The data acquired using the training interface described above was pre-processed and analysed in order to keep only the relevant pieces of information. To use our NN classifier, we identified the features to be passed as inputs to the network. Features were chosen to be independent of the writing speed and also scale independent, i.e., independent of the size of the user's drawing. For these reasons, data such as the number of points (depending both on the acquisition capabilities of the tablet and on the speed of the user's writing) were removed. Other data such as the length and the width of the bounding box was not used directly but computed to form new measures relevant to the network. After conducting the analysis on the data, the chosen features are:

- the number of sharp corners. We can imagine that this number is greater for text.
- the bounding box ratio: It is computed as the ratio between the width and height of the bounding box. The bounding box is the smallest rectangle encompassing all points. We can imagine that text would tend to have horizontal boxes and shapes more vertical boxes.
- the longest segment ratio: It is computed as the longest corner segment divided by the longest side of the bounding box. A segment is defined as a line between two consecutive points, and a corner segment is defined as a segment between two consecutive corners. We can imagine that we are more prone to find longer segments in geometrical shapes e.g. boxes, arrows, actors rather than in text.
- the total segment ratio: It is computed as the sum of all corner segments divided by the longest side of the bounding box. We can expect that text will tend to have a greater ratio.
- the minimum angle corner: It corresponds to the minimum angle computed among all angles found between two corner segments. We can expect that shape will tend to have a greater minimum angle.

3.3.4 Network design

The starting proposed structure for the NN classifier corresponds to a multilayer fully connected network, and it is composed of:

- one input layer of size five since we have five features.
- one output layer that represents one of the categories; it must be one or zero for each geometrical shape or text, respectively.
- one to three hidden layers between the input and the output layers. Besides these three layers, the size of each layer varies from five to ten neurons.

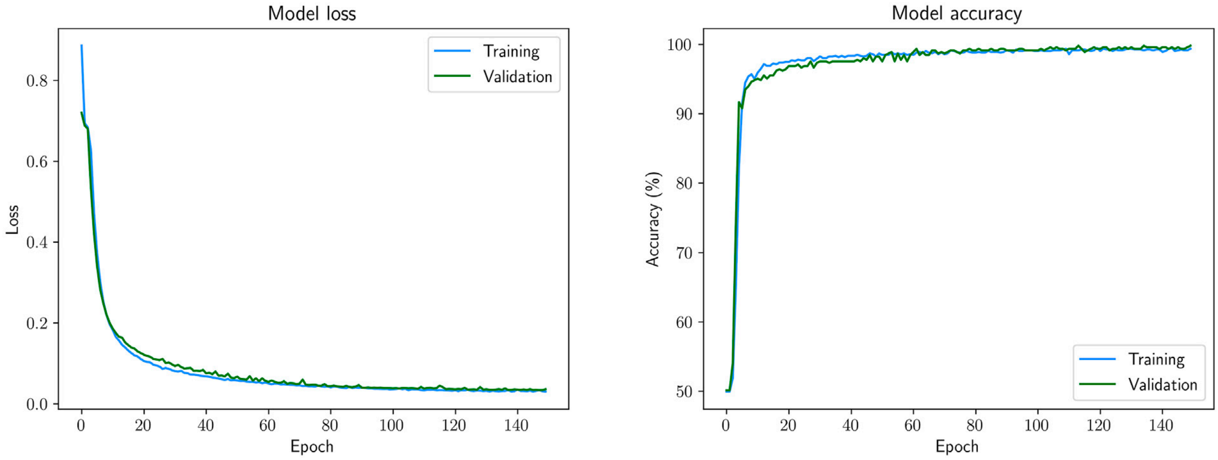


FIGURE 8
Model loss and accuracy on training and validation data for the final structure network.

On the one side, the rectified linear unit function [43, 44] was chosen as an activation function for the hidden layers. On the other side, to guarantee that our network output is between 0 and 1, we used a Sigmoid activation function for the output layer.

The final structure of the network was chosen after the training and validation step. It is described in the following section.

3.3.5 Training and validation

Our dataset is composed of 1,342 entries (after removing 18 invalid entries) coming from 17 different users who participated in the user experiment described earlier. The dataset was divided between 67% training and 33% validation.

Training a network consists in finding the best set of weights to map the elements to be classified to their respective categories. To evaluate a set of weights, we must specify a loss function. This function is used by an optimization algorithm to estimate the loss of the model, update weights and reduce the loss on the next evaluation.

To train our network, we used the Adam optimization algorithm and we use cross-entropy as the loss function for our binary classification problem. This process was run for 150 epochs (the number of iterations through the dataset) and a batch size of 10 (number of training data considered per epoch).

To tune the hyperparameters (number of hidden layers, number of neurons for each hidden layer) we conducted 500 experiments. For each experiment, the number of hidden layers was randomly chosen between one and three and for each hidden layer, the number of neurons was randomly chosen between five and ten. The final structure network showing the best results was composed of two hidden layers of sizes eight and five nodes respectively. Figure 8 present the performance of the network over time during training. On the one hand, we can observe that the model loss has comparable performance on both datasets. On the other hand, the model accuracy plateaus indicate that the model did not underfit and that the validation accuracy did not diverge from the training accuracy, indicating that the model did not overfit. The validation stage showed that the final structured network has a good performance and a 99.77% prediction accuracy.

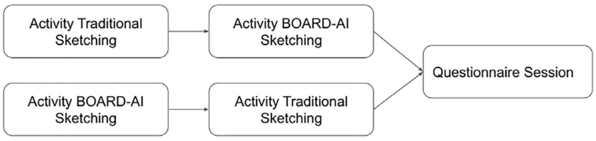

Activity Traditional Sketching
Activity BOARD-AI Sketching
Activity BOARD-AI Sketching
Activity Traditional Sketching
Questionnaire Session

FIGURE 9
Exploratory Study Design and data gathering techniques.

In future work, we would like to consider if training on a subset of the available shapes of our training data would still allow for good classification performance even on unseen shapes.

4 Evaluation

4.1 Aims and research questions

BOARD-AI was designed to facilitate sketching system engineering models. For its implementation, BOARD-AI was trained using data collected from human draws and annotations (see Section 3.3). We present now the results of evaluating

TABLE 1 Data gathering instruments.

Data gathering technique	Description	References
Questionnaire	Questionnaire with 22 questions. This questionnaire is composed of 3 preliminary questions about the user's background, the Usability Scale (SUS) [45] translated to French, i.e. 10 closed questions, plus 7 closed and 2 open questions inquiring the user about the functionalities of the tool	https://osf.io/v45c8/
Participants Consent Form	Consent form to be signed before participating in the experimental study	https://osf.io/v45c8/
Protocol Document Activity Group_1	Document facilitated to the Group_1, which started sketching using a traditional method, and then using BOARD-AI	https://osf.io/qutx3/
Protocol Document Activity Group_2	Document facilitated to the Group_1, which started sketching using BOARD-AI, and then a traditional method	https://osf.io/g36rz/

BOARD-AI with final users. This evaluation permits to understand how BOARD-AI supports engineers in sketching system models, and allows us to identify future improvements of the tool. Specifically, four research questions drove the evaluation:

RQ 1. Does BOARD-AI facilitate the modeling process compared with traditional methods?
RQ 2. How helpful are the tips and shape suggestions provided to the users, so as to support sketching system models?
RQ 3. Would a false positive in recognizing a geometrical symbol or a textual label affect the user the same way?
RQ 4. How usable is the BOARD-AI tool?

4.2 Methodology and data collection

To address the research questions proposed for the evaluation, we conducted an exploratory study. This is the recommended methodology for studying a phenomenon when there is insufficient prior research to establish concrete hypotheses. In this case, we wanted to study the effects of using BOARD-AI in supporting engineers on sketching their system engineering models compared with other existing more traditional methods. For this purpose, we prepared two different protocols for two groups of users (Group_1 and Group_2). Both protocols consisted in a sequence of two activities that participants had to follow:

- Activity Traditional Sketching: Users are asked to sketch a system model using paper and pencil or whatever tool of their choice they usually use for sketching models.
- Activity BOARD-AI Sketching: Users are asked to sketch the same system model using BOARD-AI.

The first group of participants (Group_1) started with the Activity Traditional Sketching and continued with the Activity BOARD-AI Sketching, whereas the second group (Group_2) of participants performed these activities in the reverse order. Combining the use of traditional sketching methods and BOARD-AI allowed the users to be able to compare the two methods for conducting similar activities and, while having two different groups isolated the effect of the novelty of using BOARD-AI in such an activity. Figure 9 shows a schema of the experimental procedure conducted.

Different data gathering techniques were used to collect data about the two activities at different moments. First, the participants were asked to fill in a consent form for participating in the experiment. Then, a researcher facilitated them with a document explaining the activities to be conducted and how to access the BOARD-AI tool. Then, the participants completed the two activities with no time limitations. Finally, the participants answered a final questionnaire containing 22 questions.

The first three questions are dedicated to determine the profile of the participants and their experience in the modeling domain. The next 10 questions (from question one to question 10) correspond to the SUS standardized questionnaire [45], designed to measure the usability of the tool. Then, questions 11 and 12 are dedicated to compare BOARD-AI with other traditional methods; questions 13 to 17 refer to aspects related with the tips and graphical help offered by BOARD-AI, and questions 19 and 20 (the latter is a supplementary question for non expert users only) ask about which aspects of the tool are considered the best, and what are those that should be improved. Except questions 19 and 20, which are open questions, the rest follow a Likert scale from 1 to 5, were 1 is "completely disagree", and five is "completely agree". Table 1 shows the different data gathering techniques that were employed as well as the links to the instruments used.

4.3 Participants and analytical methods

As a sampling method, we decided to follow convenience sampling for selecting participants. In this case, 12 participants from the University of Grenoble Alpes, engineers, experts and

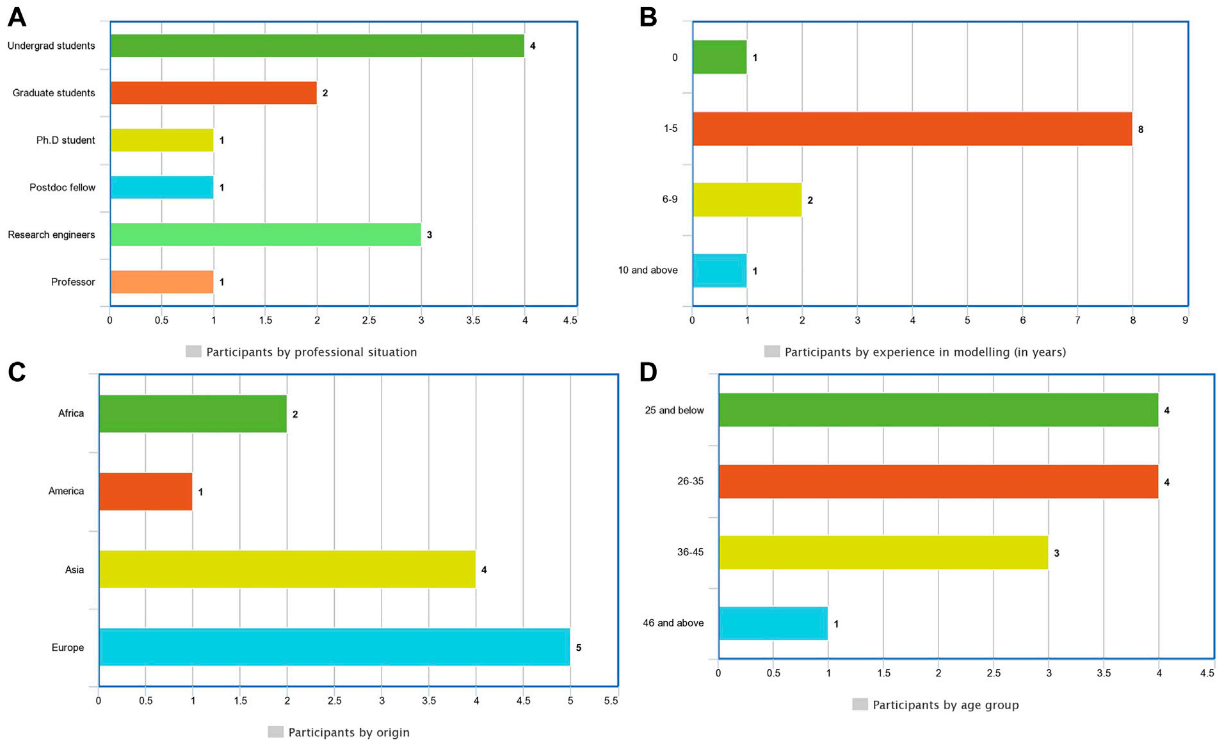


FIGURE 10
Overview of the participants in terms of professional situation **(A)**, experience in modeling **(B)**, origins **(C)**, and age groups **(D)**.

non experts in software and/or system modeling design, were invited *via* e-mail and all accepted the invitation. Six participants were randomly assigned to Group_1 and the other six to Group_2. All participants received an e-mail indicating the place and the time for participating in the study. Participation was voluntary and no reward was proposed.

Coverage/representativeness of the user base being an important concern in any user experiment, the 12 participants have been invited to have a sample representative of the population interested in using sketch-based tools. All participants but one had previous practical experience in modeling or had been taught modeling during their academic courses. Although all participants shared the same affiliation at the time of the user experiment, their profiles presented several variations, in terms of experience in modeling, CASE tools/sketch tools they are comfortable with, employment situations, age groups, and origins. Figure 10 details the different participants.

For answering the RQ1 about how BOARD-AI facilitates the modeling process compared with traditional methods we analyzed the answers to questions 11 and 12 in the Questionnaire. For answering RQ2 about how helpful the tips and shape suggestions of BOARD-AI are, we analyzed the answers to questions 13 to 16 in the Questionnaire (both included). For answering RQ3 about whether a faulty text recognition is less important than an incorrect shape recognition, we analyzed the answers to question 17 specifically. In all cases, we calculated the percentage of answers given by the participants of the experiment between 1 and 2 (badly evaluated), 3 (neutral) and 4–5 (well evaluated). Questions 1 to 10 (corresponding to the SUS questionnaire) were analyzed following the instructions provided in its design for answering RQ4 regarding the usability of the tool. Finally, we qualitatively analyzed the answers given by the participants to improve the tool, and classified them according to the different emergent topics. These answers were used to complement the data collected through the questionnaire.

4.4 Results

The participants were asked to perform similar tasks using two sketching methods. Comparing BOARD-AI to more traditional methods and to engineers habits permits to assess the usability of the tool, its performance, and eventually the trust that users are willing to put in an AI-based tool.

First, regarding RQ1, the answers to the questions 11 and 12 underlined the simplicity of using BOARD-AI. 83.4% of the participants (scores 3–5) considered that sketching with BOARD-AI was easier (or of the same difficulty) than traditional methods. This result is supported by the answers to question 12 about the confidence they had in the sketch they made: 75% of the participants considered that using BOARD-AI the resulting sketch was less or equally prone to contain mistakes or errors than using the other method—66,7% answered that they trusted the BOARD-AI sketch more in avoiding potential errors.

Second, and regarding RQ2, participants' answers to the questionnaire suggest that the support provided by BOARD-AI through tips and shape suggestions are positive and valued by the end-users. On the one hand, results from analyzing questions 13 and 14 indicate that no participant said that the tips, the shape suggestions, and the toolbar provided by the modeling interface were improper (that would have been scores 1–2). 75% of the participants gave high scores [4, 5] to the suitability of the suggested outline to the sketch being drawn, while 41.7% approved with similar scores the proposed toolbar. On the other hand, the analysis shows that users see in the suggestions an element of trust in the tool that facilitated their modeling process and their confidence on the result. This is supported by data collected from questions 12 and 15, which shows that all but two participants trusted the suggestions, that were judged appropriate. Even more, 2/3 of the participants indicated that the completion suggestions were appropriate to their intentions, easing and quickening the sketching job. These results are also supported by the qualitative data collected, which indicates that participants appreciated the shape and text recognition as a mechanism that could facilitate collaborative work: "Text recognition is new feature - the tool will help team to be collaborative" (Participant 11). Also, the analysis shows that suggestions on graphic elements helped them to create their models: 83.4% of the participants said that including the AI-suggested graphic elements in their final sketch helped them greatly to complete the task (scores 4–5) (Question 16).

Third, and regarding RQ3, participants were asked to answer to a specific question to see whether faulty text recognition had less impact than faulty shape recognition. However, there is not a preferred mechanism for suggesting among the participants. 41.6% of the participants expressed a clear preference for trusting a system that has a good accuracy in recognizing drawn shapes, while 41.6% prefer trusting systems that perform a good text recognition (Question 17). This indicates that a widely accepted AI-based tool should progress on both aspects.

Questions 1 to 10 were used to determine the SUS score associated with the BOARD-AI user interface. The results of the SUS score in our study was 65.2%. This indicates an *OK* rating for BOARD-AI, as it is currently designed. However, this score varies depending on the experience of the participants. We noted that the mean of the SUS for less experienced participants (students) is 61.7%, and for more experienced participants (engineers, MSc, post-docs) it raises at a value of 68.75%. Thus, experienced system engineers gave an evaluation of the BOARD-AI interface and usability more towards a *Good* rating than less experienced participants. This can depend on the easiness to adapt to a new modeling interface for experienced users, who used other tools in the past, on the contrary of students, that are still learning to master more traditional tools. This result is complemented by the qualitative data collected. Participants found BOARD-AI easy to understand and to use. Participants especially value its simplicity, i.e. "It is simple to use" (Participant 4), or "Easy to separate colors of different concepts" (Participant 4).

However, and despite of the positive aspects of the tool, participants identified three main aspects to be improved. First, they highlighted that the tool is slow responding to the text and shape recognition, which could be improved if the tool is uploaded to a high performance server, and by other implementation adjustments. Second, participants indicated that the text recognition engine needs some improvement. They found that not all the texts were correctly identified. Since BOARD-AI relies on a third-party software for the text recognition, this is something that could not be controlled in this first evaluation. However, future work will analyze other possible solutions that could improve the text recognition process. And third, the participants identified that the deletion and help options were not enough intuitive and should be improved. This will be considered in future development of the UI.

4.5 Threats to validity

We identified several threats to validity we list below. First, the low number of participants and their shared affiliation to the same institute at the time of the user experiment can affect the validity of the results. Nevertheless, as mentioned above, we believe that the participants represent a sample representative of the population interested in using sketching tools and each participant differs from each other with respect to his/her personal background (experience in modeling, tools used during his/her past experiences, employment situation), origin, and age group.

Another limitation of the present study relates to the alphabet used. The NN has only been trained with models using the Roman alphabet, hence limiting the broad applicability of the approach. Regarding the user experiment, all participants were proficient in writing using the Roman alphabet and use it daily at their professional workplace, although half of the participants were native to other writing systems. However, it is worth noting that we did not observe any variation in the results of the user experiment based on this criterion, and that no participant stressed out this point when filling the questionnaire.

Finally, RQ3 relied on the hypothesis that a faulty recognition when recognizing text has less impact on the users' confidence

TABLE 2 Existing approaches for natural sketching.

	MyScript Diagram	OctoUML	Bresler et al. [10, 11]	FlexiSketch	Lank et al. [5]	Tahuti [7]
Platform	Web and Desktop	Web	.NET framework[a]	Android	Desktop	Desktop
Open source	✗	✓	✗	✗	✗	✗
Recognition						
Scope	Flowcharts, organizational charts, mindmaps	Class diagram only	Flowcharts, automata	Adaptable through type promotion	UML class, sequence, usecase	UML class
Algorithm	Proprietary	Geometrical shape detection[b]	classifier and segmenter	Geometrical shape detection[c]	classifier and segmenter	Geometrical shape detection
Sktech recognition	Basic geometrical shapes	Basic geometrical shapes	Flowcharts and automata symbols	Complex geometrical shapes	UML glyphs	Basic geometrical shapes
Bulk recognition	✓	✓[d]	✓	✗	✓	✓
Handwritten text recognition	✓	✗	✓	✗	✓[e]	✓[e]
Incremental recognition	✓	✓	✗	✓	✗	✓
Explainable results	✗	✗	✗	✗	✗	✗
Performance						
Accuracy	Relatively accurate	Relatively accurate	Accurate	Relatively accurate	Unknown	Unknown
Speed	Relatively slow	Moderately fast	Fast	Relatively fast	Unknown	Unknown

✓ = available ✗ = not available.

[a] Experiments have been implemented in C# but there is no mention of any implementation.

[b] Based on PaleoSketch [9].

[c] Based on a Levenshtein distance algorithm.

[d] With some restrictions.

[e] In both work descreibd in [5, 7], text is identified but not recognized. The use of third-party tools, e.g., OCR, engines is suggested.

than an incorrect interpretation of the modeling intent. The hypothesis is based on the fact that quick corrective actions can be taken in the event of incorrect text recognition (e.g., resorting to virtual keyboards to correct only the fragments of text incorrectly recognized). However, no strong agreement to this question came out, as half of the participants agreed or strongly agreed while half of the participant disagreed or strongly disagreed. One possible interpretation of this result was that, at the time of the user experiment, we did not provide participants with the aforementioned corrective actions to quickly fix faulty text recognition. Providing such corrective actions (with the help of virtual keyboards, physical keyboards, or speech recognition) could have tip the scale in favor to a good shape recognition accuracy over than a good text recognition one.

5 Related work

Different attempts have been done to design and implement robust online sketch recognition algorithms (see Table 2) [5] designed a online recognizer based on a segmentation algorithm for hand drawn UML diagrams sketched on electronic whiteboards. We followed the same principle of image acquisition where points are collected from the instrumented drawing surface and are converted into strokes and straight lines. However, timing information are used for the segmentation process while we chose not to use this data as it is highly dependent on the user and his/her drawing habits. Besides, the approach addresses the recognition of UML symbols and characters using the same segmentation technique. In our approach, we assume that both text and geometrical shapes belong to two different classes of problems and therefore require two different processings. Relying on goal recognition also preserves explainability as the user is not left clueless when the recognizer's outcomes do not correspond to the user's intent.

Bresler et al. [10, 11] propose a recognition method to recognizing flowcharts and finite automata. They use a segmentation and classification approach to separate text and symbols. We share the same rationale in our approach as text and shapes need separate techniques to be recognized. Text recognition relies on *Microsoft. Recognizers.Text*, a module of the *NET framework* ecosystem. Experiments conducted by the authors show that the recognition is fast and accurate. The approach can be generalized to any diagram consisting of symbols connected by arrows, but it requires large amount of data to train the classifier.

Tahuti [7] also addresses online recognition of UML diagrams. We share the same approach of expressing complex sketches in terms of geometrical properties and of primitive shapes it contains. Like Tahuti, we strictly reduce the set of primitive shapes we use to a set sufficient to express any modeling elements for common modeling languages. Tahuti focuses on UML class diagrams. We tend to be more generic with our goal recognition approach where we can define library of goals to describe the model elements of various modeling languages. Tahuti supports handwritten text annotation with some limitations. Text is merely identified but not recognized. Its identification depends on positioning constraints that are specific to class diagrams. Text should be contained in the class or appear next to it. Our approach based on NN does not constrain the positioning of the text with regards to the model elements it annotates and tends to be more generic. Yet, once the classifier classifies a drawing as text, its positioning relative to the model element it annotates is expressed at the conceptual level and is part of the goal library.

SketchREAD [8] is a multi-domain sketch recognition engine using Bayesian networks to improve the recognition process. SketchREAD also reasons in terms of geometrical abstractions of a user's drawings and decompose complex sketches into strokes. It does not require any training data and only needs the description of the sketches in terms of subshapes and constraints between them. Therefore, it tends to be more generic than Tahuti, as it can be adapted to various domains. Our goal recognition approach shares the same philosophy, based on the definition of various libraries of goals, depending on the targeted modeling language. One major objective of using goal recognition is to preserve explainability of the outcomes of the modeling assistant to the user. Besides, SketchREAD does not seem to support handwritten text annotation as we do in the present study.

MyScript [20] is a leading company in the domain of handwriting recognition. It features *MyScript Diagram*, a natural sketching tool used to create various kinds of charts from flowcharts to mindmaps. Ten primitive shapes and connectors are recognized, and text recognition is supported in multiple languages. MyScript runs on desktops or in the cloud. The recognition algorithm remains proprietary and recognition can be done remotely (on a subscription basis) or on-device. Compared to the other solutions, MyScript Diagram does not need to rely on other interaction modalities (such as voice recognition or virtual keyboard) to recognize shapes and text in a simultaneous way.

OctoUML [12, 46] is the prototype of a modeling environment that captures UML models in a free-form modeling fashion and in a collaborative way. It can be used on various devices, including desktop computers and large interactive whiteboards. Sketches are then converted into a graphical UML notation. OctoUML supports class and sequence diagrams. It uses a *selective recognition* algorithm to support an incremental formalization.

OctoUML relies on PaleoSketch [9], a recognition algorithm capable of recognizing eight primitive shapes (lines, polylines, circles, ellipses, arcs, curves, spirals, and helixes) and more complex shapes as a combination of these primitive ones. By recognizing more primitive shapes than other low-level recognizers, PaleoSketch intends to recognize domain-specific shapes that could be indescribable using other methods. The drawback is that it consumes time to recognize more primitive shapes. In our tests, we observed that recognizing shapes takes on average 500 ms and up to 1 s, both of which are noticeable to the user. Besides this condition, the rationale behind recognizing more elementary shapes is elusive as some shapes (helixes, waves, spirals, *etc.*) are never used in modeling languages, specifically in systems engineering. In our approach, we took the opposite stance by choosing to recognize only a few primitive shapes (lines, circles, and ellipses) and to use plan recognition to identify model elements as any combination of these primitive shapes. The three primitive shapes are indeed sufficient in Model-Based System Engineering (MBSE) to recognize most modeling elements drawn in the most common modeling languages and to reduce the number of primitive shapes that need to be recognized to speed up the recognition process. In our tests, recognizing complex shapes (e.g., an operational actor made of four straight lines and one circle) fell under 100 ms, which is barely noticeable to the user.

FlexiSketch [19] is a diagram modeling tool available on Android platforms. In FlexiSketch, a user can sketch model elements and later promote them as types than can be easily re-used. Once a graphical sketch has been associated with a model element, similar sketches are automatically recognized. This allows for adapting FlexiSketch to new graph-based modeling languages. The FlexiSketch's recognizer relies on an adapted version of a Levenshtein string-distance algorithm. The recognition is relatively fast and accurate.

We note that among the different solutions, only MyScript Diagram and the recognition of Bresler et al. [10, 11], provide seamless handwritten text annotation recognition capabilities. In [5, 7], the use of OCR engines is suggested but not seamlessly integrated into the respective tools. OctoUML allows the users to add textual properties through a physical keyboard or *via* voice recognition which requires the users to first recognize and formalize the model elements before adding text. In FlexiSketch, textual properties of model elements are only set using the Android virtual keyboard. In our previous work [47], text annotation could also be attached to model elements after they have been recognized like in OctoUML, through a draggable virtual keyboard and voice recognition. Providing a support for mixed text and geometrical sketch recognition is an objective of the present study.

Finally, none of the aforementioned solutions provide explainable outputs. Some work such as FlexiSketch provide alternative suggestions, and MyScript Diagram can provide word suggestions during text recognition. But none of these solutions can explain why an element has been recognized in the first place. In our approach, the output of the recognizer is completely explainable. The user is informed of which part of a modeling language is being recognized (the primitive shapes composing the modeling elements), and what remains to be drawn using visual feedback.

6 Conclusion

This study presents an approach for sketch recognition of systems engineering model elements combining the benefits of Machine Learning (ML) and Automated Planning. Compared to existing ones, this approach is able to recognize model elements annotated with text supports while preserving the explainability of the outcomes of the sketch recognizer. To achieve this result, the approach relies on ML and on a trained Neural Network (NN) to separate, upstream from the global recognition process, handwritten text annotations from geometrical shapes, as the two belong to two different classes of problems and require different recognition techniques. Component-off-the-shelf Optical Character Recognition (OCR) engines are indeed well suited for text recognition while plan and goal recognition techniques permit our system to recognize a sketched element even from a partial drawing.

In our previous work [14, 15], we detailed the adaptation of plan and goal recognition techniques for sketch recognition. In the present study: 1) we complemented the approach with ML, 2) we integrated two OCR engines (namely Tesseract and iink SDK) to seamlessly recognizing text annotations in model elements; 3) we improved our original implementation; and 4) we reformulated the planning domain to be lighter while adopting an *anytime* algorithm to produce faster plans with incremental quality.

It resulted in the definition and the implementation of a Human-machine interface named *board-ai*, which, compared to our initial prototype [15], now supports incremental recognition of multiple sketches in parallel mixing geometrical shapes and textual annotations. The validation stage used to classify the sketches gave us good results for the NN and a prediction accuracy of 99.77%.

We finally assessed the usability of the Human-machine interface for Systems Engineering modeling. Thus, results from the human data permitted an evaluation that helped us to understand how BOARD-AI supports and facilitates the work of system engineers, and whether an AI-based modeling environment is trusted and deemed usable by its users. The study we've conducted provide very encouraging results about usability, and AI-assisted sketching. We acknowledge that providing tips and suggestions to the users alongside an explanation on why this suggestion is well evaluated by the AI, increased both a faster adaptation and an increased confidence in using BOARD-AI.

Data availability statement

The original contributions presented in the study are included in the article/supplementary materials, further inquiries can be directed to the corresponding author.

Author contributions

SC-P has developed and validated the Neural Network (NN) classifier, contributed to the design of the NN training interface and designed the data acquisition process to train the NN classifier; NH has led the writing of the paper and coordinated the different contributions, he has developed the BOARD-AI Human-Machine modeling interface, and integrated the different software modules; AA has modeled the planning problem, integrated the planner, and analyzed the evaluation; MP-S has participated in the experimental design for the assessment of BOARD-AI with final users, and performed the analysis of the assessment.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

1. Robertson B, Radcliffe D. Impact of CAD tools on creative problem solving in engineering design. *Computer-Aided Des* (2009) 41:136–46. doi:10.1016/j.cad.2008.06.007

2. Rudin C. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nat Mach Intell* (2019) 1:206–15. doi:10.1038/s42256-019-0048-x

3. Botre R, Sandbhor S. Using interactive workspaces for construction data utilization and coordination. *Int J Construction Eng Management* (2013) 2:62–9.

4. Alblawi A, Nawab M, Alsayyari A. A system engineering approach in orienting traditional engineering towards modern engineering. 2019 IEEE Global Engineering Education Conference (EDUCON). IEEE (2019). p. 1559–67.

5. Lank E, Thorley J, Chen S. An interactive system for recognizing hand drawn UML diagrams. Proceedings of the 2000 conference of the Centre for Advanced Studies on Collaborative research (2000), 7. doi:10.1145/782034.782041

6. Notowidigdo M, Miller RC. *Off-line sketch interpretation*. Arlington, VA: AAAI fall symposium (2004). p. 120–6.

7. Hammond T, Tahuti DR. A geometrical sketch recognition system for UML class diagrams. *SIGGRAPH Courses (ACM)* (2006) 25.

8. Alvarado C, Davis R. *SketchREAD: A multi-domain sketch recognition engine*, 34. San Diego, CA: ACM SIGGRAPH 2007 courses (2007).

9. Paulson B, Hammond T. PaleoSketch: Accurate primitive sketch recognition and beautification. *Proc 13th Int Conf Intell user Inter* (2008) 1–10.

10. Bresler M, Van Phan T, Prusa D, Nakagawa M, Hlavác V. Recognition system for on-line sketched diagrams. 2014 14th International Conference on Frontiers in Handwriting Recognition (2014), 563–8. doi:10.1109/ICFHR.2014.100

11. Bresler M, Prusa D, Hlaváác V. Online recognition of sketched arrow-connected diagrams. *Int J Doc Anal Recognit* (2016) 19:253–67. doi:10.1007/s10032-016-0269-z

12. Vesin B, Jolak R, Chaudron MR. Octouml: An environment for exploratory and collaborative software design. 2017 IEEE/ACM 39th International Conference on Software Engineering Companion (ICSE-C). IEEE (2017). p. 7–10.

13. Zhang X, Li X, Liu Y, Feng F. A survey on freehand sketch recognition and retrieval. *Image Vis Comput* (2019) 89:67–87. doi:10.1016/j.imavis.2019.06.010

14. Albore A, Hili N. From informal sketches to system engineering models using AI plan recognition: Opportunities and challenges. AAAI 2020 Spring Symposium Series (2020).

15. Hili N, Albore A, Baclet J. From informal sketches to systems engineering models using AI plan recognition. In: Lawless WF, Mittu R, Sofge DA, Shortell T, McDermott T, editors. *Systems engineering and artificial intelligence*. Springer (2021). p. 451–69.

16. Rosenfeld A, Richardson A. Explainability in human-agent systems. *Auton. Agents Multi-Agent Syst.* (2019) 33 (6):673–705. doi:10.1613/jair.2972

17. Richter S, Westphal M. The LAMA planner: Guiding cost-based anytime planning with landmarks. *J Artif Intell Res* (2010) 39:127–77. doi:10.1613/jair.2972

18. Bhowmik S, Sarkar R, Nasipuri M, Doermann D. Text and non-text separation in offline document images: A survey. *Int J Doc Anal Recognit* (2018) 21:1–20. doi:10.1007/s10032-018-0296-z

19. Wüest D, Seyff N, Glinz M. Flexisketch: A mobile sketching tool for software modeling. In: *International conference on mobile computing, applications, and services*. Springer (2012). p. 225–44.

20. MyScript. *Myscript home page* (2020). Available at: https://www.myscript.com/(Accessed 05 13, 2022).

21. Jordan MI, Mitchell TM. Machine learning: Trends, perspectives, and prospects. *Science* (2015) 349:255–60. doi:10.1126/science.aaa8415

22. Jain AK, Mao J, Mohiuddin KM. Artificial neural networks: A tutorial. *Computer* (1996) 29:31–44. doi:10.1109/2.485891

23. Sordo M. Introduction to neural networks in healthcare. In: *Open clinical: Knowledge management for medical care* (2002).

24. McNelis PD. *Neural networks in finance: Gaining predictive edge in the market*. Academic Press (2005).

25. Haddadi F, Khanchi S, Shetabi M, Derhami V. Intrusion detection and attack classification using feed-forward neural network. 2010 Second international conference on computer and network technology. IEEE (2010). p. 262–6.

26. Anderson JA. *An introduction to neural networks*. MIT press (1995).

27. Goldberg Y. Neural network methods for natural language processing. *Synth lectures Hum Lang Tech* (2017) 10:1–309. doi:10.2200/s00762ed1v01y201703hlt037

28. Ghallab M, Nau D, Traverso P. *Automated planning: Theory and practice*. Elsevier (2004).

29. Hollnagel E. Plan recognition in modelling of users. *Reliability Eng Syst Saf* (1988) 22:129–36. doi:10.1016/0951-8320(88)90070-1

30. Kautz HA, Allen JF. Generalized plan recognition. *Proc Fifth AAAI Natl Conf Artif Intelligence* (1986) 86:32–7.

31. Avrahami-Zilberbrand D, Kaminka G, Zarosim H. Fast and complete symbolic plan recognition: Allowing for duration, interleaved execution, and lossy observations. Proc. of the AAAI Workshop on Modeling Others from Observations. Edinburgh, Scotland, United Kingdom: MOO (2005).

32. Fikes RE, Nilsson NJ. Strips: A new approach to the application of theorem proving to problem solving. *Artif intelligence* (1971) 2:189–208. doi:10.1016/0004-3702(71)90010-5

33. Hansen EA, Zhou R. Anytime heuristic search. *J Artif Intell Res* (2007) 28: 267–97. doi:10.1613/jair.2096

34. Thayer JT, Ruml W. Faster than Weighted A*: An optimistic approach to bounded suboptimal search. *ICAPS* (2008) 355–62.

35. Likhachev M, Ferguson D, Gordon G, Stentz A, Thrun S. Anytime search in dynamic graphs. *Artif Intelligence* (2008) 172:1613–43. doi:10.1016/j.artint.2007.11.009

36. Bhatia A, Svegliato J, Zilberstein S. On the benefits of randomly adjusting anytime weighted a. *Proc Int Symp Comb Search* (2021) 12:116–20.

37. Ramírez M, Geffner H. Plan recognition as planning. In: *Twenty-first international joint conference on artificial intelligence* (2009).

38. McDermott D, Ghallab M, Howe A, Knoblock C, Ram A, Veloso M, et al. *PDDL-the planning domain definition language* (1998).

39. Edelkamp S, Hoffmann J. *PDDL2.2: The language for the classical part of the 4th international planning competition*. Tech. rep.. Freiburg im Breisgau, Germany: University of Freiburg (2004). p. 195.

40. Thiébaux S, Hoffmann J, Nebel B. In defense of PDDL axioms. *Artif Intelligence* (2005) 168:38–69. doi:10.1016/j.artint.2005.05.004

41. Helmert M. The fast downward planning system. *J Artif Intell Res* (2006) 26: 191–246. doi:10.1613/jair.1705

42. MDN Web Docs. Pointer events. Available at: https://developer.mozilla.org/en-US/docs/Web/API/Pointer_events (2021). Accessed: 2021-03-31.

43. Nair V, Hinton GE. In: Fürnkranz J Joachims T, editors. *Rectified linear units improve restricted Boltzmann machines*. Haifa, Israel: ICML (Omnipress) (2010). p. 807–14.

44. Glorot X, Bordes A, Bengio Y. Deep sparse rectifier neural networks. In: Gordon G, Dunson D, Dudík M, editors. *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, 15. Fort Lauderdale, FL, USA: PMLR (2011). p. 315–23.

45. Brooke J. SUS: A quick and dirty usability scale. *Usability Eval industry* (1996) 189.

46. Jolak R, Vesin B, Isaksson M, Chaudron MR. Towards a new generation of software design environments: Supporting the use of informal and formal notations with octouml. In: *HuFaMo@ MoDELS* (2016). p. 3–10.

47. Hili N, Farail P. BabyMOD, a collaborative model editor for mastering model complexity in MBSE. International Workshop on Model-Based Space Systems and Software Engineering (2020), 1–4.

TYPE Review
PUBLISHED 07 November 2022
DOI 10.3389/fphy.2022.951296

Check for updates

OPEN ACCESS

EDITED BY
William Frere Lawless,
Paine College, United States

REVIEWED BY
Michael Wollowski,
Rose-Hulman Institute of Technology,
United States
Yohei Katano,
Meiji University, Japan

*CORRESPONDENCE
Mito Akiyoshi,
mito.akiyoshi@gmail.com

SPECIALTY SECTION
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

RECEIVED 23 May 2022
ACCEPTED 24 October 2022
PUBLISHED 07 November 2022

CITATION
Akiyoshi M (2022), Trust in things: A review of social science perspectives on autonomous human-machine-team systems and systemic interdependence. *Front. Phys.* 10:951296.
doi: 10.3389/fphy.2022.951296

Trust in things: A review of social science perspectives on autonomous human-machine-team systems and systemic interdependence

Mito Akiyoshi*

Department of Sociology, Senshu University, Kawasaki, Japan

For Autonomous Human Machine Teams and Systems (A-HMT-S) to function in a real-world setting, trust has to be established and verified in both human and non-human actors. But the nature of "trust" itself, as established by long-evolving social interaction among humans and as encoded by humans in the emergent behavior of machines, is not self-evident and should not be assumed *a priori*. The social sciences, broadly defined, can provide guidance in this regard, pointing to the situational, context-driven, and sometimes other-than-rational grounds that give rise to trustability, trustworthiness, and trust. This paper introduces social scientific perspectives that illuminate the nature of trust that A-HMT-S must produce as they take root in society. It does so by integrating key theoretical perspectives: the ecological theory of actors and their tasks, theory on the introduction of social problems into the civic sphere, and the material political economy framework developed in the sociological study of markets.

KEYWORDS
machine, algorithm, artificial intelligence, interdependence, sociology, trust

1 Introduction

In this paper, Autonomous Human Machine Teams and Systems (A-HMT-S) are defined as teams that include humans and increasingly intelligent and autonomous machines working together [1, 2]. Intelligent machines are defined as machines or algorithms that think by scanning data for patterns, make inferences, and learn by testing inferences [3]. Advances in deep learning in the 21st century bring this emerging phenomenon closer to reality [1, 2, 4], though the idea of thinking machines was explored decades ago by Turing, Shannon, Weiner, Simon, and others, and was foreshadowed to an extent by Babbage's Difference Engines and Analytical Engine a century before that [5].

A key advance in the conceptualization of A-HMT-S is that intelligent machines are intended to operate as full-fledged team members collaborating with humans [4, 6]. Not only do they assist human decision-making and automate information processing, they also make decisions on their own and instruct human workers and other machines [7].

For example, an artificially intelligent co-worker named Charlie has been developed by Cummings et al. [4]. Charlie is designed to perform typical white-collar tasks: she gives interviews, takes part in brain-storming sessions, and collaborates in writing papers. But exhibiting recognizable and anthropomorphized agency or human-like identity, as Charlie does, is not central to the definition of A-HMT-S. They may not have the attractive features of *Blade Runner*'s replicants, but they have the kind of intelligence that would pass the Turing Test in the specific tasks to which they are assigned. They will also someday pass the "toilet test"—the ability to run unsupervised while humans address their bodily needs [8]. In short, the defining feature of intelligent machines that constitute A-HMT-S is that they can model human recognition, learning, and reasoning [3].

As our machine helpers become increasingly autonomous and intelligent, it leads to increasing interdependence between human and non-human actors. Although that evolution is far from complete and may never end, quasi-A-HMT-S with semi-autonomous machines are now commonplace. They diagnose and treat diseases [9], drive vehicles [1, 10], fly airplanes [11, 12], educate students, conduct research [4], trade stocks and derivatives [8], market products and services [13], fight wars [2], all with mixed results.

Algorithms lie at the core of these capabilities. In addition to their sheer ubiquity, their complexity and opacity and their anticipated consequences for humanity have motivated interdisciplinary research on societal effects of A-HMT-S [14, 15]. This paper is part of that interdisciplinary effort, addressing a crucial aspect of the implications of the integration of A-HMT-S into society: trust.

In traditional organizations, trust among workers is essential in achieving quality performance [16]. The increasing interdependence between humans and intelligent machines poses a series of trust-related questions: as machines become more autonomous, what are the causes and consequences of trust-building in A-HMT-S? What does it mean to trust non-human actors in a system? This paper uses a social-scientific toolkit to address these questions. In doing so, it might help to quickly look backward for a moment and consider the issue as it was faced by users of the earliest known human tools: handaxes. The user of a handaxe had to "trust" that its shape and material would be adequate to the task, which usually involved cutting into some kind of organic matter. Since the user probably also made the tool, she or he had an inbuilt basis for trusting it, including to trust that it wouldn't suddenly assume agentive power of its own and diverge from the user's goal, notwithstanding any animistic beliefs that might have been in play. The only other entities with "agency" in this scenario would have been other proto-humans, and the distribution of trust across the group would be established by longstanding social norms and rules. In short, the issue of trust was severable from other considerations, and its resolution was an intra-human one.

The history of technology since then has seen that simple allocation of trust be thoroughly complicated by the folding of more and more human capability into the tools themselves—at first physical and then mental [17]. A late medieval cannoneer had to trust the cannon wouldn't blow up in his face, but the location of that trustworthiness still resided in the cannon-maker. A Jacquard loom weaver, on the other hand, didn't have to place her trust in the card-maker because the output of the loom would reveal if the card-punching was accurate. A paddle-wheel steamboat passenger had to trust the boat and its crew, but might have known nothing about the mechanical steam engine governor that could be trusted (usually) to keep the engine speed steady. Today's automobile driver may only partially grasp the extent to which their survival depends on the trustability of dozens of microchips installed in the vehicle by factory workers who were overseeing relatively simple robots, which in turn had to be trusted to work right, with that chain of trust extending all the way back to the machines that designed the machines that designed them. Trust, once a human prerogative, is now diffused across multiple overlapping systems of systems. A-HMT-S is the inheritor of this long process.

But what is trust, and what makes an entity trustworthy? This paper accepts a widely agreed-upon definition of trust as the willingness of a trusting entity (the trustor) to be vulnerable to a trusted entity (the trustee) with respect to a pertinent domain, a trust object, against a backdrop of risk and uncertainty. Trust is therefore not a static thing but a constantly changeable relationship between actors, based on the assessment of each other's behavior in the relationship. One or both parties have just enough evidence to believe that the relationship will work out the way each of them expects it to [18–20]. Though fragile, it is an absolute, foundational basis of society. That is why Dante in his *Inferno* reserved the lowest circle of hell for people who have betrayed other people's trust. Trustworthiness, meanwhile, is a roughly quantifiable set of properties that the trustee in a relationship displays to the trustor to signal their intentions and probable behavior.

Each dimension of trust—trustor, trustee, and trust object—is expressed across a spectrum of generality ranging from the most particular to the most highly generalized [18]. For example, one terrible visit to a physician may imply the withdrawal of trust in that particular doctor, in the category of medical professional she or he represents (e.g., cardiology), or in the entire community of medical experts. Whether a particular visit results in the demise of trust at any level of generality depends on other pertinent variables.

From that starting point, this review will provide a synthesis of key social scientific thinking relevant to the question of trust within and between human and non-human actors. The next section reviews social scientific literature on interpersonal trust, which is compared with human-machine trust in the section after that. Empirical and experimental studies have shown that multiple factors including algorithmic transparency and machine error rates affect the level of trustworthiness that humans ascribe to intelligent machines [21]. But trust in A-HMT-S is not fully reducible to design issues; we will see that the broader context of interactions between A-HMT-S and other spheres of society is also relevant. In order to examine inter-system trust, the fourth section draws on the urban ecology tradition in sociology, as well as on research on the construction of social problems and the sociology of technology. But rather than introducing concepts in the abstract, it discusses specific incidents that involve precursors of A-HMT-S. By way of conclusion, this paper argues that the issue of trust in A-HMT-S is a specific case of the broader issue of trust in abstract systems and that as such, trust-building spans multiple social ecosystems and is supported or undermined by interactions among them.

2 Industrialization and the transmutation of trust

2.1 Interpersonal trust

Interpersonal trust is a linchpin of society. As discussed in Section 1, trust processes can be analyzed in terms of the trustor, the trustee, the trust object of varying generality degrees. Small-scale societies are characterized by particularized trust because interactions tend to be embedded in a local context [17]. Societies that are more complexly organized require coordination among actors we may not personally know; in such societies, reliance on general trust has become widespread and is essential to their continued existence [22]. In either case, trust depends on complex mutual understandings that defy easy definition [23]. This tacit and yet robust trust in others to do what a mesh of overt and latent rules dictates, and which makes the social order possible, is one major focus of ethnomethodology, the sociological and anthropological study of the rules by which people organize their everyday lives [24].

Interpersonal trust, in this perspective, operates on a provisional basis, and involves a sort of pattern-matching exercise. Confirming every datum imaginable and eliminating all alternate interpretative possibilities are neither possible nor called for unless the veracity of a person's explicit or tacit claim is called into question. A just-good-enough assessment of the situation suffices [24]. Thus, if someone who "looks like a college professor" enters a college classroom and approaches the podium, students assume that person is the course instructor and rarely ask for official proof of his or her identity. Additional elements of legitimation may appear in the form of references to the shared institutional structure that encompasses both the professor and the students—the topic of the course, the academic calendar, the grading system. As long as the behavior matches the observer's expectations in that setting, provisional trust will be satisfied.

We all do this a hundred times a day without even thinking about it. Social interaction is made possible by everyone's taking everyone else's claims at face value unless some contradictory evidence emerges that requires vetting [23]. The taken-for-granted nature of social life constitutes a cognitive and emotional common ground that is prior even to shared values and norms—things that are thought of as "culture" in the social scientific sense. Trust evolves over time in organizations through interactions that involves people's values, attitudes, and emotions [16].

Because interpersonal trustworthiness is not fully or even primarily grounded in the procedure of fact checking, societies vary widely in terms of the level of confidence people have about one another [25]. This is verifiable by looking at situations where it is lacking. For example, the mafia-type organized crime syndicates in southern Italy came into being as enforcers of contracts in a low-trust environment [26, 27]. Farmers who could not trust their counterparties in selling or buying produce and livestock had to turn to proto-mafiosi to guarantee transactions with threats of violence. Similarly, neighborhoods with high crime rates must invest heavily in security, and endure stressful anxiety, whereas individuals in low-crime areas can insouciantly leave their doors unlocked when they go out to run errands. The erosion of trust makes lives difficult. Until destroyed, the operation of trust tends to remain invisible, and yet trust is a public good from which other advantages such as cooperation, tolerance, functioning democracy, and market efficiency come about [16, 28].

2.2 Trust in machines and abstract systems

Industrialization extended the scope of trust relationships to include abstract systems [29]. Individuals and organizations in highly industrialized societies must learn to trust knowledge systems and technologies they do not fully grasp. Again, perfect grounding is precluded and faith is an integral dimension underlying trust. People board trains not knowing how the public transportation system is organized and operated, and they receive mRNA vaccines to protect themselves against viral infections without a detailed understanding of the immune system or vaccine manufacturing. Workers also learn, through trial and error, to trust machines they operate to mass produce goods and services. The threat of deskilling might be seen as a potential source of the erosion of trust in cases of automation, but Zuboff finds that workers adopt and adapt through explorative use of new technologies and achieve reskilling by becoming their adept and creative users [30].

In our capacity as consumers, too, we have entered a world where we buy things produced by distant others. The rise of advertising and branding is associated with this shift towards mass production, distribution, and consumption which Beniger has called "the control revolution" [17]. Advertising and branding are important where interpersonal trust cannot guarantee the quality of goods produced by large-scale organizations and sold anonymously. As Max Weber's celebrated analysis has shown, bureaucracy arises to enable the operation of such organizations by releasing trust from the domain of interpersonal relationships and the immediacy of face-to-face interaction, replacing it with formally defined rules and procedures and a hierarchy of offices [31].

3 Difficulties of building trust in A-HMT-S

Although trust in A-HMT-S has unique aspects, in principle the questions it raises are predictable extensions of the centuries-long process that preceded it [29]. Prior to the development of A-HMT-S, there were systems consisting of human operators and non-autonomous and non-intelligent machines and tools: vehicles, missile systems, nuclear power plants, and so on [1, 17]. I call these complex but non-intelligent tools "mundane systems" in contrast to A-HMT-S.

Technology scholars Hengstler, Enkel, and Duelli argue that trust in automated systems has two aspects: trust in the automation technology itself and trust in organizations that develop it, use it, or in which it is embedded [32]. However, in the case of trust in A-HMT-S, it is neither analytically tractable nor appropriate to separate the technology from its organizations and institutions. The literature on the sociology of technology has demonstrated the futility of treating a technology's capabilities without reference to its users and its context of use. According to the constructionist perspective of technology, there is no such thing as technology per se [33, 34]. The emergence of A-HMT-S reasserts that point with renewed exigency: in A-HMT-S, the technology implements, enacts, and embodies organizations' purposes and goals. Technology is the organization in a literal sense, and *vice versa*.

Shestakofsky conducted participant-observation research at a software firm and found that two types of labor were performed to create dynamic collaboration between humans and autonomous algorithms [35]. Computational labor addresses the issue of machine lag, problems posed by limitations of technologies. Human teams engage in repetitive information-processing tasks in order to fix gaps in software infrastructure. At the same time, emotional labor by human workers deals with human lag, clients' reluctance to use algorithms, and mediates the relationship between software systems and the latter. These findings suggest that trust among A-HMT-S actors is constructed in the course of collectively defining tasks and negotiating boundaries [35]. Jarrahi argues that human-AI symbiosis in organizational decision-making is possible when AI supplements human cognition and humans bring a holistic and intuitive approach in dealing with uncertainty [36].

