TYPE Original Research
PUBLISHED 20 January 2025
DOI 10.3389/frobt.2024.1426282

Check for updates

OPEN ACCESS

EDITED BY
Qi Lu,
The University of Texas Rio Grande Valley,
United States

REVIEWED BY
János Botzheim,
Eötvös Loránd University, Hungary
Philip M. Dames,
Temple University, United States

*CORRESPONDENCE
Gal A. Kaminka,
galk@cs.biu.ac.il

RECEIVED 01 May 2024
ACCEPTED 12 November 2024
PUBLISHED 20 January 2025

CITATION
Kaminka GA and Douchan Y (2025)
Heterogeneous foraging swarms can be better.
Front. Robot. AI 11:1426282.
doi: 10.3389/frobt.2024.1426282

Heterogeneous foraging swarms can be better

Gal A. Kaminka* and Yinon Douchan

Department of Computer Science, Gonda Brain Research Center, and Nanotechnology Center, Bar Ilan University, Ramat Gan, Israel

Introduction: Inspired by natural phenomena, generations of researchers have been investigating how a swarm of robots can act coherently and purposefully, when individual robots can only sense and communicate with nearby peers, with no means of global communications and coordination. In this paper, we will show that swarms can perform better, when they self-adapt to admit heterogeneous behavior roles.

Methods: We model a foraging swarm task as an extensive-form fully-cooperative game, in which the swarm reward is an additive function of individual contributions (the sum of collected items). To maximize the swarm reward, previous work proposed using distributed reinforcement learning, where each robot adapts its own collision-avoidance decisions based on the Effectiveness Index reward (*EI*). *EI* uses information about the time between their own collisions (information readily available even to simple physical robots). While promising, the use of *EI* is brittle (as we show), since robots that selfishly seek to optimize their own *EI* (minimizing time spent on collisions) can actually cause swarm-wide performance to degrade.

Results: To address this, we derive a reward function from a game-theoretic view of swarm foraging as a fully-cooperative, unknown horizon repeating game. We demonstrate analytically that the total coordination overhead of the swarm (total time spent on collision-avoidance, rather than foraging per-se) is directly tied to the total utility of the swarm: less overhead, more items collected. Treating every collision as a stage in the repeating game, the overhead is bounded by the total *EI* of all robots. We then use a marginal-contribution (difference-reward) formulation to derive individual rewards from the total *EI*. The resulting Aligned Effective Index (AEI) reward has the property that each individual can estimate the impact of its decisions on the swarm: individual improvements translate to swarm improvements. We show that AEI provably generalizes previous work, adding a component that computes the effect of counterfactual robot absence. Different assumptions on this counterfactual lead to bounds on AEI from above and below.

Discussion: While the theoretical analysis clarifies both assumptions and gaps with respect to the reality of robots, experiments with real and simulated robots empirically demonstrate the efficacy of the approach in practice, and the importance of behavioral (decision-making) diversity in optimizing swarm goals.

KEYWORDS

multi-agent reinforcement learning, foraging, swarm robotics, heterogeneous robots, robot diversity, difference reward, marginal contribution, game theory

1 Introduction

Distributed multi-robot systems comprise multiple robots, each under its own control (Farinelli et al., 2004; Parker, 2008). Typically, the robots are deployed to carry out tasks toward a global goal. Examples include coverage (Agmon et al., 2008a; Hazon and Kaminka, 2008; Yehoshua et al., 2016; Giuggioli et al., 2016; Rekleitis et al., 2008); patrolling (Sempe and Drogoul, 2003; Elmaliach et al., 2007; Elmaliach et al., 2008; Agmon et al., 2008b; Elmaliach et al., 2009; Basilico et al., 2009; Marino et al., 2009; Jensen et al., 2011; Portugal and Rocha, 2013; Yan and Zhang, 2016); formation maintenance (Kaminka and Glick, 2006; Kaminka et al., 2008; Michaud et al., 2002; Fredslund and Mataric, 2002; Desai, 2002; Desai et al., 1998; Kaminka et al., 2013; 2016; Balch and Arkin, 1998; Lemay et al., 2004; Michael et al., 2008); multi-agent path planning (Yu and Lavalle, 2015; Sharon et al., 2015; Stern et al., 2019) or navigation (Fox et al., 1997; van den Berg et al., 2011; Snape et al., 2011; van den Berg et al., 2008; Guy et al., 2010; Bouraine et al., 2014); order picking (Wurman et al., 2008; Hazard and Wurman, 2006); sustainable agricultural foraging (Song and Vaughan, 2013); and more (Kaminka et al., 2010).

Necessarily, the robots share resources (at the very least, the space of their work area), and thus, a fundamental challenge is the challenge of *multi-robot coordination*. As robots cannot act completely independent of others, they must coordinate their actions with other robots in order to avoid and resolve conflicts over resource use. Such coordination necessarily introduces some overhead into the workings of the robots, either by design or by *ad hoc* necessity.

Multi-robot coordination, therefore, both *supports* and *competes* with the achievement of the goals of the robots. Managing the coordination is a necessary component of multi-robot systems and can be done in a variety of ways. Distributed approaches that rely on joint decision-making by the robots [e.g., Gage, 1992; Parker, 1998; Kaminka and Frenkel, 2005; Xu et al., 2005; Zlot and Stentz, 2006; Vig and Adams, 2006; Kaminka and Frenkel, 2007; Kaminka et al., 2007; Dias and Stentz, 2000; Dias et al., 2004; Gerkey and Mataric, 2002; Gerkey and Matarić, 2004; Goldberg et al., 2003; Farinelli et al., 2006; Parker and Tang, 2006; Tang and Parker, 2007; Liemhetcharat and Veloso, 2013; Sung et al., 2013] require high communication availability and the capability of robots to assess not just their own state but also those of others. When such high-bandwidth communications are possible, these approaches can be very effective.