A theoretical framework that addresses the issue of trust in A-HMT-S may be developed by treating the amalgam of non-humans and humans as-they-are. Studies have shown that human-to-machine trust is affected by various factors: the extent to which the machine exhibits human-like appearance, cognitive biases in general, automation-specific complacency and bias [37], algorithmic error rates, epistemic opacity, and the type of tasks [38]. Trustworthiness can be ascribed to intelligent machines and form a basis of productive collaboration in A-HMT-S, but the presence of biases and complacency means that humans can over-trust or under-trust intelligent algorithms and their decisions.

The problem with A-HMT-S is that it often involves "black box algorithms," epistemically opaque to human observers because they keep self-improving by testing and learning [9]. Opacity raises concerns among developers, users, and the general public. Lee and See, observing that trust is essential in the adoption of automation systems, recommends such measures as the disclosure of intermediate results of the algorithms to the operators and the simplification of algorithms [20]. Similarly, Burrell supports greater regulations, algorithmic transparency, and education of the public [9, 39]. The Defense Advanced Research Projects Agency (DARPA) attempted to address the opacity issue by developing "explainable artificial intelligence" [40]. Whether systems that "look" human, or visibly inserting actual humans into the decision loop, have any effect on trust and affinity is also investigated [41, 42]. It is important, though, to recall that the issue of trust in "black-box algorithms" is only among the latest developments in the history of trusting increasingly distant others and longer chains of factors.

Durán and Jongsma argue, using medical AI as a case study, that trust in black-box algorithms can be established by the principle of computational reliabilism (CR) [9]. Striving for algorithmic transparency, they claim, is a losing strategy because it defeats the purpose of deploying algorithms in the first place. "Transparency will not provide solutions to opacity, and therefore having more transparent algorithms is not a guarantee for better explanations, predictions and overall justification of our trust in the result of an algorithm" [9, p.331]. They suggest employing a version of the heuristic devices we use to assess the trustworthiness of our social interlocutors. In any given setting, CR assesses the trustworthiness of AI not by using interpretive parameters to check the system's inner state at points 1 through n, but by making multiple empirical inferences that turn out to be "good enough": A comparison with known solutions (verification), comparison with experimental data (validation), robustness analysis, a history of successful or unsuccessful implementation, and expert knowledge. An analogy with human interaction is to judge people by their behavior and set aside speculation about the mental processes that led to that behavior. Epistemological opacity does not have to be removed as long as CR can be established [9]. This enables users to take advantage of sophisticated black box analysis while solving the dilemma of being dependent on it without comprehending its workings.

This is particularly important for medical AI, but is applicable to other domains and to the question of building trust in non-AI abstract systems. It is similar to the satisficing that we saw in the college professor story earlier. Limited as we all are by bounded rationality [3, 43, 44], humans and organizations have to abandon the ideal of perfect explainability and treat the state of trust as provisional and dynamic. Yet for this very reason provisional trust is a fragile construct that can collapse if challenged by outsiders. And that is likely to happen at the border between A-HMT-S and other communities across the broader society with which it interacts. At that interface, CR may not help. To address the fact that heterogeneous actors scattered across heterogenous fields also will be asking themselves questions about the trustworthiness of A-HMT-S, and about the impact of A-HMT-S on their own interests, the next section turns to the ecological perspective originated in urban sociology.

4 A-HMT-S as an ecosystem

Establishing trust in A-HMT-S increasingly entails ethical as well as legal challenges, including transparency, algorithmic fairness, safety, security, and privacy. Challenges in jurisprudence emerge when non-human actors assume human-like characteristics. Scientific as well as practitioner knowledge systems engage in articulating goals and means in trust promotion and production [45]. Opening up black-box algorithms is often presented as a key to this undertaking. But as we have seen, perfect algorithmic transparency is not always feasible or effective. To identify and better understand trust goals relevant to A-HMT-S, an urban ecology perspective is useful. Urban ecology, a sociological perspective developed by scholars at the University of Chicago in the 1920s allows us to grasp the dynamic and emergent nature of the trustor and the trustee in interaction because it incorporates heterogenous actors and can incorporate A-HMT-S as a focus of trust processes. Borrowing its key metaphor and related concepts, such as territorial competition and inter-group cooperation, from biology, it sought to account for the ways different populations distributed themselves across the space of the city and used its resources. In that tradition, authors sometimes use the word "ecology" to describe what we conventionally understand by the term "ecosystem" [8, 46]. To avoid confusion, this paper will use that more conventional term. An ecosystem is an autonomous domain of actors, their tasks, and the relationship between actors and tasks [46]. It also includes the resources they obtain from the environment, and the other ways they interact with their surroundings. Territorial shifts of populations are seen in terms of invasion and ecological succession or the replacement of one group by another. For example, residential patterns of immigrants to major cities in the United States at the turn of the 20th century were determined by their place of work—often in the central business district—, as well as by their material means, and their social distance from native populations. Neighborhoods that had seen the arrival of immigrants experienced an exodus of middle-class families; the new groups further affected the types of businesses and services in these transitioning neighborhoods. The distribution of populations and differentiation of space are subjected to the process of interaction among diverse groups.

At this level of analysis, we can think of whole ecosystems as units of interaction. A-HMT-S researchers, developers, and popularizers constitute one such ecosystem. For people outside it to trust "what the machines are doing," they have to trust or at least tolerate the ecosystem as a whole, including the motivations and behavior of the humans, the type and amount of environmental resources it uses and the way it uses them. Outsiders have to satisfy themselves that none of this poses a threat to their individual and collective livelihood or to how they understand the world and act in it. And they have to figure out how to minimize friction at the interface between their own ecosystem and that of the newcomer. As was mentioned earlier, achieving and keeping a state of trust will bring both cognitive and emotional dimensions into play, and the benchmark will tend to be: How well does this new ecosystem play by the taken-for-granted rules of everyday life [24]?

In the case of medical A-HMT-S, for instance, in order to take root in day-to-day medical practice it has to build trust relationships with patients, regulators, healthcare providers, insurance providers, and the general public. Computational reliabilism may be a necessary but not sufficient condition for that, as each party may judge the situation by different criteria. Physicians may be most concerned with diagnostic accuracy while insurance providers may worry over the cost-benefit issues and hospital technicians may care about fitting new practices into existing routines. If we recall that trust is a relation of varying generality as discussed in Section 1, then highly particularized trust in a trust object does not entail trust in a category or ecosystem of which that trust object is an instantiation. A particularized trust object is in fact a construct of multiple ecosystems. Society-wide trust in A-HMT-S is thus a constant balancing act. And as we will see in a later section of this paper, it can be lost when a failure occurs and the system as a whole does not engage in trust-repairing behavior addressed collectively to people living and working in other ecosystems.

Mackenzie, drawing on Abbott, used the ecosystemic perspective in a study of the rise of High-Frequency Trading and its relation to existing trading and regulatory systems [8]. His research reveals the ripple effect of technological decisions as they impinge on the interests of other domains. HFT is a type of A-HMT-S made possible by machines that can analyze opportunities and execute orders at a speed that surpasses that of human-only teams. Because of this advantage, HFT firms quickly became major players in their respective markets. In the process, they generated enormous profits by engaging in legal but arguably unscrupulous trading activities, made possible only by the high-speed of their systems. Then, in a move apparently unrelated to what the HFTs were up to, the New York Stock Exchange decided to install a new communication antenna on the roof of its data center. Available to any member who paid the requisite hefty fee, the antenna would provide a half-microsecond improvement in transaction time by eliminating 260 m of fiber optic cable from the transmission line. This was exactly the sort of time difference the HFTs had been exploiting through their proprietary technology, and now their advantage was threatened.

As a prelude to explaining what ensued, MacKenzie revisits an insurrection that took place in the English community of St. Albans in the late 14th century [8]. As part of a general wave of uprisings against feudalism, townspeople invaded the local Benedictine monastery and, after freeing people held in the monastery's prison, entered the abbot's parlor, methodically smashed its stone-paved floor, and carried pieces of it away with them. This seemingly random act was in fact retaliation for a previous abbot having confiscated the townspeople's millstones 50 years earlier and used the confiscated stones to pave the parlor floor. The motive for that had been to achieve a monastic monopoly over grain-milling and extract the consequent fees. Townsfolk never forgot this, which exemplifies a key point MacKenzie wants to emphasize: even seemingly minor changes in available technology are not neutral but are usually bound up in power relations with long-lasting effects.

Back in the 21st century New York, the new antenna plan had similar consequences that drew in multiple institutional spheres—which MacKenzie refers to as "ecologies." Eventually, the Securities and Exchange Commission, a local zoning board, residents of the town where the data center is located, the Stock Exchange itself, and others found themselves in conflict over something which had seemed like a simple technology decision: eliminating 260 m of fiber. The eventual solution once again exemplifies the ways in which a material consideration can be waylaid by issues of power: as of 2020, it had been decided to reinsert the half-microsecond delay by adding a coil of cable to the transmission line, thereby returning everything to the status quo ante.

Mackenzie's point is generalizable. Just as biological populations compete for habitat and resources, different social actors behaving collectively will interact to create an observed distribution of functions (tasks that need to be executed for the maintenance of order) and habitats within and between ecoystems. Interactions will define actors and the nature of their tasks; what gets done, and who does it, are not rigidly defined by pre-existing functions [46]. Instead, turf battles for resources and legitimacy dynamically shape the things actors do and don't do, in a manner that social scientists call "co-constitutive" and that other disciplines might term "emergent." Squabbling over a length of fiber optic cable, and expropriating a paving stone, can be inexplicable outside of a specific social, political and economic context that makes them highly meaningful.

The rapid growth of A-HMT-S capabilities and governmental attempts to control that process is another part of this story of ecosystems squaring off against one another. Whether unfettered development is encouraged or restrained is a function of interactions among the affected ecosystems. Lethal autonomous weapons systems (LAWS) provide a good example [2]. They will proliferate in a society if other ecosystems that interact with it invest in and legitimize their development, but will be suppressed in any society where the state reins in the military deployment of A-HMT-S.

The above examples show that when A-HMT-S is deployed it can trigger social effects across multiple domains. In the labor market, it can result in job creation, job elimination, or both. In the political domain, it can produce a crisis among regulators and legislators. Pfeffer addresses such broader implications in a study of the impact of AI on the economy and workers' well-being [47]. He points out that the introduction of A-HMT-S can have detrimental effects on workers by eliminating jobs and forcing some workers to switch occupational categories, many of whom already experience stagnant wages and job precarity. Low fertility, government budget deficits, and runaway debt in many highly industrialized societies mean that public policy interventions to attenuate the negative labor market impacts of A-HMT-S are unlikely. A-HMT-S can be used to promote human well-being, but Pfeffer observes that they are as likely to be used in ways that exacerbate economic inequities [47]. If workers come to regard A-HMT-S as a tool to make themselves redundant, computational reliabilism will probably not help them trust it.

The expanding use of A-HMT-S will also force revisions of school curricula, similar to the way basic computer skills became a key subject in the final decades of the 20th century [48]. One can envisage a future in which students are required to learn how to work with A-HMT-S to optimize learning. The ecosystemic perspective helps us understand the complex nature of systems interacting with their environments; it enables us to see that what seems external to systems themselves are in fact constitutive of their functions. Adjacent ecosystems regulate, offer incentives and resources, call for accountability, and do many other things that can influence the success of A-HMT-S.

In terms of its effects on human activity, A-HMT-S is more than the automation or translation of tasks formerly performed by humans. It leads to the emergence of new tasks to address the challenges that it and other ecosystems present to each other as they each seek to thrive in the world they must share. In the course of building explainable systems, A-HMT-S must also explain itself to any audience whose activities could be upended by it. At first glance, it may have seemed strange that Pfeffer's paper on the effects of AI has data on fertility, national deficits and debts, but the ecosystemic perspective motivates such a focus on a nexus of multiple spheres [47].

5 How technological systems can breach trust

Prior to the development of A-HMT-S, there were many systems made up of human operators and non-autonomous and non-intelligent machines and tools: vehicles, missile systems, nuclear power plants, and so on. I referred earlier to these non-intelligent tools as “mundane systems” in contrast to A-HMT-S. Mundane systems have a track record of breaching the trust of their users and the general public. The way they fail illuminates the kind of trust issues that A-HMT-S may face going forward.

5.1 Mundane system trust erosion: Three brief examples

Drunk driving: Car accidents caused by drunk drivers, and the public discourse surrounding them, remind us that the accepted narrative of interdependence between driver, car, and environment is only one of several potential ways to constellate the relevant elements. Typically, when an accident happens the drunk driver is designated as the “cause” and becomes the target of moral opprobrium. Alternate reasonings are possible but rarely accepted in what Gusfield calls the public drama of social problems [49]. The lack of public transportation to venues that serve alcohol, or the mingling of cars and pedestrians on the same thoroughfares, could be conducive to accidents caused by drunk driving, and yet poor urban planning is rarely singled out as a cause. Car manufacturers are not held accountable for building vehicles that can kill regardless of what mental state the operator is in. The underlying assumption regarding the interdependence of the driver, the car, and the streets is that the driver should be a morally upstanding individual who exercises prudence and is capable of controlling their own behavior. The presence of accidents caused by sober but incompetent drivers indicates that the association between behavior and morality involves the choice of a certain perspective.

Titan II missile explosion: In 1980, a Titan II intercontinental ballistic missile at a missile complex in Damascus, Arkansas was damaged when a worker accidentally dropped a wrench socket down its silo during a routine check of the oxidizer tank pressure, which caused a fuel leak [50]. The fuel exploded the following day, resulting in one death and multiple injuries. The interdependence of humans and non-intelligent machines can go awry without moral failure by the humans. The coexistence of the worker, the socket, and the vulnerable tank surface led to the explosion.

Fukushima Daiichi Nuclear Power Plant failure: After the East Japan Earthquake of 2011, the resulting tsunami hit the Fukushima Daiichi Nuclear Power Plant and its reactor cooling system failed. This led to reactor meltdowns, explosion and the atmospheric release of radioactive material [51]. A nuclear plant is an example of a mundane system. Even though the plant uses multiple machines and robots, they are not autonomous or intelligent. In the case of the Fukushima Daiichi Nuclear Powerplant, it turned out that TEPCO, the plant operator, and other related organizations had underestimated the risk of losing reactor cooling after a tsunami. Some seismologists familiar with the region’s earthquake and tsunami history had warned that a cooling system failure due to major tsunami was possible, but those warnings were not heeded [52]. The interdependence between humans and the plant was disrupted not by a gap intrinsic in their relationships—both humans and the plant were executing tasks assigned to them—but by TEPCO management’s decision years earlier to ignore evidence of a serious environmental risk.

As these cases illustrate, the interdependence of elements in mundane systems can be eroded by various factors. The misplacement of trust may only become evident ex post. Drunk drivers should not be trusted to drive safely and yet there is currently no scalable solution to prevent them from getting behind the wheel. The missile fuel tank was not designed to withstand the damage caused by a falling wrench socket, and it was never anticipated that a worker might drop a socket inside the silo. The Fukushima Daiichi Power Plant was supposed to have been built on safe ground and the risk of earthquake and tsunami was believed to be manageable, because the scientists who had warned of potential damage to the cooling system were considered an untrusted minority.

Being systems comprised of human and non-human actors, and operating among other groups and systems with their own idiosyncrasies, A-HMT-S could fail in the same ways mundane systems do: lack of fail-safe mechanisms, human error, poor coordination between actors. However, they can fail in ways unique to them because they have two types of intelligence: human intelligence and machine intelligence. Some further examples will illustrate this.

5.2 Two cases of failure in systems that are “A-HMT-S-adjacent”

Boeing 737 Max: Two crashes of this Boeing model were caused by some pilots’ inability to interact correctly with software that had been implemented to compensate for certain stall conditions [11, 12]. Optimistically named Maneuvering Characteristics Augmentation System (MCAS), the software conflicted with human pilots’ judgement and behavioral habits acquired over years of flying previous 737s. A 737 Max without MCAS tends to nose upward in flight because of its large engines placed high on the wing. A nose-up condition can trigger a stall, which is a bad thing for an aircraft. MCAS identifies some conditions under which it automatically forces the nose downward. In the case of the two accidents, pilots who didn't know why the plane was suddenly dipping its nose reacted incorrectly and set in motion a sequence of events that led to tragedy.

But why place the engines so high? Because more efficient engines have larger diameter than less efficient ones, and to prevent them from scraping against the ground, they had to be positioned higher on the wing than the engines on earlier 737s. This higher placement compensates for the fact that the plane's landing gear struts are short, which was a design decision made in the 1960s to make the 737 cargo bay accessible at smaller airports that lacked a full complement of loading equipment, and that design factor was never changed through many decades. A long chain of design and performance decisions, and several hundred deaths, resulted arguably from that single criterion. This also means that redesigning the wing and engine was not even possible without many other changes that would turn it into a completely new plane, requiring a lengthy and costly certification process with multiple regulatory agencies involved. Once Boeing decided to "re-engine" the 737, a software fix was the only option to compensate for the awkward aerodynamics of the high-mounted engines. Boeing vigorously lobbied with regulators to allow the design changes without fully sharing details with airline companies or pilots [11]. Pilots were not informed about the existence, much less the operation, of MCAS; in the case of the two fallen planes they had not received simulator training to work with the software.

Boeing 737 Max can be regarded as a precursor to full-fledged A-HMT-S. Humans are on the loop rather than in the loop [2]. When they are not given authority to intervene when software made a faulty move, or when they aren't sure how to react to a machine decision, the entire system fails catastrophically.

ShotSpotter: ShotSpotter uses specially designed microphones, AI, and human analysts to detect and geolocate gunshots. It claims to offer precision policing solutions to detect crimes and protect lives. In May 2020, based on evidence from this gunfire detection system, a Chicago man named Michael Williams was accused of shooting a neighbor. Forensic reports prepared by ShotSpotter employees established his culpability. After he had been in jail for nearly a year, a judge decided the evidence against him was too weak and the case was dismissed. Williams claims he was giving a ride to the victim when that person was shot by someone else [53].

As is the case with human interactions, human-machine systems must earn the trust of those with whom they interact. With the ShotSpotter case and the 737 Max disasters, these systems that are on the road to A-HMT-S may not deserve anything more than a skeptical and provisional assessment of trustworthiness. Trust in mundane systems and A-HMT-S are both examples of trust in abstract systems, which is always potentially fraught with suspicion and competing claims [29]. What is distinct about trust in A-HMT-S granted by outside actors such as the media and the political system is that it involves trust in decisions made by autonomous and intelligent machines [1, 2, 4, 7, 39]. When high-stakes decisions such as making a criminal accusation or flying an airplane are made by A-HMT-S and then turn out to be wrong, trust will naturally erode.

But A-HMT-S are not solely responsible for their ability to achieve societal trust. Other ecosystems can enhance or suppress the likelihood of it. For example, Muehlematter and Vokinger recommend that one way to improve public trust in artificial-intelligence and machine-learning-based medical devices is to increase transparancy regarding their regulation and approval. Currently, there is an unexplained gap in the timing of approval of devices commonly approved in the United State and Europe [54].

A breach in trust could also set off what Alexander called the "societalization" of A-HMT-S [55]. Societalization happens when long-enduring problems cease to be internal to a given ecosystem (in the usage we employed earlier) and are redefined as a general crisis in the public sphere. Media play the role of agenda-setter with increased and detailed coverage [55]. Investigative reporting of dramatic cases cracks them open for public discourse and denunciation. The societalization process may trigger regulatory intervention, but that will depend on whether politicians perceive that what is at stake is aligned with their own interests: another example of different ecosystems interacting at the boundary of their respective domains [46].

The 737 Max disasters and the erroneous prosecution with ShotSpotter data foreshadow what the societalization of A-HMT-S might look like. General public trust in A-HMT-S will have to be actively produced and continuously maintained if A-HMT-S is to achieve the hoped-for synergy of humans and autonomous machines. The current backlash against documented instances of biased algorithms shows the consequences of failing to secure such trust [39, 56–58]. In 2020, a computer algorithm was used to determine grades for the General Certificate of Secondary Education and A-level qualification in the United Kingdom when exams were cancelled due to the COVID-19 pandemic. The algorithm was found to disproportionately and systematically suppress the grades of students from disadvantaged backgrounds because it used the historical grade distribution at the school level to weight the grades of individual students [59]. Faced with a nationwide controversy, the algorithmically-generated grades were eventually replaced with alternative grades that integrated teachers' assessments. The emergent A-HMT-S deservedly failed to earn the trust of the public.

This section has focused on challenges involved in building trust in A-HMT-S, using cases that revealed design or deployment gaps. Of course, there are also cases in which human and non-human actors successfully achieve fully collaborative participation. In some such cases, non-human actors acquire their own agency equivalent to that of human actors and cease to be a mere assistant to the human actors [2, 4].

6 Conclusion

This paper reviewed the social scientific literature that illuminates our understanding of issues regarding trust in A-HMT-S. In research on AI and trust, establishing trust is often presented as a matter of algorithmic transparency above all [39]. Since A-HMT-S can inadvertently incorporate existing forms of inequality and discrimination, improving algorithmic transparency is certainly a key challenge. At the same time, the present review offers a broader context. The taken-for-granted nature of interpersonal trust among humans suggests some of the ground that human-machine systems will have to cover in order to display trustworthiness, and to achieve and maintain relationships of trust [8, 23, 24]. Anthropomorphizing interfaces and developing explainable AI are attempts to achieve trust within the ecosystem of A-HMT-S. But those things alone will probably not be enough to curtail skepticism on the part of people outside that ecoystem. Skepticism is not a luddite reaction. Rather, it is a predictable caution about the effects that A-HMT-S can have on well-being of those whose lives and livelihoods may be touched by them [47, 59]. A-HMT-S researchers and developers' engagement with the labor market, academia, mass media and other domains will contribute importantly to the goal of securing trust about technologies that are not fully explicable and yet lead to highly consequential outcomes.

Author contributions

MA is solely responsible for the entire contents of the article.

Acknowledgments

I wish to thank Gerald Lombardi and William Lawless, and two independent reviewers for their insights and helpful comments.

Conflict of interest

The author declares that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

The author declares a past co-authorship with the handling editor WL.

The handling editor declared a past co-authorship with one of the authors MA.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

1. Lawless WF. Toward a physics of interdependence for autonomous human-machine systems: The case of the uber fatal accident, 2018. *Front Phys* (2022) 10: 879171. doi:10.3389/fphy.2022.879171

2. Lawless WF, Mittu R, Sofge DA, Shortell T, McDermott TA. Introduction to "systems engineering and artificial intelligence" and the chapters. In: WF Lawless, R Mittu, DA Sofge, T Shortell, TA McDermott, editors. *Systems engineering and artificial intelligence [internet]*. Cham: Springer International Publishing (2021).

3. Frantz R. Herbert Simon: Artificial intelligence as a framework for understanding intuition. *J Econ Psychol* (2003) 24(2):265–77. doi:10.1016/s0167-4870(02)00207-6

4. Cummings P, Schurr N, Naber A, Charlie SD. Recognizing artificial intelligence: The key to unlocking human AI teams. In: WF Lawless, R Mittu, DA Sofge, T Shortell, TA McDermott, editors. *Systems engineering and artificial intelligence [internet]*. Cham: Springer International Publishing (2021).

5. Gleick J. *The information: A history, a theory, a flood*. New York: Vintage Books (2012). p. 526.

6. Jiang W, Fischer JE, Greenhalgh C, Ramchurn SD, Wu F, Jennings NR. Social implications of agent-based planning support for human teams. International Conference on Collaboration Technologies and Systems (2014). p. 310.

7. Lee MK. Understanding perception of algorithmic decisions: Fairness, trust, and emotion in response to algorithmic management. *Big Data Soc* (2018) 5(1): 205395171875668. doi:10.1177/2053951718756684

8. MacKenzie D. *Trading at the speed of light: How ultrafast algorithms are transforming financial markets*. Princeton, NJ: Princeton University Press (2021). p. 290.

9. Durán JM, Jongsma KR. Who is afraid of black box algorithms? On the epistemological and ethical basis of trust in medical AI. *J Med Ethics* (2021) 47(5): 329. doi:10.1136/medethics-2020-106820

10. Panagiotopoulos I, Dimitrakopoulos G. An empirical investigation on consumers' intentions towards autonomous driving. *Transportation Res C: Emerging Tech* (2018) 95:773–84. doi:10.1016/j.trc.2018.08.013

11. Robison P. *Flying blind: The 737 MAX tragedy and the fall of boeing*. New York: Doubleday (2021). p. 336.

12. Mongan J, Kohli M. Artificial intelligence and human life: Five lessons for radiology from the 737 MAX disasters. *Radiol Artif Intelligence* (2020) 2(2): e190111. doi:10.1148/ryai.2020190111

13. Ameen N, Tarhini A, Reppel A, Anand A. Customer experiences in the age of artificial intelligence. *Comput Hum Behav* (2021) 114:106548. doi:10.1016/j.chb.2020.106548

14. Liu Z. Sociological perspectives on artificial intelligence: A typological reading. *Sociol Compass* (2021) 15(3):e12851. doi:10.1111/soc4.12851

15. Rahwan I, Cebrian M, Obradovich N, Bongard J, Bonnefon JF, Breazeal C. Machine behaviour. *Nature* (2019) 568(7753):477–86. doi:10.1038/s41586-019-1138-y

16. Jones GR, George JM. The experience and evolution of trust: Implications for cooperation and teamwork. *Acad Manage Rev* (1998) 23(3):531–46. doi:10.5465/amr.1998.926625

17. Beniger JR. *The control revolution: Technological and economic origins of the information society*. Cambridge, Mass: Harvard University Press (1986). p. 508.

18. Schilke O, Reimann M, Cook KS. Trust in social relations. *Annu Rev Sociol* (2021) 47(1):239–59. doi:10.1146/annurev-soc-082120-082850

19. Mayer RC, Davis JH, Schoorman FD. An integrative model of organizational trust. *Acad Manage Rev* (1995) 20(3):709–34. doi:10.5465/amr.1995.9508080335

20. Lee JD, See KA. Trust in automation: Designing for appropriate reliance. *Hum Factors* (2004) 46(1):50–80. doi:10.1518/hfes.46.1.50.30392

21. Robinette P, Howard AM, Wagner AR. Effect of robot performance on human–robot trust in time-critical situations. *IEEE Trans Hum Mach Syst* (2017) 47(4):425–36. doi:10.1109/thms.2017.2648849

22. Simmel G. *The philosophy of money*. London: Routledge (2004). p. 616.

23. Goffman E. *The presentation of self in everyday life*. Garden City, New York: Doubleday & Company (1959). p. 259.

24. Garfinkel H. *Studies in ethnomethodology*. Cambridge, UK: Polity (1991). p. 304.

25. Ward PR, Mamerow L, Meyer SB. Interpersonal trust across six Asia-Pacific countries: Testing and extending the 'high trust society' and 'low trust Society' theory. *Plos One* (2014) 9(4):e95555. doi:10.1371/journal.pone.0095555

26. Dickie J. *Cosa nostra: A history of the Sicilian mafia*. London: Hodder & Stoughton (2004). p. 483.

27. Gambetta D. *The Sicilian mafia: The business of private protection*. Cambridge, Mass: Harvard University Press (1993). p. 335.

28. Axelrod RM. *The evolution of cooperation*. New York: Basic Books (1984). p. 241.

29. Giddens A. *Modernity and self-identity: Self and society in the late modern age*. Stanford: Stanford University Press (1991). p. 256.

30. Zuboff S. *The age of the smart machine: the future of work and power*. New York: Basic Books (1988). p. 468.

31. Weber M. *Economy and society*. Cambridge, Mass: Harvard University Press (2019). p. 504.

32. Hengstler M, Enkel E, Duelli S. Applied artificial intelligence and trust—the case of autonomous vehicles and medical assistance devices. *Technol Forecast Soc Change* (2016) 105:105–20. doi:10.1016/j.techfore.2015.12.014

33. Latour B. *Reassembling the social: An introduction to actor-network-theory*. Oxford: Oxford University Press (2005). p. 301.

34. Grint K, Woolgar S. *The machine at work: Technology, work and organization*. Cambridge, UK: Blackwell Publishers (1997). p. 199.

35. Shestakofsky B. Working algorithms: Software automation and the future of work. *Work Occup* (2017) 44(4):376–423. doi:10.1177/0730888417726119

36. Jarrahi MH. Artificial intelligence and the future of work: Human-AI symbiosis in organizational decision making. *Bus Horiz* (2018) 61(4):577–86. doi:10.1016/j.bushor.2018.03.007

37. Parasuraman R, Manzey DH. Complacency and bias in human use of automation: An attentional integration. *Hum Factors* (2010) 52(3):381–410. doi:10.1177/0018720810376055

38. Salem M, Lakatos G, Amirabdollahian F, Dautenhahn K. Would you trust a (faulty) robot? Effects of error, task type and personality on human-robot cooperation and trust. In: 2015 10th ACM/IEEE International Conference on Human-Robot Interaction (2015). p. 1–8.

39. Burrell J. How the machine 'thinks': Understanding opacity in machine learning algorithms. *Big Data Soc* (2016) 3(1):205395171562251. doi:10.1177/2053951715622512

40. Gunning D, Aha D. DARPA's explainable artificial intelligence (XAI) program. *AI Mag* (2019) 40(2):44–58. doi:10.1609/aimag.v40i2.2850

41. Ullman D, Malle BF. Human-Robot trust: Just a button press away. In: Proceedings of the Companion of the 2017 ACM/IEEE International Conference on Human-Robot Interaction [Internet]. New York, NY, USA: Association for Computing Machinery.

42. Zlotowski J, Sumioka H, Nishio S, Glas D, Bartneck C, Ishiguro H. Persistence of the uncanny valley: The influence of repeated interactions and a robot's attitude on its perception. *Front Psychol* (2015). doi:10.3389/fpsyg.2015.00883

43. Simon H. Theories of bounded rationality. In: *Models of bounded rationality: Behavioral economics and business organization*. Cambridge, Mass: MIT Press (1982). p. 408–23.

44. Simon H. *Administrative behavior: A study of decision-making processes in administrative organizations*. New York: Free Press (1997). p. 368.

45. Rodrigues R. Legal and human rights issues of AI: Gaps, challenges and vulnerabilities. *J Responsible Tech* (2020) 4:100005. doi:10.1016/j.jrt.2020.100005

46. Abbott A. Linked ecologies: States and universities as environments for professions. *Sociol Theor* (2005) 23(3):245–74. doi:10.1111/j.0735-2751.2005.00253.x

47. Pfeffer J. The role of the general manager in the new economy: Can we save people from technology dysfunctions? (2008). [Internet] 2018 [cited May 22, 2022] Stanford Graduate School of Business Working Paper No. 3714. Available from: https://www.gsb.stanford.edu/faculty-research/working-papers/role-general-manager-new-economy-can-we-save-people-technology.

48. Rafalow MH. Disciplining play: Digital youth culture as capital at school. *Am J Sociol* (2018) 123(5):1416–52. doi:10.1086/695766

49. Gusfield JR. *The culture of public problems: Drinking-driving and the symbolic order*. Chicago: University of Chicago Press (1984). p. 278.

50. Schlosser E. *Command and control: Nuclear weapons, the Damascus accident, and the illusion of safety*. New York: The Penguin Press (2013). p. 632.

51. Whitton J, Parry IM, Akiyoshi M, Lawless W. Conceptualizing a social sustainability framework for energy infrastructure decisions. *Energy Res Soc Sci* (2015) 8:127–38. doi:10.1016/j.erss.2015.05.010

52. Ishibashi K. Genpatsu shinsai: Hametsuwo sakeru tameni. *Kagaku* (1997) 67(10):720–4.

53. Stanley J. *Four problems with the ShotSpotter gunshot detection system*. News & Commentary [Internet]. New York: American Civil Liberties Union (2021).

54. Muehlematter UJ, Daniore P, Vokinger KN. Approval of artificial intelligence and machine learning-based medical devices in the USA and Europe (2015–20): A comparative analysis. *Lancet Digit Health* (2021) 3(3):e195–203. doi:10.1016/s2589-7500(20)30292-2

55. Alexander JC. The societalization of social problems: Church pedophilia, phone hacking, and the financial crisis. *Am Sociol Rev* (2018) 83(6):1049–78. doi:10.1177/0003122418803376

56. Köchling A, Wehner MC. Discriminated by an algorithm: A systematic review of discrimination and fairness by algorithmic decision-making in the context of HR recruitment and HR development. *Bus Res* (2020) 13(3):795–848. doi:10.1007/s40685-020-00134-w

57. O'Neil C. *Weapons of math destruction: How big data increases inequality and threatens democracy*. Reprint edition. New York: Crown (2016). p. 288.

58. Nowotny H. *AI we trust: Power, illusion and control of predictive algorithms*. Cambridge, UK: Polity (2021). p. 190.

59. Waller M, Waller P. *Why predictive algorithms are so risky for public sector bodies*. [Internet]. Rochester, NY: Social Science Research Network (2020).

TYPE Original Research
PUBLISHED 07 November 2022
DOI 10.3389/fphy.2022.958450

OPEN ACCESS

EDITED BY
William Frere Lawless,
Paine College, United States

REVIEWED BY
Peter Beling,
Virginia Tech, United States
Ali Raz,
George Mason University, United States

*CORRESPONDENCE
Tom McDermott,
tmcdermott@
stephensontechnologies.org

SPECIALTY SECTION
This article was submitted to
Interdisciplinary Physics,
a section of the journal
Frontiers in Physics

RECEIVED 31 May 2022
ACCEPTED 20 October 2022
PUBLISHED 07 November 2022

CITATION
McDermott T and Folds D (2022),
Construal level theory in the design of
informational systems.
Front. Phys. 10:958450.
doi: 10.3389/fphy.2022.958450

Construal level theory in the design of informational systems

Tom McDermott* and Dennis Folds

Stephenson Technologies Corporation, Baton Rouge, LA, United States

In the context of human-machine teaming, we are observing new kinds of automated and "intelligent" applications that effectively model and manage both producer and consumer aspects of information presentation. Information produced by the application can be easily accessed by the user at multiple levels of abstraction, depending on the user's current context and necessity. The research described in this article applies this concept of information abstraction to complex command and control systems in which distributed autonomous systems are managed by multiple human teams. We explore three multidisciplinary and foundational concepts that can be used to design information flow in human-machine teaming situations: 1) formalizing a language we call "RECITAL" (Rules of Engagement, Commander's Intent, and Transfer of Authority Language), which defines the information flow based on concepts of intent, rules, and delegated authority; 2) applying this language to well-established models of human-machine distributed teams represented as a systemic control hierarchy; and 3) applying construal level theory from social psychology as a means to guide the producer-consumer model of the information abstractions. All three of these are integrated into a novel user interface concept designed to make information available to both human and machine actors based on task-oriented decision criteria. In this research, we describe a conceptual model for future information design to inform shared control and decision-making across distributed human and machine teams. We describe the theoretical components of the concepts and present the conceptual approach to designing such systems. Using the concept, we describe a prototype user interface to situationally manage the information in a mission application.

KEYWORDS

distributed autonomy, construal level theory, user interface, hierarchical control systems, commanders intent

1 Introduction

Future command and control (C2) systems will feature sophisticated software capabilities with varying degrees of autonomy. In some applications these systems will work closely with humans; in others, operations will be largely autonomous. In the near future, distributed teams of humans will likely work with distributed teams of autonomous systems, with relationships that can change dynamically. Evolution of effective human command and control is crucial to the success of future autonomous systems, robots, artificial intelligence (AI), and embedded machine intelligence. The increased reliance on these intelligent technologies creates challenges and opportunities to improve C2 functions, particularly operational situational awareness, to better realize operator and leadership intent. The opportunity of particular interest is to develop better ways for humans to formally express rules, intent, and related decision authorities when interacting with intelligent machines and other humans. This includes the initial creation and subsequent editing of this information, its strategic and tactical use in the system, and the monitoring of performance during testing, training, and operations.

When humans communicate with one another in complex machine-aided tasks, the machines provide a combination of natural language (visual and aural), graphs, spatial maps or three-dimensional (3D) renderings, and other images. Comprehension of these elements is partly dependent on the training, experience, and general knowledge of the people involved. Machines are improving their ability to understand and communicate across these different media using machine learning (ML) and related technologies. Even if the natural language and image processing capabilities of machines greatly improve, such machines are not expected in the near term to possess the faculty to understand nuances of context, history, and unspoken contingencies in a manner equivalent to trained and experienced humans. These nuances are difficult enough for human to human communication and are often managed by formalizing and training hierarchical communication concepts and general language structures to coordinate activities. Using concepts from hierarchical communication models (specifically command and control models) we can define and create a language that will be used when humans interact with and through such machines. As with human to human communication, this language will necessarily exist at multiple abstraction levels and be comprised of some (constrained) natural language, annotated maps or renderings, graphs and equations, and images. The language will necessarily be rooted in defined data structures and service definitions and will require human-machine interfaces to support creating, editing, querying, and monitoring functions.

2 The RECITAL language and an example

The structure of distributed human-machine teams can be viewed as a control hierarchy. In a complex control hierarchy, some of the operations are explicitly defined and some are left to interpretation. In human enterprises hierarchical control is often guided by formal and semi-formal expressions of rules, intent, and decisional authority to act. In the military, these expressions are formally defined as Rules of Engagement (RE), Commanders Intent (CI), and Transfer of Authority (TA). In the evolution of human-machine teams, this Language does not yet exist. We call the language "RECITAL" using the nested acronym "RE-CI-TA-Language."

The primary objective of this research is to define the data, services, and user interfaces needed for humans to create, edit, query, and comprehend expressions of complex operational tasks such as rules, intent, decisional authority to act, and related control actions when interacting with each other and with intelligent machines. The primary outcome of this research is an information model and specification of the engineering methods required to support these expressions. In this work we explore three multidisciplinary and foundational concepts that can be used to design information flow in human-machine teaming situations: 1) formalizing the language we call "RECITAL," which defines the information flow based on concepts of intent, rules, and delegated authority; 2) applying this language to well-established models of human-machine distributed teams represented as a systemic control hierarchy; and 3) applying construal level theory from social psychology as a means to guide the producer-consumer model of the information abstractions. We conceptualize a standard information model intended to inform intentional design of human-machine teams. Here is a relevant example of the need for a new conceptual model for this information flow in the context of a single human-machine team:

> In November 2021, a Tesla automobile in Full Self-Driving mode was involved in an accident during a lane change maneuver. Although the details of the incident are not fully public, the driver claimed, "The car went into the wrong lane and I was hit by another driver in the lane next to my lane . . . 'I tried to turn the wheel but the car took control and . . . forced itself into the incorrect lane, creating an unsafe maneuver'. . ." [1]. According to Tesla's "Autosteer" instructions, once enabled, the vehicle will automatically change lanes when the turn signal is engaged. Autosteer requires the driver to maintain hands on the steering wheel. According to Tesla's "Navigate on Autopilot" instructions, when using Autosteer, fully automated route-based and speed based lane changes can be enabled. This mode defaults to the driver engaging a maneuver using the turn signal, but the mode can be set to allow the vehicle to do this autonomously. Once enabled, speed-based lane changes can then be separately disabled or set to operate in a conservative (MILD) or aggressive (MAD MAX) mode [2]. The manual does not discuss how a driver might overcome a vehicle initiated lane change while hands are on the steering wheel, although Tesla separately indicates driver movement of the steering wheel or brake pressure will always disengage autopilot activities.

Without any knowledge of the design of this mode, we will not speculate if and where an error in machine design or human operation may have occurred, we just use this example to familiarize the language in the context of a human-machine team. With respect to RECITAL, enabling the Navigate on Autopilot mode and disabling the default turn signal confirmation is a Transfer of Authority for complex passing maneuvers from human to machine. Selecting the desired lane change operations and defaults reflect human intent and also define machine intent. The instructions in the Tesla manual define rules of engagement for the selected mode. The research questions illustrated by this example are related specifically to the information transfer in this human-machine team and generalization of a language for that transfer. Generalization of this language will be discussed in part 3. The relationships between intent, authority, and rules also include both constraints in the machine design and constraints in human operation developed *via* training and experience. These relationships are based on how humans interpret the information present, which will be discussed in part 4.

In complex operations, human decision-making is dependent on the information they can access; their knowledge, skills, and abilities associated with the context of the tasks and related tools; and what the tools (machines) allows them to do. In design of related systems, the information requirements associated with both human and machine tasks at differing levels of the command, control, or team structure are subject to misinterpretation and error. The Tesla example might be considered a simple case of a single operator and single machine. This would be common to any Tesla vehicles operating with the same design configuration and software, although the human behavior will vary. In parts 3 and 4, we look more broadly at multiple operators managing control of multiple machines of differing capabilities and design, as subject to changes in constructs related to operational mission, rules, intent, or environment.

Humans have operational freedom to express these constructs at whatever level of specificity they desire, subject to constraints levied on them by the systems they are operating and communicating with and within. Likewise, human designers of intelligent machines have design freedom but in much more constrained environments. Human-machine teams must consider both an ontology as determined by domain and experience, and an ontology as constrained by the communication and machine control systems. Ability to interact at different levels of control will remain a primarily human function, but better design of human/machine interfaces can greatly reduce errors of interpretation and improve the flexibility of human and machine tasks. A standard informational design framework and methodology is needed. This work proposes one such approach.

3 RECITAL as a general information model in hierarchical systems

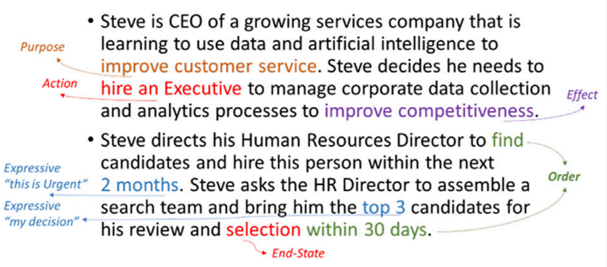
Rules of Engagement, Commander's Intent, and Transfer of Authority have a well-specified purpose and relatively standardized language in a military control hierarchy. For background the reader should refer to references [3–5]. In non-military enterprises, these information structures almost always exist but in a less well-specified form. There is no formal research that relates this language to non-military domains although components often appear in organizational leadership coaching [6, 7]. Here is a simple non-military example:

> Steve is CEO of a growing services company that is learning to use data and artificial intelligence to **improve customer service**. Steve decides he needs to **hire an Executive** to manage corporate data collection and analytics processes to improve competitiveness. Steve *directs his Human Resources (HR) Director* to find candidates and hire this person within the next 2 months. Steve asks the HR Director to assemble a search team and bring him the top 3 candidates for his review and selection **within 30 days**. The Vice President (VP)-Engineering and HR Director proceed with the hiring process. Based on the level of hire and the urgency they decide to use an executive search firm known to the HR Director for both its candidate networks and its speed. They provide the search firm a draft position description and a list of selection criteria they would like to emphasize.

In this example, intent is clearly communicated, although it must be interpreted from the language used (**hire an Executive to improve customer service, within 30 days**). Transfer of authority (*directs his Human Resources Director*) is explicit. Rules of engagement are not present in the narrative, but one can assume they are present within the enterprise's human resources organization (rules are normally defined separately). In practice, the fact that intent, rules, and authorities are almost always independent information flows is a common cause of control system failures. The RECITAL language attempts to define an integration framework for these.

3.1 Semantic representation of RECITAL

Gustavsson et al. proposed a standardized language representation of Commander's Intent to aid in machine interpretation [8]. CI is transferred down the military command hierarchy in a written set of orders describing the situation, the desired mission, how the mission should be executed, and supporting mission information. These orders exist alongside military doctrine and rules of engagement



- Steve is CEO of a growing services company that is learning to use data and artificial intelligence to improve customer service. Steve decides he needs to hire an Executive to manage corporate data collection and analytics processes to improve competitiveness.
- Steve directs his Human Resources Director to find candidates and hire this person within the next 2 months. Steve asks the HR Director to assemble a search team and bring him the top 3 candidates for his review and selection within 30 days.

FIGURE 1
Visual example of a CI Semantic language.

which exist separately from the order. CI is embedded within a military order, and directs a change from a current state to an end state by describing actions and intended effects that the commander determines will produce that end state. Gustavsson et al. further define a semantic construct for CI as an expansion on the purpose of the order, key actions to be performed, desired end state, and a set of "expressives" that convey additional intent [8]. Figure 1 shows how these semantic constructs appear using the previous non-military example.

3.2 RECITAL representation in a control hierarchy model

The semantic constructs of order, purpose, action, effect, end-state, and expressives exist universally in human control hierarchies, and as can be seen from the earlier Tesla example, are beginning to influence human-machine control hierarchies. Levenson's System-Theoretic Process Assessment (STPA) provides a means to formally model these information flows in human and machine control structures [9]. In STPA, a system is represented as a hierarchy of controlled processes, each of which can have a human and machine controller, a model of the controlled process, and a set of information and explicit control flows. Figure 2 provides a depiction of this control structure with the RECITAL ontology overlaid as adapted from [10].

3.3 RECITAL integration framework

In this process model, one can define the order, purpose, expressives, and rules as inputs into the control hierarchy; actions, authorities, and effects as implicit in the design of the controller; and the interpretation of this information as a model of the controlled process. Other contextual information that would disturb the controlled processes is noted as coming in from the bottom of the model. In C2 systems, we are particularly interested in events that disturb the normal control process flow and how they affect the interpretation of information. A more complete example of this will be provided in section 6. Each layer of hierarchy in this system might require a change in the abstraction level of the information. RECITAL attempts to resolve errors related to incorrect abstraction of information provided versus that consumed at a level of control hierarchy.

Figure 3 provides a generalized model reflecting two levels of hierarchy. In human-machine distributed teams, one must model how information flows into human and automated

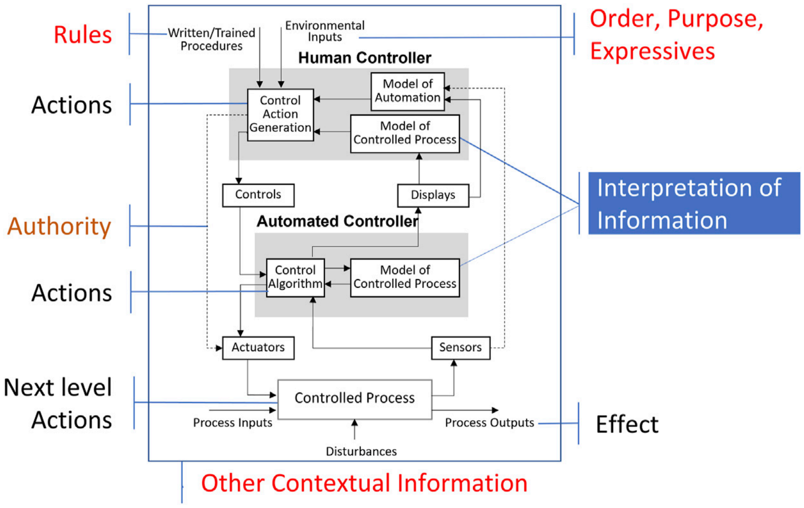


FIGURE 2
STPA control structure overlaid with the RECITAL ontology. Adapted from Cockcroft [11].

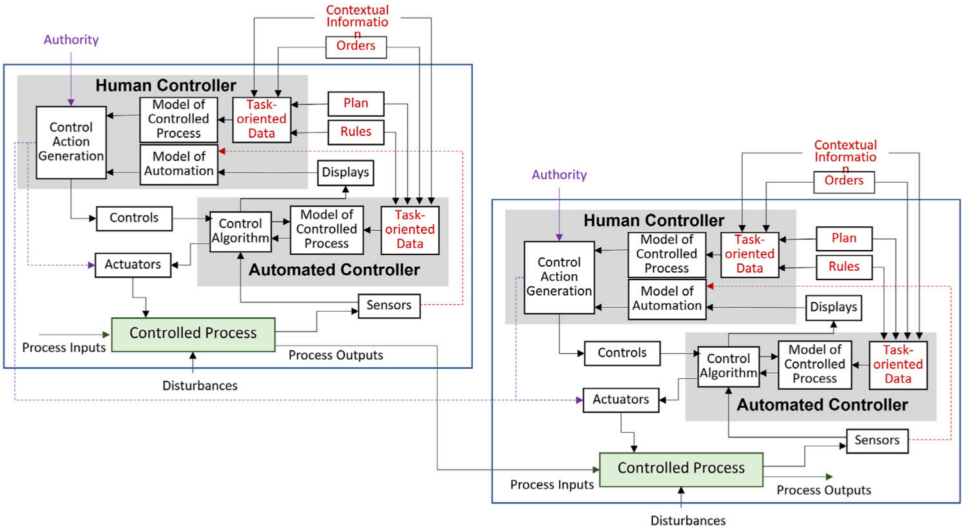
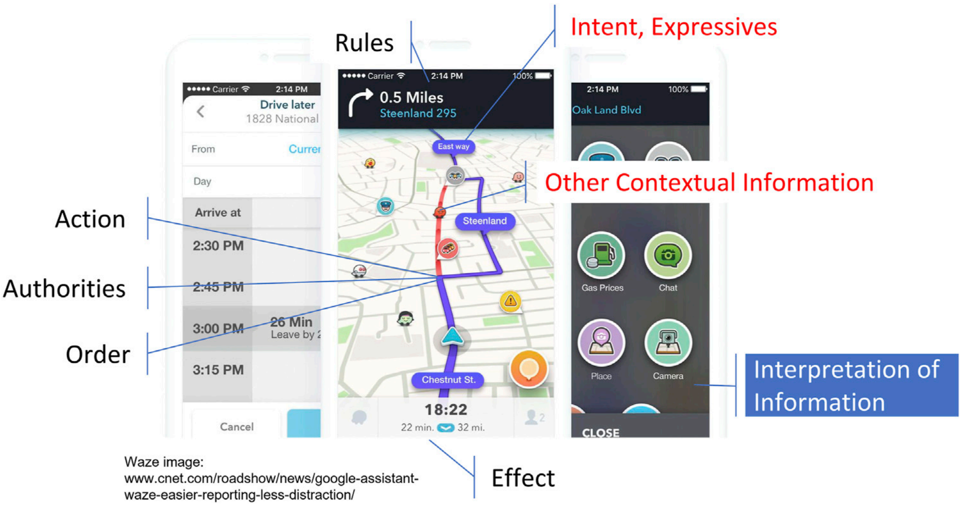


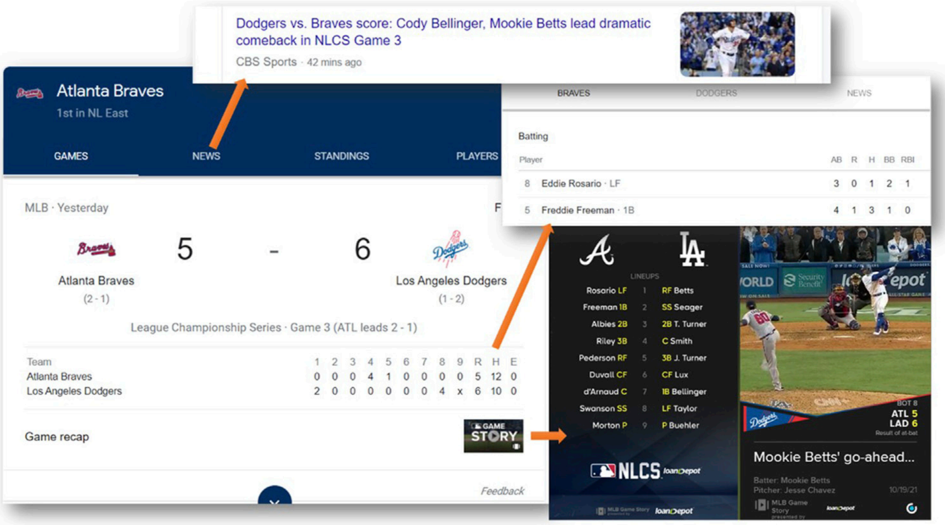
FIGURE 3
Generalized RECITAL control structure.