Under settings in which communications are limited in bandwidth and range (e.g., as the number of robots in a group increases), swarm robotics methods offer a promising approach to manage the coordination between robots. Here, robots necessarily coordinate *ad hoc* and *locally*, with little or no communications (Hamann, 2018; Hénard et al., 2023). Swarm robotics approaches have been applied various tasks, some similar to those discussed above: coverage (Batalin and Sukhatme, 2002; Osherovich et al., 2007); foraging (Goldberg and Matarić, 1997; Rybski et al., 1998; Balch, 1999; Vaughan et al., 2000; Zuluaga and Vaughan, 2005; Rosenfeld et al., 2008; Kaminka et al., 2010; Douchan and Kaminka, 2016; Douchan et al., 2019); and flocking, formation maintenance, and collective motion (Balch and Hybinette, 2000; Mataric, 1994; Moshtagh et al., 2009; Bastien and Romanczuk, 2020).

With few exceptions (see Section 2 for a discussion), swarm robotics research has investigated settings in which swarms are homogeneous; every robot has the same capabilities as others. Ignoring stochastic elements in perception, actuation, and decision-making components, different robots would respond in an identical manner, given the same local state in which they find themselves.

In this paper, we show how swarms can perform better when they self-adapt and specialize so that their behavioral roles become heterogeneous: given the same settings, different robots in the swarm learn to respond differently.

We focus on spatial coordination in swarm foraging. This is a canonical task for swarm robotics researchers, with many practical applications (see Section 2). We may model this task as an extensive-form fully cooperative game, in which the swarm goal is an additive function of individual contributions (collected items) (Kaminka et al., 2010). As robots cannot share the same spot at the same time and must avoid and resolve collisions, they must coordinate *spatially*, acting so as to not collide and continue their task normally if a collision occurs. Theoretically, if robots could predict future collisions and their effects, they could use such a model to make optimal collision-avoidance decisions. In practice, individual robots cannot coordinate or communicate globally and, thus, cannot select actions that are optimal for the swarm as a whole.

To compensate for missing global information, Kaminka et al. (2010) presented a multi-agent independent-learner reinforcement learning approach, where a reward function, called the effectiveness index (EI), uses only local information: the time between collisions, which is easily measured by each robot independently. Robots can individually and independently use EI to adapt their collision-avoidance strategies, dynamically diversifying their behavioral responses.

Unfortunately, although the use of EI proved effective in some cases, its effectiveness is brittle (as we show). All too often, robots learned policies that minimize their individual time spent on collisions (improving their own EI rewards) but at the expense of others. This degraded the performance of the swarm as a whole. In such cases, the individual and collective utilities are said to be *mis-aligned*.

To address this, we re-examine how swarm-wide (global) utility is related to individual actions. First, we show that the total coordination overhead of the swarm (total time spent on collision avoidance, rather than foraging *per se*) is directly related to the total utility of the swarm: the less collective overhead, the more items collected. Then, we transform the extensive-form game to a fully cooperative repeated game with an unknown horizon. Treating every collision as a stage in the repeating game, we show that this collective overhead is bounded by the *total EI of all robots* over all stages. These two results are conjectured, but unproven, in previous work.

We then derive an *aligned* individual reward function, called the *aligned effective index* (AEI). This derivation is done from first principles: given the total EI (a global measure), we derive for each robot its marginal (individual) contribution to this value. This is done by having each robot estimate the swarm utility when it is a member of the swarm and when—hypothetically—it is not (a counterfactual). This derivation step follows difference-reward formulations (Tumer et al., 2002; Tumer et al., 2008) but differs from them in the assumptions required for estimating the counterfactuals for physical robots that only measure time. The resulting individual reward—AEI—has the property that each individual can estimate the impact of its decisions on the swarm: individual improvements translate to swarm improvements. Although AEI is derived anew, it provably generalizes EI as introduced in earlier work, adding to it a component that computes the effect of counterfactual robot absence. Different assumptions on this computation lead to bounds on AEI from above and below, which we present. Although the theoretical analysis clarifies assumptions and principled results, experiments with real and simulated robots highlight gaps with respect to the reality of robots. We explore several experimental settings (simulated and real robots), using various approximations of AEI and using both discrete-time and continuous-time Q-Learning algorithms. The experiments empirically demonstrate the efficacy of the approach in practice.

The results show that in the general case, the swarm as a whole achieves maximal results when its members become specialized through learning, i.e., they become *behaviorally diverse*: their responses to potential collisions differ, and it is that diversity that achieves maximal results. This conclusion complements those of others, investigating mechanical diversity or capability diversity in swarms (Dorigo et al., 2012; Kaminka et al., 2017; Berumen et al., 2023; Adams et al., 2023).

This paper is organized as follows: Section 2 provides background and motivation for the foraging the task, as well as a review of related work; Section 3 details the theoretical model; Section 4 discusses its approximation in the reality of robotics in practice; Section 5 presents the results from extensive simulation and real robot experiments; and Section 6 concludes with a discussion on the implications and scope of the work.

2 Motivation and background

We discuss the background and context for this study. First, we motivate the focus on multi-robot swarm foraging and commercial variants in Section 2.1. We then present a view of swarm foraging from the perspective of the single swarm member (Section 2.2). This allows us to place previous and existing work in context and also to present the opportunity for using learning for improving foraging. Section 2.3 focuses on investigations of this opportunity and their relation to the techniques reported here.