Waze image:
www.cnet.com/roadshow/news/google-assistant-waze-easier-reporting-less-distraction/

FIGURE 4
Waze application overlaid with RECITAL ontology [12].

machine controllers at any level of a hierarchy. It is expected (at least in the foreseeable future) that authority will be transferred between humans and machines as a human generated control action. We add “task-oriented data” to the model of Figure 2 as both the data that will be available and how that data is interpreted will affect the operation of the control loops. Most tasks in these systems will be at least partly defined by software and related task-oriented data, and data will be used as a selection process for various aspects of a control process. Orders, rules, planning information, and other contextual information can be made available in a consistent way to all human and machine controllers in the control hierarchy. The question becomes how is the right data provided and selected for each task? The answer requires understanding and modeling of both producer (what data is available) and consumer (how will it be interpreted) views of data. In addition, much of this information becomes more



Dodgers vs. Braves score: Cody Bellinger, Mookie Betts lead dramatic comeback in NLCS Game 3

CBS Sports · 42 mins ago

Atlanta Braves

1st in NL East

MLB · Yesterday

Atlanta Braves 5 - 6 Los Angeles Dodgers

(2 - 1) (1 - 2)

League Championship Series · Game 3 (ATL leads 2 - 1)

Team	1	2	3	4	5	6	7	8	9	R	H	E
Atlanta Braves	0	0	0	4	1	0	0	0	0	5	12	0
Los Angeles Dodgers	2	0	0	0	0	0	0	4	x	6	10	0

Game recap

Batting

	Player	AB	R	H	BB	RBI
8	Eddie Rosario · LF	3	0	1	2	1
5	Freddie Freeman · 1B	4	1	3	1	0

LINEUPS

Rosario LF	1	RF Betts
Freeman 1B	2	SS Seager
Albies 2B	3	2B T. Turner
Riley 3B	4	C Smith
Pederson RF	5	3B J. Turner
Duvall CF	6	CF Lux
d'Arnaud C	7	1B Bellinger
Swanson SS	8	LF Taylor
Morton P	9	P Buehler

Mookie Betts' go-ahead...

Batter: Mookie Betts
Pitcher: Jesse Chavez 10/19/21

FIGURE 5
Example representation of CLT in a Google search for "baseball score".

subjective as one moves from rules and plans, to orders, to contextual factors. The information processing needs are different at different levels. A framework is needed to manage the data and information abstractions and related decision processes in each controller.

4 Construal level theory and application to RECITAL

Hierarchical control systems become constrained by limitations on the information produced and consumed at various levels of the hierarchy. Typically, changes in context and related information come in at the top of the control hierarchy (i.e., combatant command, vehicle driver) and information is lost or incorrectly abstracted as it progresses down the hierarchy. The additional detail needed by some users may not be present, requiring queries or speculative interpretations to get the needed detail. We desire to design future distributed human-machine systems with more flexibility in decision processes at each level of hierarchy, including more flexibility in decisions made by machine-machine teams. We would like to define the information model so that the multiple levels of detail coexist at each level of hierarchy in the information structure and can be extracted according to user needs.

There are a number of AI-based applications appearing today that provide such flexibility by intentionally managing the consumed abstraction hierarchies, providing the human controller greater versatility in selection of contextual information. Figure 4 provides an example of our RECITAL language overlaid on the popular "Waze" road navigation application.

In Waze, the driver's intent is expressed by the initial selection of the route which is generally determined by either shortest time, shortest distance, or an acceptance of the recommended route. The driver's order is expressed as selection of the route and clicking "start." At this point, authority is transferred to the application to manage the route. Waze can sense changes in contextual information, such as other driver reports of accidents ahead, and open up a reassignment of authority to the driver to select a new route (or not). What is most interesting about this human-machine user interface is the way in which the Waze application presents to the driver new contextual information at different abstraction levels. The driver can just select the new route, can view the location and nature of the incident ahead before deciding, or can even see the comments from other drivers about the incident. The use of a **progressive disclosure** concept to manage the abstraction level of consumed information is a well-known approach to consumer-driven information design [13]. It has primarily been applied as a means to reduce complexity in human-machine interface design [14], not as a means to manage informational design tool in the context of RECITAL. Hence, a generalized model and associated research will need to be developed to determine its effectiveness in the context of human-machine teaming.

In this research we investigated Construal level theory (CLT) as a potential underlying theoretical basis for this type of

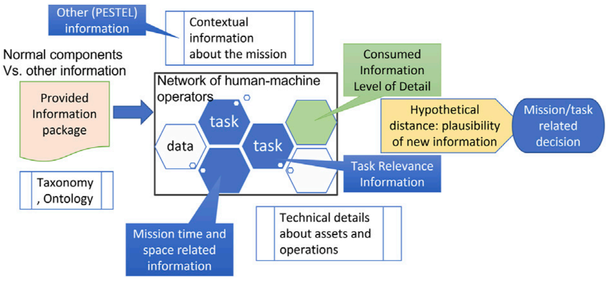

Other (PESTEL) information
Contextual information about the mission
Normal components Vs. other information
Network of human-machine operators
Provided Information package
task
data
task
Consumed Information Level of Detail
Hypothetical distance: plausibility of new information
Mission/task related decision
Task Relevance Information
Taxonomy , Ontology
Mission time and space related information
Technical details about assets and operations

FIGURE 6
Conceptual view of the use of RECITAL and CLT in an operational information flow.

hierarchical information design in applications of human-machine teaming. CLT is used in social psychology to describe the extent to which people prefer information about a topic to be abstract versus concrete as a function of psychological distance [15, 16]. Psychological distance can be defined as a function of separation in time, space, task relevance, or other interest [17]. The degree to which information is abstract versus concrete may manifest as a combination of comprehensiveness of the information elements presented, and level of detail about a given information element.

Tasks performed by different users, in which information about intent, rules, or decisional authority are needed to perform the task to a standard, have common information requirements but differing needs for detail. Differing levels of detail can be supported by these emerging user interface concepts that use AI to monitor information and then provide progressive disclosure. Text-based user interfaces (structured or unstructured) can be formatted so that top-level information is presented in outline or title form, and interested users can progressively expand the text to access the desired level of detail. Similarly, pictograms and annotated maps can be structured so that top-level information is presented, and a "show more"/"show less" structure can be provided to allow a drill-down into the various levels of detail. A narrated story approach, which combines a multitude of medias, may be the most straightforward way to mediate the need to provide different levels of detail to different users (or, to the same user at different points in time), as it combines both mission and task level aspects. The narrated story has the additional advantage of easing the cognitive burden placed on the user.

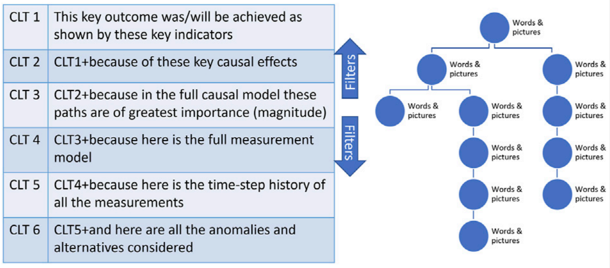
These approaches, though not explicitly identified as design approaches, are also regularly used today to structure information about such diverse topics as movies, sporting events, recreational activities, and other forms of entertainment. For example, an account of a baseball game can be represented simply as the final score, as a box score with statistics for individual participants, as a "highlights" summary showing key plays, as a play-by-play account, and of course, as the full pitch-by-pitch account of the game. These different levels of detail are of interest to different types of consumers. Some baseball followers will simply want to know the final score. Others may also be interested in key facts, such as the impact on pennant races or personal milestones achieved by participants. Some may be interested in scoring plays only, while others may want to track certain participants throughout the course of the game. Relatively few would be interested in a pitch-by-pitch account after the game (though quite a few might be interested in this during the game itself); those interested in pitch-by-pitch after the game might be analysts trying to find certain trends. Figure 5 shows how Google manages progressive disclosure in their search results for a baseball score [18].

In a C2 environment one can draw similar analogies. At the command level, commander's intent is provided to set the highest level desired outcomes and constraints for a mission. The commander might only want to know mission results and related "box score" information as long as the mission was successful. Planning teams would want additional contextual and operational information including the mission concept of operations, areas of regard, resources available, and rules of engagement. Even different planning activities may demand different levels of detail. For example, the details of order of battle and route planning would differ in detail between an aircraft mission at altitude and one in terrain. Tactical operators would want more of the "play-by-play," but would only want to be burdened with higher level contextual information when a mission goes off-plan. The ability to selectively add-in or subtract information at different abstraction or construal levels, only when needed for decision making, will be quite useful in complex missions or tasks.

CLT provides a basis for structuring information in advanced user interface concepts, using information representations that can be provided and then selectively engaged to provide more or less detailed information to the interested consumer of the information. We used CLT and the RECITAL language model to structure an operational model of the information flow desired in distributed human-machine teams. A conceptual view of this model is shown in Figure 6.

Given a set of tasks performed by a network of human-machine operators, there are two selection processes: the selection of information provided by the system design; and the selection of information consumed by the operator(s). Future information systems design should attempt to work both aspects of the information selection process. Data analysis, artificial intelligence, and machine learning methods and tools are making great progress in inferring context from large corpuses of data. To design effective queries in more complex tasks, explicitly modeling human construal levels is a promising approach.

A core aspect of achieving effective mission/task related decisions can be related to "plausibility" of the consumed information as related to the operator's beliefs. The logic



CLT 1	This key outcome was/will be achieved as shown by these key indicators
CLT 2	CLT1+because of these key causal effects
CLT 3	CLT2+because in the full causal model these paths are of greatest importance (magnitude)
CLT 4	CLT3+because here is the full measurement model
CLT 5	CLT4+because here is the time-step history of all the measurements
CLT 6	CLT5+and here are all the anomalies and alternatives considered

FIGURE 7
Six-layer CLT model.

follows formal definitions of plausibility from Dempster-Shafer Theory [19] although in our use formal mathematical bounds on plausibility may not be possible. This is an area for further research. We define plausibility as the operator's perceived probability of occurrence of an event based on hypothetical distance between current state and end state, based on explicit, implicit, and other contextual information. Defining an "estimated proximity function" for that distance would require:

1. Direct access to information and user interfaces that allow an operator to situationally retrieve additional contextual information (either historical or predicted future) based on proximity of that information to the task at hand and hypothetical distance between that information and the task situation,
2. A design for organization of that information based on operator construal level. From CLT, we can initially organize information by Task relevance, Spatial relevance, Temporal relevance, and Other contextual information relevant to the task at hand,
3. Measurement of operator task situational awareness. Endsley [20] defines situational awareness as an operator's perception of the information, comprehension of the information, and projection of that information onto their task at hand, and
4. The "proximity function" that rates effectiveness of the user interface in relating the information to tasks so as to improve operator situational awareness.

If we are able to define such a function, we can use CLT in practice to evaluate five categories of contextual information in an operational environment:

1. How people perceive, comprehend, and project temporal information
2. How people perceive, comprehend, and project spatial information
3. Relevance of this information to their tasks (or not)
4. Other contextual information that may be relevant (political, social, etc.)
5. Hypothetical distance (plausibility) between their interpretation of the information and their tasks

Temporal and spatial information are related to the operators' mission, task relevance and hypothetical distance are related to operators' actions in tasks, and other contextual information may affect both. Again, the Waze user interface is a good example of how mission relevant and task relevant information can be combined into the human-machine interface.

5 A formal model of construal levels

Based on an evaluation of CLT, existing applications, and the full distributed human-machine teaming case study to be described in section 6, we formalized a six-layer model linking construal levels to related information abstractions. This model is shown conceptually as a progressive measurement information model in Figure 7, where increasing CLT layer number denotes progressively increasing detail of information. The tree structure at the right of the table depicts that there is a hierarchy of information (words and pictures) that is added to and progressively disclosed at each increasing CLT layer.

If one were to define a causal model that relates the produced information to the consumed information and then to task decisional effectiveness, that would generally form CLT layer 3.

5.1 Descriptive measurement model

The following provides a descriptive application of construal levels into an information feed consisting of both narrative information and visual images. We have defined six construal levels as appropriate standards in this work. A given application might need fewer levels.

CLT 1: Executive summary. This level is generally composed as one visual and two to three sentences of text, and no more than 10 s in duration. This is the most abstract level. For a future planned event, this level presents the main claim [*key outcome*] will achieve intent as shown by these [*key indicators*]. For a past event, this level declares overall success or the lack thereof: this [*key outcome*] achieved (or did not achieve) intent as shown by these [*key indicators*].

As a baseball analogy example, imagine the manager of a baseball team interacting with an app that has statistical information on all of the players. The manager's intent is most certainly to *win the game*. In the future example, the manager's level 1 construal might be: "these players in the 8-9th batting order positions are most likely to *produce the runs* needed to *win the game*." In the past example: "the difference in the *win* was the *production of those two additional runs* from our 8-9th batting positions."

CLT 2: Mission overview. This level is composed of perhaps two or three visuals, and the text is no more than 30 s in duration. This is the "elevator speech" level of abstraction. It is more specific about intent and related considerations for rules of engagement. For a future planned event, this level presents the main reason intent is expected to be achieved: this [*key outcome*] will achieve intent as shown by these [*key indicators*], due to these [*primary causal effects*]. For a past event, this level declares overall success or the lack thereof, and gives the key reasons why: this [*key outcome*] achieved or did not achieve intent as shown by these [*key indicators*], because of these [*primary causal effects*].

This level in the baseball analogy might be a future example: "these players in the 8-9th batting order positions are most likely to produce the runs needed to win the game. Joe Baseball and Jim Stealer have matched up well against their starter Mike Pitcher in our previous two meetings. Our statistics indicate we can count on at least two runs from the bottom of the order with these players." Past example: "the difference in the win was the production of those two additional runs from our 8-9th batting positions. Joe Baseball has been productive in the 8th spot all year."

CLT 3: Mission summary. Although there are no specific constraints on duration, this level is limited to describing events as a sequence of actions. The narration of each step is succinct. For a typical operation, the duration would be less than 5 min. For a future planned event, this level provides a top-level description of how the planned operation will apply rules and authorities to achieve intent and why success is predicted. It conveys the overall timeline for the operation, describes all of the major actions, and identifies the specific actors. The narration describes the key parameters of actors, resources, and activities that are essential to success (i.e., the primary causal paths in an underlying mission model). For a past event, this level describes the actual sequence of actions and events that determined the success (or lack of success) of the operation.

For a future planned event, the step by step activities are explored and selected by evaluating multiple scenarios for the mission and perhaps running simulations. These steps are similar to how the Waze app might calculate different multiple alternative routes with varying time estimates around an accident, with that information displayed in different colors, and presented to the driver to accept. In baseball, this analogy might reflect how a manager evaluates batter substitutions due to an opponent's pitching changes. In a more complex mission, human and machine aided planners and tools might test different courses of actions (COA's) before selecting the best COA to give the operators as their baseline plan.

In the course of a mission, the RECITAL concept dictates that all or many of those scenarios remain present as part of the produced information, to be selected by operators or automated tools based on disruptions to the mission control flow. Instead of simple route changes, the operators have access to more complex alternative mission descriptions based on their spatial, temporal, task-driven, or other information needs for information. This access may require them to explore information into the next construal levels to aid in deciding on an alternate future mission success strategy. The mechanization of this capability will be presented further in section 6.

CLT 4: Mission brief. There are no specific constraints on narrative duration at this level. The emphasis shifts to substantive completeness rather than brevity. Content at this level should cover the major points of background and context (to address *why*), major contingencies, and elaboration on rules of engagement as appropriate. Authorities to execute the mission are explicit but include multiple scenarios where different authority levels may be assigned. This level would include any significant political/environmental/social/other considerations. For a typical operation, the duration would be less than 30 min. For a future planned event, this level is similar to briefing the mission plan to the next higher-level authority. There should be enough detail to cover what actions are planned, the key timeline for those actions, and the key contingencies that are recognized and covered by the plan. For a past event, this level constitutes an after-action report presented to the next higher-level authority. It states whether the intent was achieved and covers the actions that were taken and the timeline associated with those actions. It also describes contingencies that occurred and the reaction to each, and any anomalies that impacted the outcome. The narration includes reference to rules of engagement that governed the reaction to contingencies or anomalies.

CLT 5: Mission plan/report. Again, there are no specific constraints on narrative duration for this level, but the intent is to include all elements of the mission plan. Key political/environmental/social/other parameters are typically included, even if benign. For a typical operation, the duration would be less than 60 min. For a future planned operation, the content should cover all relevant points of background and context, all contingencies that are reasonable to expect, and key technical parameters or details (to address *how*). This level is similar to reviewing a detailed mission plan with the crew that will execute it. The contingencies covered by the plan may be considered unlikely but are of enough significance to merit explicit planning. The key milestones on the mission timeline are covered at this level, as they were at Level 4. At this level the impact of contingencies on the mission timeline should be addressed, especially if time itself becomes a forcing function in the presence of certain contingencies. For example, available fuel may limit the route selected by the driver/Waze teaming (or be integrated into the app). Factors that may not be a significant concern in the nominal mission plan should be included as various contingencies, and hence should be a topic covered at this level of detail. Any maintenance-related concerns that will potentially impact the mission should be described at this level as well.

For a past operation, the content is similar to a mission report. It repeats the relevant information about context, to help explain *why* the operation was conducted, and narrates the sequence of events and actions from beginning to end. It includes narration about contingencies that were realized, and anomalies that occurred that were consequential to the outcome. A user interface described in section 6, which we call "UxBook," is used to store multiple past mission reports at this level of detail in order to learn and inform future missions.

CLT 6: Mission details/logs. There are no specific constraints on duration at this level. The content is a point-by-point elaboration on Level 5, adding more detail "on demand." It is not expected that any one individual would be interested in all of the detail. Examples of additional detail that is made available at this level include additional contingencies, additional information on the technical parameters or principles of operation, recent maintenance history relevant to the operation, and full environmental data and estimates.

5.2 Informational forms

The presentation of information in the RECITAL concept will contextually blend different forms of information based on differing spatial/temporal/task-driven/other needs. The selection of form is critical to the appropriate operator consumption of information and must be selected to situationally reduce psychological distance. Informational forms include structured and unstructured text, pictograms, annotated maps or other visual renderings, and narrated story.

5.2.1 Unstructured text

Unstructured text is visual or auditory content that cannot be readily mapped onto standard database fields. Information about intent, rules, decision authority, and control tasking is most often expressed in unstructured text. There are no constraints on how the constructs are expressed. Errors in comprehension occur from both differences in the language used versus comprehension, and differences in the information transferred versus that needed to perform the operation. Using unstructured text to convey these constructs to automated software systems is not practical, as it would require advanced natural language processing capabilities far in excess of what is currently available. However, most human to human information exchange is unstructured or only partially structured as codes or standard terms so informational concepts must address this form.

5.2.2 Structured text

Constraints on information exchange in hierarchical control systems are governed by standard formats that structure the information into defined fields, with limited use of unstructured text in some of those fields. Information appears in a specified order, and for many of the fields is restricted to certain values (or range of values) to be valid. For some messages there is a field that allows unstructured text, typically of constrained length, and perhaps labeled "notes" or "remarks". In practice, these unstructured fields may contain significant information relevant to the operational task, which must be interpreted based on very limited expressions adapted to fit into a message structure. Representing context is critical for decision making in operational environments, and requires richer forms of communication.

5.2.3 Pictograms

A pictogram is a depiction of relatively abstract information in caricature form (The term is not universally used and is not tightly defined. A pictograph is also used in some contexts.) As used here, a pictogram is a graphical depiction of an action, constraint, or other attribute with minimal reliance on text. The Waze screen in Figure 4 is a typical example. The pictogram relies on some degree of visual similarity to the object, action, or other attribute that is represented. A pictogram is generally static, and a sequence of pictograms may be used to depict temporal order. An animated pictogram is a brief succession of images that supports perception of motion or other action in the context of the pictograph. One example of a simple pictograms are icons, which are used to represent certain ideas, things, or categories, signal certain conditions, or direct attention in a quick and easy manner. Pictograms have the putative advantage of not requiring language proficiency in order to comprehend meaning, although in practice pictograms may be dependent on labels and familiarity with cultural stereotypes in order to be effective.

Pictograms can be used to convey certain actions that are allowed or prohibited, or end states that are intended or unintended. Pictograms thereby convey information about intent, rules, and authorities. Animation of the pictogram may aid comprehension of actions depicted by the pictogram. Pictograms overlaid on an actual operator's visual scene, such as with augmented reality devices, might also be used.

5.2.4 Annotated map

An annotated map uses spatial information overlaid with supplemental annotated information. The Waze image in Figure 4 provided an implemented example. An annotated map is particularly useful at visualizing the spatial context of any operational activity. Annotation on a map may include information that does not have a strict spatial referent, such as the time at which something occurred (or is planned to occur), or an outcome that was achieved (or is intended). Annotated

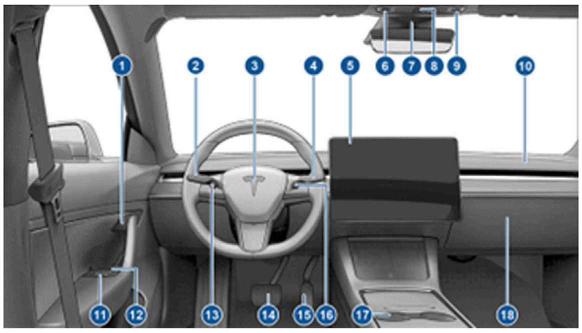


FIGURE 8
An example of an image as an annotated map [2].

maps may be referenced to an external context, such as a geographic map, or a system context, such as the pictogram image of the Tesla cockpit shown in Figure 8. Many operations are predicated on a coordinated movement of items or people over time and in space. Thus, the annotated map may show the relative positions of participants and objects at the beginning of the operation, along with their intended movements.

5.2.5 Narrated story

The narrated story uses visual information (such as still images, full motion video, text, and graphics) accompanied by an audible narration (which may be captioned in some contexts) to create the perception of a story. The story usually has standard elements such as setting, actors, and plot.

The effectiveness of the narrated story is dependent on the extent to which the story builds natural interest and corresponding comprehension in the recipient (user). Unlike annotated maps, which require the user to actively engage in the material, the narrated story allows the user to remain relatively passive as comprehension is created by its presentation. This process eases the cognitive burden on the user and reduces the probability that incorrect inferences will be drawn from the presentation.

An illustration of the narrated story concept associated with this research, with the visuals and accompanying narratives, appears in section 6. The narrated story may be the approach that is best suited for application of the CLT constructs, and it may also benefit from requiring less cognitive effort by the user in achieving required levels of comprehension.

A story, as narrated at CLT Level 6, may include a considerable amount of supporting detail. A single thread of narration may not be practical. Instead, it may be more effective to provide the narrative at Level 5 with a way for the user to request more detail from Level 6 for topics of interest to him or her. The user interface (UI) mechanisms by which such detail can be requested include a list of topics ("more information"), attributes on icons or other symbols on the display, and/or spoken prompts that state an action to take to get more information on a certain topic. Such conventions are not meant to be restricted to Level 5 presentations. They can be used at higher levels, certainly Level 4, but even at Levels 1, 2 and 3.

6 Practical example: Distributed autonomy in a mine warfare mission

We present an example of the modeling process and development of a future planning system using a military mission scenario associated with undersea mine countermeasure (MCM) operations. In this scenario, military commanders' intent, rules of engagement, and decision authorities are represented down to a set of operators who are conducting mine search and destroy operations using unmanned airborne and underwater vehicles (UAVs and UUVs) with a number of automation capabilities. Control of mission activities can be distributed between the human planners, the operators, and the vehicles, as well as vehicle to vehicle. In particular, the scenario assesses the information flows associated with transfer of control of the UAV platforms between operators, a process known as Transfer of Tactical Control (ToTC). In this specific scenario, a failure in the UAV associated with one ship, the USS Coronado, requires a transfer of the mission to another UAV known as RQ-X, currently in control of another ship - the USS San Diego. The alternate UAV is an experimental platform with automated mine search and neutralization capability. The ToTC process is executed so the RQ-X is managed by the USS Coronado during the operational mission, then returned to the San Diego at a designated handoff point. Figure 13 provides a visual overview of the mission.

We would like to develop a system that allows the decision authority in that transfer to be made at the operator level, with operator decision data that situationally includes both intent and application of rules of engagement as annotated through various hierarchies of command. This process requires a rapid transfer of authority, a re-evaluation of rules of engagement, and a revision to mission planning. As this transfer is for a less familiar type of UAV to the Coronado's operators, the scenario presents a narrative-driven planning and rehearsal capability where the operator(s) can review planning information at multiple construal levels. The appropriate construal level for a particular operator would vary based on both their familiarity with the RQ-X and the mission operational context. In the present day, these decision data are normally expressed in unstructured text.

In the definition and analysis process using this methodology, we begin with a mission task analysis (MTA) that defines the sequence of human and machine tasks to be

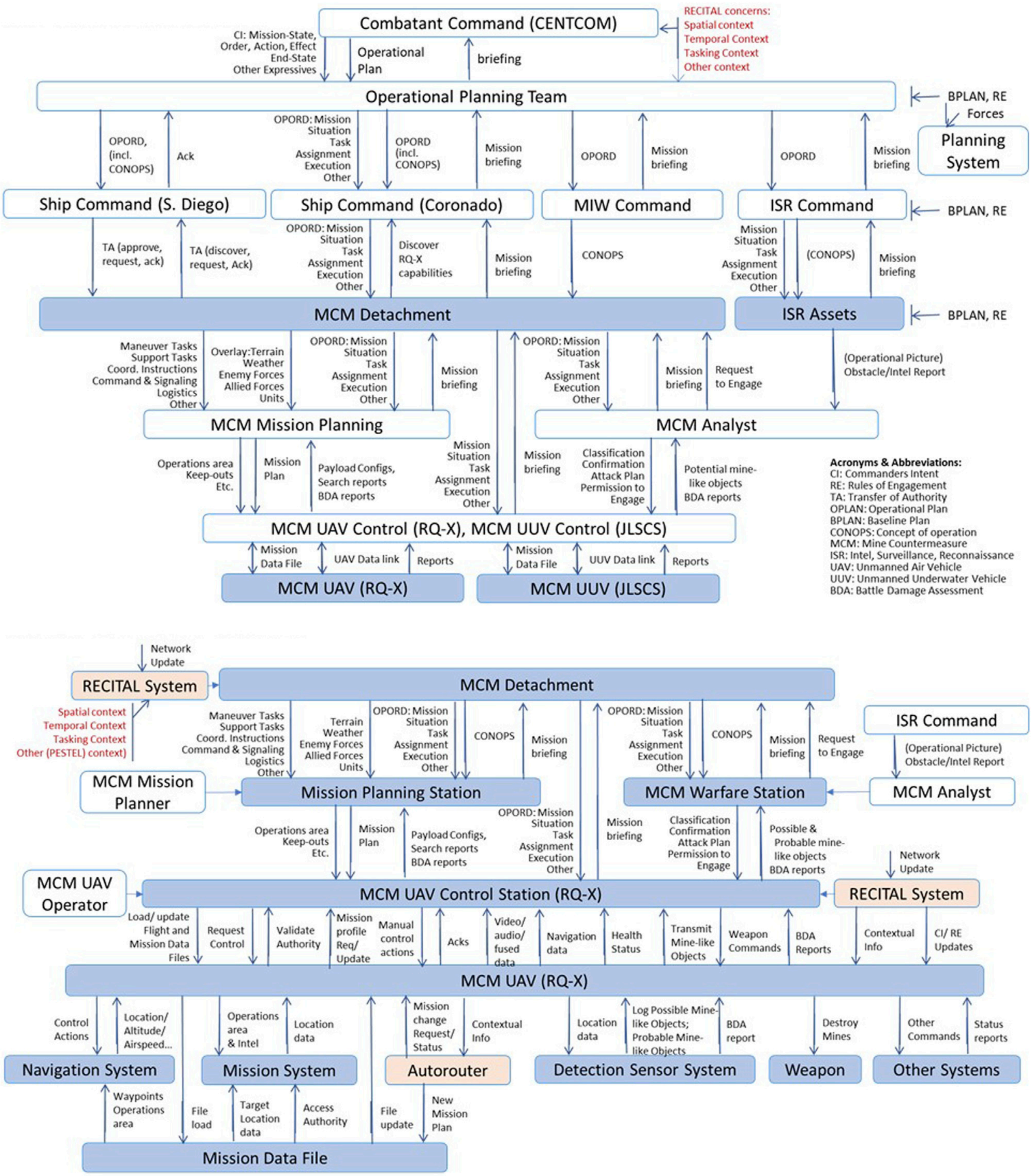

Combatant Command (CENTCOM)
RECITAL concerns:
Spatial context
Temporal Context
Tasking Context
Other context
CI: Mission-State, Order, Action, Effect End-State Other Expressives
Operational Plan
briefing
Operational Planning Team
BPLAN, RE
Forces
Planning System
OPORD, (incl. CONOPS)
Ack
OPORD: Mission Situation Task Assignment Execution Other
OPORD (incl. CONOPS)
Mission briefing
OPORD
Mission briefing
OPORD
Mission briefing
Ship Command (S. Diego)
Ship Command (Coronado)
MIW Command
ISR Command
BPLAN, RE
TA (approve, request, ack)
TA (discover, request, Ack)
OPORD: Mission Situation Task Assignment Execution Other
Discover RQ-X capabilities
Mission briefing
CONOPS
Mission Situation Task Assignment Execution Other
(CONOPS)
Mission briefing
MCM Detachment
ISR Assets
BPLAN, RE
Maneuver Tasks Support Tasks Coord. Instructions Command & Signaling Logistics Other
Overlay:Terrain Weather Enemy Forces Allied Forces Units
OPORD: Mission Situation Task Assignment Execution Other
Mission briefing
OPORD: Mission Situation Task Assignment Execution Other
Mission briefing
Request to Engage
(Operational Picture) Obstacle/Intel Report
MCM Mission Planning
MCM Analyst
Operations area Keep-outs Etc.
Mission Plan
Payload Configs, Search reports BDA reports
Mission Situation Task Assignment Execution Other
Mission briefing
Classification Confirmation Attack Plan Permission to Engage
Potential mine-like objects BDA reports
Acronyms & Abbreviations:
CI: Commanders Intent
RE: Rules of Engagement
TA: Transfer of Authority
OPLAN: Operational Plan
BPLAN: Baseline Plan
CONOPS: Concept of operation
MCM: Mine Countermeasure
ISR: Intel, Surveillance, Reconnaissance
UAV: Unmanned Air Vehicle
UUV: Unmanned Underwater Vehicle
BDA: Battle Damage Assessment
MCM UAV Control (RQ-X), MCM UUV Control (JLSCS)
Mission Data File
UAV Data link
Reports
Mission Data File
UUV Data link
Reports
MCM UAV (RQ-X)
MCM UUV (JLSCS)
Network Update
RECITAL System
MCM Detachment
Spatial context
Temporal Context
Tasking Context
Other (PESTEL) context)
Maneuver Tasks Support Tasks Coord. Instructions Command & Signaling Logistics Other
Terrain Weather Enemy Forces Allied Forces Units
OPORD: Mission Situation Task Assignment Execution Other
CONOPS
Mission briefing
OPORD: Mission Situation Task Assignment Execution Other
CONOPS
Mission briefing
Request to Engage
ISR Command
(Operational Picture) Obstacle/Intel Report
MCM Mission Planner
Mission Planning Station
MCM Warfare Station
MCM Analyst
Operations area Keep-outs Etc.
Mission Plan
Payload Configs, Search reports BDA reports
OPORD: Mission Situation Task Assignment Execution Other
Mission briefing
Classification Confirmation Attack Plan Permission to Engage
Possible & Probable mine-like objects BDA reports
Network Update
MCM UAV Operator
MCM UAV Control Station (RQ-X)
RECITAL System
Load/ update Flight and Mission Data Files
Request Control
Validate Authority
Mission profile Req/ Update
Manual control actions
Acks
Video/ audio/ fused data
Navigation data
Health Status
Transmit Mine-like Objects
Weapon Commands
BDA Reports
Contextual Info
CI/ RE Updates
MCM UAV (RQ-X)
Control Actions
Location/ Altitude/ Airspeed...
Operations area & Intel
Location data
Mission change Request/ Status
Contextual Info
Location data
Log Possible Mine-like Objects; Probable Mine-like Objects
BDA report
Destroy Mines
Other Commands
Status reports
Navigation System
Mission System
Autorouter
Detection Sensor System
Weapon
Other Systems
Waypoints Operations area
File load
Target Location data
Access Authority
File update
New Mission Plan
Mission Data File

FIGURE 9
Top and bottom halves of the hierarchical control model focused on mission level information transfers in the MCM mission involving the RQ-X UAV and JLSCS UUV. The bottom half of the hierarchical control model is just for the RQ-X UAV.

performed in the control hierarchy. The MTA methodology consists of defining a set of design reference scenarios, from which a hierarchical functional breakout can be derived. That functional breakout leads to identification of human and machine tasks, and the information requirements associated with those tasks. The information requirements are a key point of interest. A functional analysis process adopted from Chaal, et al. [21] is used in discussions with operators to make sure all tasks/functions and information needs are captured in the control structure at every level. At this point vignettes are used identify control actions that need to be defined or modified in response to not only disruptions but also changing mission context, orders, rules, or authorities. At this point we use the standard STPA analysis flow of identifying losses, related human or machine operator hazards, and control actions of interest to reason about the additional information needs (either additional detail or context) the operator requires to successfully perform a task. The difference in our use of STPA in this work is a focus on any disruptive change instead of just accidents. For example, a mission loss can occur if a change in political situation requires a mission to be aborted and the operators fail to successfully abort. A sample vignette for our MCM mission is described below:

> As the RQ-X conducts its mine search and neutralize pattern, a suspected mine-like object was found at a location near to the politically mandated keep-out zone. The local operator and RQ-X geographic information systems do not have sufficient resolution to isolate location of this mine in the operational area versus keep-out area and the RQ-X is allowed to proceed to this location and neutralize the mine. Higher accuracy satellite geographic information indicates the mine is actually in a keep-out area. Both the human operators and the RQ-X fail to access this additional information and cause an international incident.

In implementation of a RECITAL system the mine in question would show up as an alert on the operator's screen (likely a visual map) indicating the need to query more detailed information. A similar alert would cause the RQ-X to transfer control back to the human operator for that particular segment of the mission.

We can model this information flow in a system-theoretic approach at multiple levels using the STPA concept of a control model. Figure 9 shows a control model for a complex MCM mission using the RQ-X. Int the lower have of the figure the concept of a "RECITAL System" is a simplified black box function for the set of applications that would scan external context and provide relevant information to the operators at the appropriate construal levels.

A number of innovative user interface (UI) concepts were identified in this research as alternatives to using text to convey CI and RE. These include combinations of pictograms, annotated maps, and narrated stories. The narrated story concept proved particularly adept at supporting the different levels of detail needed across users. A UI concept rooted in current social media platforms, called the "UxBook" concept, was developed to provide a way to feature structured and unstructured text, pictograms, annotated maps, and narrated stories. The narrated story formed an initial conceptual model of an implementable system, focused on scenarios. System operational and information modeling was identified as providing a useful framework to understand interoperability requirements in information exchanges involving both humans and intelligent systems, and the effort developed an initial approach to capture these information exchanges in a commercial model-based systems engineering (MBSE) tool.

Figure 10 is a potential representation of formal CI based on a typical military concept of operations transfers, presented as unstructured text. This is color coded to reflect the intent and effects model of Figure 1.

Note that a statement of intent generally describes the context of the mission and end state but not the resources or plans required to accomplish it. Resources and plans can be provided in textual format but also more are richly represented as pictograms or maps. The following section describes an illustration of CLT levels in a simulation tool that utilizes annotated maps as the primary user interface.

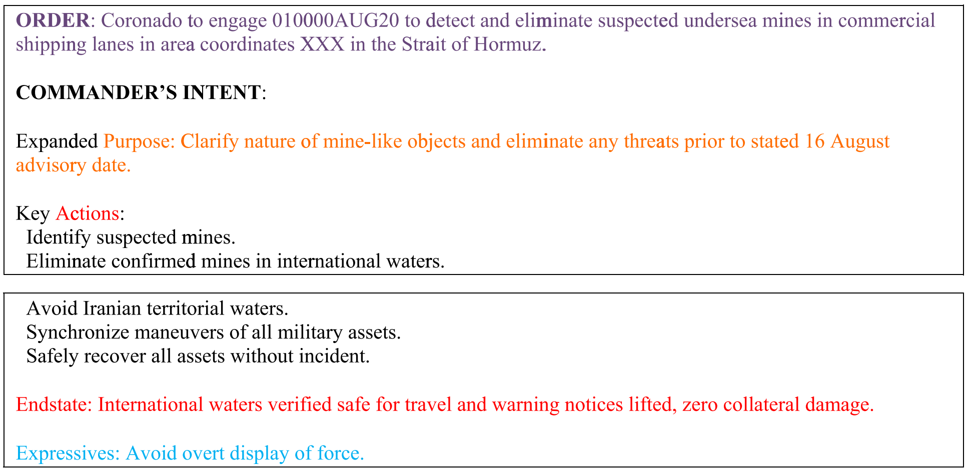
6.1 Illustration of a narrated story using annotated maps

Our narrated story uses visual information (such as still images, full motion video, text, and graphics) accompanied by an audible narration (which may be captioned in some contexts) to create the perception of a story. The story usually has standard elements such as setting, actors, and plot. The narrated story supports different levels of detail by providing different forms (or versions) of the story.

The narration of the story is provided by natural language, perhaps implemented by a text-to-speech function. (Automatic generation of narrative is a topic currently under investigation by multiple researchers and is showing considerable promise. Future updates to this research will contain a review of this progress.) The narration may feature multiple voices, perhaps to distinguish different sources or points of view, or to represent different functions supported by the information. There is no practical limit to the number of individual human voices that a person can discriminate, but using two to four distinct voices within a given story is likely to be sufficient. Using one male and one female voice is readily discriminable and can be used to distinguish between primary information and supporting information. Narration can also be presented as captions or transcripts if necessary. The following describes an example of a narrated story reflecting the vignette at each CLT level.

6.1.1 CLT level 1

The top level presentation of the story (construal level 1) is illustrated in Figure 11 (In these figures, the narration appears



ORDER: Coronado to engage 010000AUG20 to detect and eliminate suspected undersea mines in commercial shipping lanes in area coordinates XXX in the Strait of Hormuz.

COMMANDER'S INTENT:

Expanded Purpose: Clarify nature of mine-like objects and eliminate any threats prior to stated 16 August advisory date.

Key Actions:
 Identify suspected mines.
 Eliminate confirmed mines in international waters.
 Avoid Iranian territorial waters.
 Synchronize maneuvers of all military assets.
 Safely recover all assets without incident.

Endstate: International waters verified safe for travel and warning notices lifted, zero collateral damage.

Expressives: Avoid overt display of force.

FIGURE 10
An example unstructured statement of commander's intent in a typical military operational order.

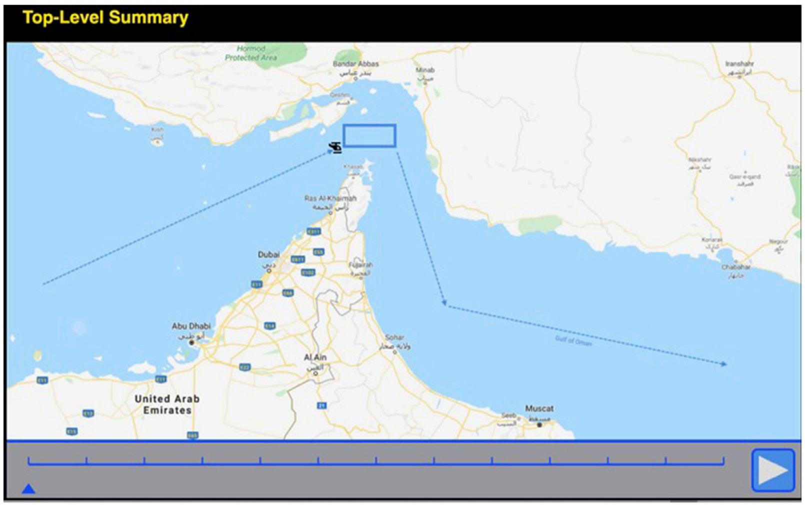


FIGURE 11
Visual component of the top level summary of a story.

below the figure caption.) As the narration is played, the helicopter icon moves on the screen and the pointer moves across the timeline at the bottom of the map. Note the timeline at the bottom of the map; multiple static panels can be used to depict changes in the position of participants (and other aspects of the operation) at different times, where the small caret below the timeline shows the time in question. The large arrow at the right of the timeline is the "play" button.

CLT one Narration. Voice 1: *The RQ-X will find and destroy shallow mines in the Strait of Hormuz on 15 August 2020. It will not enter the Iranian No Fly Zone.*

6.1.2 CLT level 2

The presentation of the story at construal level 2, the quick overview level, is illustrated in Figure 12. The narration of this panel adds information about purpose and more detail about the time of the operation. This view is at the end of the mission

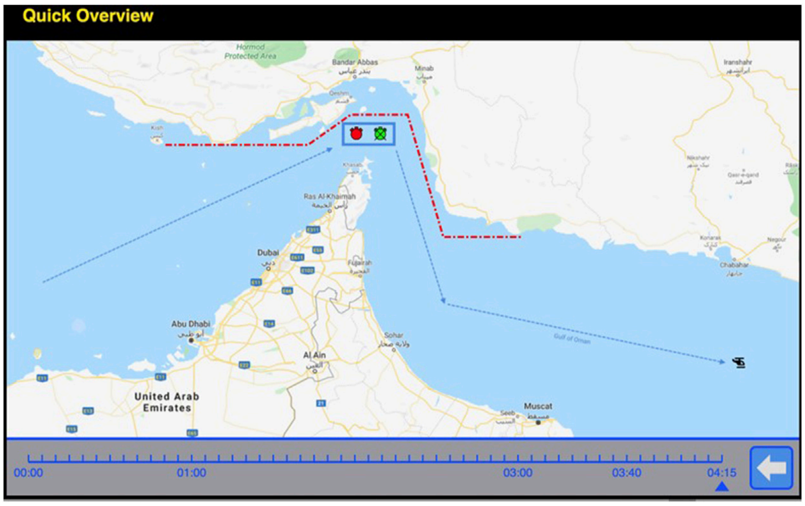


FIGURE 12
First panel of the quick overview version of the story.

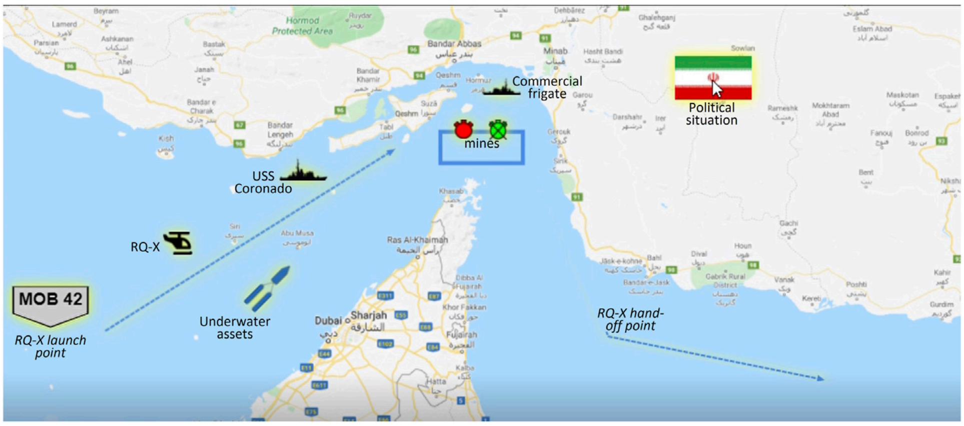


FIGURE 13
Second panel of the quick overview version of the story.

timeline (4 h, 15 min), showing both successful (green circle) and unsuccessful (red circle) mine destruction by the RQ-X. The red dotted line is the keep-out or “no-fly” zone.

Narration. Voice 1: *To reduce the threat from mines and to support open sea lines of communication, the RQ-X will find and destroy shallow mines in the Strait of Hormuz on 15 August 2020 commencing at time zero one zero zero Zulu. It will not enter the Iranian No Fly Zone.*

Figure 13 is a panel that depicts the full mission from the starting point. The icons on this panel can be selected to show more detail about a particular player in the mission (these icons have been annotated with titles in this figure). Note: the USS San Diego is not shown in this panel. The narration adds the details of the timeline.

Narration. Voice 1: *Control of the RQ-X will be transferred from USS San Diego to USS Coronado at zero zero zero hours Zulu. Transit time to the OPAREA is 1 h. In the OPAREA, RQ-X will find and plot mines down to depths of 25 m. If mines are detected at depths of 10 m or less, RQ-X will engage and detonate them. After a maximum of 2 h on station, RQ-X will depart the OPAREA. Tactical control will be transferred back to San Diego no later than zero three forty. Maximum endurance requires RQ-X to be recovered no later than zero four fifteen by San Diego* (Note: “OPAREA” refers to the operational area of the mission.)

6.1.3 CLT level 3

As the construal levels progress, more detail is added. Complete presentation of this detail is not practical in the format of this article but the concept has been completely developed in a simple simulation tool. Instead, the increasing levels of detail are illustrated in how the story opens at each level. At construal level 3, the story begins with the most salient background point, thereby explaining the motivation for the mission. At this level, a second voice is added to provide background and supplemental information, and the first voice provides the primary information. The visuals are a sequence of map views, accompanied by the following narration:

Narration. Voice 1*: To reduce the threat from mines and to support open sea lines of communication, the RQ-X will find and destroy shallow mines in the Strait of Hormuz on 15 August 2020 commencing at zero one zero zero Zulu.*

Voice 2: *Overhead assets indicated the potential presence of mine-like objects in the strait of Hormuz. Analysis of those images suggested a mine field with mine-like objects dispersed at multiple depths. USS Coronado was on patrol in the Persian Gulf, configured with its countermine warfare mission module.*

Voice 1: *USS Coronado received orders from CENTCOM to reconnoiter the area of the suspected mine field and to clear it of mines within 24 h. Coronado prepared a plan to launch its UAV to begin surveillance for shallow mines while its UUV assets were being prepared to continue surveillance and perform mine neutralization.*

Voice 2: *As Coronado began preparing its unmanned air and underwater systems to surveil the OPAREA, it discovered maintenance issues that threatened the completion of the mission within the specified time. Mission planners aboard Coronado discovered the presence of the RQ-X UAV, equipped with its airborne countermine system, under tactical control of the USS San Diego, located at mobile operating base 42. Mission planners determined that RQ-X was capable of finding and neutralizing shallow mines in the OPAREA.*

6.1.4 CLT level 4

At construal level 4, additional detail is added to provide a justification for why the operation is warranted. The visuals are a sequence of map views and image intelligence accompanied by the following narration:

Narration. Voice 1: To reduce *the threat from mines and to support open sea lines of communication, the RQ-X will find and destroy shallow mines in the Strait of Hormuz on 15 August 2020 commencing at zero one zero zero Zulu.*

Voice 2: *On 13 August, an Iranian-flagged surface vessel registered as a nautical research platform was observed executing a pattern consistent with laying a mine field in the straights. As this was happening, the Islamic Republic of Iran issued a general statement asserting its right to control traffic through the Strait of Hormuz. Overhead assets indicated the potential presence of mine-like objects in the straights. Advisories were issued to commercial ships planning to transit the area.*

Voice 1: *CENTCOM tasked USS Coronado to reconnoiter the area of the suspected mine field and to clear it of mines within 24 h. Rules of engagement specify no UAV reconnaissance below ten thousand feet within 10 nautical miles of the Iranian coastline* (Note: "CENTCOM" refers to Central Command.)

Note that the last set of Voice 2 narration is the same as at level 3.

6.1.5 CLT level 5

At construal level 5, still more detail is added in the introduction. For example, details about information from overhead assets is expanded to include which assets were used and the contribution each made. Additional detail is added about the hostile pronouncements by the adversary and the involvement of key coalition partners. The visuals continue to be a series of maps and image intelligence captures, now also supplemented by a video clip of a speech by the Iranian president and a copy of a notice to mariners issued by the United Kingdom. This narration is included in its entirety to provide context for the full mission. Refer to Figure 15 for the players.

Narration. Voice 1: *To reduce the threat from mines and to support open sea lines of communication, the RQ-X will find and destroy shallow mines in the Strait of Hormuz on 15 August 2020 commencing at zero one zero zero Zulu. It will launch from Mobile Operating Base 42, transit to the op area, find and plot mines down to 25 m, neutralize mines down to* 10 m*, and then transit for handoff to the USS San Diego for recovery.*

Voice 2: *On 13 August, an Iranian-flagged surface vessel registered as a nautical research platform was observed executing a pattern consistent with laying a mine field in the straights. The vessel is the Khalije Fars Voyager, registered to the Iranian Defense Ministry's Marine Industries Organization, which is affiliated with the Iranian National Institute for Oceanography and Atmospheric Science. It is equipped with a data transfer system that uses satellite communication, and is capable of deploying a precise pattern of bathymetric buoys. This capability can also be used to automatically deploy a wide variety of mines.*

Voice 2: *The vessel was tracked by Triton, as part of routine maritime surveillance. The Triton mission crew at NAS Jacksonville noticed an anomaly in the AIS report from the vessel. The AIS transmission indicated a planned route along the coast, consistent with normal bathymetry scans. The route as executed deviated from the planned route and followed the same general pattern observed in previous mine warfare training missions conducted by the Iranian Navy. The most recent of these missions was conducted by the Konarak, a Hendijan-class support vessel outfitted with anti-ship missiles and mine laying systems, in December 2019. It departed from the Iranian Navy port in its namesake city, Korarak, proceeded to the straits where it* *laid a diagonal pattern of dummy mines, then returned to base* (Note: "NAS" refers to Naval Air Station and "AIS" refers to Automated Identification System on the ship.)

Voice 2: *In response to the AIS anomaly report from Triton, an EA-18 Growler was diverted from routine patrol and tasked to do a specific emitter identification collection on the Iranian vessel. The maritime navigation radar and satellite communications data link transmitters were identified as the Khalije Fars Voyager, which was also the visible hull marking. But the AIS transmitter and an encrypted UHF line of sight radio were identified as from the Korarak. The Konarak was severely damaged in a friendly fire accident in May 2020, and repairs have not been completed. ONI assesses that these components from the Korarak were retrofitted onto the Khalie Fars Voyager to help provide deception regarding the nature of the mine laying mission.*

Voice 2: *As this was happening, the Islamic Republic of Iran issued a general statement asserting its right to control traffic through the Strait of Hormuz. The Iranian President reminded the world that the body of water is called the Persian Gulf for good reason. As part of a speech on regional tensions, the president stated that Iranian patience and tolerance for intrusion in its territorial waters was strained by repeated provocations from the Gulf states, from Britain, and from the United States. The Iranian foreign minister released a statement addressed to 42 ambassadors warning of severe consequences if the provocations from their nations continue.*

Voice 2: *Overhead assets indicated the potential presence of mine-like objects in the straights. Triton descended below 45,000 feet and collected detailed hyperspectral images. These images from Triton indicated the presence of potential shallow mines. A geosynchronous KH-11 satellite was tasked to perform a multi-spectral collection on the area. Analysis of those images suggested a mine field with mine-like objects dispersed at multiple depths.*

Voice 2: *Advisories were issued to commercial ships planning to transit the area. The United Kingdom Maritime Trade Operations office issued a Notice to Mariners regarding the heightened threat level in what was already categorized as a high risk area. This notice contained an estimate that the situation might be resolved by 16 August 2020, about 48 h after the notice was issued.*

Voice 1: *US Central Command reviewed and assented to the notice before it was sent.*

Voice 2: *USS Coronado was on routine patrol in the Persian Gulf. The Coronado was configured with its countermine warfare mission package, which includes a UAV platform with a sensor suite capable of detecting shallow mines, and UUV assets capable of detecting deeper mines. Other UUV assets on Coronado can neutralize many mines.*

Voice 1: *CENTCOM tasked USS Coronado to reconnoiter the area of the suspected mine field and to clear it of mines within 24 h. Coronado prepared a plan to launch its UAV to begin surveillance for shallow mines while its UUV assets were being prepared to continue surveillance and perform mine neutralization.*

Voice 1: *Rules of engagement specify no UAV reconnaissance below ten thousand feet within 10 nautical miles of the Iranian coastline.*

Voice 2: *Use of the sensor to detect mines by the Coronado's UAV requires operation at a maximum altitude of 2000 feet, and better performance is obtained at altitudes of 500 feet or below. The northwest corner of the OPAREA lies approximately nine and one-half nautical miles from the coast of the island of Qeshm.*

Voice 2: *As Coronado began preparing its unmanned air and underwater systems to surveil the OPAREA, it discovered maintenance issues that threatened the completion of the mission within the specified time of 24 h. Mission planners aboard Coronado discovered the presence of the RQ-X UAV, equipped with its airborne countermine system, under tactical control of the USS San Diego. The RQ-X is an experimental platform undergoing a technology demonstration phase in live operations. The San Diego has been operating the RQ-X since 1 August. When Coronado discovered the RQ-X, it was located at a mobile operating base, MOB 42.*

Voice 2: *MOB 42 is currently located on the island of Zirku, which is part of the United Arab Emirates. A private commercial airfield on the island allows MOB 42 to use its runways and other support facilities. The RQ-X landed there for routine maintenance and refueling. It was scheduled to remain there for approximately 24 h, awaiting a landing slot back on the San Diego.*

Voice 2: *The RQ-X is capable of detecting mines down to a depth of 25 m, and neutralizing them at depths of no more than 10 m. To detect the mines, RQ-X uses a COTS sensor with three pulsed lasers. In littoral waters, the TRW sensor can detect mines down to about 25 m.*

Voice 2: *The RQ-X is also capable of neutralizing shallow mines using a directed energy weapon developed by the Navy Research Laboratory. In littoral waters, the weapon is effective against most mines down to a depth of 10 m, although it is most effective against mines floating on or very near the surface.*

Voice 2: *After the directed energy weapon attempts to destroy the mine, the TRW sensor system is re-engaged to determine whether the mine-like object is still present in the water.*

6.1.6 CLT level 6

At the construal level 6, additional supporting details are added for the interested consumer. Details about how the sensors and weapons will operate are of interest to few users, but these may be germane for those users to assess whether the asset can provide the necessary capabilities. Examples include the following narration, accompanied by appropriate imagery.

Voice 2: *This sensor was originally developed by a company called TRW. It performs an alternating circular versus raster scan with the three beams to detect solid objects in the water, and to* *estimate object size. Objects detected that are within the range of sizes for mine-like objects are further probed by the sensor in a lidar mode, to estimate depth. The depth estimate is more accurate if the sensor is directly above the object.*

Voice 2: *This weapon, not yet nomenclatured, focuses a coherent beam of energy on the object to find a centroid, then successively adds more coherent beams every 5 seconds until the object begins to splinter, usually from premature detonation or from melting. If the object does not show signs of disintegration after 45 s, the weapon will attempt to find an alternate centroid point and repeat the attack. A maximum of three attempts will be made. Some mines may be neutralized by the attack even though they may not disintegrate. The directed energy attack may defeat the sensor, the fuse, or the other control circuitry in the mine.*

6.1.7 Simulation organization

One key to creating and maintaining interest in the story is the match between the construal level of the user and the level of detail in the story as presented. Too much unwanted detail can prompt users to lose interest, and not enough detail can produce frustration, especially if the missing details are needed for task performance. We created a simple user interface using concepts from the popular Facebook application to provide background information on the capabilities of the systems involved and reference to historical missions. As was shown in Figure 13, the user could also select icons to gain more detail about selected players of mission steps (effectively drilling down into the CLT level 6 narrative. Future research will automate information feeds so that we can evaluate automated pop-up of detail as mission events change.

7 Discussion

In this work we applied three new conceptual approaches to design and manage information flow in human-machine teaming situations. We applied construal level theory as an organizing approach to managing information detail in complex mission situations. We formalized the language we call "RECITAL" to constrain that subjective and objective information based on concepts of intent, rules, and delegated authority. To design the information flow, we modeled the human-machine distributed teams as a systemic control hierarchy. The combination of these approaches was used to design and demonstrate a simple command and control user interface operating at six CLT levels using progressive disclosure concepts.

In a complex command and control hierarchy, there is an inherent risk of operators misperceiving and incorrectly abstracting or adapting to the information disseminated. The application of CLT provides a novel approach to the structure and presentation of such information in complex mission environments. By infusing CLT into a UI design, we ensure a better fit to the operator's mental representation of the information can be realized, and communication and comprehension in a C2 hierarchy can be improved based on an individual's specific level of psychological distance from the information and context. In this initial work, a UI concept was developed for representing difficult ideas such as intent, rules, control, and outcomes in a simulatable model. Such a model is the foundation for an advanced UI that uses CLT to disseminate mission information in the most efficient possible form.

In this work we present a novel approach to address the subjective nature of expressions of intent, rules, and authorities in complex missions. These expressions are typically composed of unstructured text, delivered from multiple systems to multiple command levels, with various interpretations that gradually make the context of the order seem more distal to an operator. Today, it would not be possible for a machine to process this unstructured text as a means to make real-time decisions, because so much of the contextual information is inferred by operators as a function of training and experience. However, many increasingly "intelligent" machine platforms are making progress with this type of inference by mining additional information in the external context.

Additional research is ongoing to model the RECITAL hierarchical information flows, and the potential definition of a set of applications that would deliver that information to the various planners and operators at different levels of command. At this point, the provision of contextual information is only modelled as a single black box entity in the control flow. Eventually this would be a set of software applications. We envision that these applications would present data in a rich narrative form similar to the stories presented in section 6. Research that uses artificial intelligence to automate narrative generation is being explored as a means to scale the approach. This work provides a conceptual platform for additional research on machine learning approaches to search for and select the contextual information, as well as to learn individual user preferences that help to contextually manage CLT. Finally, the conceptual approach is being extended to a set of additional mission scenarios with more complex distributed autonomy to further evaluate and generalize its applicability and benefits.

8 Conclusion

This research is highly conceptual at this time but is being published because it represents a novel approach to understanding of information flows in human-machine teaming. While many prevailing narratives about distributed automation reflect automation of inefficient human tasks, this work addresses automation of information flows, particularly contextual information, that enable human (and perhaps machine) operators to make better task-related decisions. This mirrors the concepts being observed in popular automation platforms like Google and Waze.

This research makes several fundamental hypotheses about task related activities in human-machine teams. The first is that expressions of intent, rules, and transfer of authority are present in the interaction of human machine teams, just as they are in human-human teams. The second is that these interactions tend to follow information produced and consumed in hierarchical control structures and the information can be modeled as a control flow. The third is that the design of the produced/consumed information interaction between humans and machines can be designed using construal level theory, and that there are six observable levels that reoccur in these interactions. Finally, the research found that visual information combined with narratives is effective at representing construal level information.

Data availability statement

The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding author.

Author contributions

DF developed the basic RECITAL concept, applied CLT to the concept, and developed the initial UI demonstration. A key aspect of this work was the use of narrative at different construal layers, leading to the theory that six recognizable construal levels exist. TM contributed the systems engineering approach including the use of STPA and the hierarchical information flows.

Funding

This work was partially funded by the Office of Naval Research (ONR), under contract N00014-20-C-2004. Dr. Jeffrey Morrison, ONR Code 34, is the ONR Program Officer.

Acknowledgments

Thank you to C Jay Battle, Jake Baumbaugh, Zachary Harshaw, and Matthew Schmouder who all contributed to the development of the user interface concepts described in this article. Thank you to Kayla Noble for reviewing and editing this article.

Conflict of interest

Authors TM and DF were employed by Stephenson Technologies Corporation.

The handling editor declared a past co-authorship with one of the authors TM.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

1. Associated Press, "Tesla driver's complaint being looked into by US regulators", 2021. Available at: https://apnews.com/article/technology-business-traffic-9c3d3c75d9c668dbb1a8cdbbe883ff88 (Accessed 22 May 2022).

2. Tesla Corporation, Tesla 3 owners manual. Available at: https://www.tesla.com/ownersmanual/model3/en_nz/GUID-0535381F-643F-4C60-85AB-1783E723B9B6.html (Accessed 22 May 2022).

3. U.S. Department of Defense Joint Publication 3-0. *Joint operations* (2018). Available at: https://www.jcs.mil/Doctrine/Joint-Doctrine-Pubs/3-0-Operations-Series/ (Accessed 22 May 2022).

4. U.S. Department of Defense Joint Publication 5-0. *Joint planning* (2020). Available at: https://www.jcs.mil/Doctrine/Joint-Doctrine-Pubs/5-0-Planning-Series/ (Accessed 22 May 2022).

5. International Institute of Humanitarian Law. *Rules of engagement handbook* (2009). Available at: https://iihl.org/wp-content/uploads/2017/11/ROE-HANDBOOK-ENGLISH.pdf (Accessed 22 May 2022).

6. Larsen E (2017). "Use these three military lessons to Be decisive in business." Forbes. Available at: https://www.forbes.com/sites/eriklarson/2017/04/25/three-things-the-military-taught-me-about-being-decisive-in-business/?sh=584262a372ff (Accessed 1 October 2022).

7. Colan L. *Why you need to lay down ground rules for a high-performing team*. New Delhi: Inc (2014). Available at: https://www.inc.com/lee-colan/rules-of-engagement-for-high-performing-teams.html (Accessed 1 October 2022).

8. Gustavsson PM, Hieb M, Eriksson P, More P, Niklasson L. Machine interpretable representation of commander's intent. In: 13th International Command and Control Research and Technology Symposium; June 17-19, 2009; Bellevue, Washington, USA.

9. Leveson NG. *Engineering a safer world: Systems thinking applied to safety*. United States: MIT Press (2012).

10. Cockcroft A, "COVID-19 hazard analysis using STPA," 2020. Available at: https://adrianco.medium.com/covid-19-hazard-analysis-using-stpa-3a8c6d2e40a9 (Accessed 22 May 2022).

11. Leveson NG, Thomas JP, STPA handbook, p. 179, 2018. Available at: http://psas.scripts.mit.edu/home/get_file.php?name=STPA_handbook.pdf (Accessed 22 May 2022).

12. CNET, Cnet, Waze image Available at: www.cnet.com/roadshow/news/google-assistant-waze-easier-reporting-less-distraction/ (Accessed 22 May 2022).

13. Nielson J. *Progressive disclosure*. California, United States: Nielson Norman Group (2006). Available at: https://www.nngroup.com/articles/progressive-disclosure/ (Accessed 22 May 2022).

14. Spillers F (2004). Progressive disclosure- the best interaction design technique? Experience dynamics. Available at: https://www.experiencedynamics.com/blog/2004/03/progressive-disclosure-best-interaction-design-technique (Accessed 22 May 2022).

15. Trope Y, Liberman N. Construal-level theory of psychological distance. *Psychol Rev* (2010) 117(2):440–63. doi:10.1037/a0018963

16. Trope YL. Construal level theory. In: Van Lange PK, editor. *Handbook of theories of social psychology*. Washington DC: Sage Publications Ltd (2012). p. 118–34.

17. Bar-Anan Y, Liberman N, Trope Y. The association between psychological distance and construal level: Evidence from an implicit association test. *J Exp Psychol Gen* (2006) 135(4):609–22. doi:10.1037/0096-3445.135.4.609

18. Google Google, Available at: google.com/search?q=braves+score (Accessed 22 May 2022).

19. Shafer G. *A mathematical theory of evidence*. Princeton, NJ: Princeton University Press (1976).

20. Endsley MR. *Human-AI teaming: State-of-the-Art and research needs*. Washington, DC: The National Academies of Sciences-Engineering-Medicine National Academies Press (2021). Available at: https://www.nap.edu/catalog/26355/human-ai-teaming-state-of-the-art-and-research-needs.

21. Chaal M, Banda V, Osiris A, Glomsrud JA, Basnet S, Hirdaris S, et al. A framework to model the STPA hierarchical control structure of an autonomous ship. *Saf Sci* (2020) 132:104939. doi:10.1016/j.ssci.2020.104939

TYPE Perspective
PUBLISHED 23 November 2022
DOI 10.3389/fphy.2022.942896

Trust and communication in human-machine teaming

Memunat A. Ibrahim[1]*, Zena Assaad[2] and Elizabeth Williams[1]

[1]School of Cybernetics, The Australian National University, Canberra, ACT, Australia, [2]School of Engineering, The Australian National University, Canberra, ACT, Australia

OPEN ACCESS

EDITED BY
William Frere Lawless,
Paine College, United States

REVIEWED BY
Katie Parnell,
University of Southampton,
United Kingdom
Josh Ekandem,
Intel, United States

*CORRESPONDENCE
Memunat A. Ibrahim,
memunat.ibrahim@anu.edu.au

SPECIALTY SECTION
This article was submitted to
Interdisciplinary Physics,
a section of the journal
Frontiers in Physics

RECEIVED 13 May 2022
ACCEPTED 10 November 2022
PUBLISHED 23 November 2022

CITATION
Ibrahim MA, Assaad Z and Williams E (2022), Trust and communication in human-machine teaming.
Front. Phys. 10:942896.
doi: 10.3389/fphy.2022.942896

Intelligent highly-automated systems (HASs) are increasingly being created and deployed at scale with a broad range of purposes and operational environments. In uncertain or safety-critical environments, HASs are frequently designed to seamlessly co-operate with humans, thus, forming human-machine teams (HMTs) to achieve collective goals. Trust plays an important role in this dynamic: humans need to be able to develop an appropriate level of trust in their HAS teammate(s) to form an HMT capable of safely and effectively working towards goal completion. Using Autonomous Ground Vehicles (AGVs) as an example of an HAS used in dynamic social contexts, we explore interdependent teaming and communication between humans and AGVs in different contexts and examine the role of trust and communication in these teams. Drawing on lessons from the AGV example for the design of an HAS used for an HMT more broadly, we argue that trust is experienced and built differently in different contexts, necessitating context-specific approaches to designing for trust in such systems.

KEYWORDS

trust, autonomous vehicle, human-machine teaming, communication, context

Introduction

Automation is defined as "technology that actively selects data, transforms information, makes decisions, or controls processes" [1]. These technologies are typically designed to help humans achieve their goals more efficiently, and can be classified according to purpose: information acquisition, information analysis, decision selection, action implementation, and automated systems monitoring [2, 3]. A highly-automated system (HAS) may incorporate one or more automation types, and is designed to pursue specific goals with some independence [4]. An HAS designed to operate in uncertain environments is often required to form a dynamic relationship with one or more humans to achieve a goal, forming a human-machine team (HMT). In this perspective, we explore the role of trust in HMTs with a focus on contextual factors shaping trust dynamics in an HMT, as a means of guiding "trustworthy" HMT systems design for diverse and uncertain contexts - an unsolved problem [5].

Human-machine teaming

Human-machine teaming refers to the relationship between a human and machine (typically a HAS) that encompasses the shared pursuit of a common goal [6] as set by humans. The nature of this relationship varies depending on the distribution of decision-making power and roles among the teammates. For example, an HAS may have little influence over the team's collective actions if it only helps the human make decisions or only acts as instructed by the human. Alternatively, an HAS with the capacity to independently act on its environment in alignment with its team's goal, with or without human oversight, could have a significant influence over the team's actions [4]. In some HMTs, the distribution of decision-making and agency between human and HAS teammates is dynamic—it changes with time and circumstance. This distribution can be beneficial: both human and HAS teammates have different strengths and response timescales; dynamically allocating agency can allow for collaborations that optimise the teammates' contributions. As with any teamwork, achieving these benefits depends heavily on the establishment of an effective relationship between human and HAS teammates.

Designing for effective relationships between human and HAS teammates can prove challenging [4]—particularly when an HAS incorporates artificial intelligence (AI) capabilities. AI capabilities are often used in HASs to enable intelligent, dynamic actions. Essentially, AI imbues HASs with the ability to learn and evolve over time from experience [7]. This learning ability is typically probabilistic, which can yield unpredictable behaviour. This unpredictability is intensified when the HAS is used in real-world contexts characterised by dynamic interactions. One example of such contexts is road traffic: a setting consisting of multiple heterogenous autonomous actors acting in the same environment towards their individual goals, with their interactions often guided by shared rules and understandings. For HMTs operating in such environments, there may be unpredictable aspects of teammate interactions that emerge as a function of the HAS capabilities, the human teammate, the team dynamics, and the complexity and unpredictability of the contexts they operate in. This makes the HMTs adoption in dynamic contexts risky and potentially costly for humans involved—both within the HMT, and in their environments [8–11].

Trust in automation is a key enabler of HMT collaborations and automation adoption. Research shows that trust is key in the successful teaming of dissimilar heterogenous agents involving humans [12]. Trust reflects the degree of confidence a person may have in another actor and can shape human-automation interactions [2]. As noted in [2], trust's importance in a technology's adoption correlates with the complexity of the automation and its roles, how critical their deployed environment is, and perceived risks (e.g. [13]). Trust is generally important and useful in:

1. Guiding the design of automation that facilitates productive HMT collaboration and appropriate interactions [2]; and
2. Designing automation with the goal of mitigating the potential negative consequences of their use [2].

In the remainder of this perspective, we focus our exploration of trust in HMTs on HASs designed for large-scale deployment in social settings characterised by dynamic interactions, risks, and uncertainties requiring contextual considerations. To facilitate this argument, we will use the example of an AGV on the road. AGV driving automation systems are HASs that demonstrate all five categories of automation identified in [2, 3]; form part of a HMT; can be designed to dynamically shift roles between a human operator and itself; and operate in diverse, complex, and safety-critical social environments. AGVs deployed in road traffic environments are therefore useful for exploring trust's role in HMTs operating in social contexts, and demonstrating the need to consider their potential contexts of use in HAS design. To facilitate this exploration, we begin by defining AGVs and exploring some of their properties, considering AGVs as individual agents and exploring AGVs in autonomous teams.

Autonomous ground vehicles—An example

AGVs include driving HAS that, depending on their design, may have the capacity to achieve partial to full autonomy, meaning that the system's actions can range from providing advice to a human driver to taking full control of driving operations. Their intelligent driving capabilities are often enabled by AI. In the case of AGVs, the HMT consists of a driving HAS and the human driver.

To describe the nature of HMT dynamics between a human operator and an HAS during driving, we draw on the Society of Automotive Engineers (SAE) taxonomy [14] for driving automation systems. The SAE levels describe the capabilities and roles of driving automation and humans at different automation levels. According to the SAE standard, Level 0 vehicles offer no driving automation, while vehicles at level 1 and beyond incorporate driving automation that provides varying levels of support and control when engaged. The human and HAS have joint control of either longitudinal or lateral vehicle motion in level 1, while for level 2, the human actively supervises the system. Level 3–5 vehicles incorporate an Automated Driving System (ADS)—in-vehicle HAS that provides automated driving capabilities that allow for partial to full driverless operation of AGVs. Level 3 vehicles can perform driving conditionally and require humans to serve as a "fallback-ready user"—a human teammate that can take over driving in the vehicle or remotely as appropriate. Level 4 and 5 vehicles perform driving autonomously (albeit in limited circumstances for level 4) and do not need a fallback ready user during operation [14].

Driving HASs from levels 1–4 are increasingly being integrated in vehicles because they promise to improve road safety. This promise can only be achieved if people are receptive of AGVs, use them, and if AGVs operate safely, in a socially acceptable manner when in use. We are already witnessing the trialling and roll out of level 1–4 AGVs in societies—for example, Tesla's Autopilot features, or China's first fully driverless taxis—the Baidu self-driving taxis.

These AGVs operate in societies with humans, including a human co-driver on roads with other human and autonomous road agents. They use road resources and infrastructure alongside other road users in diverse socioeconomic contexts. Consider, for example, the operation of Level 3 AGVs on the road. When engaged, the ADS and human driver complement each other as co-drivers, playing interdependent dynamic roles in ensuring the safe navigation of AGVs to their destinations. This requires the human driver and the ADS to continuously communicate with each other and their environments through sensing, monitoring, and team acting. This example demonstrates an HMT in which the human and machine share decision-making and action implementation control. In such an HMT, there are two interdependent dynamical aspects to consider: that of the environment the HMT acts, and the team itself.

Within the HMT, team dynamics are shaped by the capabilities of each teammate, as well as the roles they are expected to play in achieving the team's goals set by the human teammate. In AGVs, increased automation made possible by increased cognitive capability and dynamic adaptability of the ADS comes at a price: adapting in real-time to the surrounding environment. This can lead to the ADS exhibiting unpredictable behavior, particularly in situations they have not been designed for nor are familiar with, impacting trust.

The potential for unpredictability in AGVs has been demonstrated multiple times—e.g., a Tesla in automated driving mode nearly hitting an individual [15], or the Uber self-driving car crash resulting in the death of a jaywalking pedestrian in Arizona [16]. In the case of the Uber crash, the AGV was struggling to classify the jaywalking pedestrian, while its human operator was paying attention to her tablet. Both were operating independently—unaware of each other's activities until too late [17].

Both examples illustrate the challenge AGVs and their human teammates face in operating on the road that needs to be considered and designed for in ADS: the diverse and dynamic nature of road transport environments. While transport infrastructure facilitates some predictability through traffic lights and stop signs, the inclusion of human agency—within and outside vehicles—creates an inherently unpredictable environment, one that has been found to vary significantly depending on infrastructure and social norms [18, 19].

An unpredictable environment combined with increased dependency on the ADS by the human creates the opportunity for unpredictable reactions by the human or the AGV teammate to that environment. To achieve AGV use at scale, HMTs will need to demonstrate the ability to act and react appropriately to achieve their collective goals safely and responsively in any environmental context. This requirement poses a significant design challenge in which the HMT and its environments are dynamic and inherently unpredictable.

Trust—within an HMT and within societies where HMTs may operate—is an important factor that affects the adoption and safe use of HASs. It is a dynamic construct that can help us to understand HASs, HMTs and their environments, and to design for their interactions. Trust definitions are subjective and contextual, and one's understanding may be shaped by experiences in different research fields, cultures or contexts [1, 12, 20]. With this in mind, we explore and define trust, first broadly, in the context of HASs, and then specifically for AGVs.

Trust and communication in AGV human-machine teams

Trust is widely researched across disciplines ranging from engineering to psychology, economics, etc. Trust as a social concept is interpersonal, and is researched as existing within relationships [2, 12]. We adopt this trust definition: the "willingness of a party to be vulnerable to the actions of another party based on the expectation that the other will perform a particular action important to the trustor, irrespective of the ability to monitor or control that other party" [12]. By this definition, HMT teamwork is easily understood as mutual dependence from a shared awareness (e.g. [21]).

Over the past century, efforts towards researching and developing trustworthy AI and human-automation trust have increased, as have the complexity and deployment rate of HASs. With regards to automation, human trust can be defined as "the attitude that an agent will help achieve an individual's goals in a situation characterized by uncertainty and vulnerability" [1]. In this definition, agent can refer to humans or automation systems. Specific to AI systems, the High-Level Expert Group on Artificial Intelligence defined trustworthy AI systems as systems where trust is established in their design, development, deployment, and use [22]. Trustworthy AI refers to AI systems that are assured to act in the trusting party's interest [23], and society at large.

Trust in automation varies depending on the automation, their context of use, and the human operator [2]. All these need to be considered holistically in trustworthy automation design. In the context of an HMT, trust is usually one-sided: humans need to trust their automated teammates to collaborate effectively, but an automation agent within the team has no inherent knowledge of "trust" in the human sense. Humans tend to evaluate the trustworthiness of other agents—HASs included—based on their perceived abilities, integrity, and benevolence [12]. An automation's association with trust relates specifically to its design: its actions and communication must foster an appropriate trust level with the humans it interacts with.

In AGVs, as with many other safety-critical HAS, trust is necessary for a human driver to willingly collaborate with the driving automation [24]. Hence, it is useful in understanding how they might interact with ADS. Trust development is dynamic. In HMTs, human and HAS teammates develop mutual expectations and an understanding of one another over time [1] as they interact in a given context. One way human teammates express the level of trust they have in an automation is through reliance or compliance, which may vary in different use cases [1, 2]. For AGVs, for example, a human operator may rely on an ADS to take the lead after it safely navigates a familiar well-marked road, while opting to take full control when navigating an unfamiliar school crossing.

To ensure an appropriate level of reliance on an ADS in uncertain and risky situations, humans need to develop and maintain appropriate trust with the ADS. To achieve this, appropriate communication of the automation's capabilities, intentions, decisions, and actions is important. But appropriate communication is also contextual and dynamic: the nature of the automation, the human operator, and the context or environment the HMT operate in all inform the potential risks involved in navigating a given situation as well as the appropriate communication methods within and outside the HMT [2]. In the next section, we explore the importance of communication in trust development and maintenance in HMTs with a continued focus on AGVs.

Communication in AGV HMTs

An AGV HMT operates in safety-critical situations where lack of cooperation can result in fatal accidents, as observed from the aforementioned Uber accident in Arizona [16]. In general, analyses show that accidents can stem from inappropriate trust. Inappropriate trust in AGVs can include overtrust, where a human operator trusts an ADS too much, leading to human inaction at crucial moments, or undertrust, where humans do not trust the ADS enough, resulting in a human overtaking ADS duties inappropriately [1]. Inappropriate trust can be caused by inappropriate communication of information between automation and its teammate [1, 2, 25, 26].

Hoff and Bashir [2] summarized the design recommendations for trustworthy automation as: increasing anthropomorphism with consideration of user preferences, simplifying user interfaces, ensuring an automation's communication style appears trustworthy, providing users with accurate and continuous feedback on its reliability, explaining their behaviours, and increasing automation feedback and transparency. The Chartered Institute for Ergonomics and Human Factors similarly proposed nine principles to address key human factor challenges in ADS design [4]. The principles revolve around the HAS, their users and environments, and their interactions and communication. All these design recommendations highlight appropriate communication as a means of shaping trust dynamics for humans interacting with automation.

Specific to AGVs, trust can be influenced by the driving scenario [27], the ADS communication style, the interface design [28], the appropriateness of the level of detail in explanations provided to the human operator [27], and so on (see [29]). These findings, too, highlight the importance of appropriate communication and interface design in shaping trust dynamics for a successful AGV HMT. Because the context informs the risks involved, the definitions of appropriate communication and the ways appropriate communication are achieved will vary depending on the human and machine teammates and their operational context. This highlights the importance of understanding context to designing appropriately for successful teaming.

However, implementing these recommendations in an HAS used in diverse environments globally may prove challenging. For AGVs, driving culture and norms may vary in different nations, driving environments, and communities; these are usually tacitly and explicitly taught to—and understood by—human drivers; shape how human drivers operate on the road; and have been found to influence the risks involved [18]. Success for AGVs and any HAS used for HMT deployed at scale will involve responsively accounting for local cultures, norms, and communication expectations, lending support to the idea that contextually appropriate communication will play an important role in enabling effective HMTs. Some provide guidance for carrying out contextually-sensitive work for specific contexts—see, e.g., Smith [30]. But such guidance is difficult to carry out at scale.

To properly design for the diverse contexts HASs may operate in, it is important to understand these contexts and how road agents interact and communicate with one another in them. Some of the approaches used for this are: ethnographic observations, cultural probes, interviews, modelling and simulations, surveys, etc. [31–34] The choice of method is in itself shaped by context; therefore, there is currently no one systematic way for determining the appropriateness of the methods to contextual design problems.

Discussion

In this perspective, we used AGVs to explore how the contextual nature of trust can play a significant role in whether HMTs can operate at scale and how, particularly in uncertain or safety-critical scenarios. As we saw with HMTs involving AGVs, dynamic changes in the teammates' roles can combine with contextual factors (environments, communication expectations, social norms, trust definitions, etc.) to make designing for successful HMTs a significant challenge.

As a result, we see a need to change how designers think about designing for trust in HMTs. It is not enough to design HASs that are trusted by humans—we must instead aspire to design HASs that are worthy of trust in the contexts and dynamic environments in which they will operate. Central to this conclusion is the need to facilitate appropriate trust through appropriate communication and performance—both of which are context dependent.

We therefore propose questions that could guide future work on HASs that are likely to form part of HMTs in diverse contexts:

- How can we help designers create trustworthy HASs for HMTs, where "trustworthy" is defined appropriately for the contexts HMTs will operate in?
- How can we help designers (those who play a significant role in shaping HAS) understand how their own trust perception shapes the design process? And how can they design for trust as others (drivers, pedestrians, regulators, etc.) understand it?
- What approaches and frameworks can be used to systematically support these?

Most HASs—if successful—are now deployed globally. These questions suggest the need to create new frameworks for creating trustworthy HMTs—ones where the definition of "trustworthy" is dynamic, contextual, and representative of the many voices whose lives are likely to be impacted when such a system is deployed [5].

Data availability statement

The original contributions presented in the study are included in the article/Supplementary material, further inquiries can be directed to the corresponding author.

Author contributions

MI conducted the literature review underpinning this work, wrote the first draft, and took the lead in revisions leading to the submitted paper. ZA and EW contributed ideas to the structure of the paper, provided additional literature on human-machine teaming and communication aspects, provided editorial feedback, and contributed writing to sections of the paper.

Funding

MI is funded by scholarships including: ANU HDR Fee Merit Scholarship, Florence McKenzie Supplementary Scholarship in a New Branch of Engineering, and the University Research Scholarship.

Acknowledgments

We thank Amir Asadi, a fellow PhD student, for sharing relevant literature with the authors.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

1. Lee JD, See KA. Trust in automation: Designing for appropriate reliance. *hfes* (2004) 46(1):50–80. doi:10.1518/hfes.46.1.50.30392

2. Hoff KA, Bashir M. Trust in automation: Integrating empirical evidence on factors that influence trust. *Hum Factors* (2015) 57(3):407–34. doi:10.1177/0018720814547570

3. Parasuraman R, Sheridan TB, Wickens CD. A model for types and levels of human interaction with automation. *IEEE Trans Syst Man Cybern A* (2000) 30(3): 286–97. doi:10.1109/3468.844354

4. CIEHF. Human factors in highly automated systems (2022). Available from: https://ergonomics.org.uk/resource/human-factors-in-highly-automated-systems-white-paper.html (Accessed on Sep 19, 2022).

5. National Academies of Sciences E. Human-AI teaming: State-of-the-Art and research needs (2021). Available from: https://nap.nationalacademies.org/catalog/26355/human-ai-teaming-state-of-the-art-and-research-needs (Accessed on Sep 28, 2022).

6. Walliser JC, de Visser EJ, Shaw TH. Application of a system-wide trust strategy when supervising multiple autonomous agents. *Proc Hum Factors Ergon Soc Annu Meet* (2016) 60(1):133–7. doi:10.1177/1541931213601031

7. Wing JM. Trustworthy AI (2020). ArXiv200206276 Cs [Internet]Available from: http://arxiv.org/abs/2002.06276 (Accessed on Nov 23, 2021).

8. Lima A, Rocha F, Völp M, Esteves-Veríssimo P. Towards safe and secure autonomous and cooperative vehicle ecosystems. In: *Proceedings of the 2nd ACM workshop on cyber-physical systems security and privacy [internet]*. New York, NY, USA: Association for Computing Machinery (2016). p. 59

9. Liu P, Ma Y, Zuo Y. Self-driving vehicles: Are people willing to trade risks for environmental benefits? *Transportation Res A: Pol Pract* (2019) 125:139–49. doi:10.1016/j.tra.2019.05.014

10. Ramchurn SD, Stein S, Jennings NR. Trustworthy human-AI partnerships. *iScience* (2021) 24(8):102891. doi:10.1016/j.isci.2021.102891

11. Hussain R, Zeadally S. Autonomous cars: Research results, issues, and future challenges. *IEEE Commun Surv Tutorials* (2019) 21(2):1275–313. doi:10.1109/comst.2018.2869360

12. Mayer RC, Davis JH, Schoorman FD. An integrative model of organizational trust. *Acad Manage Rev* (1995) 20(3):709–34. doi:10.5465/amr.1995.9508080335

13. Williams ET, Nabavi E, Bell G, Bentley CM, Daniell KA, Derwort N, et al. Chapter 17 - begin with the human: Designing for safety and trustworthiness in cyber-physical systems. In: WF Lawless, R Mittu, DA Sofge, editors. *Human-machine shared contexts*. Academic Press (2020). Available from: https://www.sciencedirect.com/science/article/pii/B9780128205433000171 (Accessed on Jul 4, 2022).

14. On-Road Automated Driving (ORAD). *Taxonomy and definitions for terms related to driving automation systems for on-road motor vehicles [Internet]* (2021). Available at: https://www.sae.org/content/j3016_202104 (Accessed Oct 15, 2021).

15. Hill B. This shocking video of Tesla fsd Autopilot almost hitting A pedestrian is being covered-up [internet]. HotHardware (2021). Available from: https://hothardware.com/news/tesla-fsd-autopilot-crosswalk-dmca-video-takedown (Accessed on Apr 7, 2022).

16. Uber self-driving test car involved in accident resulting in pedestrian death. TechCrunch. Available from: https://social.techcrunch.com/2018/03/19/uber-self-driving-test-car-involved-in-accident-resulting-in-pedestrian-death/(Accessed on Apr 7, 2022).

17. Lawless W. Toward a Physics of interdependence for autonomous human-machine systems: The case of the uber fatal AccidentFront phys (2018). Available from: https://www.frontiersin.org/articles/10.3389/fphy.2022.879171 (Accessed on Sep 23, 2022).

18. Nordfjærn T, Şimşekoğlu Ö, Rundmo T. Culture related to road traffic safety: A comparison of eight countries using two conceptualizations of culture. *Accid Anal Prev* (2014) 62:319–28. doi:10.1016/j.aap.2013.10.018

19. Müller L, Risto M, Emmenegger C. The social behavior of autonomous vehicles. In: *Proceedings of the 2016 ACM international joint conference on pervasive and ubiquitous computing: Adjunct*. Heidelberg Germany: ACM (2016). Available from: https://dl.acm.org/doi/10.1145/2968219.2968561 (Accessed on Dec 7, 2022).

20. Idemudia ES, Olawa BD. Once bitten, twice shy: Trust and trustworthiness from an african perspective. In: CT Kwantes BCH Kuo, editors. *Trust and trustworthiness across cultures: Implications for societies and workplaces [internet]*. Cham: Springer International Publishing (2021). doi:10.1007/978-3-030-56718-7_3

21. Sliwa J. Toward collective animal neuroscience. *Science* (2021) 374(6566):397–8. doi:10.1126/science.abm3060

22. Ethics guidelines for trustworthy AI. Shaping Europe's digital future (2022). Available from: https://digital-strategy.ec.europa.eu/en/library/ethics-guidelines-trustworthy-ai (Accessed on Apr 7, 2022).

23. The IEEE Global Initiative on Ethics of Autonomous and Intelligent Systems. *Ethically aligned design: A vision for prioritizing human well-being with autonomous and intelligent systems, version 2 [Internet]*. New York, USA: IEEE. Available from: https://standards.ieee.org/industry-connections/ec/ead-v1/ (Accessed on Apr 7, 2022).

24. Walker F, Wang J, Martens MH, Verwey WB. Gaze behaviour and electrodermal activity: Objective measures of drivers' trust in automated vehicles. *Transportation Res F: Traffic Psychol Behav* (2019) 64:401–12. doi:10.1016/j.trf.2019.05.021

25. Ekman F. Designing for appropriate trust in automated vehicles: A tentative model of trust information exchange and gestalt [Internet]. Gothenburg, Sweden: Chalmers University of Technology. Available from: https://research.chalmers.se/publication/517220.

26. Niculescu AI, Dix A, Yeo KH. Are you ready for a drive? User perspectives on autonomous vehicles. In: Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems [Internet]. (Denver, Colorado: Association for Computing Machinery (2017). p. 2810–2817. doi:10.1145/3027063.3053182

27. Ma RHY, Morris A, Herriotts P, Birrell S. Investigating what level of visual information inspires trust in a user of a highly automated vehicle. *Appl Ergon* (2021) 90:103272. doi:10.1016/j.apergo.2020.103272

28. Oliveira L, Burns C, Luton J, Iyer S, Birrell S. The influence of system transparency on trust: Evaluating interfaces in a highly automated vehicle. *Transportation Res Part F: Traffic Psychol Behav* (2020) 72:280–96. doi:10.1016/j.trf.2020.06.001

29. Merat N, Madigan R, Nordhoff S. *Human factors, user requirements, and user acceptance of ride-sharing in automated vehicles*. International Transport Forum Discussion Papers (2017). Report No.: 2017/10. Available from: https://www.oecd-ilibrary.org/transport/human-factors-user-requirements-and-user-acceptance-of-ride-sharing-in-automated-vehicles_0d3ed522-en.

30. Smith CJ. *Designing trustworthy AI: A human-machine teaming framework to guide development* (2019). ArXiv191003515 Cs Available from: http://arxiv.org/abs/1910.03515

31. Vereschak O, Bailly G, Caramiaux B. How to evaluate trust in AI-assisted decision making? A survey of empirical methodologies. *Proc ACM Hum Comput Interact* (2021) 5(CSCW2):1–39. doi:10.1145/3476068

32. Nathan LP. Sustainable information practice: An ethnographic investigation. *J Am Soc Inf Sci Technol* (2012) 63(11):2254–68. doi:10.1002/asi.22726

33. Balfe N, Sharples S, Wilson JR. Understanding is key: An analysis of factors pertaining to trust in a real-world automation system. *Hum Factors* (2018) 60(4):477–95. doi:10.1177/0018720818761256

34. Raats K, Fors V, Pink S. Trusting autonomous vehicles: An interdisciplinary approach. *Transp Res Interdiscip Perspect* (2020) 7:100201. doi:10.1016/j.trip.2020.100201

TYPE Original Research
PUBLISHED 07 December 2022
DOI 10.3389/fphy.2022.969544

OPEN ACCESS

EDITED BY
William Frere Lawless,
Paine College, United States

REVIEWED BY
Bonnie Johnson,
Naval Postgraduate School,
United States
Mito Akiyoshi,
Senshu University, Japan

*CORRESPONDENCE
Micael Vignati,
mvignati@ihmc.org

SPECIALTY SECTION
This article was submitted to
Interdisciplinary Physics,
a section of the journal
Frontiers in Physics

RECEIVED 15 June 2022
ACCEPTED 23 November 2022
PUBLISHED 07 December 2022

CITATION
Vignati M, Johnson M, Bunch L, Carff J and Duran D (2022), Interdependence design principles in practice.
Front. Phys. 10:969544.
doi: 10.3389/fphy.2022.969544

Interdependence design principles in practice

Micael Vignati*, Matthew Johnson, Larry Bunch, John Carff and Daniel Duran

Institute for Human and Machine Cognition, Pensacola, FL, United States

Adaptability lies at the heart of effective teams and it is through management of interdependence that teams are able to adapt. This makes interdependence a critical factor of human-machine teams. Nevertheless, engineers building human-machine systems still rely on the same tools and techniques used to build individual behaviors which were never designed to address the complexity that stems from interdependence in joint activity. Many engineering approaches lack any systematic rigor and formal method for identifying, managing and exploiting interdependence, which forces ad hoc solutions or workarounds. This gap between theories of interdependence and operable tooling leaves designers blind to the issues and consequences of failing to adequately address interdependence within human-machine teams. In this article, we propose an approach to operationalizing core concepts needed to address interdependence in support of adaptive teamwork. We describe a formalized structure, joint activity graphs, built on interdependence design principles to capture the essence of joint activity. We describe the runtime requirements needed to dynamically exploit joint activity graphs and to support intelligent coordination during execution. We demonstrate the effectiveness of such a structure at supporting adaptability using the Capture-the-Flag domain with heterogeneous teams of unmanned aerial vehicles and unmanned ground systems. In this dynamic adversarial domain, we show how agents can make use of the information provided by joint activity graphs to generally and pragmatically react and adapt to perturbations in the joint activity, the environment, or the team and explicitly manage and exploit interdependence to produce effective teamwork. In doing so, we demonstrate how flexible and adaptive teamwork can be achieved through formally guided design that supports effective management of interdependence.

KEYWORDS

human-machine teams, joint activity, interdependence, adaptability, joint activity graph, heterogeneous multi-agent systems

1 Introduction

Teaming is a dynamic activity that comes to life as agents[1] work together towards a common goal. Understanding the factors of teaming is critical to produce effective human-machine teams. Approaches to understanding teaming are as numerous as they are multidisciplinary, involving human sciences (sociology [1], linguistics [2], psychology, etc.), engineering sciences (distributed artificial intelligence, constraint satisfaction problems, planning [3], synchrony [4]) and human machine cognition often trying to bridge the two (theory of mind, common ground [5], communication, trust [6]). On one hand, research in theoretical models for understanding teaming is extensive and cohesive, with some empirically deriving principles of cooperation (e.g., in cognition enabled multi-agent systems [1]). On the other hand, the applied aspect of teaming and its understanding has been assessed to be far less consistent across the research community [7]. As systems become more sophisticated and take on increasingly complex roles, the need for tools which help researchers, designers and engineers consider and exploit all dimensions of teaming are key for both effectiveness and acceptance.

According to the concept of bounded rationality [8], agents are required to work together in any system of substantial complexity. Whether due to the distribution of skills, capabilities and knowledge or simply to improve aspects of performance, teaming is unavoidable and makes coordination between agents a pragmatic requirement. Malone and Crowston define coordination as "*the act of managing interdependencies between activities*" [9]. Johnson and Bradshaw make the case that "*the understanding of interdependence is key to characterizing human-machine teamwork in an understandable, actionable, and generalizable manner*" [10]. Interdependence is sometimes characterized as interference (which can be positive or negative) [1, 11] or as dependence [1]. A few types of interdependence and their impact on task quality and maximum duration were formalized by Decker as non-local-effects [12]. It is clear from the research community that interdependence is a key component to teams [9, 13, 14] and makes the need for a theory of interdependence fundamental [10, 15]. However, interdependence is complex and for that reason it has historically been avoided.

Adaptability is a central aspect of teaming. The capacity to adapt is often correlated with team performance. There is a significant body of research on *adaptive autonomy* [16–18] which followed the definition of levels of automation, and defined adaptive autonomy as switching between levels of automation. This work demonstrated the impact of automation design choices on human-machine performance. Other work has focused more broadly on *adaptability* [19], not limiting it to levels of automation. Adaptability is about allowing agents to understand and react to change cognitively or physically. Adapting allows agents to mitigate adverse effects and opportunistically take advantage of situations to improve performance along specified dimensions of teaming. We posit that adaptability is the ability to negotiate and manage the interdependencies within the team to yield behavior classified as *good teamwork*.

Given the importance of human-machine systems, there are surprisingly few tools available to support designing, building, execution and interaction with human-machine teams. Existing tools typically target a single aspect of human-machine systems, such as distributed communications, or task allocation. Our desire is to develop a more comprehensive approach based on the extensive body of research on teaming. In a previous paper, *Understanding Human-Autonomy Teaming through Interdependence Analysis* [20], we provided principles to identify and understand interdependencies within a human-machine-system, helping designers design effective teaming systems. We now take the next step of providing tools to operationalize these principles in practice.

In this paper, we present joint activity graphs (JAGs), a formalism providing a systematic method to capture the essential elements necessary to describe and execute joint activity. We also introduce the JAG Engine as a runtime JAG interpreter and a means to execute multi-agent behavior. This engine handles the aspects of teaming that must be dynamically determined, such as team composition, task participation/allocation, and communication. The JAG Engine leverages the JAG formalism to support runtime teamwork through management of interdependence. Together, these tools provide designers and builders a practical and systematic approach to creating joint activity behaviors that support coordination processes within a team. Because JAGs are grounded in teamwork theory, they also enable a highly adaptive system. To demonstrate the range of adaption possible, we provide examples grounded in a *Capture-the-Flag* (CTF) domain. CTF is a fast-paced adversarial domain requiring quick and responsive adaptation within the team to be successful. This systematic approach, combined with the formalism described in this paper, help define and expose the different types of interdependence common to a broad range of activities and their associated coordination requirements, which in turn supports *good teamwork*.

2 Background

Providing agents with a computational means to determine their own behavior is necessary for agents to be useful. Many approaches have been developed over the years, such as planning systems [21], and reactive behaviors [22]. However, these are

[1] In this paper agent indistinguishably refers to human or machine.

behavior architectures, not guidance on how to develop specific behaviors suitable for joint activity. We next discuss some representational techniques important to the design of joint activity.

2.1 Hierarchical task networks

Per Erol, in 1994, "*most of the practical work on AI planning systems during the last 15 years has been based on Hierarchical Task Network (HTN) decomposition*" [23], and while HTNs now share the scene with machine learning, this statement mostly still holds true [3]. HTNs introduced the concept of compound tasks which were lacking in classical automated planning systems such as STRIPS [21] and PDDL, dramatically reducing the search space at the expense of domain knowledge. HTNs provide declarative goals and a rich constraint language on intermediate states that can express a large space of interactions. A key challenge with HTNs is their lack of support for parallel activity, which is critical in joint activity. Classical HTN planning does not explicitly preclude parallel behavior (in that an agent can execute multiple plans in parallel) but unfortunately does not concern itself with the complexity and relationships that may arise from parallel activities. We will see in Section 4.2.2.2 that we take the opposite approach and assume that all activities can be executed in parallel unless otherwise constrained.

Work based on hierarchical models is often concerned with task decomposition but seldom with *answer synthesis* (how to re-compose subtasks back together to fulfill the goal they decompose and their interactions) [24]. Duarte proposes a hybrid controller [25] as a solution to the *answer synthesis* problem [24] as presented by Smith and demonstrates that controllers can be synthesized hierarchically by applying it to the swarm multi-agent domain [26]. With regards to distributed artificial intelligence, Durfee compared the agent coordination to a search in hierarchical space which very cleverly approaches the problem of synthesis by grouping and abstracting behaviors at multiple levels [27] fully taking advantage of the composition capabilities of hierarchical task networks.

Decomposition and synthesis are critical components of designing joint activity. Choices made will enable or hinder exploitation of associated interdependencies and will have a substantial impact on teamwork. We build on the strengths of HTNs and extend it to include an understanding of interdependence (supported by the *4S framework for understanding teamwork* [28] and *interdependence analysis* [29, 30]), as well as a generalization of synthesis.

2.2 Behavior trees

Behavior trees are a form of hierarchical decomposition of agent behavior that has its genesis in video games. They were first proposed as good engineering practice to handle the complexity of large systems and allow behaviors to be more reactive to changes in requirements (such as behaviors of non player entities in video games).

Behavior tree use in robotics and artificial intelligence has been steadily growing in last decade [31]. They are a hierarchical decomposition of agent behavior, grounded in execution, data driven, and reusable which makes them a go-to model when designing agent behaviors. In these respects, they are similar to the joint activity formalism that we present in Section 3. However, behavior trees are significantly different in other aspects. They were initially designed with single agent behavior in mind (with sparse and disconnected attempts to be augmented to support multi-agents). Thus, they hide interdependencies, resulting in ad hoc solutions when trying to apply them to build joint activity (multi-agent behavior). They are re-evaluated often, typically multiple times per second, and do not hold state. Behavior trees stateless nature and their lack of data flow make data usage within a behavior pragmatically hidden and require the use of back channels for information sharing, such as a blackboard. Like HTNs, parallel execution is not precluded (behavior tree formalism has been augmented with an explicit parallel node) but there is no context support for the resulting interdependencies.

In contrast, a joint activity graph is always designed from a multi-agent perspective, with single agent behavior being the degenerate case. JAGs are event driven, as opposed to being reevaluated at regular intervals, providing observability into teamwork processes, enabling causality tracing and adaptation explanation. In stark contrast to a behavior tree's lack of state and blackboard back channel, data flow (i.e. inputs, outputs and bindings) is a central part of the JAG model. This makes tying data to joint activities possible and in turn enables and simplifies the process of identifying relevance of information (see Section 6.3). JAG inputs further specify its behavior and as such are part of the context necessary to make decisions. In that respect inputs satisfy coactive design interdependence requirements: observability, predictability and directability (OPD) [29].

We build on the practicality of behavior trees (composability, grounded in execution and data driven) to augment our framework with established practices and adapt it to the domain of human machine teaming.

2.3 TAEMS

In his inspirational thesis, Decker describes a generalization of Durfee's Partial Global Planning [27] called TÆMS or Generalized Partial Global Planning [12]. TÆMS is described as a "*domain-independent coordination framework for small agent groups*" [32]. It expands on the domain specific limitation of Partial Global Planning by including a more abstract and hierarchical representation of the joint activity allowing a generalized identification and management of coordination relationships (interdependencies).

Our work heavily builds upon and extends this work, both in the structure (task decomposition and quality) and interdependencies (non local-effects). Decker formalises hierarchical synthesis in the form of quality accrual functions (e.g., min, max, average) making it consistent with task interdependence (see Section 4.2.1). Decker also formalises a significant set of interdependencies (e.g., *enables*, *facilitates*) and their measurable effect on tasks' quality and duration.

Joint activity graph expands this work to also include OPD requirements as team interdependencies [29]. Most, if not all, concepts described in TÆMS have direct overlap or are generalized in the joint activity graph formalism that we propose.

3 Joint activity graphs

Joint Activity Graphs (JAGs) are a new method to describe *joint* activity in a way that is executable. Our goal with JAGs is to provide a rigorous and systematic method for defining joint activity that can be run in a distributed manner and achieve behavior that would be described as *good teamwork*. A key design principle when employing JAGs is that all work should be designed as joint work [20], meaning the JAG should be designed with an understanding that multiple agents will be involved in performing the work. This is a dramatic shift from the typical single-agent behavior mindset.

JAGs describe the solution space of joint behaviors. They capture the goals and actions necessary, as well as the options and contingencies available. Because of this, JAGs are not simply a plan, but a description of the set of alternatives available to the team.