2.1 Swarm foraging: an exemplary swarm task

The motivation for our work arises from the scientific study of a canonical multi-robot task: *foraging* (Balch, 1999; Winfield, 2009; Zedadra et al., 2017; Lu et al., 2020). This is a task where a group of robots is deployed to repeatedly search for objects of interest (*items*) and, when found, for transporting them to one or more collection points (*homes*). Foraging is a canonical multi-robot problem because it raises challenges in multiple aspects of multi-robot systems:

- Management of communications between robots, e.g., with respect to where items may be found. Communications are often non-existent or limited in range and bandwidth; they may be stigmergic, as in the case of ant trail pheromones.
- Effects of population changes (robot death/birth) and various types of individual failures.
- Scalability of methods as groups grow in size
- Collision handling and avoidance as robots inevitably crowd around home locations and sometimes in areas with high item density.

We cannot do justice to a full survey of multi-robot coordination, even if we limit ourselves to foraging. Some surveys of interest on swarms in general (Hamann, 2018; Dorigo et al., 2021; Hénard et al., 2023) and foraging in particular (Winfield, 2009; Zedadra et al., 2017; Lu et al., 2020) may be found elsewhere. We discuss the most closely related work below.

Our focus is on a swarm version of foraging, where robots do not rely on communications for coordination and have little knowledge of the state of others other than their bearing within some limited local range. In other words, we assume that robots can find items, transport them, and repeat the process. They can sense the bearing (angle) to others within a limited range so that they may attempt to avoid collisions or resolve them if they occur. Other than this sensing capability, we only assume they have their own internal clocks (which are not globally synchronized), so they may, for instance, measure the time from a previous collision.

The robots have mass and cannot pass through each other, in contrast to theoretical investigations of so-called “particle agents” (Löffler et al., 2023). We make no assumption as to the self-mobility of the items themselves, although in our experiments, the items were static [see the studies by Rosenfeld et al., 2008; Hahn et al., 2020 for examples of foraging while needing to track targets].

Foraging has largely been investigated in settings where transporting an item requires a single robot, and we maintain this assumption here. However, other investigations have broken away from this assumption (and others noted above). Adhikari (2021) and Lee et al. (2022) discussed dynamic robot chains (“bucket brigades”), in which robots pass items from one to the other to avoid congestion, utilizing multiple robots even for a single item. Pitonakova et al. (2014), Ordaz-Rivas et al. (2021), and Ordaz-Rivas and Torres-Treviño (2022) addressed collective transport tasks in foraging, where multiple robots are required in order to move a single object. From this perspective, AVERT (Amanatiadis et al., 2015) is also a related system. It is a four-robot system designed to transport wheeled vehicles by having each robot attaches itself to a wheel, lifting the vehicle and carrying it together.

Almost all investigations of foraging swarms, including ours reported here, are of fully cooperative systems, where robots are assumed to be cooperative, and coordination is a challenge that arises out of their limited capabilities. This assumption stands at the basis of many applications of foraging for physically searching areas (Schloesser et al., 2021; Aljalaud and Kurdi, 2021), search-and-rescue operations (Francisco et al., 2018; Suarez and Murphy, 2011; Pham and Nguyen, 2020), and humanitarian mine clearance (McGuigan et al., 2022). Albiero et al. (2022) surveyed a few dozen investigations of agricultural applications of swarm robotics, of which a large number discuss foraging variants or highly related technologies. Dorigo et al. (2021) examined new application areas for swarms in general, foraging in particular. These include future applications in precision agriculture (e.g., harvesting), industrial monitoring and inspection, civil protection and response to natural disasters, and molecular robotics for medicinal and clinical intervention.

Recent studies break away from the assumption of fully cooperative swarms. They are motivated by future applications of foraging, where robots are self-interested (e.g., manufactured or deployed by different organizations). In such cases, the robots have to be incentivized to cooperate (Van Calck et al., 2023) and coordinate under conditions requiring privacy protection (Ferrer et al., 2021) and proof-of-work (Pacheco et al., 2022).

One specific application of foraging, *order picking*, is of particular interest here. It is a highly successful commercially significant variant of foraging, where robots collect items in a logistic warehouse, in order to fulfill customer orders (e.g., arriving via the web) (Wurman et al., 2008; Hazard and Wurman, 2006). Automated robotic order picking is one of the key technologies developed by Amazon Robotics, after it was acquired by Amazon in its takeover of Kiva Systems (for 775 million dollars; at the time, Amazon's second-largest acquisition). This system was built to replace most human labor in a logistics warehouses[1]. In such settings, robots must engage in spatial coordination, e.g., while moving in the passageways along shelves, or when arriving at the packing stations with the collected items.

Order picking is a complex task and is interesting from a number of different technological perspectives. From a pure swarm perspective, it may be looked at as a form of foraging: robots individually look for items of interest, pick them up, and bring them to a target area. From a centralized control perspective, order picking can be viewed as a particularly challenging continual N-robot motion planning task, where a central server computes non-colliding paths for the robots.

From a scientific point of view, both perspectives raise interesting challenges worthy of investigation. Centralized algorithms for planning multi-robot paths can guarantee optimal paths free from collisions. However, such planning is computationally intractable (Yu and Lavalle, 2015; Sharon et al., 2015; Stern et al., 2019). In contrast, swarm methods are simple to deploy and robust to population changes, although typically sacrificing the ability to prevent all collisions. This raises the need for online collision-handling methods, which, in swarm settings, are often *myopic* and far from optimal. Necessarily, they respond to a collision with little or no ability to consider future collisions and inevitable crowding (e.g., around the target areas).