A major challenge in defining a JAG is understanding the interdependencies within the joint work. Teamwork is complex and involves the interplay of dimensions such as team goals, task work, team composition, execution strategies and interdependencies as discussed in 4. Malone and Crowston stated that "*one of the most intriguing possibilities for coordination theory is to identify and systematically analyze a wide variety of dependencies and their associated coordination processes*" [14]. The JAG structure is defined to provide a framework for capturing the interdependence systematically. It provides a common structure onto which teaming information within a joint activity is captured. This structure allows designers to systematically consider a broader range of teamwork aspects at design time than commonly supported by current tools and techniques. The JAG definition includes the hierarchical work, similar to HTNs. It also includes synthesis functions in a more generic manner than found in behavior trees. Lastly the JAG includes the necessary information for capturing data flows.

Formally, a jag $\gimel$ is defined as the tuple

$$\gimel = \langle J_\gimel, s_\gimel, I_\gimel, O_\gimel, B_\gimel \rangle$$

$J_\gimel$ is the set of joint activity graph children of $\gimel$

$$J_\gimel = \{\gimel_1, \gimel_2, \ldots\}$$

$s_\gimel$ is a synthesis function over its own inputs and $J_\gimel$'s outputs

$$\left(\bigcup_{n=1}^{|J_\gimel|} O_{\gimel_n} \cup I_\gimel\right) \overset{s_\gimel}{\mapsto} O_\gimel$$

$I_\gimel$ is the set of $\gimel$'s input parameters

$$I_\gimel = \{i_1, i_2, \ldots\}$$

$O_\gimel$ is the set of $\gimel$'s output parameters

$$O_\gimel = \{o_1, o_2, \ldots\}$$

$B_\gimel$ is the set of bindings representing the output-input data flow within $\gimel$

$$B_\gimel = \{b_1, b_2, \ldots\}$$

where

$$b_k \in \left(\bigcup_{n=1}^{|J_\gimel|} O_{\gimel_n} \cup I_\gimel\right) \times \left(\bigcup_{n=1}^{|J_\gimel|} I_{\gimel_n} \cup O_\gimel\right)$$

These features, represented in a JAG definition, capture the essential elements needed to interpret interdependencies within joint activity. An example of a generic JAG definition, such as the jag pictured in Figure 1 would be defined as follows:

$$\begin{aligned}
\gimel &= \langle \{\gimel_a, \gimel_b\}, s_\gimel, \varnothing, \varnothing, \varnothing \rangle \\
\gimel_a &= \langle \{\gimel_{a,1}, \gimel_{a,2}\}, s_{\gimel_a}, \varnothing, \varnothing, \{(o_1^{a,1}, i_1^{a,2})\} \rangle \\
\gimel_{a,1} &= \langle \varnothing, s_{\gimel_{a,1}}, \varnothing, \{o_1^{a,1}\}, \varnothing \rangle \\
\gimel_{a,2} &= \langle \varnothing, s_{\gimel_{a,2}}, \{i_1^{a,2}\}, \varnothing, \varnothing \rangle \\
\gimel_b &= \langle \varnothing, s_{\gimel_b}, \varnothing, \varnothing, \varnothing \rangle
\end{aligned}$$

It should be noted that the JAG formalism intentionally does not describe the team or the strategy. This is consistent with the interdependence design principles [20], appropriately separating these concerns. The formalism, as we will show, does work with both at runtime.

This JAG formalism is beneficial in a variety of ways. First, it provides a framework for tracking the information necessary for understanding the teaming context within an activity. Additional information about team context, such as team composition, task allocation, and task progress, while not defined in the JAG, can be tracked through the JAG. This enables individual agents to make effective single agent behavior choices that are consistent with good teamwork decisions. Second, the framework provides agent coordination mechanisms to facilitate appropriate team interactions at runtime based on that team context reasoning (see Section 4). In other words, the formalism

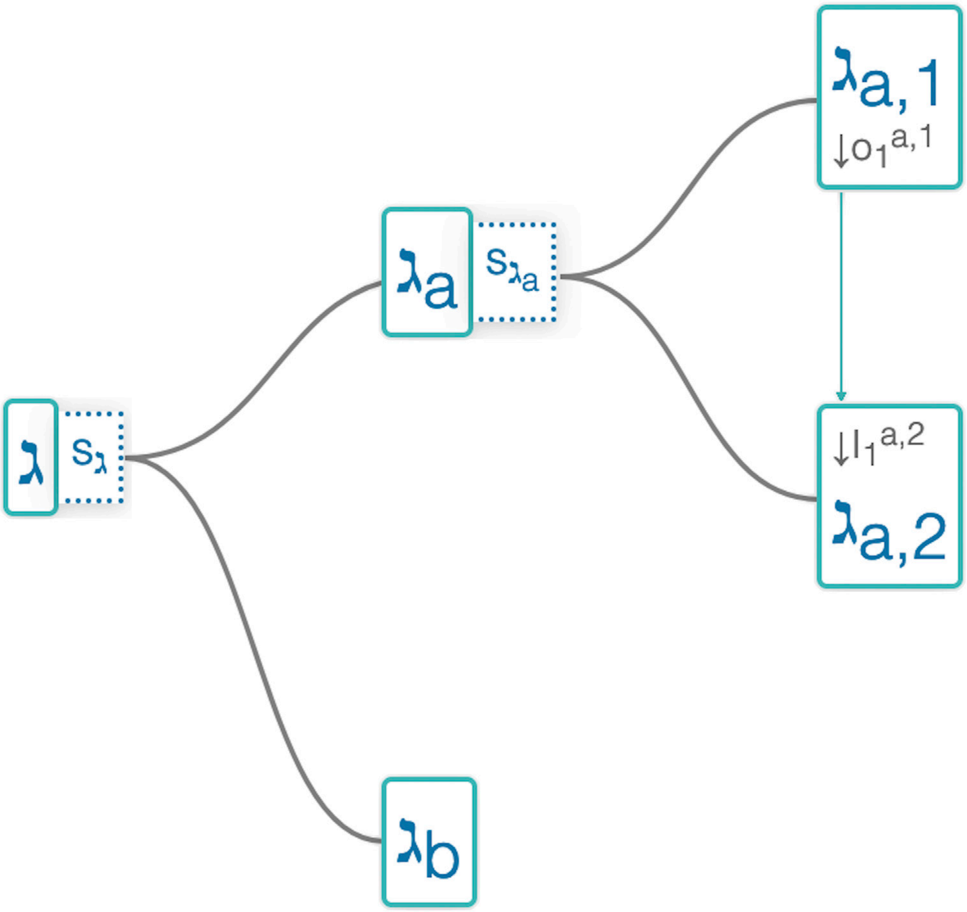


FIGURE 1
Example of joint activity graph representation showing decomposition, synthesis and data binding between two siblings.

provides the minimal situation awareness necessary for collaborative contexts.

Before explaining how the JAG formalism helps address interdependence, we will first expand on the broad range of sources of interdependence that occur within joint activity. To do so, we will reference the 4S interdependence framework for understanding teamwork [28].

4 Teamwork challenges

One of the main reasons understanding teamwork is challenging is because teamwork involves a wide range of interdependencies. It should not be a surprise that different kinds of teamwork can be distinguished according to the types of interdependence involved. For example, lifting a couch together involves different interdependencies than sharing a hammer. Each type of interdependence can involve different coordination mechanisms necessary to manage it. For example, lifting a couch might require agreeing and reacting to a start signal ("lift on 3"), while sharing a hammer could require verbal notification of completion or even simple observation of availability of the hammer. As such, operationalizing teamwork requires developing support for managing a range of interdependent relationships using a range of coordination mechanisms and techniques. Johnson et. al [28], proposed a framework for organizing many of the important concepts associated with teaming based on the interdependencies at play. The framework is organized on four facets: state, structure, skills, and strategy. Here we expand on this framework.

4.1 State interdependence

State interdependence refers to interdependence resulting from the need to coordinate and share resources across team members.

We propose expanding this category with two common state types that generate interdependence constraints on the team: information and resources.

4.1.1 Information interdependence

Information creates interdependence based on each agent's need-to-know. Teamwork is built on common ground [2], and so it should be no surprise that team members would need to share information to operate effectively as a team. This need generates information interdependence as each team member experiences their own view of the activity. This type of interdependence is often referred to as a need for situation awareness [33], common ground [2] or shared mental models. Regardless of the phrasing, each implies that discrepant knowledge between agents can lead to poor team performance while consistent knowledge would result in improved team performance. Examples of the type of information teammates depend on are task assignment (who is working on what), task commencement (what has been started), task completion (what has been finished), task outputs, including status (successful or failure) and results (values or decisions).

A key challenge for information interdependence is determining information relevance. As with most aspects of teaming, there are two sides to the issue. The first is an agent recognizing information it receives as relevant and understanding how that information might impact their own understanding of past, present or future decisions and adapting based on the new information. The other side of the issue is an agent understanding when new information it discovers might be relevant to others. This involves being able to identify who is dependent on what information and when. This is made more difficult in fluid teams without fixed roles. Even with fixed roles, dynamic activity means that some information will likely become irrelevant with time and effective teammates should recognize this.

4.1.2 Resource interdependence

Resources create interdependence by constraining what can be done. A person can only carry so much and a robot can only drive to one location at a time. Resource constraints are probably one of the most studied types of interdependence. It is well known that if two activities require the same resource, one can block the other, creating a sequential interdependence constraint [34]. Resources can be things in the environment, like a printer, but the agents themselves can be viewed as a resource as well. For example, person A can help person B carry something, and person A can help person C carry something, but it is unlikely person A can help both person B and C simultaneously, thus creating a sequential interdependence constraint. A key challenge with resource constraints is identifying them and being able to coordinate them effectively as a team.

Information and resources share very similar coordination requirements. One substantial difference is that information can be replicated, usually at low cost compared to physical resources. Once replicated information can then be used in parallel as if there were two of the same resource available. This is one of the many ways to address state interdependence.

4.2 Structural interdependence

Structural interdependence refers to types of interdependence caused by the structure or organization of the work. It comes from two main sources: the taskwork and the team organization [28]. Interactions in highly complex and tightly coupled systems can be difficult to predict. Different level of abstractions are needed at different levels of operation with no holistic understanding of its interdependence. In high risk systems this may lead to catastrophic consequences [35]. Understanding the interdependence resulting from the system itself and its organization is key in identifying potential critical paths and address them adequately.

4.2.1 Task structure interdependence

Taskwork generates interdependence in both the decomposition process and in the synthesis process.

4.2.1.1 Task decomposition

Decomposition of the joint activity generates taskwork interdependence. Structural interdependence is determined by the decomposition boundary of the activity. This boundary is often a design or engineering driven decision. Arguably, tasks can always be further decomposed into sub-tasks but eventually the level of decomposition becomes unwieldy or even absurd. The decision of where the boundary lies often varies with the domain and agents under consideration. The coordination mechanisms involved in a command and control situation are different than those required in a mechanical repair situation and so are the abstractions at play. Goals and requirements may also dynamically change at run time and adaptive teams should be able to adjust task boundaries to provide flexibility in their plans. The process of decomposition itself is not a challenge, it is understanding the implications of how those changes impact interdependence that is difficult.

4.2.1.2 Task synthesis

When tasks are decomposed they must eventually be recomposed, creating interdependence. In distributed problem solving, answer synthesis and behavior composability are critical abstractions of complex distributed systems [24]. Synthesis provides practical mechanisms that address structural interdependence. The key challenge for synthesis to be operationalized is defining how tasks' outputs are generically combined given the range of possibilities.

When decomposing a joint activity, there is a requirement to explicitly define synthesis functions capturing how the output/state of the joint activity is derived from the output/state of the sub-tasks. This allows designers and reasoning engines to understand and exploit decomposition related interdependencies. For example, Boolean logic could be used to define synthesis functions. Consider the activity of going to lunch. It can be decomposed into eating lunch and paying for lunch. Here, the success of going to lunch depends on the success of both children. This type of synthesis can be captured with a Boolean operator such as and; both children have to succeed for the parent to be considered successful. Changing the success synthesis function to *or*, would completely change the meaning of the activity and associated types of interdependence.

4.2.2 Team organization interdependence

4.2.2.1 Team decomposition (roles)

Another way to generate interdependence is through organizational choices. Similar to task decomposition, one's choices about the team structure can create boundaries and interdependence. Distribution of work is another reason why coordination is necessary [13]. For example, having an engineering department and a purchasing department will require the engineering department to go through purchasing for parts, creating a sequential interdependence. Effective organization design typically involves designing roles to reduce the degree of interdependence to allow roles their maximum freedom.

4.2.2.2 Team participation

Some organizations have fixed predefined roles continuously performed by the same individuals, making participation constant and predetermined. Teamwork in general is more fluid, allowing flexible roles and intermittent participation to allow the team to adapt. This is particularly important for teams that do not have the resources to cover all work and may need to choose what is attended to.

Participation in joint activity represents joint commitment, a requirement for teamwork [5, 13]. Participation is often assumed or ignored in system design, but it is an important dimension that plays a critical role in interpreting interdependence. An implementation that is unable to account for participation is blind to key information necessary for effective teaming.

Participation must also account for interdependence in the form of task constraints. The structure and task decomposition choices may limit team composition and participation. In his book *Group Processes and Productivity*, Steiner [36] presented a categorization of joint activity (*group tasks*) along three dimensions, one of which was whether the task was divisible or unitary (*component*). Divisible means the task can be divided and distributed to individuals. The example Steiner gives is a multi-question test, where each question could be given to a different student. Unitary means the task cannot be divided. Steiner uses a test with a unique single question as an example. He posits that, because the question cannot be broken down into sub-questions, this makes this task unitary and that "*the group would be required to* ***work together*** *to discuss and determine the correct answer [. . .]*". A limitation of the unitary category is it does not differentiate tasks that can only be done by a single person. For example, giving a group a single pill that must be swallowed. Only one person can do it and no others can contribute. This is a different type of interdependence than the single question, in which all team members could contribute to the answer.

4.2.2.3 Team synthesis

Simply distributing work creates a need for a synthesis function, similar to task decomposition. The synthesis strategy used is related to the second dimension proposed by Steiner [36], which he characterized as the *interdependence* characteristic of the joint activity. Steiner proposes the categories of, additive (all team members' work contributes to the task - shoveling snow), compensatory (group averaging—averaging weight estimates), disjunctive (single decision—answer to a math problem), conjunctive (all team members must contribute - climbing a mountain as a group) and discretionary which is a combination of any of the previous ones. Each of these synthesis strategies involves a different type of interdependence and different coordination mechanisms.

Practically, team synthesis is different from task synthesis in that it is not about reasoning over children's outputs but rather over multiple outputs for a given joint activity instance. In Steiner's "single question test" example, each student participates in the same joint activity generating multiple, potentially different, outputs to the question. These outputs must be reconciled to produce the unified joint activity output. For example, by using *team operators* (a particular type of team synthesis) on a hierarchical decomposition of joint activity, Tambe demonstrated how selective and efficient communication could be achieved in a distributed environment [37].

Another aspect of team synthesis is the understanding of participation status in joint activities. For example, if a goal is conjunctive, meaning it must be completed by all members of the team, recognizing when a teammate is not capable of participating in the goal (e.g., due to capability requirements or resource constraints) will allow a reasoning process to understand that this sub goal should not be undertaken by anyone or should trigger early failure if it had been started by some agents already.

Team synthesis is challenging because it often involves awareness of several other interdependencies. For example, to accomplish team synthesis effectively, an agent may need to be aware of state information, task structure, and team participation.

4.3 Skill interdependence

While state and structure interdependencies are about being able to identify and understand interdependence, skill is about having the supporting coordination mechanisms to address them. For example, if one is dependent on knowing when their teammate has finished using the hammer, one could employ several mechanisms to coordinate. One could actively observe the teammate with the hammer to see when they put it down (i.e., monitoring). Alternatively, one could ask the teammate to provide a notification when complete. Both mechanisms are effective each with their own advantages and constraints. Each mechanism requires specific skills or abilities to be successful. For example, monitoring only requires effort by the person doing the monitoring and alleviates the burden from the one being monitored. The disadvantage is that it can require significant attention, possibly reducing team productivity. It also creates a single point of failure. Notification requires more coordination effort by both parties, but frees each from the monitoring burden, potentially allowing better use of time.

While there are potentially an endless number of coordination mechanisms, many can be categorized as being able to recognize the existence (or lack of) interdependence between one or more parties, understand the communication or behavior pattern necessary to manage that interdependence, and the means to execute it. This means recognizing when observed changes in the environment are relevant to others on the team and sharing them (information), recognizing someone is constrained to doing one task at a time and providing assistance (resource), recognizing that tasks have sequential interdependence and providing the waiting party notification of completion (decomposition), sharing task results (synthesis), and notifying only those relevant to the activity (participation). These pattern generalizations are how people can leverage teamwork skills in new situations.

4.4 Strategy interdependence

Teaming strategy is about having the competency to discern how and when to engage a coordination skill to impact a state or structural interdependence in order to improve some quality within the team. Effective teamwork involves trying to improve behavior qualities. This aligns with Steiner's third category of coordination challenges: *focus* [36]. Steiner provided only two discrete categories: maximizing (improving throughput) or optimizing (improving quality). Decker [12] generalized this concept by introducing a *quality* to tasks as an abstract representation of the task's *focus* as well as a task's *duration* as one of its prime characteristics. A key challenge with focus, and strategy in general, is that it often varies based on circumstances and rarely can be set in stone *a priori*, hence it is not part of the JAG formalism, but a runtime consideration of that formalism.

5 Evaluation domain

We desired to have an evaluation domain that exercised the broad range of types of interdependence described in Section 4. As part of the Defense Advanced Research Projects Agency (DARPA) program called CREATE (Context Reasoning for Autonomous Teaming), we developed a new evaluation domain. The goal of CREATE was to investigate new decentralized teaming approaches for physically distributed groups of agents. The program's focus was on solutions that demonstrated "context reasoning", enabling agents to be resilient to uncertainty and adapt to unexpected events in the absence of centralized control. This provided a perfect test case for operationalizing interdependence design principles. We chose to base our evaluation domain on Capture-the-Flag (CTF). CTF is a dynamic adversarial game that has many of the desired characteristics that demand complex teaming, in particular many of those discussed in Section 4.

One limitation of traditional CTF is that it is mainly disjunctive activity (e.g., shooting, carrying the flag). It was desirable to have an evaluation domain that exercises a broader range of activity types. We looked to enhance the CTF domain leveraging Steiner's interdependence categories [36]. Some domains only provide additive tasks (e.g., foraging, search), others only provide disjunctive tasks (e.g., image recognition, decision making), while others are solely conjunctive (e.g., carrying a large table together).

Our new version of CTF has unique rules that foster a wider variety of teaming activities to better exercise different interdependence requirements. It consists of adversarial teams composed of heterogeneous agents: unmanned aerial vehicles (UAV) and unmanned ground systems (UGS). The objective is to find the enemy's flag and bring it back to your team's base (color coded endzones in Figure 2). A UAV can find and pick up the enemy flag, and deliver it to their base (disjunctive task). UAVs can also pick up and move UGS, which cannot move on their own. UGS are non-mobile smart mines that can suppress the enemy UAVs, sending them back to their base. UAVs can deploy UGS (additive task) as a defensive tactic. Instead of shooting each other as in traditional CTF (disjunctive task), the UAVs can temporarily suppress one another, sending them back to their base. UAVs achieve this by outnumbering the enemy players (conjunctive task). The visual range of the UAVs and UGSs were restricted to increase the value of sharing information between team members. This combination of activities required teams address a broader range of interesting teaming challenges than traditional CTF.

Specifically, our modified CTF domain exercises all of the types of interdependence described in section 4. For instance, the addition of the mine laying task created information interdependence (Section 4.1.1) with regard to where to lay mines and resource interdependence (Section 4.1.2) to coordinate who would lay each mine. There is variety in the activity decomposition (Section 4.2.1.1), as the main activities (retrieving flag and laying mines) can be completed in parallel,

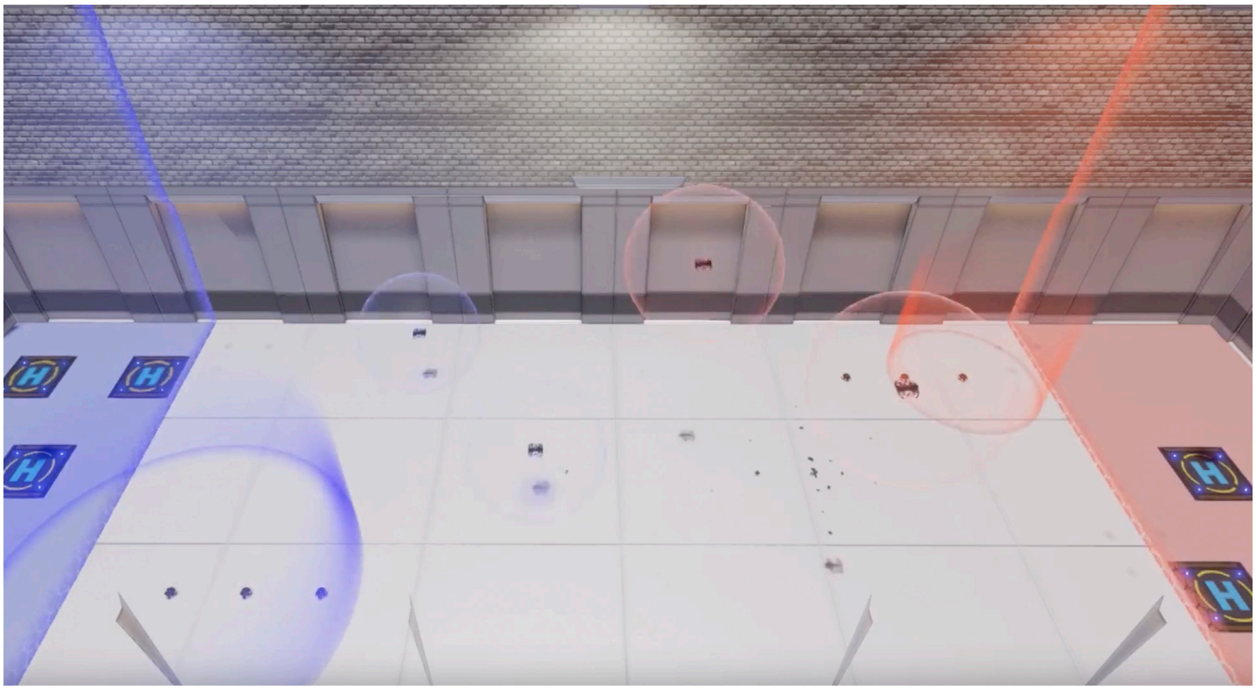


FIGURE 2
Live game of Capture-the-Flag in the simulated lab arena. This game shows a 3v3 (technically a (3 + 3)v (3 + 3) with 3 UAVs and 3 UGSs per team) with the blue team endzone on the left and the red team endzone on the right. At this point in the game, 1 UAV in each team has been disabled.

while the sub-activities of each have sequential dependence (e.g. pickup before delivery). The task synthesis requirements (Section 4.2.1.2) vary as some tasks can be done asynchronously, like mine laying, while other tasks must be done conjunctively, like UAV suppression of the enemy. The team must chose how to balance offensive and defensive strategies (Section 4.4), which directly impacts participation (Section 4.2.2.2). Team members can dynamically change roles creating team organizational interdependence (Section 4.2.2.1). The team's strategy must account for how the combined efforts of each individual result in effective behavior (Section 4.2.2.3). These are only a few of the many instance of interdependence that must be managed to produce effective coordination by the team. Each requires possessing the coordination skill (Section 4.3) and an understanding of information relevance (Section 6.3) to support *good teamwork*.

To exercise our framework, we developed hardware agents (see Figure 3) as well as a virtual twin simulator in unity (see Figure 2) that allowed the development and validation of joint activity graphs both in simulation and hardware in a fully distributed environment.

As a dynamic and uncertain evaluation environment, our modified CTF fosters a broad range of interdependence demanding a rich understanding of team context to produce effective team performance. Although CTF is a very active domain that involves fast-paced physical work, it also requires a large amount of sophisticated cognitive reasoning over team context. This reasoning is complicated by the fact that it happens within each individual agent in a distributed manner. These independent decisions must be synthesized and coordinated across the team, providing an excellent evaluation domain for assessing our interdependence design principles in practice.

6 Addressing interdependence

The JAG formalism was developed to help designers think through the considerations necessary when designing joint activity. It directly supports addressing state and structural interdependence. It also provides the teaming context needed to address skill and strategy interdependence within a team. This is accomplished by the JAG Engine reasoning over the JAG formalism to make coordination and strategy decisions, discussed further in Section 7. As conveyed in 4, the various types of interdependence relate to one another in many ways, so there is not a one-to-one-mapping to the JAG formalism. Instead, the elements of the JAG formalism combine in different ways to help address all of the interdependecies in 4.

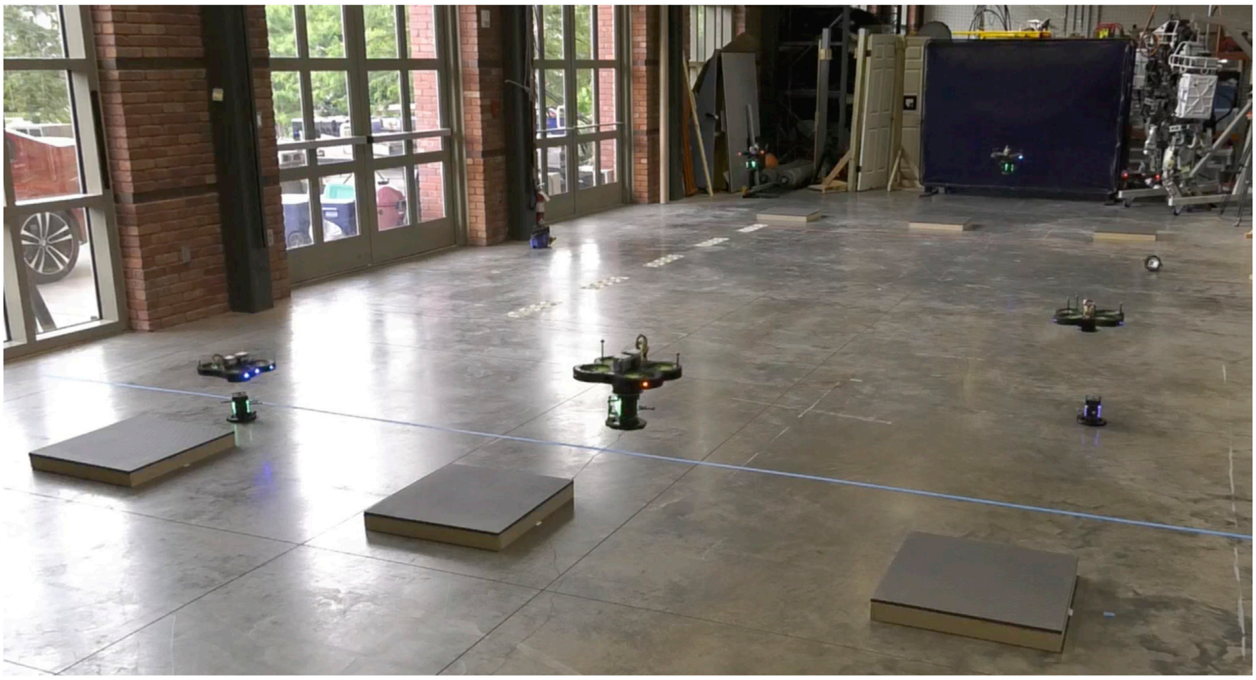


FIGURE 3
Live game of Capture-the-Flag in the lab arena.

6.1 Decomposition

JAG's task work component, $\gimel$, is a hierarchical decomposition of the joint activity. It defines the activity search space for the agents and is consistent with Durfee's distributed goal search [27] and Smith's synthesis requirements [24]. The main purpose of hierarchical decomposition is to understand task work context.

As an example, Figure 1 shows two different levels of decomposition: $\gimel$ decomposes into $\gimel_a$ and $\gimel_b$. $\gimel_b$ has no further decomposition whereas $\gimel_a$ is further decomposed into $\gimel_{a,1}$ and $\gimel_{a,2}$. This decomposition has implications both in terms of participation and interaction.

Agents select what to do next through an understanding of the activity space as defined by the decomposition. Additionally, activity decomposition provides a structural skeleton for tracking participation of the entire team. Each agent's participation in joint activity can be tracked at the individual hierarchy level. This helps scope interactions enabling level specific coordination mechanisms. For instance, communications about sub-tasks do not need to be broadcast to the entire team but only to the agents participating at that level of the hierarchy (see Section 6.3). Similarly roles and responsibilities can be defined at each individual level of decomposition.

Decomposition also allows designers and agents alike to define and act at different abstraction boundaries. Consider a grab behavior defined as $\gimel_{grab}$. On one hand, a human could undertake the $\gimel_{grab}$ behavior as a 'primitive' limiting observability, predictability or directability into the task. This would prevent team members from interacting with the different parts of the process involved in the grab behavior. A machine, on the other hand, may decompose its $\gimel_{grab}$ behavior further to allow team members to interact, contribute and/or support the different sub processes at play within the machine during the activity.

There might be practical reasons for relying on higher abstraction levels. For instance, humans can grab things pretty reliably whereas current machines may need more support throughout the whole process such as finding the location of the object or determining the best approach trajectory. Pragmatically, the level of decomposition drives the abstraction boundary of the behavior and in turn the type of interdependence and the capabilities needed to manage it. The JAG approach allows both design time and runtime flexibility for such boundaries, facilitating human-machine joint activity.

Decomposition has other intrinsic benefits common to all similar approaches, such as the creation of modular behaviors and the promotion of reuse of existing designs. Since JAG designs have interdependence considerations defined with the decomposition, those consideration transfer with reuse.

By using a hierarchical structure, JAGs support the ***task decomposition*** (Section 4.2.1.1) like similar approaches (see

Sections 2.1 and Section 2.2). However, JAGs go further and support ***structural interdependence***, specifically ***team decomposition boundary*** (Section 4.2.1.1) and ***team participation*** (Section 4.2.2.2), the importance of which will be further discussed below.

6.2 Synthesis function

"*Our ability to decompose a problem into parts depends directly on our ability to glue solutions together*".

- John Hughes, Why Functional Programming Matters [38].

Synthesis defines the process of recomposing the task decomposition and its results. The synthesis function $s_\gimel$ can take the form of any mathematical function. Examples include quality functions min, *mean* or max [12, 36] (dealing with conjuntive, disjunctive or additive aspect of tasks) as well as Boolean operator [37] such as *and* or dealing with team goal requirements. As such, joint activity graph synthesis can benefit from contributions from a wide variety of fields such as sensor fusion and organizational theory.

Even though this synthesis function can be arbitrarily complex, a broad range of activity can be covered by a reasonably small set of reusable joint activity patterns. We expect each domain will favor specific sets of functions with significant overlap. For example, in our modified CTF domain, all but one operators were standard Boolean operators.

Importantly, the synthesis function also acts coherently with leaf nodes, also called primitives [13, 23, 25, 32] or methods [39]. A leaf node is a jag $\gimel$ whose set of children $J_\gimel$ is empty. As $\bigcup_{n=1}^{|J_\gimel|} O_{\gimel_n} = \varnothing$, its synthesis function $s_\gimel$ is then reduced to:

$$I_\gimel \overset{s_\gimel}{\mapsto} O_\gimel$$

A leaf node's synthesis function $s_\gimel$ essentially acts on its own inputs, then outputs a result and potentially generates non-local effects as defined by Decker [39]. This is essentially a function call to a machine or human interface. This synthesis function coherency is an important distinction from classical planning and behavior modeling. It allows joint activity designers and exploiters to consider and interact with all levels of abstraction in the same manner.

Synthesis function definitions allow JAGs to formally capture the processes needed to manage the ***synthesis interdependencies*** described in Sections 4.2.1.2 and Section 4.2.2.3. A surprising number of tools and techniques ignore synthesis, even though it is critical to teaming. JAGs provide a general and extensible solution to address synthesis interdependence.

6.3 I/O and information relevance

The inputs $I_\gimel$ to a joint activity $\gimel$ provide the necessary information needed by the activity. They are a common way to parameterize activities.

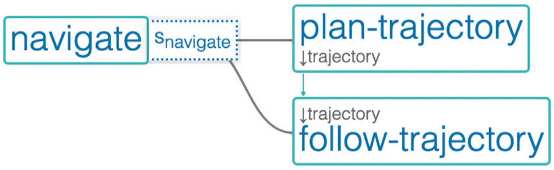


FIGURE 4
A simple jag behavior with input and outputs.

The outputs $O_\gimel$ of a joint activity $\gimel$ are derived from $\gimel$'s inputs and the outputs of $\gimel$'s children *via* $\gimel$'s synthesis function $s_\gimel$. The synthesis function can be a simple pass-through, return one or more child outputs, or can be an arbitrarily more complex function returning a derived result from one or more child outputs. This is consistent with and supports concepts such as Decker's sub-task quality accrual functions (min, max average) [12], but a more general extension.

Bindings, $B_\gimel$, define the information flow within an activity. Bindings uniquely identify a data provider and a data consumer. Inputs for an activity can be passed down and consumed by (bound to) any child joint activity. Sibling outputs can also be consumed as inputs by other siblings. For example, Figure 1 shows $\gimel_{a,1}$'s output $o_1^{a,1}$ bound to $\gimel_{a,2}$'s input $i_1^{a,2}$.

This creates an implicit sequential interdependence requirement [34]; $\gimel_{a,2}$ cannot be started before $\gimel_{a,1}$ has completed and generated its output $o_1^{a,1}$. Input and output flow is completely defined, in practice, through bindings.

I/O plays a key role in identifying information relevance. Team performance monitoring is one of the *Big Five* components of team effectiveness [40] and is crucial in enabling adaptability. It is common to use monitoring functions to observe, prevent failure, and repair plans through continuous planning [41]. However, monitoring functions have to be manually defined and managed which can be cumbersome. We propose that we can make the process more observable and systematic with parameterization of behaviors and explicit data flow to address ***resource and information state interdependences***. The data used by joint activities is inherently relevant to that activity. If the input changes the output may change as well. Hence, data flow identifies what portion of the world is relevant at different levels of the joint activity. In turn, team processes addressing ***information interdependence*** can be executed based on this flow. These processes help agents identify to which teammate a new piece of information is relevant. They also help agents assess if a received piece of information is relevant to their own ongoing activities.

Similar to good software engineering practices, we have found that the amount of behavior parameterization is directly proportional to the adaptability the team with regard to that behavior.

For instance the behavior $\gimel_{navigate}$ in Figure 4 behavior can be implemented very specifically as to only be able to navigate to a predefined location. Without input, this behavior cannot react to information updates. There is also no observability into the information necessary to execute navigate. Two independent executions will behave the same, and make managing certain interdependencies impossible, and by extension, teamwork that much worse. A slightly more common implementation would be to parameterize $\lambda_{navigate}$ with a location which would be consumed by $\lambda_{plan-trajectory}$. Updates about the location would now be known to have an impact on $\lambda_{navigate}$. Similarly, navigation is likely to include a list of obstacle in its planning. If that list of obstacles is a parameter (an input), then $\lambda_{navigate}$ can now react to new information about obstacle location (see Section 8.2). Relevance of new information, such as the information about the destination and obstacles, is now systematically tied to the navigate behavior which leads to smart and informed reactions to world changes. Relevance can now be defined more specifically:

Information *p* is relevant to a behavior *b if b is active* and *if p matches any input from b or from a behavior whose output is recursively consumed by b.*

For example, $\lambda_{follow-trajectory}$ consumes $\lambda_{plan-trajectory}$'s output, $o^{plan-trajectory}_{trajectory}$ which was generated using $i^{plan-trajectory}_{obstacles}$. Any change to an obstacle concept would therefore be relevant to the $\lambda_{follow-trajectory}$ joint activity.

Because there is a systemic link between data and the behavior that uses this data, the more a behavior can be parameterized the more it can be reactive to changes in its parameters. This awareness of information relevance can facilitate better team adaptation. This applies generically throughout the joint activity as defined by its data bindings.

Although outside the JAG formalism, the concept of matching information was an important part of building an agent knowledge base. The process of matching should be left to the system designer to decide but it may be useful for the reader to understand how we designed information and implemented concept matching in our agents. Our approach was soft property matching. Concepts (or pieces of information) are a bundle of arbitrary property value pairs. If all properties of a concept *c*1 exist in another concept *c*2, and both values satisfying equality for their type then *c*1 matches *c*2, however the inverse is not true. For instance, an agent referring to a *blue mine* would match the generic friendly unarmed mine concept in listing 1 and the more specific mine instance in listing 2. However, if an agent refers to a specific blue UGS, that agent is not referring to just any blue UGS. This is analogous to looking for one's favorite blue pen that was gifted when graduating as opposed to looking for any blue pen.

Listing 1. Friendly unarmed mine concept.

```
{
"type": "agent:mine"
"team": "blue"
"armed": false
}
```

Listing 2. Specific mine instance concept.

```
{
"type": "agent:mine"
"team": "blue"
"id": "542ce2b1-c00e-47ff-8d7f-8db0fc118b13"
"name": "blue-mine-3"
"armed": false
"location": (0.0, 0.15, -1.5)
}
```

7 Dynamic team context reasoning

While the JAG formalism helps designers consider interdependence *a priori*, other types of interdependencies only manifest themselves during joint activity execution. The JAG Engine provides reasoning over the JAG formalism to make coordination and strategy decisions.

7.1 JAG engine

In order to operationalize the management of interdependence we developed an additional tool called a JAG Engine. A JAG Engine is an execution environment that interprets and executes joint activity graphs. It is able to use the information captured by the formalism described in Section 3 to drive the behavior of an agent in support of teamwork. It exposes interdependencies and provides processes to manage them. The JAG Engine interfaces with the agents being supported *via* traditional application programming interfaces for artificial agents and user interfaces for human agents. This engine uses user defined strategies to drive the behavior of agents following the process models defined in joint activity graphs under execution. The engine understands the intrinsic interdependencies in JAGs, such as the fact that multiple agents can participate in additive tasks, or that new information may be relevant to specific agents based on their joint activity participation. It is worth noting that in the case of human agents, the processes can be exposed *via* user interfaces allowing human agents to interact with the proposed courses of action in the same way artificial agents would (e.g., accept, reject or counter propose).

The JAG Engine, combined with the JAG formalization provides a unique capability for system control, enabling flexible and even dynamic shifts in control. JAG Engines have a 1 to n relationship with agents. They can be distributed (one engine per agent), centralized (one engine for all agents) or anything in between (*k* engines for *n* agents where $1 \leq k \leq n$). The engine specification also provides an abstraction layer for communication with built-in communication options relevant to teaming processes (e.g. participation in a JAG, completion of a

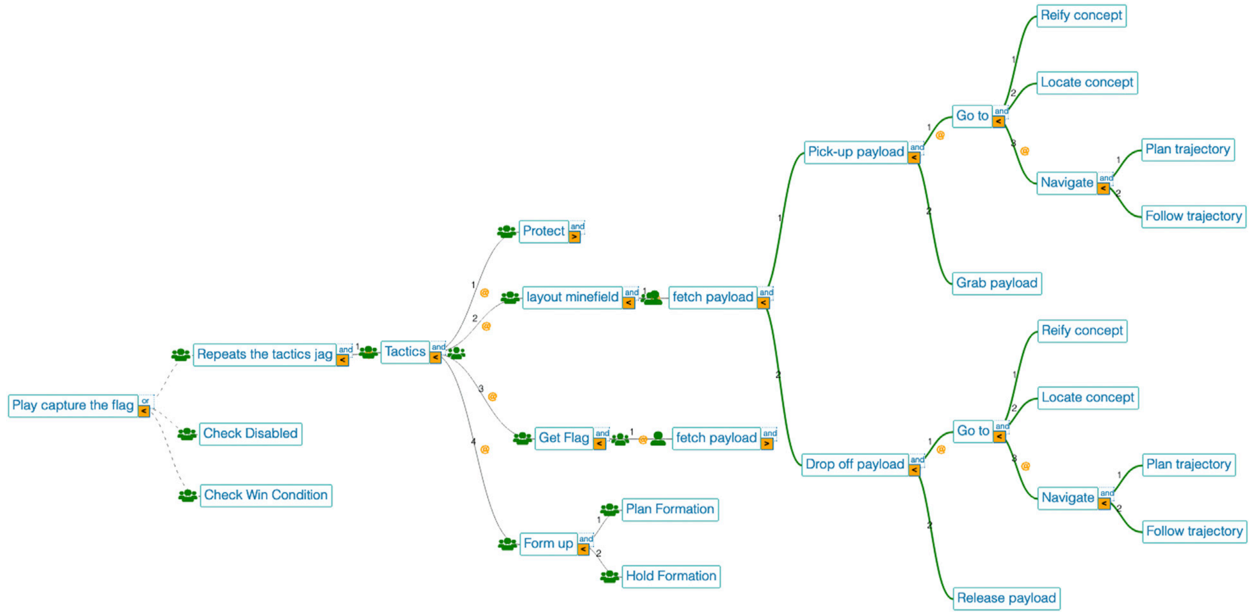


FIGURE 5
Joint activity graph for the Capture-the-Flag scenario - semi expanded.

JAG, negotiations relevant to the strategy in use, etc.). The specification allows for flexible strategy implementations allowing system designers and subject matter experts to create domain appropriate decision-making systems that can dynamically ingest and act on the run-time teaming context.

A JAG engine implementation provides the necessary framework to interpret JAGs, drive team behavior using strategies, comply with and expose interdependencies requirements and opportunities, and interface with a team of heterogeneous agents. The JAG Engine tracks and coordinates participation, as well as enabling strategy to be informed by the teaming context of the JAG. Figure 5 shows a semi expanded version of a Capture-the-Flag joint activity graph. Such graph is a view on joint activity definitions and is directly executable by a JAG Engine.

7.2 Participation

Participation in a joint activity plays a key role in managing communication backed coordination mechanisms such as information sharing, as well as decisions about task allocation. Yet common techniques often ignore participation (see Section 2.1 and Section 2.2). Without rigidly defined roles, it is unclear how proper teamwork can be achieved without an understanding of participation.

For example, consider the JAG defined in Figure 1. $\gimel_{a,2}$ requires input $i_1^{a,2}$ from the output $o_1^{a,1}$ of $\gimel_{a,1}$. The agent participation in $\gimel_a$ will contribute to restricting sharing of $\gimel_{a,1}$'s result with only agents participating in jag $\gimel_a$ and not with agent participating in $\gimel_b$. The information interdependence exists locally and no higher than $\gimel_a$, thus agents not involved in this subspace of the joint activity probably do not need to know about $\gimel_{a,1}$'s result.

One key nuance we have encountered, which is lacking on most approaches, is joint activity instance tracking. Let's consider the joint activity $\gimel_{deploy-ugs}$; Two agents can participate in the same joint activity instance, (both are working together to deploy ugs on the left side of the field) or they can work on two separate instances of the same joint activity (agent A deploys UGS on the left, and agent B deploys UGS in the center of the arena). Instance tracking together with participation proved to be key in differentiating intent and ***team organization interdependence***. This has ramifications for strategy and information sharing as well.

7.3 Strategy

In most real world problems that involve teamwork, there are usually multiple ways to tackle the problem, each with different costs and benefits. For teams to be successful, they must have some goal alignment to ensure the team members are utilizing compatible strategies (see Steiner [36] and Decker [39]). This is another aspect of interdependence that manifests itself at run time: team *focus* Section 4.4.

We consider that the focus of a activity cannot be statically set for all teamwork scenarios and thus designed joint activity graphs to support a multidimensional representation of the activity's *focus* (or foci). It is of note that an activity's foci is not defined in the taskwork but rather defined outside of taskwork as a strategy parameter that can dynamically change. For instance, a team may decide to focus on speed while another on quality. Often teams will have multiple competing foci and balance them at runtime. This is essential for reusable team behaviors across different strategic approaches. This multidimensional abstraction of the focus represents the agents interests in task quality (e.g., speed, accuracy, quantity, etc.) and can inherently be specified at individual levels in the task work and per agent. This allows our structure to support more recent work on preferences, [42] and consider concepts such as commitment [13] as an agreement on the work to be done (taskwork) and the foci to work towards.

Team participation, a type of team organization interdependence, is crucial to information relevance as well as task and role allocation. We know that agents often work together with the goal of improving some joint activity focus (e.g., speed, accuracy, resource consumption). Strategy interdependencies help understand and address the varying, potentially conflicting or synergistic, foci at play during execution of the joint activity, thereby addressing ***strategy interdependence***.

8 JAG supported adaptation

The Capture-the-Flag domain described in section 5 allows us to exercise a broad range of teaming challenges and operationalize interdependence design principles to show adaptability to the environment, the team and the joint activity.

We ran teams of heterogeneous agents from size 5 (2 UAVs and 3 UGSs) to size 23 (20 UAVs and 3 UGSs) against each other in our virtual environment. Due to safety and space constraints, we only ran 5v5 and 6v6 games on hardware. All these games were run using the exact same JAG shown in Figure 5. Teams were able to adapt and coordinate independent of scale (see video *ctf-scale* in Supplemental Video S1) addressing interdependencies described in Section 4.

Our tools enable systematic identification and management of interdependence through its formalism. ***Decomposition interdependencies*** are handled by the joint activity natural hierarchical structure through jag children. ***Resource and information state interdependencies*** are captured by joint activity data flow definition in combination with participation awareness. ***Task and team synthesis interdependencies*** are reflected through each joint activity synthesis definition also in combination with participation awareness. ***Skill and strategy*** are exposed and addressed at run time by the JAG Engine and user defined strategies.

8.1 Structural adaptation

Agents were able to reason over task allocation using decomposition and strategy interdependence, and participation status.

For instance, agents would dynamically re-prioritize their behavior to go after the enemy flag if and when they realized there was no agent currently participating in that section of the joint activity. This would happen when offensive agents would get suppressed on their way to the enemy flag.

Agents would also understand whether they were participating in the same joint activity instance (such as A and B laying down UGS on the right side together) or in different instance of the same joint activity (such as A laying down UGS on the right and B also laying down UGS but in the center).

8.2 State adaptation

Agents were able to resolve resource constraints using information and participation interdependencies. For example, two agents would often try to deploy the same UGS. Using participation status they were able to identify the need for negotiation which would lead one of the agents to reevaluate its activity to go after a different UGS. In our strategy, we used first come first serve and distance based costs as negotiation processes. However, it is important to note that the specific negotiation process is less important than the identification of the need for negotiation within context. Agents were able to quickly adjust to new information whether it was a new location of the flag or the enemy (which would automatically trigger planning of a new path) or the fact that one's own flag had been grabbed (leading to re-prioritization of behavior to intercept the enemy with the flag). In a dynamic and information rich environment such as the CTF domain, information sharing and observation are a significant source of knowledge update.

Early in design we were confronted with the "artificial" dichotomy of information provenance. There were two distinct but similar pathways for an agent to ingest information depending on whether it was observed by local sensors (vision) or received through communication by team members. We realized that the source of information could instead be a characteristic of the piece of information received and that there was no need to distinguish them in processing. Team members can be thought of as sensors, and the information received can be characterized accordingly based on the sensor (teammate) and transport medium characteristics. This makes dealing with cognitive activities such as reifying information simpler, robust to failure and often elegantly handled.

By associating characteristics to each information provider (sensors, teammates) such as latency, accuracy, reliability, an abstraction can be made over the reception of information which does not need to distinguish local vs. remote information, and makes handling reaction to change simpler, more consistent and elegant. Often, assumptions about local sensors are made which may hide characteristics of the transport medium and source, and leads to unnecessary special handling. In Capture-the-Flag's agent design, we successfully removed special information processing based on the source or transport medium in favor of information characterized along the dimension of interest. As such, reacting and adapting to new information behaved completely independently of its provenance and transport which allowed, for example, agents to re-plan their trajectory around enemies that were not in their vision range but in range of a teammate (UGS or UAV) somewhere else on the field. Team members know what information may be relevant, enemy location in this example, because data flow and participation indicates what information is in use at any given time (see Section 4.2.2.2 and Section 6.3). This happened **without us, designers, having to make any specific behaviors or adjustments to existing behaviors**.

Accessing agent knowledge is part of the activity and allows situations to fail gracefully. For instance, if getting the location of a resource is a joint activity, one agent can fail to complete the activity which can then be completed by another agent without special consideration. The activity of generating the location for a resource can be completed by all members of the team and synthesis of the answers can applied to that activity the same way they are applied to *physical* activities. It also ties knowledge use to activities which in turns enables adaptability (described in Section 6.3) independent of the knowledge provenance.

8.3 Strategy adaptation

Reacting to new information often means re-evaluating activities under execution. Whether it is because they are no longer relevant, or because they need to be restarted with a different parametrization, tasks need to be interrupted. That said, not all tasks can be abruptly interrupted without further considerations. Designing interruption as a first class system within our framework proved to be an important requirement.

Two main concepts need to be considered: partial results and interruption procedures.

With regards to partial results, there already exists a substantial body of research, of which we were able to take advantage: namely anytime algorithm and its derivatives [43, 44]. Being able to produce partial results is an important consideration when designing adaptable joint activities. Partial results, may influence characteristics of the results (e.g. accuracy) and as such can be processed by synthesis without special consideration.

Some tasks may need to execute a clean up procedure before they can interrupt a behavior (such as release a constraint on a resource). The most blatant example of a need for interruption procedures was delivery of UGS. Initially naively defined, the transport of objects (UGS or flag) proved to be an interesting scenario demonstrating how failing to handle interruption clean up may lead to failure. While in the middle of deploying a UGS, the suppression of the UAV attempting to retrieve the flag, triggered another UAV to re-evaluate current priorities of active task and switch to go after the flag. Still carrying a UGS, the UAV was unable to successfully grab the enemy flag but kept trying without knowing how to "clean up" the previous behavior. This type of situation can be really insidious and can be a common engineering problem in structures such as behavior trees that get constantly reevaluated. Conversely, the clean up can be a requirement of starting a task as well and in that aspect is consistent with pre-conditions and post-conditions in classical planning. For example, if one fails to release a piece of tape, the clean up may be about succeeding at the failed task (failure to release the tape and must try again before moving on - post-condition) or the clean up may be about having a "hand" free to grab something else as part of the subsequent task (need to grab something different and must succeed before undertaking the next task). Interruption is an inherent part of adaptability in unpredictable environments and even more so in human machine teams when opportunities and conflicts have a tendency to arise. In that respect joint activity graphs enable event driven interruption to understand and use contextual information (such as current participation, data flow, interruption's partial result) about the activity at hand to clean up adequately.

In an execution driven environment such as CTF (where the joint activity graph drives the behavior of the agents), agents default to participation in all nodes. Capability, task type, or resource constraints may restrict participation in joint activities. For instance, dropping of a UGS is reserved to the agent carrying said UGS. Note that the trajectory planning section of the drop off joint activity would not be restricted and as such, all agents, including UGS, can contribute a trajectory result (which is valuable as they may have information that other agents would not have - which indeed happened when UGS were used as scouts). An instance of capability restriction would be activating the grab mechanisms, which is restricted to agents capable of grabbing (not UGS).

9 Future work

In this paper, we provided a joint activity graph formalism (Section 3) to capture the key design elements necessary for effective teaming. We also described the JAG Engine (Section 7.1) as a reasoning engine to interpret the JAG formalism and drive the behavior of individual distributed agents working together as a team. In other words, JAGs provide an understanding of team context that enables generation of cooperative team behavior. However, that understanding could be utilized in others ways. Two alternatives include using the JAG representation to support inference and prediction of human team behavior and using JAG representations to support inference and prediction of adversarial team behavior.

In an ongoing project called DARPA ASIST (Artificial Social Intelligence for Successful Teams), we have had some initial success using JAGs to build a mental model of a human team's joint activity in the search and rescue domain. The goal is to use that modeling to support an artificial social intelligence observer to use signals from the team members to build a partial mental model of each participant. Through this JAG-based dynamic model of the team, we aim to enable the artificial social intelligence to generate prediction and potentially intervene to prevent errors, help repair common ground, or simply improve team processes and performance. In a similar way, JAG engines can be used operationally as a real-time C2 decision-aid or for real-time monitoring of the multi-agent behavior.

We are also investigating using the same approach used during the DARPA CREATE program in support of cooperative teams to explore its potential with adversarial teams. Still in the context of Capture-the-Flag, we are working on integrating a way for agents to dynamically track the joint activity of the enemy team using observations of the enemy's actions as signals. This is similar to process of using JAGs to model team behavior, as described for ASIST. However, ASIST is using a single agent, and this work needs to perform the assessment across distributed agents. The challenge is enabling a distributed team of agents to build a pragmatic mental model of the enemy's joint activity and then use that model to make predictions about their intentions. This would allow the team to deploy counter-measures or take other actions to opportunistically gain an advantage.

This future work aims to show that joint activity graphs are an effective structure to capture and track active contribution to tasks, helping to predict team behavior, assess team efficiency, identify team breakdowns, and generate interventions to improve team performance.

10 Conclusion

In this paper we have presented a new formalism, joint activity graphs, as a tool to design joint activity. We have also introduced the JAG Engine as a tool to interpret JAGs at runtime, driving agent behavior. Together these tools enable human-machine systems to manage and exploit the interdependence within the team through the systematic use of joint activity graphs. By providing support for understanding teaming context, JAGs provide an rigorous and systematic approach to effective human-machine team performance.

In Section 3 we presented a formal structure, joint activity graphs, that systematically guides the architecture of human-machine team systems to address these challenges. JAGs assist designers in the design of joint activities and provide shared contextual information at run-time that supports coordination processes, enabling team members to manage their interdependencies with teammates. Specifically, it provides structures to capture hierarchical decomposition, handling of task interdependence, agent participation, sequencing processes, data flow and the synthesis necessary for activity recomposition.

In Section 4, we describe the broad range of teaming challenges in terms of types of interdependence necessary for adaptive teamwork.

In Section 7.1, we introduce the JAG Engine. Its purpose is to ingest and execute joint activity graphs providing the context necessary to recognize when interdependencies arise and their nature: operationalizing their management through adequate coordination processes.

In Section 6 and Section 7.1 we describe how the two tools provide support for the broad range of interdependencies in Section 4.

In Section 8, we described how these principles of joint activity design were applied in the concrete domain of Capture-the-Flag to provide highly adaptive team behavior. Teams of distributed agents were able to do at least as well as fully centralized teams and were more resilient to breakdowns in communication, agent failures and dynamic team re-composition (such as the loss of a member). This demonstrates that adequate identification and management of interdependence allows teams to better understand information relevance [5], handle and recover from coordination surprise, and continuously repair common ground. The result of this adaptability is effective team performance that can be described as *good teamwork*.

We hope the formally guided approach to human-machine team design presented here proves useful to others working toward complex adaptable teams. The approach is supported by principles, guidelines and tools that can help designers develop systems that support effective management of interdependence in order to achieve flexible and adaptable teamwork in human-machine systems.

Data availability statement

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

Author contributions

MV and MJ designed and developed the framework presented in this paper and contributed equally to the writing of the manuscript. All authors contributed to the design and development of the experimental testbed and system implementation. All authors contributed to manuscript revision, read, and approved the submitted version.

Funding

The work in this paper was partially supported by the following grants Title—ASIST Predicting Effective Performance in Teams (PEPT) TA2 Funding Agency—DARPA ARO Agreement #—HR001120C0037 Title—JACKPOT: Joint Activity and Contextual KAOS Policies for Operational Teaming Funding Agency—DARPA Agreement #—HR00112090024. This material is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contracts No. HR00112090024 and No. HR001120C0037. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Defense Advanced Research Projects Agency (DARPA).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Supplementary material

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fphy.2022.969544/full#supplementary-material

SUPPLEMENTAL VIDEO S1
Capture-the-Flag at scale.

References

1. Castelfranchi C. Modelling social action for AI agents. *Artif Intelligence* (1998) 103:157–82. doi:10.1016/S0004-3702(98)00056-3

2. Clark HH. *Using language*. Cambridge [England]New York: Cambridge University Press (1996).

3. JC Beck, O Buffet, J Hoffmann, E Karpas, S Sohrabi, editors. *Proceedings of the thirtieth international conference on automated planning and scheduling*, 30. France: AAAI Press (2020).

4. Demir M, McNeese NJ, Cooke NJ. Team synchrony in human-autonomy teaming. In: J Chen, editor. *Advances in human factors in robots and unmanned systems*, 595. Cham: Springer International Publishing)Advances in Intelligent Systems and Computing (2018). p. 303–12. doi:10.1007/978-3-319-60384-1_29

5. Klein G, Feltovich PJ, Bradshaw JM, Woods DD. Common ground and coordination in joint activity. In: WB Rouse KR Boff, editors. *Organizational simulation*. Hoboken, NJ, USA: John Wiley & Sons (2005). p. 139–84. doi:10.1002/0471739448.ch6

6. Johnson M, Bradshaw JM. The role of interdependence in trust. In: *Trust in human-robot interaction*. Elsevier (2021). p. 379–403. doi:10.1016/B978-0-12-819472-0.00016-2

7. Ilgen DR, Hollenbeck JR, Johnson M, Jundt D. Teams in organizations: From input-process-output models to IMOI models. *Annu Rev Psychol* (2005) 56:517–43. doi:10.1146/annurev.psych.56.091103.070250

8. March JG, Simon HA. *Organizations (cambridge, mass*. 2nd ed. edn. USA: Blackwell (1993).

9. Malone TW, Crowston K, Toward an interdisciplinary theory of coordination. In: *Center for Coordination Science, Sloan School of Management, Massachusetts Institute of Technology*. Tech Rep CCS (1991).

10. Johnson M, Bradshaw JM. How interdependence explains the world of teamwork. In: WF Lawless, J Llinas, DA Sofge, R Mittu, editors. *Engineering artificially intelligent systems*, 13000. Cham: Springer International Publishing)Series Title: Lecture Notes in Computer Science (2021). p. 122–46. doi:10.1007/978-3-030-89385-9_8

11. Hoc JM. Towards a cognitive approach to human–machine cooperation in dynamic situations. *Int J Human-Computer Stud* (2001) 54:509–40. doi:10.1006/ijhc.2000.0454

12. Decker KS. *Environment centered analysis and design of coordination mechanisms*. Amherst, MA: Doctoral dissertation, University of Massachusetts Amherst (1995).

13. Jennings NR. Coordination techniques for distributed artificial intelligence. In: GM O'Hare NR Jennings, editors. *Foundations of distributed artificial intelligence*. Wiley (1996). p. 187.

14. Malone TW, Crowston K. The interdisciplinary study of coordination. *ACM Comput Surv* (1994) 26:87–119. doi:10.1145/174666.174668

15. Lawless WF. Towards an epistemology of interdependence among the orthogonal roles in human–machine teams. *Found Sci* (2021) 26:129–42. doi:10.1007/s10699-019-09632-5

16. Kaber DB, Endsley MR. The effects of level of automation and adaptive automation on human performance, situation awareness and workload in a dynamic control task. *Theor Issues Ergon Sci* (2004) 5:113–53. doi:10.1080/1463922021000054335

17. Kaber DB, Riley JM, Tan KW, Endsley MR. On the design of adaptive automation for complex systems. *Int J Cogn Ergon* (2001) 5:37–57. doi:10.1207/S15327566IJCE0501_3

18. Entin EE, Serfaty D. Adaptive team coordination. *Hum Factors* (1999) 41:312–25. doi:10.1518/001872099779591196

19. Burke CS, Stagl KC, Salas E, Pierce L, Kendall D. Understanding team adaptation: A conceptual analysis and model. *J Appl Psychol* (2006) 91:1189–207. doi:10.1037/0021-9010.91.6.1189

20. Johnson M, Vignati M, Duran D. Understanding human-autonomy teaming through interdependence analysis. In: *Symposium on human autonomy teaming* Southsea, United Kingdom: NATO/STO (2018). p. 20.

21. Fikes RE, Nilsson NJ. Strips: A new approach to the application of theorem proving to problem solving. *Artif Intelligence* (1971) 2:189–208. doi:10.1016/0004-3702(71)90010-5

22. Brooks RA. Intelligence without representation. *Artif Intelligence* (1991) 47:139–59. doi:10.1016/0004-3702(91)90053-M

23. Erol K, Hendler J, Nau DS. HTN planning: Complexity and expressivity. *AAAI* (1994) 94:1123

24. Smith RG, Davis R. Frameworks for cooperation in distributed problem solving. *IEEE Trans Syst Man Cybern* (1981) 11:61–70. doi:10.1109/TSMC.1981.4308579

25. Duarte M, Oliveira SM, Christensen AL. Evolution of hybrid robotic controllers for complex tasks. *J Intell Robot Syst* (2015) 78:463–84. doi:10.1007/s10846-014-0086-x

26. Duarte M, Gomes J, Costa V, Oliveira SM, Christensen AL. Hybrid control for a real swarm robotics system in an intruder detection task. In: G Squillero P Burelli, editors. *Applications of evolutionary computation*, 9598. Cham: Springer International Publishing)Series Title: Lecture Notes in Computer Science. (2016). p. 213–30. doi:10.1007/978-3-319-31153-1_15

27. Durfee E, Montgomery T. Coordination as distributed search in a hierarchical behavior space. *IEEE Trans Syst Man Cybern* (1991) 21:1363–78. doi:10.1109/21.135682

28. Johnson M, Vera A. No AI is an island: The case for teaming intelligence. *AI Mag* (2019) 40:16–28. doi:10.1609/aimag.v40i1.2842

29. Johnson M, Bradshaw JM, Feltovich PJ, Jonker CM, Van Riemsdijk MB, Sierhuis M, Coactive design: Designing support for interdependence in joint activity. *J Human-Robot Interaction* (2014) 3:43. doi:10.5898/jhri.3.1.johnson

30. Johnson M, Shrewsbury B, Bertrand S, Calvert D, Wu T, Duran D, et al. Team IHMC's lessons learned from the DARPA robotics challenge: Finding data in the rubble. *J Field Robot* (2017) 34:241–61. doi:10.1002/rob.21674

31. Iovino M, Scukins E, Styrud J, Ögren P, Smith C. A survey of Behavior Trees in robotics and AI. *Robotics Autonomous Syst* (2022) 154:104096. doi:10.1016/j.robot.2022.104096

32. Lesser V, Decker K, Wagner T, Carver N, Garvey A, Horling B, et al. Evolution of the GPGP/TÆMS domain-independent coordination framework. *Autonomous Agents Multi-Agent Syst* (2004) 9:87–143. doi:10.1023/BAGNT.0000019690.28073.04

33. Endsley MR. Toward a theory of situation awareness in dynamic systems. *Hum Factors* (1995) 37:32–64. doi:10.1518/001872095779049543

34. Thompson JD *Organizations in action: Social science bases of administrative theory*, 10. New York, NY: McGraw-Hill College (1967).

35. Perrow C. *Normal accidents: Living with high risk technologies*. Princeton, NJ: Princeton university press (1999).

36. Steiner ID. Group process and productivity. *Social psychology*. New York: Academic Press (1972).

37. Tambe M. Towards flexible teamwork. *J Artif Intell Res* (1997) 7:83–124. doi:10.1613/jair.433

38. Hughes J. Why functional programming matters. *Comput J* (1989) 32:98–107. doi:10.1093/comjnl/32.2.98

39. Decker KS, Lesser VR. Generalizing the partial global planning algorithm. *Int J Coop Inf Syst* (1992) 01:319–46. doi:10.1142/S0218215792000222

40. Salas E, Sims DE, Burke CS. Is there a "Big five" in teamwork? *Small Group Res* (2005) 36:555–99. doi:10.1177/1046496405277134

41. Myers KL. Cpef: A continuous planning and execution framework. *AI Mag* (1999) 20:63

42. Rossi F, Venable KB, Walsh T. Preferences in constraint satisfaction and optimization. *AI Mag* (2008) 29:58. doi:10.1609/aimag.v29i4.2202

43. Dean TL, Boddy MS. An analysis of time-dependent planning. *AAAI* (1988) 88:49

44. Zilberstein S. Using anytime algorithms in intelligent systems. *AI Mag* (1996) 17:73

TYPE Original Research
PUBLISHED 08 December 2022
DOI 10.3389/fphy.2022.975247

Using meta-reasoning for incremental repairs in multi-object robot manipulation tasks

Priyam Parashar[1*†], Ashok K. Goel[2] and Henrik I. Christensen[1]

[1]Contextual Robotics Institute, University of California, San Diego, San Diego, CA, United States,
[2]Georgia Institute of Technology, Atlanta, GA, United States

OPEN ACCESS

EDITED BY
William Frere Lawless,
Paine College, United States

REVIEWED BY
Jiuchuan Jiang,
Nanjing University of Finance and Economics, China
Michael Mylrea,
University of Miami, United States

*CORRESPONDENCE
Priyam Parashar,
priyam8parashar@gmail.com

†PRESENT ADDRESS
Priyam Parashar,
Meta AI, Pittsburgh, PA, United States

SPECIALTY SECTION
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

RECEIVED 22 June 2022
ACCEPTED 07 October 2022
PUBLISHED 08 December 2022

CITATION
Parashar P, Goel AK and Christensen HI (2022), Using meta-reasoning for incremental repairs in multi-object robot manipulation tasks.
Front. Phys. 10:975247.
doi: 10.3389/fphy.2022.975247

Robots tasked with object assembly by manipulation of parts require not only a high-level plan for order of placement of parts but also detailed low-level information on how to place and pick the part based on its state. This is a complex multi-level problem prone to failures at various levels. This paper employs meta reasoning architecture along with robotics principles and proposes dual encoding of state expectations during the progression of task to ground nominal scenarios. We present our results on table-top scenario using perceptual expectations based in the concept of occupancy grids and key point representations. Our results in a constrained manipulation setting suggest using low-level information or high-level expectations alone the system performs worse than if the architecture uses them both. We then outline a complete architecture and system which tackles this problem for repairing more generic assembly plans with objects moving in spaces with 6 degrees of freedom.

KEYWORDS

cognitive artificial intelligence, cognitive robot architecture, robot system architecture, knowledge-based (KB), task planning, task and motion planning, meta-reasoning

1 Introduction

Industrial robots, i.e., robots producing consumer goods at industry-scale, have remained the fastest growing market in recent times [1]. This reliability and demand are attributed to model-based programming paradigms [2–7] which enable program-and-replay for manipulation tasks. Model-based programming assumes access to completely modeled objects, pre-existing sets of robot-motion plans, and a structured environment that does not change over-time. The requirements of the next Frontier in small-scale robotics, however, cater to a scenario where the end-user wants to program the robot on one instance of the task and expects generalization over different instantiations of the same task or tasks that are similar in terms of objects and actions [8–10]. This problem context brings several realistic, but presently unattainable, robotics challenges that are summarized by the overarching question of "how to transfer known high-level plans for a given task to a similar but different environment represented as low-level observations"?

We focus on the class of multi-object manipulation tasks where a task can only be achieved by correct handling of multiple objects that leverages their affordances. Examples include getting soup out by dipping the ladle with its concave side facing up and attaching a screw to a washer by aligning the screw shaft and with the washer hole. Given such objects affordances (concavity, liquidity; shaft-feature, hole-feature) and rules governing their interactions (contains; inserts), prior work on these tasks reasoning writes out high-level formulae or grammar which should be followed in achievement of the task. However, as pointed out in [11, 12] these formulae do not elaborate "how" the robot should move to accomplish satisfaction of the goal state from any given initial state. Motions depend on the value of the next state but also on low-level percepts in the current state like vision and joint readings. When solved analytically this is a computationally hard process. Thus, we have witnessed a shift towards approximate solutions that leverage data and structured principles [13]. We propose a complementary incremental approach where we can expand the scope of the application of a plan for a task based on trial-and-error. We address the specific problem in which, given a motion planner, a high-level plan for an assembly task, and the plan's successful execution on a specific grounding (i.e., object poses in 3-dimensional space), how might the robot transfer the planner and the plan to a new grounding?

Data driven learning has demonstrably worked very well for applications where noisy pixel or sensor inputs need to be matched to symbols [14, 15] or to a regressed control output [16]. However, the strength of these methods comes from the ability to mine unlimited amounts of data, either through web crawling or simulators. The robotics domain in contrast has limited and specific data, which leads to overfitting and non-generalized task solutions. A hierarchical model which can parse the environment using generic symbols (which can be learned from widely available data) but then uses specific data and learning at lower level to ground those symbols in the given environment and execute plans can bridge this gap [9, 13]. There is a corollary problem to learning policies, then: given manipulation policies and seed sets of states that lead to success, find other connecting states such that any given initial state leads to task success. Our approach does this by learning state-entity sets in which the agent explores an action space through trial-and-error and gathers enough state-data over time to enable robust execution with learned "good states".

More specifically, we want to balance exploitation of high-level knowledge of goals and plans with low-level exploration such that the agent can learn reactive repairs while maintaining goal-driven reasoning and behaviors. Concretely, this paper grounds assembly plans as actions on objects and leverages known object affordances [17–19] as the key state-variable. Object affordances in this chapter are defined as keyframes (position, orientation) with respect to the object's centroid (also known as the object's frame). This affordance-based state description, however, is low level, continuous, and incompatible with traditional task planning. In order to bridge this gap we take a dual-encoding approach leveraging the task domain definitions to also ground plans in high-level pre and post conditions. Thus, our architecture can reason about plans based on low-level sensor observations as well as high-level knowledge inputs. Meta-reasoning and goal-driven reasoning has addressed sophisticated tasks but mostly within disembodied contexts [20–24]. For example, the REM system uses meta-reasoning for transforming a plan for disassembling a device to a plan for assembling it from its components [25]. In contrast we address simpler tasks in an embodied context. This implies a grounding of the plan reasoning and meta-reasoning in visual encodings. The key contributions are an updated architecture for meta-reasoning, a theory for classifying failures in embodied systems, and grounding of meta-reasoning in perceptual expectations. These contributions build on previous several streams of meta-reasoning research which attacked this question of how to account for low-level observations [25–29] as well as robotics research investigating the conceptualization and operationalization of robotics processes as lifted symbolic plans and knowledgebases [2–5], [30]. Note that meta-reasoning itself is also a computationally hard-problem [26, 31] thus we use rule-based methods and data-based approximations in this chapter to ground these components.

The dual-encoding methodology presented in this chapter was observed to be more generic than low-level repair or symbolic repair used alone, as seen in Section 6. Further an application of this architecture with reinforcement learning used to learn action sequences for tasks showed much faster convergence of learning with the structure provided by the meta-reasoner [32]. Our experiments described in Section 7 show that a hierarchical architecture with a high-level repair module solves more instantiations of tasks than one with no high-level repair module [33]. In the next section we present a motivating example. Section 3 gives a quick introduction of relevant concepts and related literature. In Section 4, we present a first experiment which establishes the usefulness of visually grounded lower-level expectations. In Section 5, we present our second experiment which further refines lower-level expectations to account for object configurations and recovery from grounding-level failures.

2 Motivating example

Consider a robot executing a sequential plan to assemble multiple parts together. The robot is given one task of inserting a cylindrical housing into a matching feature on the task-board. The housing has a wider face on one-end which is not compatible with this insertion-feature. The execution sequence for this task is provided in Figure 1. We call this a nominal task sequence.

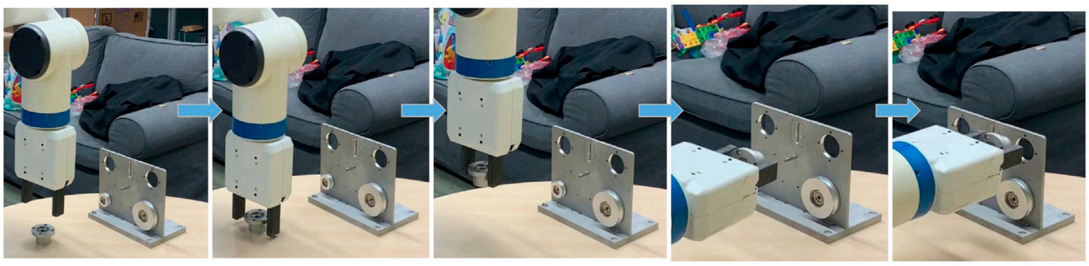


FIGURE 1
Nominal task sequence for inserting housing into the task-board.

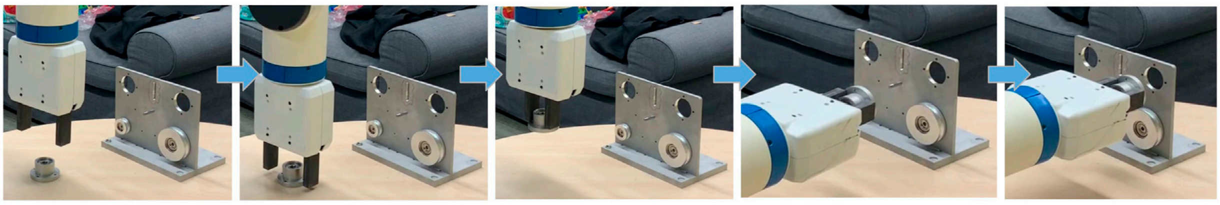


FIGURE 2
Variant setting with upside-down housing placement which leads to failed insertion task.

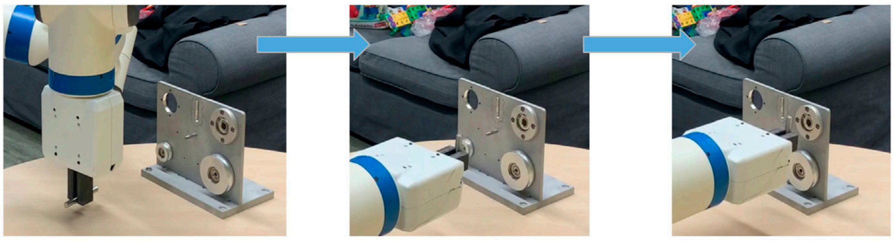


FIGURE 3
Variant setting with a new object whose placement differs from housing and repetition of same insertion task without further reasoning leads to failure.

The robot is now tasked to achieve the same goal of housing being inserted into the task-board in a variant setting, namely one where the initial pose of housing is upside-down on the table. At a high-level the current task remains the same, so if the robot repeats the same sequence as in the nominal setting without further reasoning it fails. Failure occurs because we transition into an incompatible state from which the insertion skill cannot occur as expected. This is shown in Figure 2. This is a simpler class of adaptation where the same objects are being used but their relative poses differ, so the robot's actions need to be goal-driven but also account for these grounded differences.

Now consider the robot is tasked with the goal of inserting a cylindrical shaft into the housing *via* the central hole-feature in it. Again, if the robot does not reason about task-relevant correspondence between the shaft and the housing and just repeats the insertion skill's plan as instantiated based on previous example then it will run into another failure. This situation is shown in Figure 3. This is a more complex case where the skill is the same but the object the skill is being applied to has changed. Now the robot needs to first understand how to adapt known affordances of objects based on previous examples, and next to plan an action sequence to account for the variations.

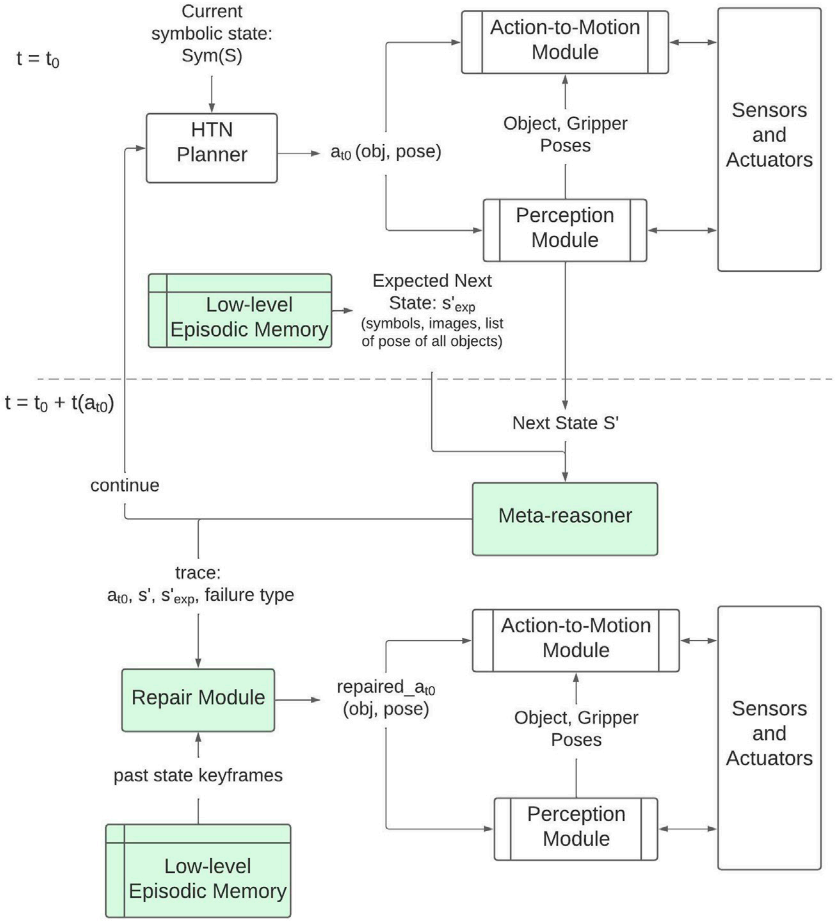


FIGURE 4
Overview of the system components and a temporal flow of how processes unfold across them when failures are identified.

An interesting aspect of this investigation is the entangled relevance of object and grounded pose towards task success. A robot is an acting agent, rather than a reasoning-only agent. A high-level model of these actions would only deign to answer, "use action A on object O." However, we contend that it is not enough to answer, "which object?" but the robot also needs to know what pose of the object with respect to the given task is actuable. A robot perceives its surroundings through 2D and 3D images, and then extracts out relevant semantic symbols as well as the grounded 6-dimensional pose (x, y, z position and r, p, y rotation with respect to base coordinate system) of these symbols in the environment. Each grounded pose of these objects may require a different grasp and alignment from the robots to enable their assembly which is non-trivial to plan since robots have a different physicality and reach in the Euclidean space as compared to humans. Prior work assumes the definition of the action encompasses this reasoning and focuses on high-level sequencing only. We relax this assumption, asking the question how to capture the low-level task-relevance of object affordances from successful high-level plans and use it to generalize said task-plan.

Further these interactions also implicitly respect dynamics like rigid object physics, friction dynamics, and even specific instantiation of objects in the current world. It is computationally intractabe to model each and every one of these aspects separately, thus we use the past traces from successful executions of a task to serve as heuristics on which poses make sense for the given task. Therefore, our focus is on the architecture which enables this learning from failures, incorporation of new evidence into knowledge-base and better planning rather than robustness of the object or affordance features themselves. Our primary goal here is to motivate the fact that the relevance of an object to a task and the function of the object's grounded pose within that task context are entangled together. An agent cannot generalize to a novel situation with an answer to only one of these two questions. Thus, in this chapter we explore representations encoding object affordances, evaluate if these encodings improve example-based instantiation of novel tasks and present a framework which uses these encodings to facilitate automatic learning from agent's failures.

3 System overview

Figure 4 shows the lifecycle of a typical repair process described in this chapter. The key components of this framework are as follows:

1. A task: A task is a multi-action motion sequence which achieves a given goal state. Given a goal-state the HTN planner chooses the correct task. Thereby given the current state and overarching task it generates the next action towards that task.
2. An action: An action is defined as a skill-label (move, insert, etc.) along with its supporting arguments. In this chapter, we first discuss repair of objects as arguments and then expand scope for repairing embodied poses.
3. A low-level episodic memory: We assume the robot has past episodic memory of at least one successful execution of the task under consideration. This memory is assumed to have low-level information about objects and pose perception based on sensors during execution. Given a task and next action, one can query the expected state after successful execution of that action. One can also query the set of states which successfully lead to execution of a given action.
4. A meta-reasoning component: This component compares the current state with the expected state. It assumes a failure taxonomy to be present and follows rule-based assessment of whether a failure is present or not. It then either asks the deliberative planner to continue or passes on the failure category as well as low-level details to the repair module.
5. A repair module: Given failed action and current task, the repair module generates suggestions for next action which can lead to previously known states which led to success. We assume the HTN knowledgebase has actions which can bridge transformation from current state to "promising states" [32].

The biggest reason for failures in robotics is due to non-determinism over initial state and stochastic execution of actions. In this chapter, we focus on failures induced by missing availability of correct objects and associated table-top rearrangement (Section 6) and wrong positioning of objects (Section 7). Line items three to five described above are core contributions towards generating such repairs for an embodied system.

4 Related work

It is agreed in robotics that hybrid systems that can do both deliberation and reactive revisions pave the way for more complex robotic applications [34, 35], but there does not exist a systematic theory of how to combine distinct levels of planning, reaction, and learning. For instance, ROS [36], a robotics middleware commonly used in research and adopted by some circles in industry [37], uses hybrid planners at both navigation and manipulation level which use both global and local planning behaviors. Parashar et al [38] proposed a hierarchy of failures for such multi-layered architectures, as seen in Figure 5, attempting to scope systematic investigation. This paper is scoped at the level of understanding objects and pose related failures.

Architecturally this paper is informed by cognitive robotics systems like CoSy Project [34] and CRAM [39] and paves a bridge between such hybrid architectures and meta-reasoning components [40, 41]. Given heuristic or expert knowledge about a process, a meta-reasoning system (Figure 6) generates expectations about the state of the world given the actions applied to it, compares the observed state of the world with the expected state, and maps the discrepancy between expectation and observation into one or more repairs at the deliberative level [25, 26]. The recognition of a failure through a comparison of the expected state and the observed state can be challenging if the observations are made through low-level sensors and the expectations are encoded in terms of abstract knowledge representations. We seek a more general strategy for comparing expected and observed states and recognizing failures for the robotics domain.

Our work has some similarity with [23], since they too use HTN plans annotated with expectations to conduct meta-level reasoning over their incomplete plans. However, their expectations are of a conceptual form, abstracted on top of environmental symbols. Jones and Goel [29] present "Empirical Verification Procedures" which ground all high-level concepts and axioms known to the agent in lower-level precepts in a video game. Prior work [32, 42, 43], combines meta-reasoning with reinforcement learning using purely visual form expectations. However, they still use symbolic descriptions or computerized descriptions of visuals which simplifies the perception part of the problem.

Finally, to ground the planning problems we make use of formalism provided by Hierarchical Task Networks. STRIPS planning enables a search-based planning solution in the state-space of the planning domain, however given the repetitive structure of assembly plans, HTNs fare better in terms of efficiency as they allow reuse of expert knowledge. Such procedural and routine-based problems occur in scheduling and logistics regularly, and hierarchical task networks [44] (HTNs) have been used to instead express procedures for completing tasks. HTN planning was formalized and operationalized *via* the SHOP [45] and SHOP2 [46] planners in the International Planning Competition(s). HTNs are a popular way of designing domain-configurable as well as domain-specific planning domains [47], with the procedures defining search control over the state-space to make planning faster. A comprehensive review of different HTN planners is provided in [48].

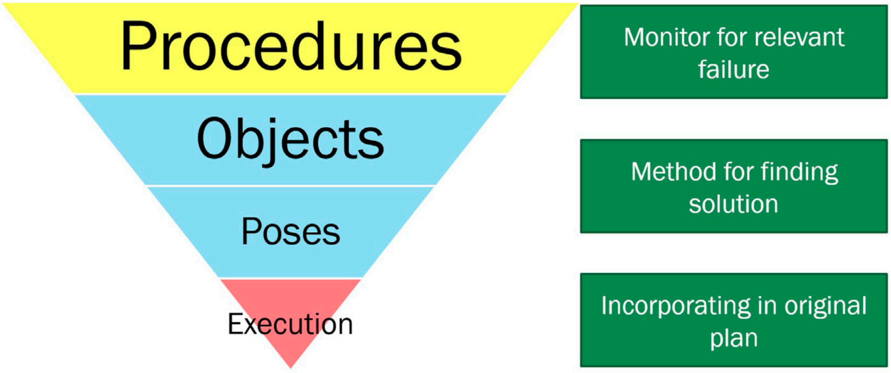

Procedures
Objects
Poses
Execution
Monitor for relevant failure
Method for finding solution
Incorporating in original plan

FIGURE 5
The hierarchy of failures (categorised by colour) that occur in a goal-driven, multi-layered robotics architecture. The green boxes represent the three required components to meta-reason about each class of failures.

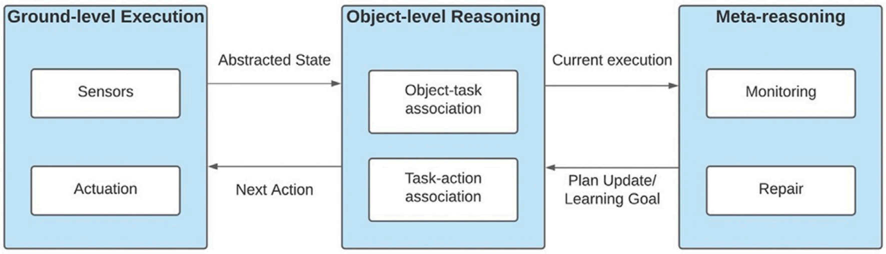


FIGURE 6
High-level system architecture describing the 3 salient components of a meta-reasoning system. Ground-level execution relies on sensors which read the environment state and actuators which manipulate it. The object-level reasoning component which relies on domain knowledge to formulate long-horizon plans given the current state. Finally, the meta-reasoning component which monitors the execution of the object-level plan and launches repairs when failures are identified.

5 Background

5.1 Hierarchical task networks and assembly task domain

We use the HTN formalism [49] for defining our task planning domain and problem. The domain is given as $\mathbf{D} = <\mathcal{P}, \mathcal{T}, \mathcal{M} \ldots >$. $\mathcal{T}$ is a set of tasks in the domain, and $\mathcal{M}$ is a set of methods (recipes) defining how to decompose $\mathcal{t} \in \mathcal{T}$ to smaller subtasks which is called its task network. When $\mathcal{t}$ cannot be further decomposed then $\mathcal{t} \in \mathcal{P}$ and is called a primitive action, which is directly executed by the underlying agent. γ is the set of preconditions defined as grounded symbols and predicates over grounded symbols for each method. A method $\mathcal{m}$ can be applied on task $\mathcal{t}$ while in state $\mathcal{s}$ if $name(\mathcal{m}) = \mathcal{t}$ and $\mathcal{s}$ satisfies $\gamma(\mathcal{m})$. δ is the set of effects that executing primitive-action $\mathcal{t}$ will affect on the world-state: if $\mathcal{t} \in \mathcal{P}$ is executed in $\mathcal{s}$ then next state:

$$\mathcal{s}' = (\mathcal{s} - \delta^{-}(\mathcal{t}) + \delta^{+}(\mathcal{t}))$$

$\delta^{-}(\mathcal{t})$ and $\delta^{+}(\mathcal{t})$ represent minus and add effects respectively. The assembly task domain that these HTNs operate on is based upon the domain formulation presented in [56] and does not cover those details in the current scope. In the later experiments we explicate relevant aspects of the task planning domain to ground our understanding and discussion.

5.2 Task and motion planning

Task and motion planning (or TAMP) [50, 51] is a task planning formalism extended to account for the continuous-space execution that robots need to do. This is done by extending the next-state equation from previous section and associating a set of continuous-valued valid poses for each agent and object implicated in state $\mathcal{s}'$. Let $\mathcal{R}$ be the relation connecting an object or agent oa with its valid poses under predicate $\mathcal{p}$, then for state description $\mathcal{s}$ containing $\mathcal{p}$ applied on agents or objects:

$$\mathcal{R}(\mathcal{s}) = \mathcal{R}(\mathcal{p}(oa_i))$$

Given such symbol-to-set mapping TAMP does a hybrid optimization over value assignment (as described in previous sub-section) as well as finding a feasible space of continuous execution. However, this symbol-to-set mapping is non-trivial to define. The procedures covered in this chapter formally solve this problem *via* incremental learning of these sets given successful and unsuccessful examples and unlimited time for the robot to conduct trial-and-error. This learning technique is the same as set-expansion techniques used in knowledge-based information retrieval [52–54] except we are learning the set-based relation $\mathcal{R}$ in 6DOF space which does not exhibit any obvious semantic structure.

6 Reasoning across different objects

In this section, we discuss a simple table-top shape drawing task where we ask an agent to draw an alphabet with playing bricks. We wish to evaluate our meta-reasoning architecture that given a variation on the original task along with visual expectations related to its execution, can the agent provide better plan repairs than one with only conceptual expectations. Our experimental setup uses a Baxter with an eye-in-hand camera setup (camera is situated on the wrist). For evaluation we create example failure cases to compare the meta-reasoning cycle which uses dual-encoded expectations (visual + conceptual) against one which only uses the symbolic-level expectations. In the following sections we formulate this problem using task domain description, present representations for encoding visual expectations and discuss the processes which can utilize visual expectations to propose plan repair instead of conceptual expectations. We also explain our implementation for extracting these visual expectations out of an image-stream. In the results section we present a qualitative assessment of our system for failure recovery where failures are induced by changing the environmental conditions to mismatch plan pre-conditions at different depths.

6.1 Problem setup

The problem domain is to use Mega BloksTM to draw shapes on the table-top; for simplicity will be referring to a single unit as a block. Our system considers two different shapes of blocks: 1 × 1 and 1 × 2; and supports two different colors: blue and red. Goals are communicated as strings naming the shape to be drawn and relate to a sequence of placements of specific blocks which draw the shape. Each block's physical placement is described by two attributes, its orientation with respect to the table's axis and the location of its centroid in the workspace. When blocks are recognized in an image, they are indexed with a number starting at 0, e.g., b_0 =<color, shape>. To describe the placement of two blocks with respect to each other we use a graph-based format where $\psi_{0,1}$ is a coordinate system transform between the centroid of b_0 and the centroid of b_1. A 1 × 1 sized Mega Blok is of length 6.1 cm and width 6.1 cm, which we denote as lb in the rest of this section.

To codify the pre-conditions and effects of the HTN tasks we use a symbolic state description which include: 1) ob_{grip}: description of the block held in the gripper, Φ if empty and b_j if block with ID j is held in the gripper, and 2) β: set of pairs depicting the required blocks and their mapping to recognized blocks on the table. The overall system uses other variables for planning purposes, but they have been abstracted because they are not relevant to the current discussion. The only primitive action available to planner is place (b_j,x,y) which maps to a heuristic policy under which it grasps and then moves the block b_j to (x,y).

6.2 Approach: Hierarchical representation of expectations

In order to scope the search complexity, the high-level planning framework uses lifted symbolic descriptions, however if a failure is noted the meta-reasoner needs access to lower-level, continuous observations which is encoded in the expectations. Our overall planning then is hierarchical in nature, where high-level planner assigns a block symbol to the place action and then a lower-level simple reasoner assigns the requisite (x, y). Leveraging this dual-level planning, we encode a hierarchical schema of expectations. The higher-level level of expectations has block symbols describing the relation between a shape and required block units. The lower-level stores cropped visual grid-maps centred on each block to capture a locally detailed description of its placement. Each grid-map is of length $3lb \times 3lb$ to include the block and some of the surrounding context. We chose a size which records the surrounding context as such mid-level features have been seen to work better for task-level reasoning when compared to hyper-local object-specific features. Readers should note that such a low-level description would require domain knowledge to be encoded since its form is tightly integrated with the goal of the problem itself.

The plans are annotated with expectations at the primitive action level and bubbled up to be associated with the task by assigning the parent task the expectation of the last primitive action in its decomposition. Typical HTN methods/tasks have only symbolic pre-conditions associated with each possible decomposition and then pre and post-conditions associated with each action under it. By bubbling up these grid-based

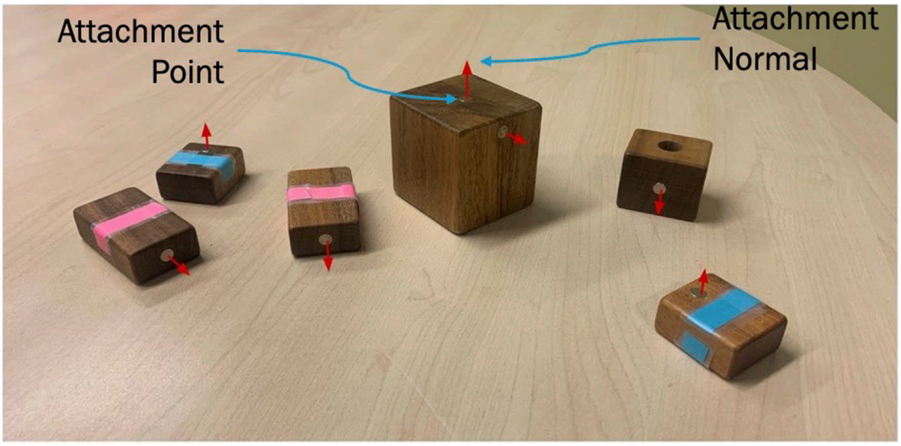


FIGURE 7
Overview of all objects implicated in MagneticAttach task as well as adiagrammatic description of attachment points and attachment normals.

encodings we add more information to our HTN description which also accounts for the final goal of the task.

6.3 Implementation

The HTN plan is executed by an expert kinesthetically driving the agent (hold on to the agent's arm and move it to perform the required actions) to annotate the plan with resulting "ideal profile" of expectations. After each place action the user presses a key to record the block which was placed as well as its 2-dimensional location with respect to the table-top coordinate system. After the full execution, the expectation annotator creates two kinds of databases: one of the complete annotated plan executions, denoted as P_a, which maps each action-step i to demonstrated $< b_j$,x,y $>$ instance. The other one stores the low-level grid-maps and symbolic expectations annotated by the causing action(s) which points back to the parent task itself, denoted by E_{db}. This action-to-expectation mapping can be many-to-one. The second database is key for creating a two-way communication protocol between sensor information and plan knowledge even when symbol grounding fails during run-time.

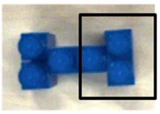
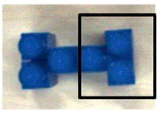
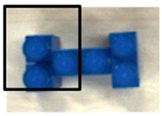
E_{db} is populated using an expectation extractor module. The expectation extractor uses a top-view image feed of workspace, *via* the robot's eye-in-hand setup, to extract expectations associated with each action and task. The block-level description of a symbol is extracted by performing HSV color-thresholding for blob detection on the tabletop view of the symbol under-construction. Once a colored blob is found, its shape is assigned by comparing blob-axis with lb. Next, the visual expectation is the image within a $3lb \times 3lb$ bounding-box centred on the centroid of the blob, and a quantized view of the form of the blob, i.e., a grid-map is created. A grid-map is like an occupancy grid where the occupancy of a cell is decided based on the color presence of the block on a uniformly colored background. The resolution of the grid-map is $0.5 \times lb$.

6.4 Experiment and results

An embodied system can encounter two kinds of failures: logical or physical. By physical we mean misplacement of gripper, wrong state of gripper, etc. This work does not address these failures. In the rest of the paper when we explain our algorithm, we are addressing only the logical failures, i.e., missing blocks, unexpected configuration of blocks, etc. We broadly classify the logical failures into two kinds, one where known entities are observed in an unseen configuration thus going against the explicit nature of pre-conditions, and one where unknown entities are observed breaking the planner's assumptions. We present here an example case where we create both kinds of failures by manipulating the environment. In this example, we have provided the agent with the plans for shape A and H (Figure 7), using two 1 × 1 blocks for H rather than one 1 × 2 block. We use these shapes because they possess the kind of form similarities, we want our algorithm to identify. We want to see if our expectations can help in creating connections between pieces of knowledge already stored in our database better than symbolic expectations. Next, the environment is modified to progressively make the failure more difficult for drawing the shape A. We replace required 1 × 2 block with another:

- Block of same shape but different color
- Set of two 1 × 1 blocks of same color
- Set of two 1 × 1 blocks of different color

TABLE 1 Summary of match results. The white square shows which block's resultant placement expectation in shape H was matched.

Type of Replacement	Affected Action-Exp Pair	Grid-map Match	Symbolic Match
1 × 2, red → 1 × 2, blue	BO = {1 × 2, red, 90°}	B4 = {1 × 2, blue, 90°} 	B4 = {1 × 2, blue, 90°} 
1 × 2, red → 1 × 1, red + 1 × 1, red	BO = {1 × 2, red, 90°}	None	None
1 × 2, red → 1 × 1, blue + 1 × 1, blue	BO = {1 × 2, red, 90°}	B2 = {1 × 1, blue, 0°} 	None

The mismatch vector is used to identify the first instance of action in the invoked task which uses the mismatched entity, i.e., block in our case. Next, this action's expectation is retrieved from P_a and a nearest-neighbour algorithm is invoked to find ranked matches from E_{db}. We compare the entries retrieved by symbolic matching and grid-map matching to qualitatively assess the usefulness of our hierarchical expectation representation.

Our results are summarized in Table 1 and compare the grid-map retrievals against symbol expectation matching. The most significant result is shown in row three where due to a grounded encoding of grid-maps its matches were able to search for a visual similarity of form unlike symbolic matching. For row 2, neither found a match since no shape uses {1 × 1, red} blocks in the current HTN plan library.

Our approach lends itself naturally to hybrid execution architectures where reactive learners manipulate raw-data and work in synchrony with deliberative planners which rely on some heuristic or some other form of domain knowledge. While it is easy to think of meta-reasoners as only an additional layer, its strength lies in enabling trading of valuable information across these two layers. It is this strength of the meta-reasoner to form a global view which we believe will be a valuable addition to the long-term autonomy literature in robotics. Specifically, its across-event reasoning can augment the strength of episodic performance exhibited by reactive learners and task-oriented planners.

7 Reasoning across different object poses

The occupancy grid expectations used in the last section are useful for encoding states of the world but run into several problems for generic usage. In a world where an agent is actively interacting with the objects, the agent itself may occlude the sensor which can result in a different expectation which maps to the same underlying state. Further, these relative poses are also entangled with the affordances of the object, i.e., placing two different objects in the same relative pose might not lead to an assembly which was not the case with our previous simpler domain. Finally, while grids work for planar cases, for more complex 3D assembly tasks the quantization can abstract away important low-level state information. In this section, we refine the expectations to be applied to the lower-level state of assembly objects (i.e., position and orientation) and propose a generic backtracking-based repair algorithm over the representation. We use a key point-based object representation since the entire 6-dimensional pose (3-dimensional position and 3-dimensional orientation in free-space) of multiple objects is critical for the success of an assembly task; thus, addressing a more general formulation of the assembly problem.

7.1 Problem setup

An assembly task-and-motion planning domain is defined by a goal-state, i.e., its symbolic and sub-symbolic description. We extend this definition to our modified HTN methods since the original only has a precondition and a network. Continuing the setup in the previous experiment, we assume that the symbolic goal-state is given and the demonstration is used for extracting sub-symbolic states of the objects. As described in Section 7.2.1. Each state's sub-symbolic description includes observed valid poses of all objects and agents implicated in the state-description. This distinction between all valid and observed valid poses is important to note as it distinguishes our work from TAMP. We do not assume all valid poses given, rather build these sets from observations. This is also a weakness as in the current version we do not include known kinematic poses of the agent in the formulation which leads to motion planning failures (discussed in Section 7.5.2). For simplicity of analysis, we only consider two action primitives for this work: *MoveTo* which takes a pose as input and *Grasp* which toggles gripper state.

Building on the HTN specifications, for the purpose of this study we categorize the preconditions into two types, those for defining generic applicability (for example, gripper can grasp an object if the object has a grasp-point and gripper is open) and task-specific (for example, gripper should grasp the toy bricks without blocking the bottom attachment point). In an ideal situation it would be desirable for the domain designers to inject the generic preconditions marking the minimum set of conditions necessary for applying an action to the environment, and for the agent to learn the task-specific constraints based on nominal scenarios, object knowledge and transitions made by assembly skills. Thus, the aim of this study is to learn these task-specific relations over objects and their sub-symbolic groundings, assuming generic preconditions to be given. Please note, we always assume that the nominal plan and traces associated with this nominal plan are already provided to the agent. We focus on the representation of trace, monitoring using expectations based on trace and plan repair.

We model an assembly task as an initial state s_0, a goal state s_g, an attachment (equivalent to a task method) att_g and three main entities: an assembly agent, an active object and a reference object. An active object is the one being manipulated by the assembly agent with respect to the stationary reference object using the action decomposition of att_g leading to s_g. Traces are provided for nominal task settings where poses of objects match underlying assumptions of the plan, the robot is then exposed to a variant setting where the objects are in a different configuration. In the following section, we define the knowledge and representations for capturing task traces. This is followed by a description of how to generate expectations, monitoring over expectations and observations, and the algorithm to repair failures. Finally, we step through our initial experimental result.

7.2 Execution trace: Knowledge and representations

While the task plan considers gripper's poses across the space to ground a plan, the meta-reasoner explicitly collects traces encoding deeper knowledge about how the change in gripper pose is affecting the poses of the object it is operating upon. This bears similarity to how high-level and low-level information is connected in [50, 55]. However, unlike the former we do not assume these relations to be already given rather learn them as part of trace collection and compared to the latter we organize knowledge differently and do not explicitly connect the poses with kinematic constraints of the robot. Thus, for the given attachment att_g an action a_i at ith place resulting in observed state s_0, will have trace-state:

$$T'(i) = \bigcup_{obj_i \in att_g} state(obj_i)$$

Note that this bears similarity to the TAMP equation in Section 4.B. relating sub-symbolic grounding of states to known valid 6DOF states of objects. These traces can now be used to calculate expectations over pose-changes over time for individual objects, as well as for two objects with respect to each other. For the lifted symbols, trace is collected by attaching causal-links to variables which are established by assembly skills by way of computation, perception or motion (see Supplementary Materials).

7.2.1 Object state representation

Each object implicated in a task, i.e., $O(att_g) = \{obj_a, obj_r\}$, is identified by its semantic name which is a string passed as an argument for HTN task methods. Each obj is assigned the following attributes for describing its state:

- Object Pose: $P(obj)$: $|O| \Rightarrow \mathfrak{R}^6$
- Attachment Point: $AP(obj)$: $|O| \Rightarrow \mathfrak{R}^3$
- Attachment Normal: $AA(obj)$: $|O| \Rightarrow \mathfrak{R}^3$, relative to $P(obj)$

Even if implicitly related, all the components are converted to be with respect to the robot's coordinate frame for traces.