We observe that regardless of the scientific lens through which we examine order picking, we find that on-board, fully autonomous collision avoidance is a strictly *necessary* component. There are two reasons for this (Wurman et al., 2008):

- First, human workers may move about the warehouse—neither do they follow trajectories planned for robots (Thrun et al., 2000) nor can they be relied on to avoid collisions in a manner compatible with the robots' choices. Moreover, human involvement may be needed in other applications as well (Schloesser et al., 2021).
- Second, even under the assumption that a planning algorithm generates perfect trajectories for the robots, and no humans are about, the possibility of electro–mechanical and communication failures (even non-catastrophic failures, such as simply slowing down as battery levels decrease) requires the robots to have on-board collision-avoidance and re-planning capabilities (Simmons et al., 1997).

In reality, therefore, robots deployed for order picking essentially carry out swarm foraging, albeit perhaps more guided in their search for items and when moving toward home. In the early version of the Kiva system, as captured by the alphabet soup simulator published by Kiva engineers, each robot was responsible for its own path-planning and collision-avoidance responses.

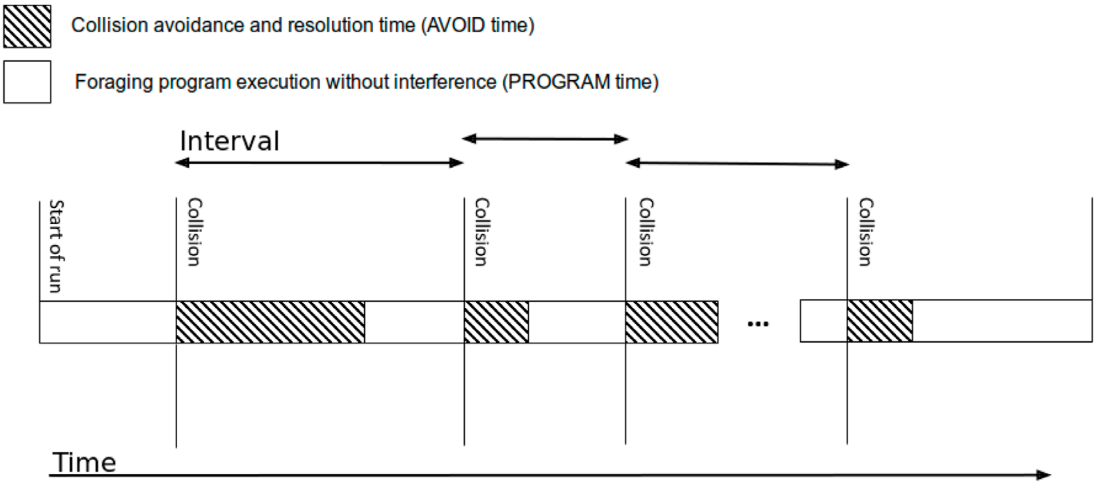
2.2 Improving foraging by improving collision avoidance

Figure 1 shows a perspective [also described by Kaminka et al. (2010); Douchan et al. (2019), although using somewhat different terminology] on the execution timeline from the perspective of a single robot engaged in foraging. The robot begins by executing its foraging activity, stopping when a spatial conflict occurs (e.g., a collision is imminent). It then selects a collision-handling method, which executes for a time. When the collision is averted, the robot can switch back to carrying out its foraging until another collision is imminent. This repeats until the robot task is externally terminated (e.g., by the need to recharge). Each interval between collisions is split into two, termed the *avoidance time* (spent by the robot actively coordinating—shown in gray) and the *program time* (no need to coordinate; the robot focuses on its task).

This view of the robot's timeline allows us to position our work with respect to others. First, many foraging methods focus on improving the productivity of the *program* phase, where the search (for items or home) takes place. This can be done by having robots (i) plan their paths better (Duncan et al., 2022; Cheraghi et al., 2020; Nuzhin et al., 2021; Jin et al., 2020), assuming some localization capabilities, or (ii) communicate information relevant to improving the search (Hoff et al., 2010; Sung et al., 2013; Pitonakova et al., 2014; Alers et al., 2014; McGuigan et al., 2022; Adams et al., 2023; Salman et al., 2024).

A second set of investigations focus on attempting to optimize an entire cycle (avoidance and program), by restricting the behavior of the robot during both program and avoidance such that collisions are minimized, and their resolution is relatively fast. For instance, Schneider-Fontan and Matarić (1998) reported on an algorithm that pre-allocates robots to different territories. Each robot operates in its territory but has the ability to pass objects to another, thus creating a bucket brigade-like structure. They also discussed re-allocating the territories once a robot fails. Goldberg and Mataric (1997) compared several different approaches to this task, measuring the amount of *interference* between the robots, as a tool for choosing an appropriate approach (see more on interference below).

[1] Human workers used to walk 20 km a day to pick ordered items in shelves in such warehouses, and labor was in short supply.



Collision avoidance and resolution time (AVOID time)
Foraging program execution without interference (PROGRAM time)
Interval
Start of run
Collision
Collision
Collision
Collision
End of run
...
Time

FIGURE 1
Single robot's timeline.

A third independent direction attempt to shorten the time spent on avoidance so as to free up time for program. The most direct approach here is to improve the collision avoidance algorithm.