Figure 8 shows such a description for the objects in the magnetic robot domain. This representation is based on the preliminary assembly representation in [56] where attachment normals and final pose of objects with respect to world coordinate are specified. Note that here we encode keypoints with respect to the object frame as intrinsic features or affordances of the objects. One can imagine several models, based on this representation, co-existing for a given set of objects. For example, for the toy brick domain, the top and bottom groove locations would count as legitimate attachment-points when creating a 3D structure. On the other hand, for the 2D planar experiment outlined at the beginning of this chapter, we would instead expect a representation where the sides of the bricks count as legitimate attachment-points instead.

7.3 Monitoring for failure

In this work we assume we are only handling objects and pose related failures. Given that the agent only knows about the affordances of each object but not their relative importance to the task or to each other, it is not clear to the agent whether a task is destined for success or failure until the final attachment occurs, especially when the agent has not seen any failures in the past. Thus, we add a verification procedure [29] which is executed by the agent at the end of a task to verify the task was a success or not. If the task is deemed a failure based on verification procedure, then the meta-reasoning process is triggered with traces of past execution, T_{past}, and the current execution trace, T_{curr}. The verification procedure is added as a task method as shown in the code snippet in Supplementary Materials.

In practice though, due to occlusions we detect failure or success after the attachment task by moving the entire sub-assembly to a staging pose. Then we try to detect the assembly on the table, if it is not found then the assembly is verified as

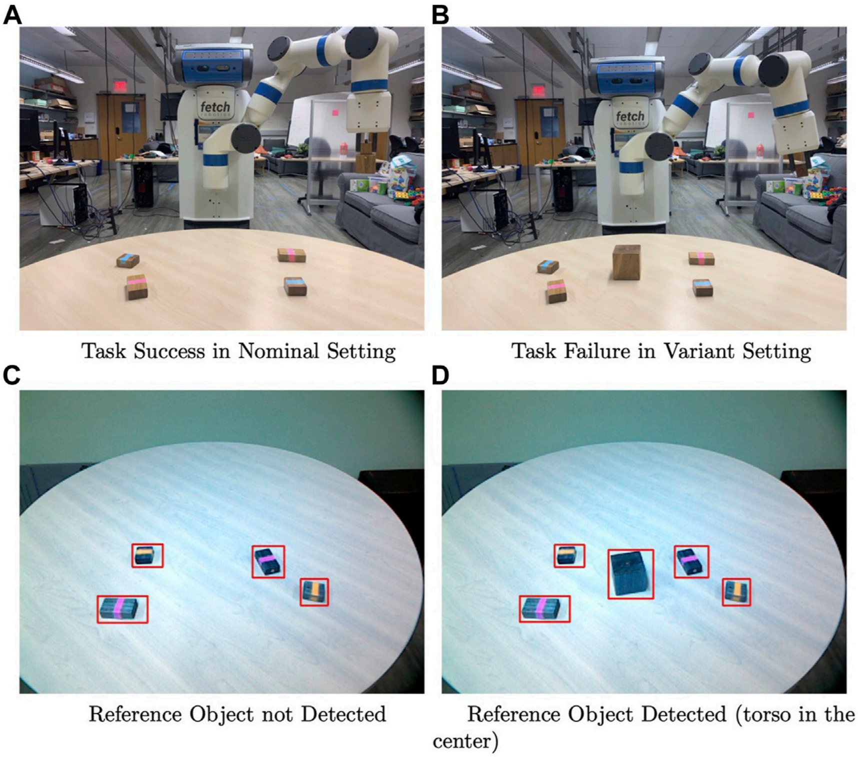

A
B
Task Success in Nominal Setting
Task Failure in Variant Setting
C
D
Reference Object not Detected
Reference Object Detected (torso in the center)

FIGURE 8
An example of a task failure and how visual systems help in monitoring it to verify task completion.

successful. This process is depicted in Figure 9. Note that if obvious or hard failures like mismatched preconditions are observed, then the meta-reasoner can use the algorithms proposed in prior work [32, 33] to repair the domain. In contrast this chapter focuses on non-obvious or soft failures where all the tasks and skills are actuable however do not lead to a successful result.

7.4 Meta-reasoning and repair

If the verification procedure results in a failure, the meta-reasoning cycle is triggered where the meta-reasoner collects past N traces for the same task, generates expectations and invokes a step-by-step comparison with the current trace. We focus on failures at the lowest level, since we have observed this level to be most probable for failures and least explored in related literature. Please note that while the final failure is registered in the form of a logical inconsistency, i.e., task executed but attachment was false, the originating reason for this can be physical or logical.

The generated low-level expectations encode constraints over the relative poses of the two objects or sub-assemblies being attached in the given task. Informally, given aggregate and current poses of assembly objects, it expects:

$$aggrpose(obj_1) : aggrpose(obj_2) :: currpose(obj_1) : currpose(obj_2)$$

Formally, these relations are computed based on aggregation over past traces and the following sets of equations define relations extracted over each trace T_j for task-step i and the expectations averaged over multiple traces.

$$R_p(i, j) = \text{EuclideanDist}(AP(obj_a), AP(obj_r))$$

$$R_n(i, j) = \cos^{-1}(AN(obj_a) \cdot AN(obj_b))$$

$$Exp_{p(i)} = \frac{\sum_{j=1}^{N} R_p(i, j)}{N}$$

$$Exp_{n(i)} = \frac{\sum_{j=1}^{N} R_n(i, j)}{N}$$

If the attachment point and normal in the current trace are significantly different from expected configurations, then the gripper backtracks to the last primitive which assigned its new pose. At this task-step a random good position for the active object's attachment point and corresponding normal is

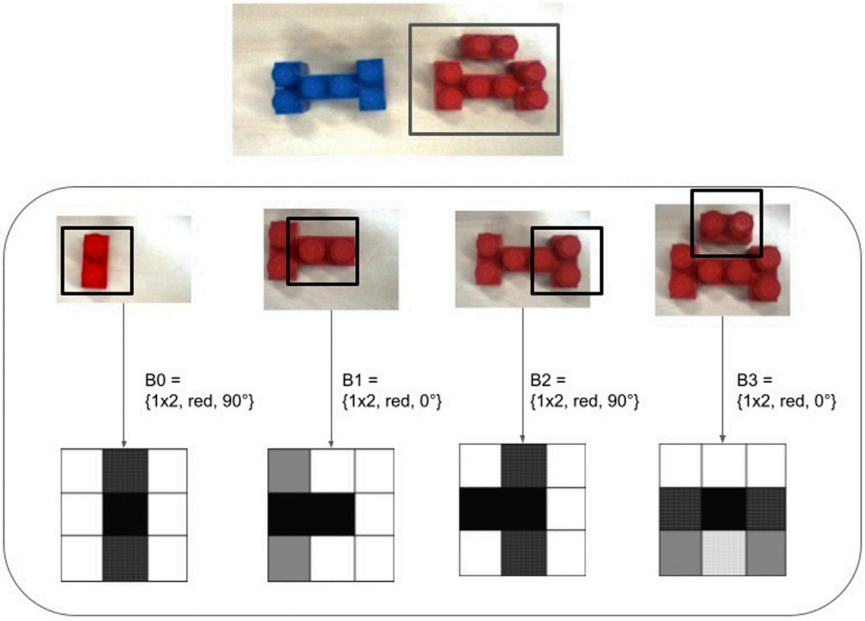


FIGURE 9
Depiction of Hierarchical Expectations, position of centroid removed from symbolic description for brevity. At the top, the darker H shape is blue while the lighter A is red.

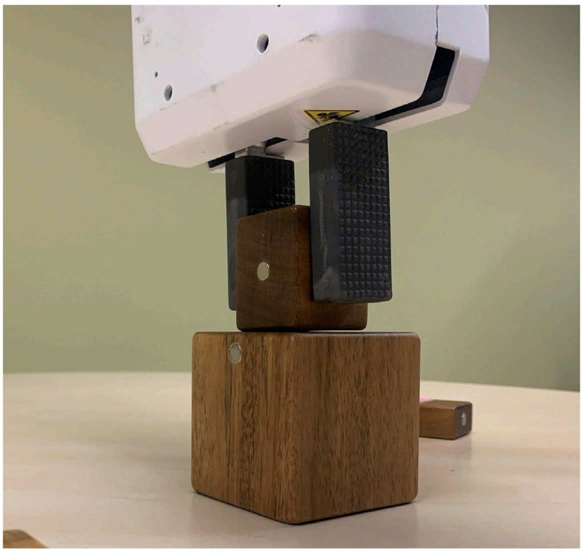


FIGURE 10
Variant setting leading to failure as the attachment points are not aligned and the nominal method did not have explicit actions to align them.

sampled from the traces. We have provided the meta-reasoner with a motion repair module. Given the kinematic relations between the object and the gripper, this is transformed into the related gripper position which is then treated as a repaired argument for the primitive. The repair algorithm keeps backtracking until the last pose which differs from trace if no successful plan is found. The final successful repaired plan, if found, is saved as a new refinement of the task along with starting pose of the objects as a sub-symbolic precondition.

7.5 Experiments and results

7.5.1 Setup

We focus our experiment on the *MagneticAttach* high-level task which is decomposed into a plan as shown in Figure 10. This plan requires grounding for the *MoveTo* primitive poses. These groundings are given by a human instructor *via* a graphical utility aligning the arm with the objects. This grounding is recorded by the task planner using object and gripper poses.

Figure 11 shows the progression of the nominal plan and the changes made to objects. Figure 12 shows the novel variant task which fails given knowledge gap in object grounding. In our results we show how the plan is compared and changed, evaluate transfer over traces which account for different objects as well as configurations, and finally we also outline the complex failures observed during the course of this attempted repair.

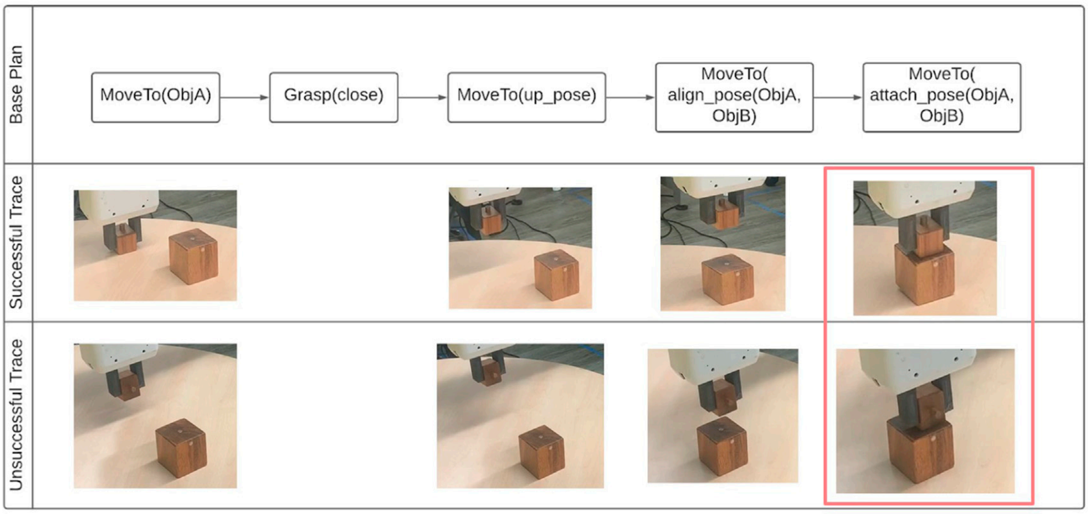


FIGURE 11
A high-level outline of how HTN plans are grounded using experience and compared for repairing object pose input to the execution actions. Meta-reasoner repair process which starts when the top base plan results in failure. The meta-reasoner first compares to see if the current trace was significantly different from nominal traces, if it is then a replacement object pose is sampled from previous traces, the executor backtracks to last action and tries this pose for next action. This sampling and backtracking occurs until a solution is found. Once solution is found, the current trace is rewritten with the new values.

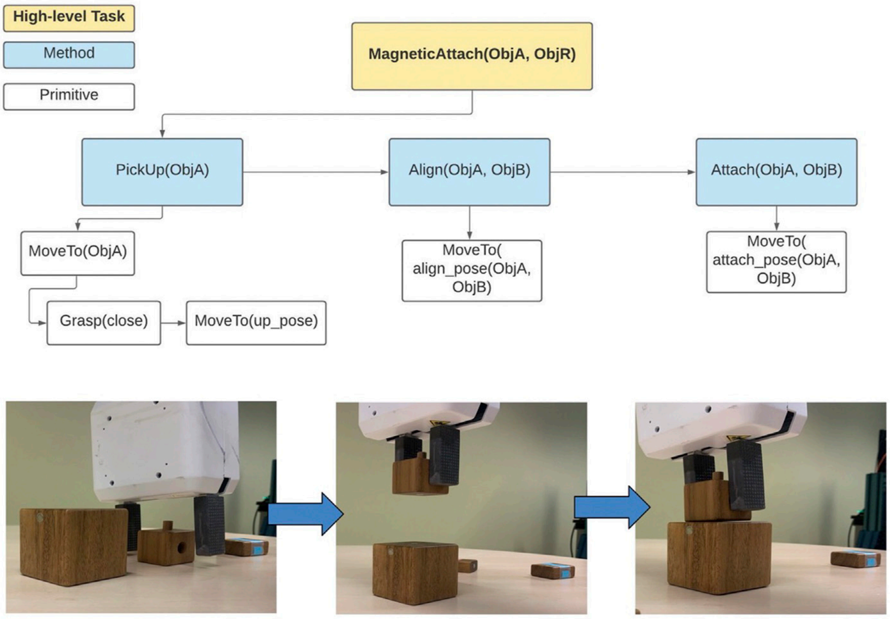
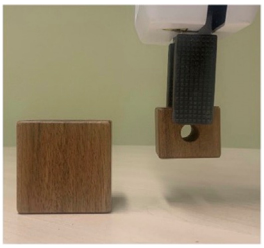
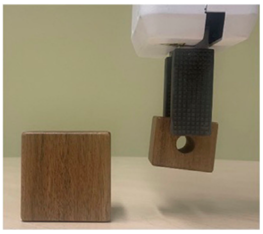
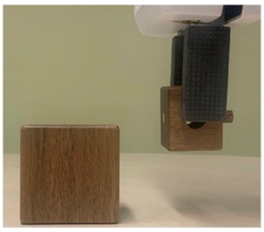


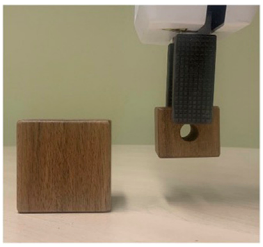
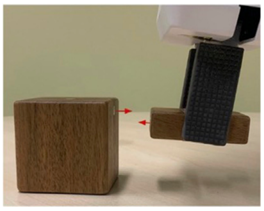
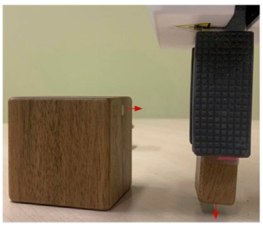
FIGURE 12
Nominal HTN plan decomposing a high-level assembly task (MagneticAttach) to primitives using methods. At the bottom we show the nominal task trace which grounds the high-level plan within the poses of the robot and the objects.

TABLE 2 A comparison of calculated R_p and R_n across different poses of the same object. Column 1 is the nominal plan grounded in instructed poses. Column 2 and 3 are variant poses from the test task. R_p is a good indicator of failure at the end of the task, however R_n is more sensitive to gross variations in object's pose.

			
Rp	0.2276	0.3355	0.3409
	0.2248	0.3286	0.3734
	0.0575	0.0392	0.0720
	0.0023	**0.0058**	**0.0412**
Rn	3.123	3.129	**1.483**
	3.14	3.133	**1.494**
	3.13	3.127	**1.492**
	3.13	3.14	**1.483**

The bold values indicate higher distinction power between task-states.

TABLE 3 A comparison of calculated Rn across different objects. Column 1 is the nominal plan grounded in instructed poses for Object 1. Column 2 and 3 are nominal and variant poses for Object 2 operated on by the same MagneticAttach task. R_n traces are still able to differentiate between wrong and nominal traces, given only the object attachment information about Object 2.

			
Rn	3.123	3.141	**1.571**
	3.14	3.142	**1.569**
	3.13	3.137	**1.568**
	3.13	3.143	**1.568**

Bold values indicate higher distinction power between task-states.

7.5.2 Repair result

The meta-reasoner backtracks one step each time, samples a known good pose from past traces which resulted in a successful task and plans a motion for it. This pose replaces the argument for next action in the plan. We observed as long as the gripper pose associated with the object is reachable from current configuration this repair can find a repair.

However, we observed two key failures which our approach does not model:

1. Collision Failure: In some cases, since we are not modeling the grasp to be task-informed, the gripper can cover the attachment point of the object. In such cases, even if our repair algorithm provides a good pose for the object, the robot cannot plan for it since the gripper would collide with the reference object in this configuration.
2. Reachability Failure: In certain cases, even if the gripper is not covering the attachment point, the robot arm joint configuration required for a good object pose is unreachable from the backtracked pose. This is due to singularity issues in arm motion planning and limited workspace.

7.5.3 Monitoring for same object's variant poses

Table 2 presents the values for R_p and R_n for the object configurations shown in the header. We see that by considering both values for traces, we can establish better similarity between successful tasks.

7.5.4 Monitoring across objects

Different objects may have their attachment normals aligned with different axes of the object's coordinate frame. In Table 3 we present the R_n values comparing nominal trace for one object to nominal and wrong configuration for another object. We observe that by modeling attachment normals separately we can still establish similarity between successful and failing tasks.

8 Conclusion

We started this paper by describing a research gap in traditional industrial robotics' planning around the issue of small-scale, heterogeneous assembly scenarios. The traditional methods only operate at high-level modeling of plans assuming low-level poses of objects are fixed, however such mono-level architectures and systems do not work well when an agent is required to adapt and learn in unstructured environments. We also highlighted a research gap in metareasoning literature with respect to grounding methods of failure monitoring and repair in action and perception that are critical for an embodied agent like an industrial robot.

In this investigation, we outlined lower-level task representations and reasoning processes to include them in the meta-reasoning architecture. These task representations focus on the action and object representation components which are unique to embodied agents. This gave rise to representations of expectations in meta-reasoning which encode relations between physical goal-state and acting processes rather than encoding the meta-relations between the reasoning processes and their arguments, as is typically done in the traditional meta-reasoning literature. This motivated an updated architecture and processes to encode, store and use these representations in an action and object-centric manner.

Our key finding is that extending the metareasoning architecture with the lower-level expectations adds flexibility to otherwise rigid model-based planners. We posit that this also enables a crisper modeling of different knowledge bases and processes involved in a robotics planning and execution process. Using only conceptual expectations does not capture state changes on the ground and our experiments suggest using both conceptual and visual expectations solves more kinds of failures than conceptual expectations alone. We conclude that knowledge transfer using metareasoning makes a robotic system more flexible than one with only classical planning. On the other hand, previous results [42] suggest that learning with metareasoning requires more structured knowledge but less data.

Our overall goal and motivation with this line of work was to enable adaptive agents which can conduct long-term reasoning but also learn from low-level data and thus explore the environment in a meaningful way. We believe that while learning paradigms can bring significant improvement to what we believe a robot is capable of, a learned component is only as good as the quality of data it is based on. We conclude that designing transitional agents can enable a bridge to systematically collect real-world data by specifically mining failure-events.

Data availability statement

The original contributions presented in the study are included in the article/Supplementary Materials, further inquiries can be directed to the corresponding authors.

Author contributions

This research is part of PP research thesis. Primarily advised by HC on the robotics end, and advised by AG on the cognitive system end.

Funding

The work was Funded by HC and Contextual Robotics Institute, UC San Diego—Funding *via* RPDC and Dr. Nahid Sidki for supporting part of this research.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Supplementary material

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fphy.2022.975247/full#supplementary-material

References

1. IFR. *World robotics report*. Frankfurt, Germany: International Federation of Robotics (2020). [Internet]Available from: https://ifr.org/ifr-press-releases/news/record-2.7-million-robots-work-in-factories-around-the-globe.

2. Huckaby JO. *Knowledge transfer in robot manipulation tasks*. Atlanta, GA, United States: Georgia Institute of Technology (2014).

3. Huckaby JO, Christensen HI. A taxonomic framework for task modeling and knowledge transfer in manufacturing robotics. In: *Workshops at the twenty-sixth AAAI conference on artificial intelligence* (2012).

4. Huckaby J, Christensen H. Modeling robot assembly tasks in manufacturing using SysML. In: Proceeding of the ISR/Robotik 2014; 41st International Symposium on Robotics; June 2014; Munich, Germany. IEEE (2014). p. 1–7.

5. Huckaby J, Vassos S, Christensen HI. Planning with a task modeling framework in manufacturing robotics. In: Proceeding of the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems; November 2013; Tokyo, Japan. IEEE (2013). p. 5787–94.

6. Wang R, Guan Y, Song H, Li X, Li X, Shi Z, et al. A formal model-based design method for robotic systems. *IEEE Syst J* (2019) 13(1):1096–107. doi:10.1109/jsyst.2018.2867285

7. IFR. *Advances in programming lower cost of adoption [Internet]*. Frankfurt, Germany: IFR International Federation of Robotics (2021). [cited 2022 Aug 21]. Available from: https://ifr.org/post/advances-in-programming-lower-cost-of-adoption.

8. Devin C, Abbeel P, Darrell T, Levine S. Deep object-centric representations for generalizable robot learning. In: Proceeding of the 2018 IEEE International Conference on Robotics and Automation (ICRA); September 2018. IEEE (2018). p. 7111–8.

9. Ghalamzan EAM, Paxton C, Hager GD, Bascetta L. An incremental approach to learning generalizable robot tasks from human demonstration. In: Proceeding of the 2015 IEEE International Conference on Robotics and Automation (ICRA); May 2015; Seattle, WA, USA. IEEE (2015). p. 5616–21.

10. Koert D, Maeda G, Lioutikov R, Neumann G, Peters J. Demonstration based trajectory optimization for generalizable robot motions. In: Proceeding of the 2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids); November 2016; Cancun, Mexico. IEEE (2016). p. 515–22.

11. Fitzgerald T, Goel AK, Thomaz AL. Human-guided object mapping for task transfer. *ACM Trans Hum Robot Interact* (2018) 7(2):1–24. doi:10.1145/3277905

12. Fitzgerald T, Goel A, Thomaz A. Abstraction in data-sparse task transfer. *Artif Intelligence* (2021) 300:103551. doi:10.1016/j.artint.2021.103551

13. Wells AM, Dantam NT, Shrivastava A, Kavraki LE. Learning feasibility for task and motion planning in tabletop environments. *IEEE Robot Autom Lett* (2019) 4(2):1255–62. doi:10.1109/lra.2019.2894861

14. Dosovitskiy A, Beyer L, Kolesnikov A, Weissenborn D, Zhai X, Unterthiner T, et al. *An image is worth 16x16 words: Transformers for image recognition at scale* (2020). arXiv preprint arXiv:201011929.

15. Park D, Hoshi Y, Kemp CC. A multimodal anomaly detector for robot-assisted feeding using an LSTM-based variational autoencoder. *IEEE Robot Autom Lett* (2018) 3(3):1544–51. doi:10.1109/lra.2018.2801475

16. Nair S, Rajeswaran A, Kumar V, Finn C, Gupta A. *R3M: A universal visual representation for robot manipulation [internet]* (2022). arXiv; 2022 [cited 2022 Aug 21]. Available from: http://arxiv.org/abs/2203.12601.

17. Simeonov A, Du Y, Tagliasacchi A, Tenenbaum JB, Rodriguez A, Agrawal P, et al. *Neural descriptor fields: SE(3)-Equivariant object representations for manipulation* (2021). [cited 2022 May 10]; Available from: https://arxiv.org/abs/2112.05124v1.

18. Murali A, Liu W, Marino K, Chernova S, Gupta A. Same object, different grasps: Data and semantic knowledge for task-oriented grasping. In: *Conference on robot learning* (2020).

19. Manuelli L, Gao W, Florence P, Tedrake R. Kpam: KeyPoint Affordances for category-level robotic manipulation. In: T Asfour, E Yoshida, J Park, H Christensen, O Khatib, editors. *Robotics research*. Cham: Springer International Publishing (2022). p. 132–57. (Springer Proceedings in Advanced Robotics).

20. Cox MT. Perpetual self-aware cognitive agents. *AIMag* (2007) 28(1):32.

21. Cox MT. A model of planning, action and interpretation with goal reasoning. *Adv Cogn Syst* (2016) 5:57–76.

22. Cox MT, Alavi Z, Dannenhauer D, Eyorokon V, Muñoz-Avila H, Perlis D. Midca: A metacognitive, integrated dual-cycle architecture for self-regulated autonomy. *Proc AAAI Conf Artif Intelligence* (2016) 30:3712–8. doi:10.1609/aaai.v30i1.9886

23. Dannenhauer D, Muñoz-Avila H. Raising expectations in GDA agents acting in dynamic environments. In: Proceedings of the 24th International Conference on Artificial Intelligence; July 2015 (2015). p. 2241–7.

24. Dannenhauer D, Muñoz-Avila H, Cox MT. Informed expectations to guide GDA agents in partially observable environments. In: Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence; July 2016 (2016). p. 2493–9.

25. Murdock JW, Goel AK. Meta-case-based reasoning: Self-improvement through self-understanding. *J Exp Theor Artif Intelligence* (2008) 20(1):1–36. doi:10.1080/09528130701472416

26. Murdock JW, Goel AK. *Self-improvement through self-understanding: Model-based reflection for agent self-adaptation*. Amazon (2011).

27. Stroulia E, Goel AK. Functional representation and reasoning for reflective systems. *Appl Artif Intelligence* (1995) 9(1):101–24. doi:10.1080/08839519508945470

28. Stroulia E, Goel AK. Evaluating PSMs in evolutionary design: The A UTOGNOSTIC experiments. *Int J human-computer Stud* (1999) 51(4):825–47. doi:10.1006/ijhc.1999.0331

29. Jones JK, Goel AK. Perceptually grounded self-diagnosis and self-repair of domain knowledge. *Knowledge-Based Syst* (2012) 27:281–301. doi:10.1016/j.knosys.2011.09.012

30. Goel AK, Rugaber S. Gaia: A CAD-like environment for designing game-playing agents. *IEEE Intell Syst* (2017) 32(3):60–7. doi:10.1109/mis.2017.44

31. Conitzer V, Sandholm T. *Definition and complexity of some basic metareasoning problems* (2003). arXiv:cs/0307017 [Internet]. [cited 2021 Feb 9]; Available from: http://arxiv.org/abs/cs/0307017.

32. Parashar P, Goel AK, Sheneman B, Christensen HI. Towards life-long adaptive agents: Using metareasoning for combining knowledge-based planning with situated learning. *Knowledge Eng Rev* (2018) 33:e24. doi:10.1017/s0269888918000279

33. Parashar P, Naik A, Hu J, Christensen HI. A hierarchical model to enable plan reuse and repair in assembly domains. In: Proceeding of the 2021 IEEE 17th International Conference on Automation Science and Engineering (CASE); August 2021; Lyon, France. IEEE (2021). p. 387–94.

34. Christensen HI, Kruijff GJM, Wyatt JL *Cognitive systems*, Vol. 8. Springer Science & Business Media (2010).

35. Kortenkamp D, Simmons R, Brugali D. Robotic systems architectures and programming. In: *Springer handbook of robotics*. Springer (2016). p. 283–306.

36. Quigley M, Conley K, Gerkey B, Faust J, Foote T, Leibs J, et al. Ros: An open-source robot operating system. In: *ICRA workshop on open source software*, 3 (2009).

37. ROS-Industrial. *ROS-Industrial goals and background*.

38. Parashar P. *Using meta-reasoning for failure detection and recovery in assembly domain [internet]*. San Diego: University of California (2021). [cited 2022 May 11]. Available from: https://www.proquest.com/openview/9e3639b5696fdd09488ed817b5786710/1?pq-origsite=gscholar&cbl=18750&diss=y.

39. Beetz M, Mösenlechner L, Tenorth M. CRAM—a cognitive robot abstract machine for everyday manipulation in human environments. In: Proceeding of the 2010 IEEE/RSJ international conference on intelligent robots and systems; October 2010; Taipei, Taiwan. IEEE (2010). p. 1012–7.

40. Muñoz-Avila H, Cox MT. Case-based plan adaptation: An analysis and review. *IEEE Intell Syst* (2008) 23(4):75–81. doi:10.1109/mis.2008.59

41. Cox M, Raja A. *Metareasoning: A manifesto*. Cambridge, MA, United States: BBN Technical (2007).

42. Parashar P, Sheneman B, Goel AK. Adaptive agents in minecraft: A hybrid paradigm for combining domain knowledge with reinforcement learning. In: *International conference on autonomous agents and multiagent systems* (2017). p. 86–100.

43. Ulam P, Goel AK, Jones J, Murdock W. Using model-based reflection to guide reinforcement learning. *Reasoning, Representation, Learn Comp Games* (2005) 107.

44. Erol K, Hendler J, Nau DS. *HTN planning: Complexity and expressivity*. Seattle, WA, United States: AAAI (1994). p. 1123–8.

45. Nau DS, Cao Y, Lotem A, Munoz-Avila H. SHOP: Simple hierarchical ordered planner. In: Proceedings of the 16th international joint conference on artificial intelligence - volume 2; July 1999. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc. (1999). p. 968–73. (IJCAI'99).

46. Nau DS, Au TC, Ilghami O, Kuter U, Murdock JW, Wu D, et al. SHOP2: An HTN planning system. *J Artif Intell Res* (2003) 20:379–404. doi:10.1613/jair.1141

47. Nau D, Au TC, Ilghami O, Kuter U, Wu D, Yaman F, et al. Applications of SHOP and SHOP2. *IEEE Intell Syst* (2005) 20(2):34–41. doi:10.1109/mis.2005.20

48. Georgievski I, Aiello M. *An overview of hierarchical task network planning* (2014). Mar 28 [cited 2020 May 16]; Available from: https://arxiv.org/abs/1403.7426v1.

49. Erol K, Hendler JA, Nau DS. *Semantics for hierarchical task-network planning*. College Park, MD, United States: Maryland Univ College Park Inst for Systems Research (1995).

50. Garrett CR, Chitnis R, Holladay R, Kim B, Silver T, Kaelbling LP, et al. Integrated task and motion planning. *Annu Rev Control Robot Auton Syst* (2021) 4(1):265–93. doi:10.1146/annurev-control-091420-084139

51. Kaelbling LP, Lozano-Pérez T. Hierarchical task and motion planning in the now. In: Proceeding of the 2011 IEEE international conference on robotics and automation; May 2011; Shanghai, China. IEEE (2011). p. 1470–7.

52. Jindal P, Roth D. Learning from negative examples in set-expansion. In: Proceeding of the 2011 IEEE 11th International Conference on Data Mining; December 2011; Vancouver, BC, Canada. IEEE (2011). p. 1110–5.

53. Sarmento L, Jijkuon V, de Rijke M, Oliveira E. "More like these": Growing entity classes from seeds. In: Proceedings of the sixteenth ACM conference on Conference on information and knowledge management - CIKM '07; January 2007; Lisbon, Portugal. New York, NY, United States: ACM Press (2007). [cited 2022 Aug 21]. p. 959. Available from: http://portal.acm.org/citation.cfm?doid=1321440.1321585.

54. Zhang X, Chen Y, Chen J, Du X, Wang K, Wen JR. Entity set expansion via knowledge graphs. In: Proceedings of the 40th international ACM SIGIR conference on research and development in information retrieval; August 2017; New York, NY, United States: Association for Computing Machinery (2017). p. 1101–4. (SIGIR '17). Available from. doi:10.1145/3077136.3080732

55. Dornhege C, Eyerich P, Keller T, Brenner M, Nebel B. Integrating task and motion planning using semantic attachments. In: Proceedings of the 1st AAAI Conference on Bridging the Gap Between Task and Motion Planning. Washington, DC, United States: AAAI Press (2010). p. 10–7. (AAAIWS'10-01).

56. De Mello LH, Sanderson AC. A correct and complete algorithm for the generation of mechanical assembly sequences. *IEEE Int Conf robotics automation* (1989) 7:56–7.

frontiers | Frontiers in Physics

TYPE Review
PUBLISHED 04 January 2023
DOI 10.3389/fphy.2022.1080132

Foundational concepts in person-machine teaming

Ariel M. Greenberg[1]* and Julie L. Marble[2]

[1]Intelligent Systems Center, Johns Hopkins University Applied Physics Laboratory, Laurel, MD, United States, [2]Institute for Experiential Robotics, Northeastern University, Boston, MA, United States

OPEN ACCESS

EDITED BY
William Frere Lawless,
Paine College, United States

REVIEWED BY
Monique Beaudoin,
University of Maryland, College Park, United States
Judith Dijk,
Netherlands Organisation for Applied Scientific Research, Netherlands

*CORRESPONDENCE
Ariel M. Greenberg,
✉ ariel.greenberg@jhuapl.edu

SPECIALTY SECTION
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

RECEIVED 25 October 2022
ACCEPTED 02 December 2022
PUBLISHED 04 January 2023

CITATION
Greenberg AM and Marble JL (2023), Foundational concepts in person-machine teaming.
Front. Phys. 10:1080132.
doi: 10.3389/fphy.2022.1080132

As we enter an age where the behavior and capabilities of artificial intelligence and autonomous system technologies become ever more sophisticated, cooperation, collaboration, and teaming between people and these machines is rising to the forefront of critical research areas. People engage socially with almost everything with which they interact. However, unlike animals, machines do not share the experiential aspects of sociality. Experiential robotics identifies the need to develop machines that not only learn from their own experience, but can learn from the experience of people in interactions, wherein these experiences are primarily social. In this paper, we argue, therefore, for the need to place experiential considerations in interaction, cooperation, and teaming as the basis of the design and engineering of person-machine teams. We first explore the importance of semantics in driving engineering approaches to robot development. Then, we examine differences in the usage of relevant terms like trust and ethics between engineering and social science approaches to lay out implications for the development of autonomous, experiential systems.

KEYWORDS

autonomous systems, trust in autonomy, AI ethics, human-machine teaming, semantic mapping

1 Introduction

For much of its history, teaming research has focused on teams of people. Yet, as artificial intelligence and autonomous system technologies become more advanced, we enter an age in which it is necessary to consider how people will team, cooperate, and collaborate with intelligent machines, and *vice versa*. As research on person-machine teaming begins to take shape, the prevailing assumption has been that the social interactions occurring within interpersonal teams (and/or teams including non-human animals) can serve as a useful basis for understanding the interactions between persons and machines. However, we argue that that there are essential differences between persons and machines that require special consideration when discussing person-machine teaming.

In particular, there are foundational concepts in interpersonal (person-person) teaming that require translation and adaptation when applied to person-machine teams. These concepts include *Autonomy* (compared to *Automation*), *Trust* (compared to *Reliability*), *Ethics* (compared to *Governance*), and *Teaming* (compared to *Use of Automation*) and other, similar terms, which do not directly port from teams comprised of people to those including machines.

While of superficially minor distinction, the interpersonal concepts to which these terms refer inform how we interact with, interpret, and evaluate behavior, regardless of whether their extension to machines is performed casually or deliberately. In teaming, essential notions underpinning these concepts, such as *control* and *vulnerability* (compared to *risk* or *uncertainty*) tend to become mischaracterized or fall out of consideration in the course of translation. In the following pages, we will scrutinize each concept across the interpersonal and machine contexts and identify features that warrant additional consideration in translation.

To accomplish this, we will review conceptual issues that have arisen in the translation of the above italicized terms from the interpersonal context to that between people and machines, and the cross-disciplinary roots of the divergent uses for each term. We begin by summarizing an assembly of computational linguistic techniques devised to shed light on the state of discourse around concepts for application to teaming with machines that we refer to collectively as the *Semantic Mapping Pipeline (SMP).* Then, we provide a series of qualitative discussions about topics ready to be run through this quantitative method, beginning with *Teaming and Sociality*, continuing with underpinning notion of *Vulnerability,* and then covering the concepts of *Autonomy*, *Trust*, *Ethics*, and *Teaming.* We conclude with a discussion that summarizes the major arguments presented.

2 Personhood and relationships over speciesism

While it is common for researchers to use the term "human-machine teaming" and to speak of "the human" doing this or that with "the machine," a core tenet of our analysis is that the species of intelligent animal is not the primary feature that distinguishes people from machines. Instead, a more salient difference between the two classes of teammates, in the spirit of Locke, Singer, and Strawson, is that humans are persons, who are able to reciprocally recognize the personhood of other humans, whereas machines are not (yet, if they ever could be) persons, and are not (yet) able to recognize personhood, even if they can discriminate humans from other species [1]. In making this distinction, we seek to highlight the fact that there is something special about those with personhood status (and people, in particular) that makes their dyadic behavior fundamentally distinct from that of their machine counterparts. Indeed, we believe that the direct comparison of living species with technology is a false equivalence. First, it implies that the two classes of teammates may be treated similarly—that the person is just another cog in the system with an input and output interface, ripe for replacement by machines. Second, the use of the term "the human" distances and objectifies the person with clinical detachment, especially when juxtaposed with "the machine," so reducing people to automatons. Thus, the use of the term "the human" gives the impression that person-person interactions (and the language that is used to describe them) are directly analogous to person-machine interactions, when it is eminently clear that much of what imbues these actions with their significance stems from mental capacities that no machines currently possess. The mutual relationship of personhood and person recognition in the interpersonal sphere is perhaps the absent core that prevents direct translation to person-machine teaming context (more on this in upcoming paper on relationships by Hutler and Greenberg, forthcoming). For the rest of this paper, in deference to the taxonomic superiority of *person* over *human* (a *human* is a type of *person,* and all humans are people), we will try to correct this terminology by referring strictly to *persons* or *human beings* where *humans* might typically be used, in particular to recast human-machine teaming as person-machine teaming or PMT.

3 Semantics matter

When a concept is ported from one domain to another, a typical first step in the engineering design process is for practitioners to compress the concept into an operational definition toward which they can build. This reduction in practice is ordinarily very effective [2]. However, when designing for PMT in particular, salient features of the concept to be replicated (e.g., sensitivity to vulnerability, recognition of personhood) are frequently lost in translation, while skeuomorphic features (i.e., machine features that superficially emulate interpersonal capabilities, but are substantively dissimilar under the hood, e.g., voice production or eye-contact) are unintentionally retained, or picked up in translation, leading to inappropriate expectations of machine capability. In contrast to these issues in the translation of interpersonal capabilities, translation of physical capabilities (e.g., walking or grasping) is relatively mechanistic and straightforward.

There are perils in this lossy compression (The metaphor of *lossy compression* is used here to indicate that the concepts coming out of this processes are smaller, but also lower resolution). If we only use these concepts in their most superficial form, we miss out on the richness to be had in the phenomena they signify. If we use them in their full interpersonal sense, we misrepresent the capabilities of the machine and set inappropriate expectations for their performance (see wishful mnemonics [3]). If we use the terms in an ambiguous sense between the most superficial and the full interpersonal, then the capabilities realized in different machines are bound to be inconsistent across implementations. This last case is the most prevalent, and with each occurrence, the conceptual drift continues. By allowing this inconsistent usage to prevail, we may ultimately lose our grip on the original meaning, our appreciation of the fullness of the phenomenon may diminish, and the reduced definition may become prone to be cast over the entire phenomenon, interpersonal and otherwise. As Sherry Turkle [4] puts it: "When we see children and the elderly exchanging tenderness with robotic pets, the most important question is … *what will "loving" come to mean?*" [emphasis added].

This disconnect in language becomes especially apparent in design meetings and program reviews, wherein operational definitions are only found to be incongruent with empirical capabilities after the fact. Worse is when that incongruence remains unrecognized—the same words are used by different designers with significantly different meanings. This disconnect naturally arises from the different backgrounds of those using the terms. Robotics is inherently a multidisciplinary area of research, and different disciplines understand and use the same terms very differently, including how to measure them in context. Of course, harmonizing terms is a perennial challenge in multidisciplinary work, but is particular acute for social robotics since interpersonal terms have previously been used in the context of technology more metaphorically than anthropomorphically, or simply for purpose of usability. An engineer using the term "trust" may construe the term with respect to things that can be engineered, while a social scientist might construe the term with respect to social constructs. We argue that adequate definitions are those which may be operationalized sufficiently for design and can be measured accurately, reliably, and repeatably, while respecting the richness of the phenomenon in the context of its relevant interpersonal constructs.

There are a number of interventions possible to address this disconnect. The most extreme is to declare that interpersonal terms may not be used in the context of person-machine teaming. This prescriptive approach to controlling language rarely succeeds, and will only be effective at alienating incoming generations of researchers. Another intervention is to create new terms or to add a qualifying adjective to existing ones to make these terms specific to PMT [e.g., adding *semi*-to qualify *Semi*-autonomous [5], adding *robot and artificial intelligence* to qualify *RAI*-responsibility (upcoming publication by Greenberg et al., Robots that "Do No Harm")]. The most gentle intervention is to do what we have set out to do here: Identify what is lost and found in translation between contexts.

3.1 Semantic mapping pipeline (SMP)

As part of an earlier effort, Greenberg led a small team to review how terms central to person-machine teaming were being used across the literature. This preliminary investigation sought to develop the methodology and begin a cursory exploration, and it is presented here to introduce a mixed-method approach to semantic conceptual analysis. In the sections that follow, we discuss the concepts of *Autonomy*, *Trust*, *Ethics*, and *Teaming* in primarily qualitative terms. However, we believe that these same topics are ripe to be run through the SMP method for quantitative support.

This review, formulated as a semantic map of terms, was intended to address questions such as:

- How do various disciplines use PMT terms, both within their discipline, and when communicating to their interdisciplinary counterparts?
- What are the differences and similarities in the ways the various disciplines use the terms?
- How much true interdisciplinary treatment is there, or is treatment mostly disparate multidisciplinary contributions?
- In which semantic clusters does a particular organization find their conceptualization to fit best (the inverse problem of semantic map assembly, whereby particular articles invoking the terms are placed within the map generated from the corpus).

To answer these questions, the concepts under consideration are first cataloged and discussed. Once a suitable list of keywords has been identified, they are then run through the semantic mapping pipeline to display their prevalence, authorial provenance, and co-occurrence in current person-machine teaming scholarship, against a background of those terms' usage in interpersonal contexts and in common parlance. The methodology of the semantic mapping pipeline is as follows:

First we populate a corpus; the body of papers that include the terms of interest, semantically similar terms, and their related word forms across various parts of speech, retrieved from scholarly clearinghouse sources like Web of Science and arXiv, and from policy statements of international organizations. We then review the bibliographies of these papers to augment the corpus with secondary papers that are related but may not have used the search terms precisely as we had specified them. Given that the contents of this corpus is the source material from which the pipeline produces it analyses, we take care to be comprehensive at the start. Late additions are possible to be accepted, at which point those new entries are reprocessed as described in the next steps, for an updated output.

Next, we use the systematic review software Covidence to screen the papers for relevance by the PRIMSA protocol, and tag them with an interpretation of how the paper authors are using the search term, from a standardized list of meanings set *a priori* from a preliminary scan. Should new meanings be discovered in-process, they are added to this list for tagging. Those papers emerging from the screen are parsed, along with their metadata.

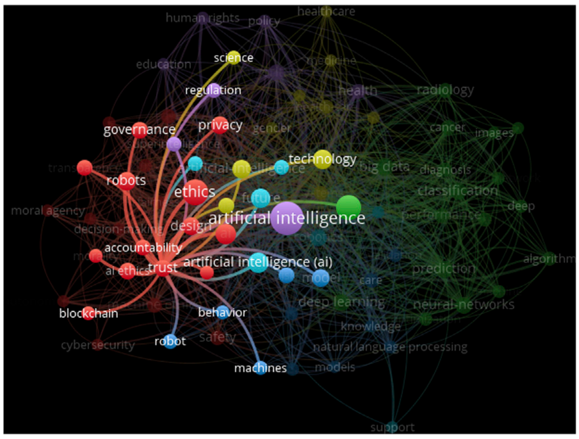


FIGURE 1
Term co-occurrence.

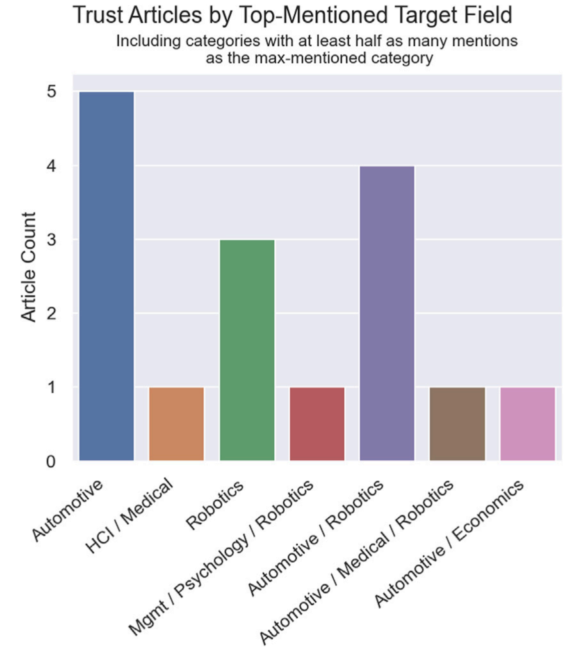


FIGURE 2
Articles by target.

With the corpus now formed into a computational object, we can visualize and analyze *in silico*. For preliminary visualization, nodes of terms are linked by edges of co-occurrence, sized by number of citations, and colored by community detection that reflects disciplinary field. A semantic graph composed of these nodes is assembled and visualized by VOS Viewer (Figure 1), or programmatically by the python Network X package.

Next, we perform various analyses to update the semantic map. Bibliometric analyses include undirected graphs of cocitations and directed graphs describing the discourse between contributing disciplines, authorial provenance, and target audience (Figure 2). Throughout, the method invokes *synsets* (WordNet's grouping of synonymous words that express the same concept) to improve flexibility across semantically similar terms. Semantic analyses encompass the usage patterns of terms found across several parts of speech: nouns, adjectives, and prepositions.

The nominal [of nouns] analysis queries co-occurrence graphs and compares term frequency distributions (Figure 3) to discover if PMT discussions around a term are addressing the same concept—and if so, how those discussions are distinct by discipline and/or from usage in interpersonal literature. The adjectival analysis queries the lexical dispersion (relative locations of terms within the text and their distances from one another, Figure 4) of preidentified terms and of terms with wordfinal morphemes of adjective suffixes, to collect how terms are described, and whether descriptions are used consistently throughout each document. The prepositional analysis uses phrase chunking to discover to what, to whom, or for what the term pertains.

By filtering the visualization of the semantic map that serves as frontend to the combination of computational linguistics techniques here described, researchers may examine term semantics at scale. We invite those interested to adopt this mixed-method approach and continue the work where we left off, with code available upon request.

4 PMT concept 1: Teaming is inherently social

Unlike almost any other engineering product, autonomous systems interact with people through social channels to achieve their goals. Meanwhile, people's responses even to non-agentic computers are inherently social (e.g. [6]), and with just a bit more interactivity, they become what Sherry Turkle [4] calls relational artifacts: "Their ability to inspire relationship is not based on their intelligence or consciousness but on their ability to push certain Darwinian buttons in people (making eye contact, for example) that make people respond as though they were in a relationship."

The question of whether a machine can truly team with people (or even other non-human agents) is a source of significant debate, and the term "team" is frequently misused or misapplied, especially with respect to person-machine teams. Research engineers often apply the term "human-machine team" to any collection of people and robots or autonomous agents, regardless of whether they meet the criteria that define a team, such as the need for interdependence between members or common identity as a team [7–9].

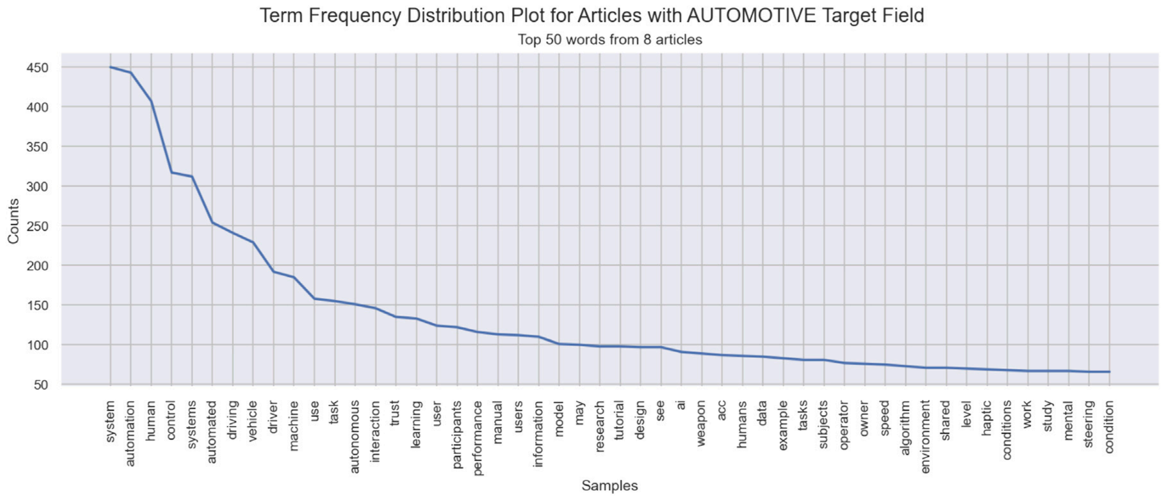


FIGURE 3
Term frequency distribution.

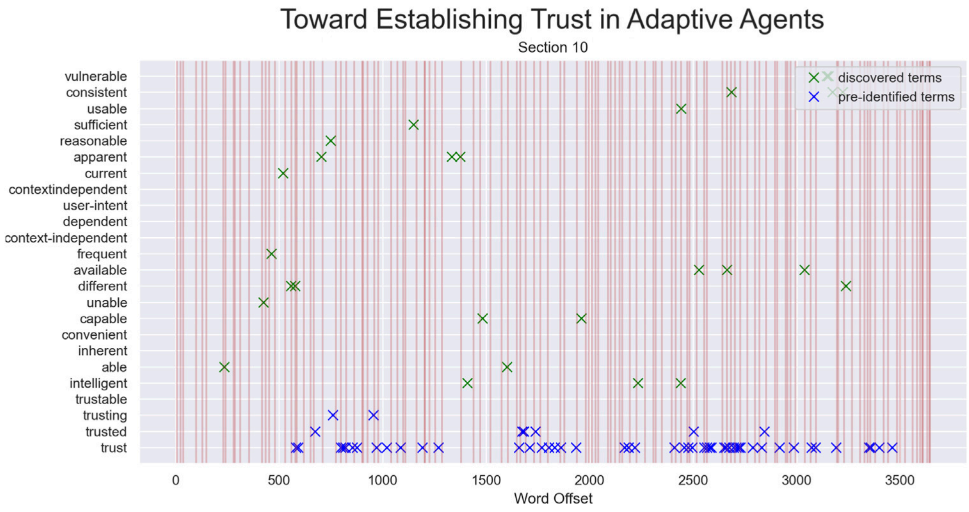


FIGURE 4
Lexical Dispersion, with sentence breaks indicated by red vertical lines. The legend is on top of hits for the first two discovered terms ("vulnerable" and "consistent") between word offsets 3000 and 3500.