Not all collision-avoidance algorithms are a good fit for swarm foraging. For example, algorithms in the *reciprocal velocity obstacle* (RVO) class of navigation methods (Snape et al., 2011; Guy et al., 2010; van den Berg et al., 2008) plan ahead based on the space of admissible relative velocities to nearby obstacles and robots. They therefore assume knowledge of others' velocities and shapes—a challenging task in many cases (e.g., when using vision only). To guarantee collision-free paths (within a specific horizon), the optimal reciprocal collision avoidance (ORCA) algorithm (van den Berg et al., 2011) also requires that all agents use ORCA, which fails when humans are involved. In contrast, the *passively safe partial motion planning* (PassPMP) algorithm (Bouraine et al., 2014) provides some guarantees on collision safety, without making such assumptions. This comes at a cost of non-trivial computation of predicted trajectories.

A related approach, presented by Danassis and Faltings (2018), is called *CA*^{3}*NONY* and intended for domains where an optimal behavior will be to anti-coordinate[2], i.e., that each agent must choose an action that differs from other agents' actions in order for the outcome to be optimal. Here, agents are being *courteous*: If an agent collides with another agent, i.e., chooses the same resource at the same context, it backs off from this choice with a constant probability. In addition to this social convention, the agents maintain a distributed bookkeeping scheme that prevents them from monopolizing resources, causing each agent to choose only one resource for one context. Although this algorithm guarantees optimal behavior, it assumes that the reward is shared between all agents, an assumption that breaks with no communications between the agents.

Other algorithms appear to work relatively well in swarm robots, in practice. However, these offer no guarantees at all. These are essentially reactive algorithms that respond to a collision, with no or very little planning with respect to the task goal of the robot or the group, i.e., these are necessarily *myopic* algorithms. On the other hand, such algorithms are extremely simple to implement and use (both in practice and from a computational perspective) and are generally task-independent (because they do not use information about the goals of the task).

We use several such myopic algorithms in this research. The *dynamic window* algorithm (Fox et al., 1997) is a coordination method that uses limited planning in the space of admissible velocities. This method is capable of making decisions based not only on external constraints like obstacles and other robots but also on internal constraints like maximal velocity and acceleration. We use a dynamic window variant as one of the algorithms in the experiments. One reactive algorithm is the *noise* algorithm presented by Balch and Arkin (1998). Given a collision, the robot repels itself away from the collision, with some directional noise. Rosenfeld et al. (2008) described the *repel* method. As the name suggests, once a robot collides with another robot, it repels itself backward for an arbitrary time interval or distance.

More sophisticated algorithms introduce stochasticity into the decision-making. A reactive algorithm named *aggression* was described by Vaughan et al. (2000) and improved by Zuluaga and Vaughan (2005). When robots use this coordination method, the robot with the highest "aggression factor" gets the right of way, while the other backs off.

It is now understood that while each method is effective in some settings, no method is always effective (Rybski et al., 1998; Rosenfeld et al., 2008; Erusalimchik and Kaminka, 2008; Douchan and Kaminka, 2016). The results in foraging show that the swarm-wide utility—the number of collected items of a specific

[2] The definition of coordination in Danassis and Faltings (2018) differs from the definition of coordination in our work. We define coordination as the need to take an action due to an interaction between agents. They define coordination as a *consensus*: where agents need to choose the same action in order to achieve optimal results.

coordination method—depends on the density of the system. For all methods, the system-wide utility declines once some density is reached. However, the density in which this occurs differs from one method to the next. Certainly, some methods do better than others, but none are superior to others in all densities.

The performance of the swarm as the group grows in size mimics the *law of marginal returns* in economics: adding more robots does not necessarily increase productivity. Goldberg and Mataric (1997) attempted to capture the cause for this, by defining *interference*, a global signal which varies in the working space of the system denoting how much robots interfere with each other, e.g., due to lack of coordination. Later, Lerman and Galstyan (2002) drew a theoretical connection between interference and task performance. This suggests that if robots act based on the global interference signal, they might improve productivity. The problem is that in practice, this signal cannot be individually computed (as it involves internal measurements from each robot) or made public without communications.

2.3 Learning to coordinate in handling collisions

Inspired by the study of interference and attempting to find a way to use it despite not having access to the global (group-wide) information required, Rosenfeld et al. (2008) showed that in foraging with a fixed group size, areas of high density of robots correlate negatively with group performance. In addition, the higher the cost robots invest on coordination methods the less the group performance will be. They defined the *likelihood of collision* around a robot as the ratio between the area of a circle of fixed radius around it and the total area robots take inside this circle. They represented the cost of coordination by the *combined coordination cost* (CCC), a weighted average of all coordination costs of a robot like time and fuel. They showed a strong negative correlation between the CCC and group performance for a fixed group size.

Rosenfeld et al. (2008) then proposed an offline adaptive algorithm for the problem of multi-robot coordination, based on their CCC measure. This algorithm arbitrates between a set of coordination methods by using methods with larger CCC when the likelihood of collision is high and methods with lower CCC when the likelihood of collision is low. It does so by sorting the set of coordination methods from the one with lowest to the one with highest CCC and sets thresholds based on the likelihood of collision to determine what method to choose. The adaptation was done by tuning the aforementioned threshold. They used two separate variants for this adaptation: hill climbing and gradient learning; each one of them tunes the thresholds differently based on the group performance. The CCC measure was not developed theoretically, despite the empirical success of using it as the basis for learning (offline).

More generally, there is much work on utilizing learning to improve multi-robot (and multi-agent) coordination, mostly focusing on *multi-agent reinforcement learning*, which is often used in the context of planning and decision-making. Indeed, this is the approach we take in this paper: to improve coordination by using learning to adjust which reactive coordination method is to be used in each conflict. We only describe it here in brief and refer the reader to previous studies (Kapetanakis and Kudenko, 2002; Hernandez-Leal et al., 2019; Kober et al., 2013; Zhang et al., 2021; Kuckling, 2023; Fatima et al., 2024) for a deeper explanation. There are several investigations that are closely related to this approach, which we describe below in detail.

Claus and Boutilier (1998) showed different variations of RL techniques in multi-agent domains and the difficulties that arise when using them. They divide learners into two different types: independent learners (IL) and joint-action learners (JAL). ILs learn actions with no knowledge about the actions of other agents, while JALs learn with knowledge about the actions of all other agents. To ground RL use in multi-agent systems, Claus and Boutilier discussed learning in the context of game theory models. They showed that even simple RL algorithms lead to non-intuitive results, depending on the settings of the game. In particular, they examined both IL and JAL agents in several identical-interest matrix games (where, in every action profile, every agent gets the same utility). For both ILs and JALs, they showed that the agents converge to a Nash equilibrium, which is sub-optimal in terms of welfare. They also show that different learning parameters such as the learning rate or exploration rate can make the system converge to different equilibrium points. As we are interested in maximizing the global utility (the group goal), this is a serious challenge, which has been undertaken in many investigations.

Kaminka et al. (2010) attempted to utilize an online adaptation mechanism for the same purpose. They introduced the first version of the reward function we discuss in this paper, the EI. This basic version measured the ratio between the resources (including time) spent in collision avoidance (avoidance time, in Figure 1) and the total resources spent in a single interval between collisions (sum of the avoidance and program time in the interval). Using a stateless Q-learning variant (Claus and Boutilier, 1998) with this basic EI as a reward and using a large learning rate so as to adapt quickly to changing conditions, they demonstrated successful foraging in many different settings. Their experiments revealed that while the robots did not converge to a specific policy (individually, i.e., robots often changed their selected action), their choices are heterogeneous and often lead to improved results (e.g., compared homogeneous policies or random mixed choices).

Despite the empirical success of the EI measurement using reinforcement learning, it comes with no guarantees. Indeed, our research work began by applying the framework to the pick ordering domain, which turned out to be not at all trivial or necessarily successful (Douchan and Kaminka, 2016). We therefore sought to ground the EI in theory and, along the way, developed a more general EI reward function; the EI introduced by Kaminka et al. (2010) is strictly a special case. The general EI introduced in this paper provides guarantees up to explicit assumptions, as well as a thorough discussion of approximation methods that can be used in practice and are motivated by the theory. It adds a component by which the individual agent estimates its effect on the swarm, allowing the general EI reward to align the individual and swarm utilities.

Godoy et al. (2015) described a method using reinforcement learning techniques with ORCA (van den Berg et al., 2011). It improves on either using only ORCA or only reinforcement learning. They presented the ALAN framework that uses a reinforcement signal composed of two factors: a goal-oriented and politeness factor. The goal-oriented factor is based on the direction cosine of the velocity vector of the robot and the displacement vector of the goal from the robot. The politeness factor is based on the vector cosine between the preferred velocity vector, and the vector ORCA will output in the current robot's situation. The final reinforcement signal for the ALAN framework is a weighted sum of the goal-oriented factor and politeness factor. This work has both similarities and dissimilarities to our work. In a similar manner to our work, this work uses reinforcement learning in order to choose the best action in any given time. However, ALAN can only choose between alternatives within ORCA and does not provide guarantees on performance, as we do here.

Wolpert and Tumer (1999) described the COIN framework, which models multi-agent systems where agents work to maximize global utility but with little or no communications between them. They show that if agents can estimate the *wonderful life utility*—how the agent's actions (or lack thereof) impact global utility—then it is possible to use reinforcement learning to improve global utility in a guaranteed manner, in various multi-agent domains (Tumer et al., 2002; Agogino and Tumer, 2008; Wolpert et al., 1999). However, this relies on knowing the global utility and/or the value (payoff) of others' actions. In practice, this is often not possible, so approximations are made (Turner et al., 2008; Devlin et al., 2014). In an earlier version of the work reported here (Douchan et al., 2019), we built on the COIN work by showing how to approximate the wonderful life utility in practice, in multi-robot swarm settings. We also briefly discussed a connection to game theory. Here, we extend these results and focus on the heterogeneous nature of the resulting optimal swarms. In addition, necessarily, because we work with physical robots, and given the focus on using timing information, the approximations we take here are different from those made elsewhere; they are discussed in context in the next sections.

3 Swarming in (game) theory

We begin in Section 3.1 by introducing an abstract game-theoretic model of multi-robot tasks carried out by a swarm of robots. We then make incremental modifications to this abstract model, to bring it closer to the reality of physical interacting robots, when the robots cannot communicate (Sections 3.2, 3.3). Finally, in Section 3.4, we address the challenge of learning optimal actions according to the game-theoretic model we introduced. For the benefit of the reader, we include a nomenclature of the symbols in Appendix A.

3.1 Swarm tasks as extensive-form fully cooperative games

When considering the task multiple robots (each engaging in its own coordination method arbitration), we follow Kaminka et al. (2010) in representing the task as an extensive form game between n robots. The extensive form game represents every possible outcome as a function of the sequence of parallel coordination actions taken by all robots in every collision during the run. In this context, the outcome is the utility of each of the robots in the allotted time.

The root node of the game tree represents the first collision. Given that there are n robots, the first n layers of the game tree will each represent a robot and its possible actions in the first collision. This is because we focus on non-communicating coordination methods, and thus, we treat each collision as having no information about the actions and utilities selected and gained by other robots.

The actions independently taken by players are coordination methods. The gains (payoffs) from taking them and the costs that they entail differ between robots and between collisions but are theoretically accounted for. Each action takes time.