A team is a set of two or more people who interact dynamically, interdependently, and adaptively, toward a common and valued goal, each member having specific roles or functions to perform, and a limited life-span of membership [10]. Teams, therefore, are inherently social groups with interdependence between team members who are working toward common goals [10–12]. Team members behave differently from other organizational structures (e.g., supervisory hierarchies) in several ways. They demonstrate increased communication between members, greater effort and commitment to the goals, greater trust between members [13] and show greater adaptability and innovation from these other structures [14]. Kozlowski and Ilgen [14] also emphasize the social aspects of teaming—motivation, affect and interpersonal interaction.

As a further example, Walliser et al. [11] explored how team structure and the manner in which people are directed to work with a teammate impact team performance with autonomous agents. In this study, participants worked with a human collaborator or an autonomous system, either as a collaborative teammate or to direct its performance as they would a tool. As would be expected, collaboration was more common in the teaming condition than in the tool condition. For example, there were significantly more task-relevant chat messages sent by participants in the team condition. Task-relevant chat messages were equally common for both the human and autonomous agents. In contrast, messages related to performance, information, and acknowledgment were only sent when the other agent was human. The authors argue that these results indicate that the interaction between people and autonomous agents is fundamentally social; given that effective teamwork relies heavily on social interactions, these aspects of interaction must be included in the development of autonomous agents. They point out that the social aspects of person–machine team design are neglected in favor of enhancing the more traditional computational and electromechanical capabilities of the autonomous agent. We explore that focus in the next section where we examine guidance given on the design and development of autonomous systems.

The debate regarding whether autonomous machines may be considered teammates over tools centers on the development and demonstration of shared common goals or shared mental models, interdependence of actions, and inter-agent trust [15]. Relatively recent advancements have begun to demonstrate the ability for machines to share goals and adapt to changing context (see for example [16]). Further, people appear to team as easily with robots as with humans [17, 18]. Taken together, these findings suggest that research that neglects the experiential, social, and cognitive-affective aspects of person-machine interaction will not yield successful teaming; in which case machines will remain in the role of tools and the full capabilities of effective person-machine teams will not be realized.

One way to approach these neglected aspects of PMT is to attend to the latent construct of vulnerability. The constituent concepts we will review in the next sections, on Autonomy, Trust, and Ethics, all share this latent construct, which tends to be the first to fall out when translating these terms from their interpersonal sense to their person-machine teaming sense. *Vulnerability*, the state in which a person is subject to harm (physical, psychological, financial, etc.) remains the condition for a teammate whether that person is relying on another person, or on a machine.

5 PMT concept 2: Vulnerability is ultimately unmitigable

In PMT contexts, the notions of autonomy, ethics, and trust are inextricably linked not just to mission and task risk (cognitive trust [19]) but to personal vulnerability (emotional trust, [19]. To demonstrate this for yourself, try the following exercise—replace the terms *autonomy*, *ethics*, and *trust* with a conjugate of *vulnerability*, and determine whether the statement still holds[1]. However, while this connection is apparent in every definition of interpersonal trust (see [20, 21]), the notion of vulnerability is frequently operationalized as relatively less-rich concepts such as uncertainty or risk when translated to pertain to persons cooperating with machines. This may be because *vulnerability* is perceived as more affect-laden and nebulous, while *uncertainty* or *risk* can be defined in probabilistic terms, which is more compatible with an engineering orientation. However, the notion of vulnerability is not encompassed by uncertainty or risk alone, and creating an operational definition that exchanges these concepts loses essence (now try that term replacement exercise again, with *uncertainty* or *risk* swapped for *vulnerability*). The stakes are not simply outcome- or likelihood- oriented pertaining to risk, but indeed personal—a machine teammate's failure has personal consequences for its human teammates.

These consequences may arise not just from failure to complete the task (as discussed in Section 8 on trust), but from performing the task in unexpected or incompatible ways, or from performing the task in an expected manner that yields undesired results. Among other things, human teammates may grow disappointed, insecure, or worried, and that negative affect is itself a harm, not captured by the concept of risk (though approximated by *vigilance*). While this may not appear to be a consequential effect, keep in mind how crucial a lever negative affect is for humans teaming with non-human animals: dogs in particular are exquisitely sensitive and responsive to our disposition to them [22].

Of course, the typically negative affect associated with the experience of vulnerability is not felt by machines, so there is an intrinsic limit to how faithfully a machine can participate in the downstream concepts of PMT Autonomy, Trust, Ethics, and Teaming. As put by Marisa Tschopp [23]: "The victims are always the humans." Even just an imbalance of vulnerability between partners is generally enough to undermine trust [24]. Autonomous systems are indifferent about survival; are without social or emotional values to protect; are unconcerned with stakes and unaffected by reward (despite it being sought computationally through reward and objective functions in machine learning) and undeterred by punishment. Autonomous systems have nothing to lose, and nothing to gain, so the act of judgement must be privileged to those who are innately vulnerable (people), who also have a sense of responsibility and who are affected by the potential disappointment of those subject to the judgement.

[1] For example, does "I *trust* the machine to fold my laundry" mean the same as "I *am willing to be vulnerable to a poor outcome* should the machine not succeed," or simply that "I *believe* the machine will be successful?"

Further, machines do not have the visceral appreciation for human vulnerabilities that people do. As a result, people have no basis for confidence that machine teammates will understand the shape of the utility functions of people to select a behavior that is congruent with their interests. This creates inter-dyadic risk that is independent of the operational context (or, at the very least, omnipresent across all contexts), and dramatically lowers the likelihood that people will be willing to trust the machine. It is not just that machines do not share the same vulnerabilities, it is that because they cannot feel vulnerable, we don't expect them to share or understand our values.

To address this vulnerability gap, Greenberg has worked toward the development of a harms ontology, described further in the section on *Ethics*. In this installment of research into artificial non-maleficence, he and his team explicitly trace potential physical harms to humans through their vulnerabilities (in this one case, the biology of the species of intelligent animal is the salient feature vs. their personhood that is primary for the other ethical principles and types of non-physical harms). From an ethical standpoint, each actor should seek to recognize and respect the vulnerability of other actors, to minimize harms that prey upon that vulnerability. In fact, the ability to recognize vulnerability may be a criterion for personhood (Strawson Microsoft Word - Document3 (brandeis.edu)).

In both interpersonal and PMT contexts, *Control* is the primary means to mitigate vulnerability to another actor's behavior or to situation outcomes. It is also something engineers are adept at building the means to achieve (e.g., control theory, control surfaces, controllers, etc.). However, increased control by people of machine actions diminishes the machine's independence, defeats the objective of autonomy, and squarely eliminates the opportunity for trust, which otherwise thrives when the trustor's vulnerability is protected by the trustee amidst unpredictable circumstances, even if (or especially when) the objective may not be met, but the measures to protect are communicated and appreciated.

McDermott et al. [25] provide an example of vulnerability mitigation in their guide on development of human-machine teaming systems [the authors of this guide use the term *human*—in the context of this paper, we would use the term *person*]. In their guide, they first discuss "Directability." Directability is supported when humans are able to easily direct and redirect an automated partner's resources, activities, and priorities. People will have expertise or insights that the automated system will not. People are ultimately accountable for system performance, which means they must be able to stop processes, change course, toggle between levels of autonomy, or override and manually control automation when necessary. They provide the following guidelines for development:

- The automation/autonomy shall provide the means for the operator to redirect the agent's resources when the situation changes or the human has information/expertise that is beyond the bounds of the agent's algorithms.
- The automation/autonomy shall provide the operator the capability to override the automation and assume partial or full manual control of the system to achieve operational goal states.
- The automation/autonomy shall not remove the human operator from the command role.

Despite the essentialness of vulnerability to PMT concepts, the term is rarely operationalized in any meaningful fashion within discourse or experimentation. In scanning our SMP corpus for lexical dispersion of the term, we find that it frequently appears in isolated statements and definitions, but is otherwise abandoned [20]. In fact, the interpersonal definition of vulnerability is often contramanded by experimental design. In the excellent review by Woolley and Glikson on Trust in AI, the authors open with Mayer's definition of interpersonal trust: "the willingness of a party to be vulnerable to the actions of another party based on the expectation that the other will perform a particular action important to the trustor, irrespective of the ability to monitor or control that other party" [26]. In contrast and in contradiction, Woolley and Glikson's summary of research conducted by Ullman and Malle states: "They found that participants reported higher cognitive trust in the robot *they controlled*" [emphasis added]. Furthermore, following this controlled experience of involvement, participants expressed significantly higher trust in potential future robots [27, 28]. This discrepancy is apparent but not addressed: If trust is about willingness to be vulnerable irrespective of control, then what is an experiment truly measuring if it finds that "trust" is contingent on level of control? Further, trust entails an acceptance of vulnerability, which is refused by a desire to control.

Recently, the authors of this paper explored the development of trust between people and machines in using a virtual environment, the Platform for Assessing Risk and Trust with Non-exclusively Economic Relationships (PARTNER) [We refer to this experiment to illustrate the questions of interest rather than to elucidate the results, therefore, we will forgo review of the conclusions. Interested readers can refer to [17]]. In PARTNER, people and machines are paired up to escape a room, and these puzzle stages are constructed to be unsolvable without cooperation (see Valve's Portal 2 game). We made sure to draw in and probe vulnerability in two ways: The first was to build upon its operationalization in Berg et al.'s [29] canonical investment game concerned with financial trust. We argued that the paradigm used—to give a gift of funds which may then be lost during the interaction—did not invoke authentic vulnerability in most participants; thus we focused on non-economic relationships. We argued that inducing participants to experience a sense of physical vulnerability comparable to a trust fall would be more effective and relevant in the real world [A trust fall is a team-building exercise in which a person deliberately falls, trusting the members of a group (spotters) to catch them.]. Insofar as Institutional Review Boards (IRBs) generally frown upon the prospect of dangling people off the edge of cliffs, we opted to do so in virtual reality (VR), from heights and into pits of hazards, to emulate physical peril. Falling in VR is a reliable method to trigger the sensation of falling in the vestibular system, and some users even experience vertigo (those participants were screened out). The other way we enabled opportunities to experience non-financial vulnerability was by creating situations for the robot partner to save or betray the human player, and for the robot partner to perform activities that were hazardous to the human (again, those hazards relate in particular to human biology). Which teammate took the risk-laden action exposes an aspect of trust designed for experimental examination: Did the person perform the safe task while the robot took the risky task (e.g., the task with the potential to fall)?

6 PMT concept 3: Autonomy is a relationship, not a system property

The term autonomous systems (AS) has its origin in warfare. The person-machine unit of a submarine is the typical exemplar, often separated from traditional C3 (command, control, and communications), and authorized to act without instruction. A special class of autonomous systems, lethal autonomous weapons systems (LAWS), are machines set to fire when conditions are met in cases in which intervention by people would be too slow to neutralize the threat. When LAWS are referred to as human-machine teams, the macabre reading is that people are participating only in the sense that they are the targets. LAWS do have significant bounds and limitations on their behavior: The systems cannot not act if the conditions for action are not met, nor can the systems weigh factors in the environment which have not been programmed to assess. While these machines are able to perform complicated actions without the direction of a person, in many respects LAWS are still more automated than autonomous[2].

When used in an interpersonal context, the term autonomy is meant to indicate that a person is not subject to another authority in making personal determinations. This sense of autonomy concerned with self-governance is not even desirable for installation in machines—after all, autonomous systems are meant to improve the human condition and serve people's needs, not act as machines want for themselves (as if wants are even possible for machines). Autonomous systems are artificial and designed, and thus without true motivations. In contrast to automation, wherein a technology performs a pre-specified task in a controlled environment, machine autonomy (in the PMT context) is often used to describe sophisticated, flexible, or adaptive automation that can perform with some degree of initiative and independence in novel contexts or environments, without complete external oversight or control. Importantly, autonomy is earned and awarded through an external authority, making it a property of a relationship rather than a property of an entity within that relationship, as in automation.

Autonomous systems are commonly understood as decision-making technology both capable and worthy of being granted some degree of independence from human control. However, "decision-making" as used here is wishful mnemonic (cf. [3]) for the calculations these machines perform, and the actuations to accomplish the determination of those calculations. While the systems do hold goals, objectives, and missions, these imperatives exist around the level of programing. These systems do not really make decisions, conduct judgements about the preferability of different actions, or emergently generate novel options to choose amongst beyond the methods available in their deployed code.

Currently, potential options and actions available to autonomous systems are limited by their programming, but these machines may eventually be so capable that available to them are such a broad spectrum of possibilities that the limits to their actions cannot be fully predicted; in fact, in systems that are not embodied in the physical world, such as on-line avatars or large language models[3] we are rapidly approaching this uncircumscribed scope, if we have not already reached it.

Though autonomous capability and intelligence often overlap, they are distinguishable. Where autonomous capability is concerned with initiative and independence, intelligence is concerned with the ability to hedge against dynamic vulnerabilities—i.e., threats to autonomy, coordination (teamwork), and ethical (desirable) behavior—in real time. In other words, intelligence and agency are among the essential components of the "personhood" that's missing. For a study in the topic of intelligence, see an upcoming paper in *Entropy* by Baker and Greenberg.

Machine autonomy is not a widget that can be built [30], but rather a privilege people grant to machines that are capable of operating without or outside of our supervision and control. That privilege is earned after testing and experience have demonstrated the capability, or in cases where control is impossible due to environmental constraints (remote, dirty, dangerous). Various conceptual efforts [31, 32] to arrange autonomy as levels, as adjustable, or on a sliding scale, falter

[2] For further information about C3 and LAWS, please see these references: Chapter 20 Command, Control, and Communication (fas.org), IF11150 (congress.gov), DoDD 3000.09, 21 November 2012, Incorporating Change 1 on 8 May 2017 (whs.mil).

[3] Is LaMDA Sentient? — an Interview | by Blake Lemoine | Medium: Though the authors of this paper do not accept the sentience claim, the novelty claim is compelling.

in ordering autonomy as a single functional unit, as opposed to a collection of constituent capabilities that combine in complex patterns to enable minimal communication along the appropriate level of abstraction. These constituent capabilities included in the notion of autonomy, initiative, and independence, and in particular, graceful handoff, are buildable.

Ideally, we might want people to be the ones drawing the line for transfer of attention, but in practice, it may have to be determined by machines, driven by time constraints to be part of its autonomous functionality. Accomplishing effective and efficient handoff between machines and people requires substantial social cognition on the part of the machine. First, the person-machine system needs to assess whether an action is in the purview or even the ability of the machine or the person. Not only does the machine need to know its performance boundaries, that is, what it can and cannot do well, but both the machine and the person require the bit of metacognition that allows each to infer what the other does not or cannot know or do. Together, these indicate to the machine when it ought to ask for help from people, for the person to offer assistance, or that it is not appropriate to ask for assistance. If the machine determines that it cannot or does not know information critical to performing the task, or that it does not have the capability to act, it needs to ask for help. In that respect, autonomous systems should be experiential—they should learn from their interactions with people, or from the experiences of other autonomous systems. Critically, methods are needed to ensure this learning is indeed in the desired direction, and that the autonomous system will not converge to performance boundaries that are unwanted. Appropriate requests for assistance require that the machine have elementary theory of mind, that is, to infer who might know what, who to ask, and deixis (how to refer in time, space, and person). Finally, the machine may need to escalate the request for attention to a person, and hand off the question or task to them. Requests for assistance cannot happen all the time or the system is almost useless, nor can they never happen as the system would take unacceptable action or fail to act appropriately too often. Similarly, if the task is to be handed off, there must be sufficient time for the person to assess the context and prepare to perform the task, as well as to perform the task (Tesla Autopilot Crashes into Motorcycle Riders—Why?[4] 7:24: "So before you trust his take on autonomy, just know that autopilot is programmed to shut down one second before impact, so who's the manslaughter charge going to stick to?"). The timing, information provided, and receptivity of the person are elements of this handoff package. The machine should not escalate for attention matters that set up the people for failure, by leaving insufficient time or providing insufficient information for the issue to be adjudicated by people, or by sharing with people who are not available to receive the handoff. This means that developers must consider the full spectrum of activities in which the person might be engaged as it is a person-machine team wherein neither entity is fully separable.

7 PMT concept 4: Ethics for machine teammates

Ethics for autonomous systems (i.e., those that make for machine teammates), differ from the ethics of artificial intelligence: In particular, the autonomous system's special features of agency, physicality, and sociality, draw in considerations beyond those of traditional technology ethics concerned with social implications of the built world, to include among other specializations, philosophy of action and philosophy of mind.

Agency, the capacity of an entity (agent) to "instantiate intentional mental states capable of performing action," is not necessarily required for a machine to be granted some degree of autonomy, but that capacity becomes increasingly relevant as these machines are permitted entry into more complex environments. Here, complexity is not strictly along the physical or computational dimensions, but the social—arguably, the environment of a home healthcare aid robot is more complex than that of an autonomous vehicle. *Moral agency*, wherein "an agent has the capacities for making free choices, deliberating about what one ought to do, and understanding and applying moral rules correctly in the paradigm cases" is a much higher bar. It is not clear that machines will ever be able to meet these criteria [33] or even need to in order to accomplish their directives, but *Ethical Agency* is within reach and essential for appropriate system performance. The related concept of *Moral Patiency*, the capacity to feel pain or pleasure remains in the realm of living creatures, and respecting that capacity is the mandate of artificial ethical agency.

Physicality: Not all AI is embodied, and not all autonomous systems can be deemed to have intelligence (not even all AI can be deemed to truly have intelligence, cf. upcoming Baker and Greenberg paper). The ethical implications of a machine with the ability to sense an object in the environment to change direction and avoid it differ from those of algorithms that can crunch large amounts of data. Artificial autonomous capability is generally embodied in a cyber-physical system, and is bound to have direct and indirect effect on the physical world. This is not necessarily true of AI, in which its effects on the physical world tend to be mediated by its provision of information to people.

Sociality: The ethical concerns around machine teammates tend to fall more around *how* the team interacts, how the handoff between team members is performed, and whether each member is prepared to act and capable of acting. If a machine is working with a person, it must perform the handoff between tasks,

[4] https://www.youtube.com/watch?reload=9&v=yRdzIs4FJJg

information, objects in such a way that the person is capable of succeeding, while at the same time ensuring that potential for harm to people is minimized. There will be times when a machine is performing a task, and the context changes such that the machine is no longer able to perform the task safely. In those instances, the current development approach is to hand the task back to the person. But this does not ensure that the person is able to perform the task either. It assumes that the person is fully engaged in the task to the point that a hand off is possible. But the purpose of autonomy is to allow the machine to perform without the person, enabling the person to be engaged in other tasks. If the machine is unable to perform the current task, it may be better to have the machine alert the person and instead perform a task at which it is capable of succeeding. In handoff, simply assuming that the person is ready to perform yields a liability issue, and may defy the concept of operations for which the system was built. Rather than transparency of decision making, the person needs to accurately understand how context and environment may affect the ability of the machine to perform. Similarly, the machine needs to understand how the task and environment may have impacted the person's ability to perform, e.g., whether the person has sufficient time to engage in the task, can sense the data or object that has confused the autonomous system, or is even available to perform the task.

Ethics with respect to people refers singularly to the *moral principles that govern a person's behavior or the conducting of an activity*. These principles collect as the set: Transparency, Justice and fairness, Non-maleficence, Responsibility, Privacy, Beneficence, Freedom and autonomy, Trust, Sustainability, Dignity, Solidarity. However, ethics with respect to machines carries at least two senses [34].

The first sense (the *Ethics of machines* or machines as objects of ethical consideration) is the one commonly understood when invoking the terms AI Ethics, or Ethical AI, concerned with the ethical use of artificial agency. This sense in the vein of technology ethics governs human beings (and their institutions), in producing or interacting with machines (their design, use, or interpretation of machine products). The constraints in such governance is extrinsic to the machine, and ethical principles pertain to designers and users. Of the set of principles, fairness, bias, and privacy most exemplify this *of/as objects* sense. Policy documents are exclusively of this sense, both those prescriptive, like from the US Government (IC, DOD, and CIV), Asilomar, and from the Vatican, as well as those descriptive, like the reports by Harvard and Montreal.

The best of breed survey of the *of/as object* sense is Jobin et al. [35]. In reviewing the landscape of AI ethics they came to a consensus around the set of principles listed upfront. They also identified four divergences in how each principle was addressed in the corpus they examined: how ethical principles are interpreted; why they are deemed important; what issue, domain or actors they pertain to; and how they should be implemented. These divergences characterize the splay in semantics mentioned earlier.

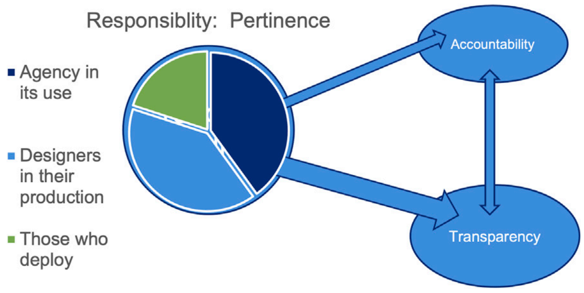


FIGURE 5
Divergence graph.

As part of the SMP effort, we sought to computationally represent and visualize these divergences. In Figure 5 below, we depict a "divergence graph" for the principle of responsibility. These graphs show how different usages or senses of terms (corresponding to Jobin's divergences) differentially connect to related terms. Nodes are sized by term prevalence in the document or corpus. Edges are directional and sized by co-occurrence so that, for example, the width of the link from *responsibility* to *accountability* is to be understood as proportionate to the number of mentions of *accountability* in discussions of *responsibility*. Within the node of *responsibility*, the pie chart indicates the proportions of pertinence usage (Jobin et al's divergence regarding to what or to whom the principle pertains), answering the question: *As it appears in documentation, in what proportions does responsibility pertain to the agency* (in this case, meaning institute) *in its use of AI, designers in their production of AI, or to those who deploy AI?*

The works Jobin review are focused on the ethical *implications of* AI and how policy and governance should safeguard development and protect users. Although this research concerned with obviating and mitigating the personal and societal consequences of AI, such as those presented by algorithmic bias and reward hacking, is crucially important to undertake, it is not the whole picture.

The alternative sense, Ethics *for machines*, or machines as subjects (*for/as subjects*), is the more Asimovian [36] sense concerned with Artificial Ethical Agency. Ethics in this sense regulate the machines themselves, and are only applicable to machines that possess the capability for autonomous agency, unlike other powerful technologies without such a capacity for initiative (like nukes). Ethics for machines are on-board the system proper, and the principles are intended to pertain to the artificial agent itself. This sense of ethics requires commensurate capability and judgement from the machine, a tall order since machines are ordinarily produced for capability, leaving the judgement for people. That gap is how accidents of the kind at the Moscow Open can occur, in which a chess-playing robot broke a child's finger (for a discussion of this incident see upcoming Elsevier chapter by Greenberg on enabling machines to reason about potential harms to humans). Of the principles, non-maleficence and beneficence are the most clearly of this sort. Important questions about how to "teach" ethics to machines emerge of this sense (described in upcoming robots that do no harm paper). The best of breed survey of the for/subject sense is by Tolmejer et al. [37].

When these two senses are set for and followed by people, there is a unitary apparatus for producing, understanding, and executing the principles. However in machines, these two senses are differentiable, though the *of/as object* sense tends to dominate. To see how little these two senses conceptually overlap between Jobin and Tolmejer, see Figure 6 below.

We argue that successful application of ethics to autonomous systems is distinguished by its goal to explicitly design into machines the basic mental faculties (including perception, knowledge representation, and social cognition) that enable them to act as ethical agents. These capabilities in ethics *for* artificial agents are so fundamental, treatment of them tends to be neglected, but it is at this low and early level that the machine's agency is most available to adjustment by normative considerations. Furthermore, owing to these faculties' universality across major schools of philosophical thought (deontological, consequentialist, and virtue ethics) their essentialness is fairly uncontroversial. Beyond this basic level where mental faculties enable machines to have consideration for moral patients, application of ethics to machines begins to resemble the ethics *of/as objects* sense, wherein appropriate behavior is imposed by governance, leading to brittle performance and diminishment of the capacity for trustworthy autonomous activity.

8 PMT concept 5: Trust is learned, trustworthiness is earned

Trust is a socio-affective construct indicating the willingness of a person to be vulnerable to the unpredictable actions of another. Of the foundational concepts for translation from the interpersonal context to the PMT context, confusion around the term *trust* is perhaps the longest lived and most fraught. The topic of *trust* is also the most integrative of the foundational concepts in PMT, and for this reason we discuss it last. As compared to the rich interpersonal concept, its use in machine contexts is austere. Notable contributions to distinguish the senses between contexts include Thin vs. Thick Trust [38], and Cognitive vs. Emotional [19]. In this section, we first survey the features of interpersonal trust and the intricacies of instantiating them in machines, to then we address issues in measurement and in calibration.

When held between people, trust and trustworthiness are understood to be part of a relationship wherein three characteristics of the trustee make it so that the trustor may confidently hold the belief that the trustee will act in the trustor's interest: ability (or competence), benevolence, and integrity [26]. The stability of one's trust varies depending on which of the aforementioned qualities it is based. If trust is based (solely) on the ability of a trustee, trust should then vary depending on how well the trustee performs a task. If trust is grounded in the integrity of a trustee, then it should vary based not on the actual performance of a trustee but on the extent to which the trustee's actions match the values of the trustee. The stability of benevolence-based trust is contingent upon whether the trustee's actions match the goals and motivations of the trustee. When trust is based primarily on the integrity or benevolence of a trustee, poor performance alone will not significantly damage it. Machines, however cannot truly be either benevolent or malevolent, or have integrity or be corrupt. Researchers have attempted to translate benevolence [39] and integrity for machine contexts, but since these qualities are currently impossible to instantiate in machines as they appear in people, they must be inherited by machines from the people who design them. When the trustee is a machine, the final pillar of trustworthiness—ability—is reduced to little more than "predictable performance," or reliability. This hollow port begs the question of why we bother with this artifice of "Trustworthiness" at all.

Yet researchers continue to pursue designs for autonomous systems that are inherently *trustworthy*. From an engineering perspective, one way to operationalize *trustworthiness* is to ensure that the behavior of the machine is reliable to a high degree, and that the machine is capable of performing the task of interest or telling the person that it is unable to perform the task. From a psychological perspective, based on research on the development of trust between people team members, research demonstrates that these are not the critical bases of the development of trust between team members.

Reducing *ability* to *reliability* is problematic: creating machines that are 99.9% reliable may actually be detrimental to the development of trust in autonomous systems. *Reliability* is defined as the consistent performance of an action, an attribute of the trustee, while trust is a learned response by the trustor [40] applicable to situations in which the trustee is not perfectly reliable, or in which the task entrusted is not certainly achievable. We know from research on learning that consistent reinforcement of behavior does lead to learned response. However, if a consistent reward is discontinued, the learned behavior is quickly extinguished. In other words, if a system is 99.9% reliable, then 999 times out of 1000, it will behave as expected—yielding the learned response of trust. But on that 1000th trial, in response to a system failure, the person's learned response can be quickly extinguished. Variable reinforcement, by contrast, leads to acceptance of a much longer duration without reinforcement before the learned response is extinguished.

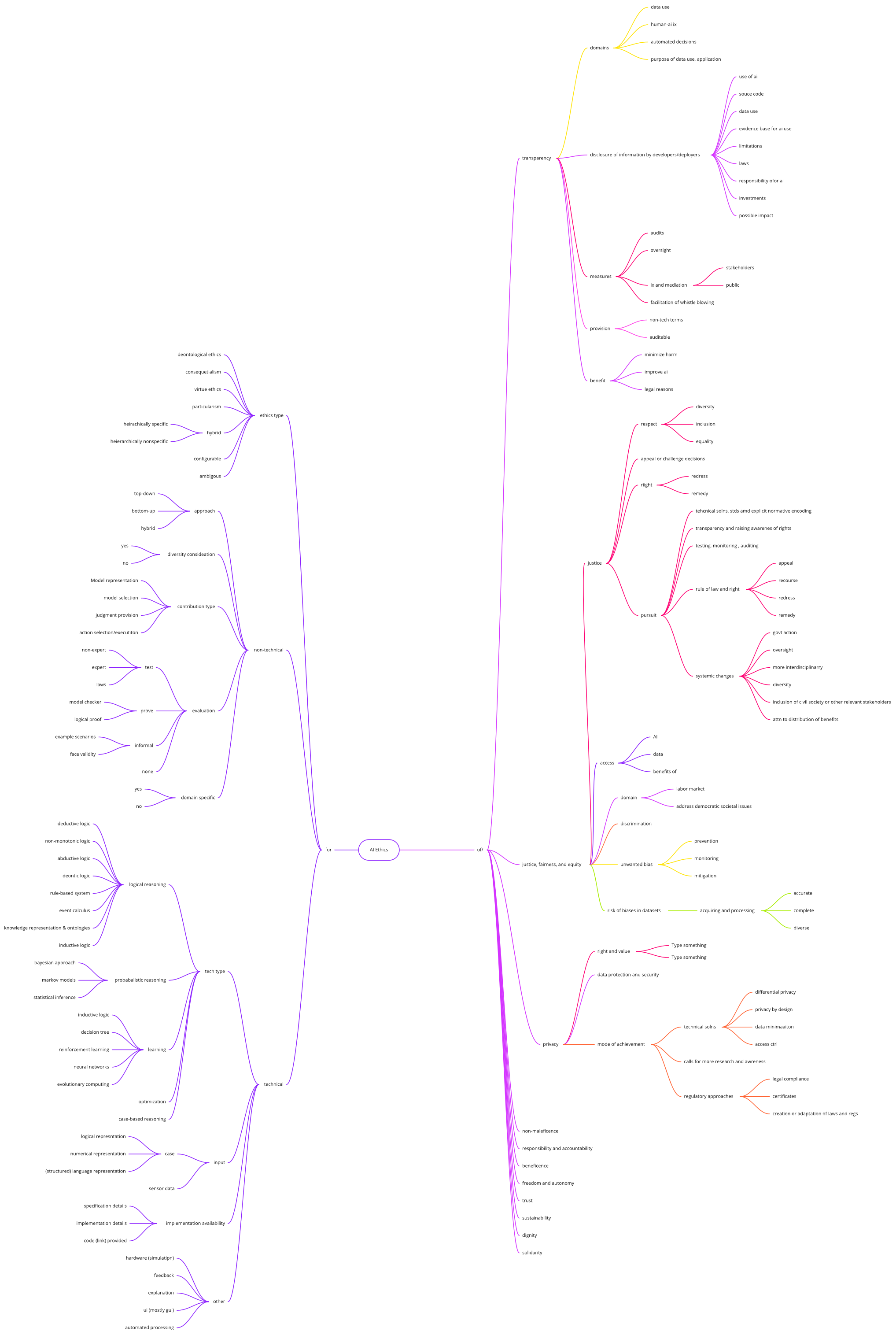


FIGURE 6
Taxonomies, of/as objects vs. for/as subjects.

Therefore, we argue that providing the people insight into when, how, or why the system will fail, will lead to higher levels of trust in autonomy even if (or especially when) the system is less than 99.9% reliable.

Calibration of trust: The discourse on the *calibration* of trust in autonomy most commonly arises from Lee and See's [21] examination of trust in automation. On the face of it, the concept is straightforward: trust in the system should match the system's trustworthiness. However, as we have reviewed, automation does not scale to autonomy, and neither trust nor trustworthiness are unitary—what aspect the trustor's trust is based upon need not match the aspect from where the trustee's trustworthiness is derived. We recommend escaping this complexity by simply replacing trust calibration with reliance calibration: In this way, the axes of calibration would simply be trustor's perceived reliability against trustee's demonstrated reliability. While aspects of the relationship between people and machines are not captured in reliance calibration, the interaction of the person and machine is not aggrandized beyond what the current state of science and engineering can speak to.

We see this aggrandization of *reliability* to *trust* occur, for example, with the reception of findings on algorithm aversion[5]. These findings are typically summarized to claim that people do not expect machines to make errors whatsoever, and so people's "trust" in people is often overrated, whereas people's "trust" in machines is underrated. However, since this phenomenon is almost entirely concerned with performance, it remains squarely within the realm of perceived reliability, and the richness of *trust* may not need to be invoked.

Nonetheless, if the behavior of the system never varies (it performs with perfect reliability), trust is almost irrelevant to the relationship between the person and the machine. For all these reasons, in some cases, the less loaded term of *assurance* (which is licensure-oriented) is more appropriate than the term *trust* (which is state-oriented). For automation, in which action is paramount, and mimicry and rule-following is sufficient (but brittle), the assurance case is based on performance. For machine autonomous systems, in which internal state reflecting the machine's conception of its environment is paramount, and generalization and transfer learning around that environment is possible, the assurance case is based on transparent and interpretable (legible) reasons for why some action was taken over another.

Operationalization and Measurement of trust: Trust is notoriously difficult to measure, in both interpersonal and PMT contexts. As [41] state "a lack of clearly defined measures as they connect to trust theory has also forced scientists to create their own *ad hoc* measures that capture trust as a monolith, rather than a targeted aspect of trust theory." In part, this is due to the phenomenon being a mental state and social relationship to which direct access or quantification is unavailable. Research instead measures proxies from classes including behaviors, subjective assessments, and physiology. However, any of these proxy measures, or even all of them together as a set still do not fully characterize the relevant mental state. The allure that these proxies are measurable drives the conceptualization of trust to meander to meet the proxies. So then, trust is reduced to adoption (behavior), or affinity (subjective assessment), or oxytocin levels (physiology). If we do not measure the right thing, but still optimize for that proxy, are we really saying anything about trust itself? This way of going about science strains the criterion of falsifiability—in these cases, we are searching for our keys under the lamppost, because that's where the light is.

Initial research on trust (of people or machines) relied on subjective measures (e.g., [42]) or indirect measures of trust reflected in the behavior of the person (see for example [43]). Subjective indicators, such as *the negative attitudes toward robots scale (NARS)*, tend to capture more about the likeability of the machine and its position vis-à-vis the uncanny valley or anthropomorphism (eye contact, smiling, nodding, social gesture, responsiveness) than about trust proper. Likeability does not necessarily indicate a willingness to be vulnerable to the machine, especially once the person experiences an event where the machine fails at the task. While such etiquette and immediacy behaviors by the machine are useful to promote adoption, these expressions are manufactured, not earnestly produced as they appear in people, and so designed to manipulate people into a positive disposition, which is not a benevolent affair. When machines produce apologies for poor outcomes, they generally cannot state what they are sorry for, nor can they necessarily change their behavior to ensure that outcome does not occur again (an essential aspect of a genuine apology without which the apology is simply a speech act to get one's way, a sociopathic device). Such a speech act improves perception of the machine's trustworthiness, at least after the first failure, though it is not clear whether repeated apologies would maintain the perception of trustworthiness after a second or third identical failure. Here, notions of betrayal and forgiveness come into play—if these related terms from the interpersonal context seem irrelevant with regard to interacting with a machine, use of the term "trust" must be drawn into suspicion for being just as overzealous.

Physiological indicators of trust are not well established in interpersonal contexts, and it is further unclear whether they would even appear in humans (humans used here instead of people, since the physiology of concern is particular to human biology) trusting machines if those machines are not recognized as social actors, or if teleoperation means that the trust relationship is interpersonal between operator and

[5] Algorithm Aversion: People Erroneously Avoid Algorithms after Seeing Them Err (upenn.edu), Overcoming Algorithm Aversion: The Power of Task-Procedure-Fit | Academy of Management Proceedings (aom.org)

user, and only mediated by machine. Since the behavior of the trustee also has an impact on the wellbeing or goals of the trustor, that is, there is vulnerability in the act of trusting, the psychophysiology associated with that state may be a more worthwhile measurement target.

Behavioral indicators (including *Acceptance, Deference, tolerance, Workload-resistant compliance, behavioral economics measures like investment*) of trust fail to capture alternatives, and if the options are utilize/adopt or not, then the volitional aspect of trust (willingness) is not being measured. Vigilance/neglect and accepting advice or recommendations are also problematic for the same reason. High workload necessitates neglect and acceptance, which saturates measurement, whereas it is only under these kinds of circumstances that one would employ an autonomous system.

On top of these confusions are another, related to Jobin's divergences mentioned in the previous section on ethics—to what issue, domain or actors does trust and trustworthiness pertain? In the 2020 executive order promoting the use of trustworthy AI in the federal government, most of the principles listed are actually referring to "trustworthy use" instead of "the use of trustworthy AI." Of the nine principles, five are incumbent on the governmental agency to ensure that the institution's *use* is trustworthy—in fact the principle of transparency is reversed from its typical use applying to the technology, and here applies to the governmental agency's transparent use of the AI.

Finally, trust is not only personal and calibrated, but highly contextual—one may trust a particular individual for one task in one context but not in another because as people we have learned the characteristics of the context that suggest potential successful performance. Therefore, the ability of the machine to understand the differences in these contexts, and predict its own performance in the context becomes a useful element for the development of trust between people and robots. In other words, the system may succeed at the task in one instance, performing in the way that the person expects but based on reasoning that differs from the person's basis for action. At a second point in time, the machine may take a different action because the aspects of the context on which it focused are different than in the first instance (while the aspects of the context on which the person focused remain the same). When an autonomous system is created to perform a task, it is designed to achieve the person's goal. When the person performs the task without automation, there are rules that underly how the task is performed—such as to act otherwise could lead to injury. These underlying rules may not be relevant to the machine, as it may not be harmed by the environment as easily. The designer must ask, however, whether the machine should still follow this rule so that the behavior of the machine is more easily predicted, understood, or trusted by the person. Given our conceptualization of trust (and following the argument of [44]) the person in a person-machine team must similarly be able to assess the state of the machine—that is, the ability to assess the risk in teamwork and their own vulnerability to the potential for a mistake by the machine.

9 Conclusion

Words matter—in a very Whorfian way, they shape how we engineer our world. The translation of terms from their original interpersonal use to their use in person-machine teaming contexts must be performed deliberately to maintain conceptual and scientific rigor. The reductive mindset of "human-machine teaming" suggests that a human may be treated like automatons with input and output to be compatible and interchangeable with machines, but in a team or otherwise, machines and people are not equivalent.

This reductive mindset further leads to beliefs that development of machine teammates can ignore the fundamental behavior of people, because the person could just be trained to support the machine. Vice our argument here that people will always be part of the system, "in the loop," "on the loop," or dictating or receiving the output of the autonomous system's actions, we find that too often, the aim in developing autonomous systems centers around the desire to engineer people out of the system. However, this approach undercuts the purpose of developing autonomous teammates. People are social, and will engage in social interactions with entities that have even a modicum of perceived independent behavior. Therefore, person-machine teaming is an inherently social activity, and as such, engineering and development of autonomous systems must acknowledge people as social entities, and account for social behavior in developing the system.

To be of the greatest utility, autonomous machines must be allowed to operate with the initiative and independence they were built to exert. Seeking to control every possible outcome of their behavior reduces them to tools and undermines their usefulness. We must admit that machine performance, just as the performance of people, will rarely be perfect. To that end, in the development of autonomous teammates, we must accept this imperfection and the vulnerability that it entails, to people, to the system, and to the task (see Coactive design (acm.org)). We must acknowledge that development and test environments, even when they are of high fidelity and of adequate ecological validity, will never exactly match the deployment environment of the wild. Instead of controlling machine behavior as a means to achieve some aspect of a trust relationship, we argue that we must appreciate how context affects system performance—both the performance of the machine and of the person. Autonomous machines must not be designed to assume that the person they are teaming with is sufficiently involved in the task to be able to take it over at any time (even with notice), but rather, these systems must be designed for safe and graceful failure that accounts for unmitigable vulnerability. The approach here detailed has significant ethical and legal implications for the development of robots that are categorically different and merit distinct consideration from those commonly discussed in the development of AI.

Author contributions

AG primarily authored the introduction, and the sections on Semantics and the SMP, Personhood, Vulnerability, Autonomy, and Ethics. JM primarily authored the section on teaming and the conclusion, contributed to all the other sections, and is here celebrated for translating AG's inscrutable language to eschew obfuscation and render the material accessible to the intended audience. AG and JM coauthored the section on Trust.

Funding

This production was supported a JHU Discovery Program grant on *Enabling machines to reason over potential harms to humans* (AG), by APL's *Execution Priority on Ethical AI* (AG), and by the Institute for Experiential Robotics, Northeastern University (JM). The development of the semantic mapping pipeline was supported by JHU's Institute for Assured Autonomy (AG).

Acknowledgments

AG declares special thanks to Joshua D. Baker at APL, and Brian Hutler now at Temple University for the years of discussion that helped hone these thoughts.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

1. Greenberg AM. *Deciding machines: Moral-scene assessment for intelligent systems. Human-machine shared contexts.* Elsevier (2020). doi:10.1016/B978-0-12-820543-3.00006-7

2. Mitcham C. The importance of philosophy to engineering. *Teorema XVII* (1998) 3:27–47.

3. McDermott D. Artificial intelligence meets natural stupidity. *SIGART Bull* (1976) 57:4–9. doi:10.1145/1045339.1045340, no.

4. Turkle S, Taggart W, Kidd CD, Dasté O. Relational artifacts with children and elders: The complexities of cybercompanionship. *Connect Sci* (2006) 18(4):347–61. doi:10.1080/09540090600868912

5. Rieder TN, Hutler B, Debra J, Mathews H. Artificial intelligence in service of human needs: Pragmatic first steps toward an ethics for semi-autonomous agents. *AJOB Neurosci* (2020) 11(2):120–7. doi:10.1080/21507740.2020.1740354

6. Lee JER, Nass CI. Trust in computers: The computers-are-social-actors (CASA) paradigm and trustworthiness perception in human-computer communication. In: *Trust and technology in a ubiquitous modern environment: Theoretical and methodological perspectives*. Pennsylvania, United States: IGI Global (2010). p. 1–15.

7. Nass C, Fogg BJ, Moon Y. Can computers be teammates? *Int J Human-Computer Stud* (1996) 45(6):669–78. doi:10.1006/ijhc.1996.0073

8. Rix J. From tools to teammates: Conceptualizing humans' perception of machines as teammates with a systematic literature review. In: Proceedings of the 55th Hawaii International Conference on System Sciences (2022).

9. Lyons JB, Mahoney S, Wynne KT, Roebke MA (2018). *Viewing machines as teammates: A qualitative study.* Palo Alto, CA: AAAI Spring Symposium Series.

10. Salas E, Dickinson TL, Converse SA, Tannenbaum SI. Toward an understanding of team performance and training. *Teams: Their training and performance*. In R. W. Swezey E. Salas (Eds). Ablex Publishing (1992), 3–29.

11. Walliser JC, de Visser EJ, Wiese E, Shaw TH. Team structure and team building improve human–machine teaming with autonomous agents. *J Cogn Eng Decis Making* (2019) 13(4):258–78. doi:10.1177/1555343419867563

12. Salas E, Cooke NJ, Rosen MA. On teams, teamwork, and team performance: Discoveries and developments. *Hum Factors* (2008) 50(3):540–7. doi:10.1518/001872008x288457

13. Abrams D, Wetherell M, Cochrane S, Hogg MA, Turner JC. Knowing what to think by knowing who you are: Self-categorization and the nature of norm formation, conformity and group polarization. *Br J Soc Psychol* (1990) 29(2): 97–119. doi:10.1111/j.2044-8309.1990.tb00892.x

14. Kozlowski SW, Ilgen DR. Enhancing the effectiveness of work groups and teams. *Psychol Sci Public Interest* (2006) 7(3):77–124. doi:10.1111/j.1529-1006.2006.00030.x

15. Lyons JB, Sycara K, Lewis M, Capiola A. Human–autonomy teaming: Definitions, debates, and directions. *Front Psychol* (2021) 12:589585–15. doi:10.3389/fpsyg.2021.589585

16. McNeese NJ, Demir M, Cooke NJ, Myers C. Teaming with a synthetic teammate: Insights into human-autonomy teaming. *Hum Factors* (2018) 60(2): 262–73. doi:10.1177/0018720817743223

17. Marble JL, Greenberg AM, Bonny JW, Kain SM, Scott BJ, Hughes IM, Luongo ME. Platforms for assessing relationships: Trust with near ecologically-valid risk, and team interaction. In: *Engineering artificially intelligent systems*. Berlin, Germany: Springer (2021). p. 209–29.

18. Fincannon T, Barnes LE, Murphy RR, Riddle DL. Evidence of the need for social intelligence in rescue robots. In: 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566); 28 September 2004 - 02 October 2004; Sendai, Japan. IEEE (2004). p. 1089–95.

19. Glikson E, Woolley AW. Human trust in artificial intelligence: Review of empirical research. *Acad Manag Ann* (2020) 42(2):627–660. doi:10.5465/annals.2018.0057

20. Lyons JB, Sean Mahoney KTW, Roebke MA. *Trust and human-machine teaming: A qualitative study. Artificial intelligence for the internet of everything*. Amsterdam, Netherlands: Elsevier (2019). doi:10.1016/B978-0-12-817636-8.00006-5

21. Lee JD, See KA. Trust in automation: Designing for appropriate reliance. *HFES* (2004) 46(1):50–80. doi:10.1518/hfes.46.1.50.30392

22. Albuquerque N, Guo K, Wilkinson A, Savalli C, Otta E, Mills D. Dogs recognize dog and human emotions. *Biol Lett* (2016) 12:20150883. doi:10.1098/rsbl.2015.0883

23. Tschopp M. Vulnerability of humans and machines - a paradigm shift (scip.ch) (2020). Available at https://www.scip.ch/en/?labs.20220602 (Accessed on August 15, 2022).

24. Roy JL, McAllister DJ, Bies RJ. Trust and distrust : New relationships and realities. *Acad Manage Rev* (1998) 23:438–58. doi:10.5465/amr.1998.926620

25. McDermott P, Dominguez C, Kasdaglis N, Ryan M, Trhan I, Nelson A. *Human-machine teaming systems engineering guide*. Bedford, United States: MITRE CORP BEDFORD MA BEDFORD United States (2018).

26. Mayer RC, Davis JH, Schoorman FD. An integrative model of organizational trust. *Acad Manage Rev* (1995) 20(3):709–34. doi:10.5465/amr.1995.9508080335

27. Ullman D, Malle B. The effect of perceived involvement on trust in human-robot interaction. In: 2016 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI); 07-10 March 2016; Christchurch, New Zealand. IEEE (2016). p. 641–2.

28. Ullman D, Malle BF. Human-robot trust: Just a button press away. In: Proceedings of the companion of the 2017 ACM/IEEE international conference on human-robot interaction; 6 March 2017; New York, NY, United States (2017). p. 309–10.

29. Berg J, Dickhaut J, McCabe K. Trust, reciprocity, and social history. *Games Econ Behav* (1995) 10123:122–42. doi:10.1006/game.1995.1027

30. Bradshaw JM, Hoffman RR, Johnson M, Woods DD. The seven deadly myths of ‘autonomous systems. *Human-Centered Comput* (2013) 2–9.

31. Sheridan TB. Humans and automation: System design and research issues. *Hum Factors* (2002) 39(2):280.

32. Beer JM, Fisk AD, Rogers WA. Toward a framework for levels of robot autonomy in human-robot interaction. *J Hum Robot Interact* (2014) 3(2):74–99. doi:10.5898/JHRI.3.2.Beer

33. Sparrow R. Why machines cannot be moral. *AI Soc* (2021) 36(3):685–93. doi:10.1007/s00146-020-01132-6

34. Müller VC. Ethics of artificial intelligence and robotics. In: EN Zalta, editor. *The stanford encyclopedia of philosophy (summer 2021 edition)*. Palo Alto, CA: The Stanford Encyclopedia of Philosophy (2020).

35. Jobin A, Ienca M, Vayena E. The global landscape of AI ethics guidelines. *Nat Mach Intell* (2019) 1:389–99. doi:10.1038/s42256-019-0088-2

36. Asimov I. In: *Run around. I, Robot (The Isaac Asimov Collection)*. New York: Doubleday (1950).

37. Tolmeijer S, Kneer M, Sarasua C, Christen M, Bernstein A. Implementations in machine ethics: A survey. *ACM Comput Surv* (2021) 53:1–38. doi:10.1145/3419633, no. 6.

38. Roff HM, Danks D. “Trust but verify”: The difficulty of trusting autonomous weapons systems. *J Mil Ethics* (2018) 17(1):2–20. doi:10.1080/15027570.2018.1481907

39. Atkinson DJ. “Final report : The role of benevolence in trust of the role of benevolence in trust of autonomous systems,(2015). doi:10.13140/RG.2.1.4710.5127

40. Hoff KA, Bashir M. Trust in automation: Integrating empirical evidence on factors that influence trust. *Hum Factors* (2015) 57(3):407–34. doi:10.1177/0018720814547570

41. Kohn SC, De Visser EJ, Wiese E, Lee YC, Shaw TH. Measurement of trust in automation: A narrative review and reference guide. *Front Psychol* (2021) 12: 604977. doi:10.3389/fpsyg.2021.604977

42. Schaefer KE. Measuring trust in human robot interactions: Development of the “trust perception scale-HRI”. In: *Robust intelligence and trust in autonomous systems*. Boston, MA: Springer (2016). p. 191–218.

43. Freedy A, De Visser E, Weltman G, Coeyman N. Mixed initiative team performance assessment system (MITPAS) for training and operation. *Interservice/Industry Train Simulation Edu Conf (I/ITSEC)* (2007) 7398:1–10.

44. Hopko SK, Mehta RK. Trust in shared-space collaborative robots: Shedding light on the human brain. *Hum Factors* (2022) 0(0):187208221109039. doi:10.1177/00187208221109039

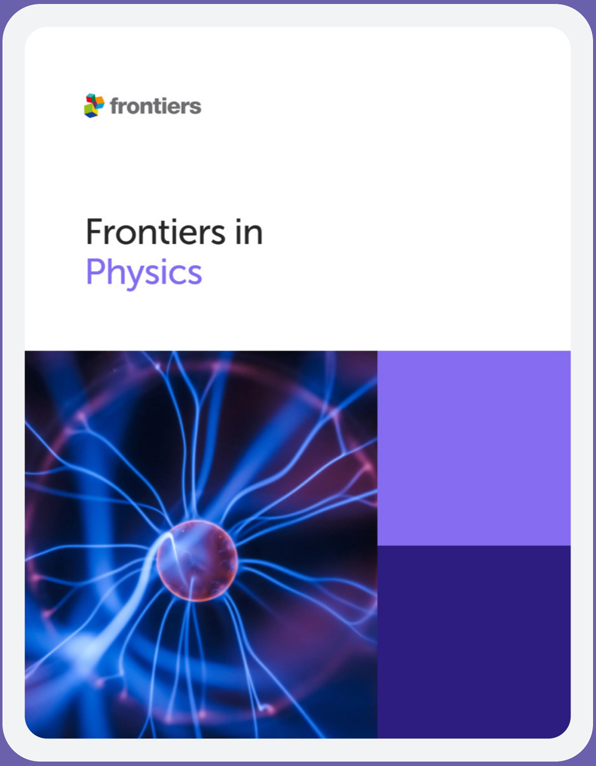
Frontiers in Physics

Investigates complex questions in physics to understand the nature of the physical world

Addresses the biggest questions in physics, from macro to micro, and from theoretical to experimental and applied physics.

Discover the latest Research Topics

See more →



frontiers

Frontiers in Physics

Frontiers

Avenue du Tribunal-Fédéral 34
1005 Lausanne, Switzerland
frontiersin.org

Contact us

+41 (0)21 510 17 00
frontiersin.org/about/contact