The next m layers represent the second collision in the same manner, and the pattern repeats until a terminal node is reached—when the time for the task is done. A terminal node represents the end of the game (task) and holds the sum of the utility of each player. Since different actions can yield different time intervals between collisions, terminal nodes can each be of different depths depending on the sequence of collisions (and associated joint actions chosen) during the game. Each such sequence is represented as a path in the game tree.

Each terminal node will hold a vector of numerical values representing the utilities of each robot in the system. As this is a cooperative task, we are interested in the sum of these utilities—in foraging this would translate to the number of items collected by all swarm members, together.

A two-player two-action example of such an extensive-form game is shown in Figure 2. It shows several paths from the root node to the terminal nodes.

3.2 From extensive-form game to normal-form games

The extensive-form model of a task run represents every possible outcome of the task run. This is only of theoretical value as no robot—or their designers—can predict the outcome of future collisions, or their timing, or their impact on global payoffs. In reality, robots only know their history of previous collisions, and the immediately imminent collision. Indeed, in swarm settings, robots cannot know of the other robots' choices (which theoretically affects their own), and thus, even this information is hidden from them.

In order for robots to make decisions based only the history and current collision, we must draw a connection between the global final utility (payoff) theoretically reached using the extensive-form game and the sequence of collisions in which the robots make collision-resolution choices. Robots may then rely on signals that are obtained during a joint collision.

To do this, we take an intermediate step and show how the extensive-form game can be expressed as a sequence of normal-form games, each representing a single joint collision. We define the following symbols (see also nomenclature in Table A1):

- s_i^j: robot i's action at the jth collision. s^j: joint action at collision j.
- $h_i^j = (s_i^1, s_i^2, \ldots, s_i^j)$: robot i's history of actions until the jth collision inclusive. History of all robots' actions until j inclusive: h^j.
- The *cost* incurred by robot i at the jth collision is denoted as c_i^j.
- The *gain* by robot i at the jth collision is denoted as g_i^j.
- $u_i^j = g_i^j - c_i^j$: the *utility* of robot i at the jth collision.

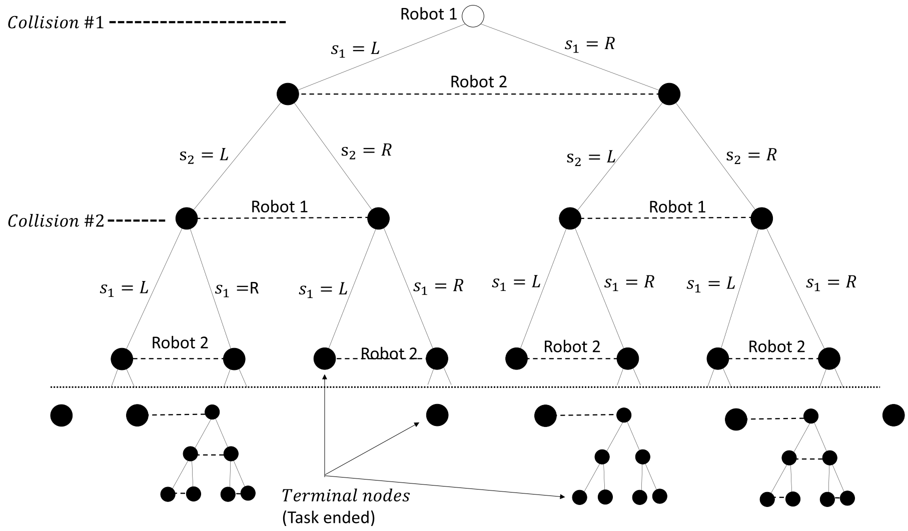


FIGURE 2
Two-player two-action task run represented as an extensive-form game for the action sets $S_1 = S_2 = \{L, R\}$. Not all terminal nodes have the same depth as different joint actions taken by the players can lead to more or less collisions.

- U: the *global utility* of the entire robot swarm during the whole run.
- C: the number of collisions during the whole run.

We start with the most general case where outcomes of a robot at the jth collision may depend on the entire history of play of all the robots up until collision j inclusive. This means that u_i^j, g_i^j, c_i^j are all functions of h^j. U now depends on the entire history of play. Given that the number of collisions for the whole task run is C, U will be a function of h^C and will be defined as the sum of utilities of every robot and every collision during the task run (Equations 1, 2).

$$U\left(h^C\right) := \sum_{i \in N}\sum_{j=1}^{C} u_i\left(h^j\right) = \sum_{i \in N}\sum_{j=1}^{C}\left(g_i\left(h^j\right) - c_i\left(h^j\right)\right). \tag{1}$$

We can look at each joint collision as a normal-form (matrix) game representing the outcomes of this collision only, rather than the whole task run. For the jth collision, the player set of this matrix is the set of robots performing the task, and the action set of each robot is its set of available coordination methods for this collision. Given the history of joint actions played up until collision j (inclusive), h^j, the payoffs of this matrix will be the sum of the utilities of the robots obtained only for the jth collision $\sum_{i \in N} u_i(h^j)$ as a function of the history of play. We call this matrix the *folded game matrix*.

We define the $\oplus$ operator between a play history and a new joint action to be the concatenation of the new joint action to the history. For $h^{j-1} = (s^1, \ldots, s^{j-1})$ and s^j, $h^j = h^{j-1} \oplus s^j = (s^1, \ldots, s^{j-1}, s^j)$. Figure 3 provides an illustration.

3.3 Global utility and folded matrices

Robots in a system have limited sensing and communication capabilities. They are unable to know the utilities of other robots, even in the same joint action. Indeed, each robot does not even know how its own action affects its own immediate utility. The only information available to a robot is from its own sensors and internal state memory.

	$s_2 = L$	$s_2 = R$
$s_1 = L$	$u_1(h^{j-1} \oplus (L, L)) + u_2(h^{j-1} \oplus (L, L))$	$u_1(h^{j-1} \oplus (L, R)) + u_2(h^{j-1} \oplus (L, R))$
$s_1 = R$	$u_1(h^{j-1} \oplus (R, L)) + u_2(h^{j-1} \oplus (R, L))$	$u_1(h^{j-1} \oplus (R, R)) + u_2(h^{j-1} \oplus (R, R))$

FIGURE 3
Example of a two-player two-action folded game matrix for the action set $S_1 = S_2 = \{L, R\}$.

In particular, the robot i can measure time. It can measure the time—denoted $A_i(s)$—it has spent in collision avoidance after having executed a collision avoidance procedure s. It can also measure the time—denoted $P_i(s)$—spent making progress on its task, undisturbed by others, once the collision is resolved.

We formally tie the avoidance and program times of the collision to the utility of the robot resulting from the collision. To do this, we assume that individual gains in avoidance time are zero (since a robot in avoidance time is handling a collision), and therefore, gains occur only in program time: $g_i(h^j) = g_i(P_i(h^j))$. We will further assume that the gains are proportional to the program time, given the history of play h^j: $g_i(h^j) = \alpha P_i(h^j)$, where α is a positive constant. We also assume that costs are constant, $c_i(h^j) = \beta(A_i(h^j) + P_i(h^j))$, where β is a positive constant. Therefore, by substituting g, c by their interval proxies using α, β, the following holds:

By definition, Equation 1

$$U\left(h^C\right) = \sum_{i \in N}\sum_{j=1}^{C}\left(g_i\left(h^j\right) - c_i\left(h^j\right)\right),$$

$$= \sum_{i \in N} \sum_{j=1}^{C} \left[\alpha P_i \left(h^j \right) - \beta \left(A_i \left(h^j \right) + P_i \left(h^j \right) \right) \right]. \tag{2}$$

3.3.1 Global utility and coordination overhead

Rosenfeld et al. (2008) empirically demonstrated that there is a strong negative correlation between coordination costs (the avoidance time in our case) and swarm performance. The more a robot, or a group of robots, spends time carrying out the task (program time) and less on coordination (avoidance time), the higher is and the higher their performance. Equation 2 formally shows this relationship.

We distinguish productive intervals (P) from coordination (collision avoidance) intervals (A) in Equation 2. Given a task run h^C (a sequence of C joint actions by the swarm members), we define the coordination overhead of the swarm is defined as follows (Equation 3):

Definition 1: The *coordination overhead (CO)* is the total amount of time the system was in avoidance time divided by the total time invested in the task run:

$$CO\left(h^C\right) \coloneqq \frac{1}{T} \sum_{i \in N} \sum_{j=1}^{C} A_i\left(h^j\right). \tag{3}$$

We show that U is a linear decreasing function of CO, i.e., minimizing CO is maximizing U. In the following, $n = |N|$, i.e., it is the number of robots.

Theorem 1: Given the assumptions on the cost and gain, U is a linear decreasing function of CO.

Proof.

$$\begin{aligned} U\left(h^C\right) &= \sum_{i \in N} \sum_{j=1}^{C} \left[u_i\left(h^j\right) \right] = \sum_{i \in N} \sum_{j=1}^{C} \left[g_i\left(h^j\right) - c_i\left(h^j\right) \right] \\ &= \sum_{i \in N} \sum_{j=1}^{C} \left[\alpha P_i\left(h^j\right) - \beta\left(A_i\left(h^j\right) + P_i\left(h^j\right) \right) \right] \\ &= \sum_{i \in N} \sum_{j=1}^{C} \alpha P_i\left(h^j\right) - \sum_{i \in N} \sum_{j=1}^{C} \beta \left[A_i\left(h^j\right) + P_i\left(h^j\right) \right] \\ &= \alpha \sum_{i \in N} \sum_{j=1}^{C} P_i\left(h^j\right) - \beta \sum_{i \in N} \sum_{j=1}^{C} \left[A_i\left(h^j\right) + P_i\left(h^j\right) \right]. \end{aligned}$$

Since T is the sum of all cycle lengths of any of the robots$'$ task run, we can write $T = \sum_{j=1}^{C} \left(A_i\left(h^j\right) + P_i\left(h^j\right) \right)$ for any robot i. Thus,

$$\begin{aligned} &= \alpha \sum_{i \in N} \sum_{j=1}^{C} P_i\left(h^j\right) - \beta \sum_{i \in N} T \qquad (T \text{ total time, identical for all robots}) \\ &= \alpha T \frac{\sum_{i \in N} \sum_{j=1}^{C} P_i\left(h^j\right)}{T} - \beta n T \qquad (n = |N| \text{ is the number of agents}) \\ &= \alpha T\left(1 - CO\left(h^C\right)\right) - \beta n T = \alpha T - \alpha T CO\left(h^C\right) - \beta n T \\ &= -\alpha T \cdot CO\left(h^C\right) + T(\alpha - n\beta). \end{aligned}$$

As α, β, T are positive constants in this setting, it follows that U is a linear decreasing function of $CO(h^C)$.

This completes the proof. □

As a result of Theorem 1, now it is possible to look at our problem as minimizing CO rather than maximizing U. However, this does not give information about how robots should choose their actions in a way that CO is minimized.

3.3.2 Coordination overhead and the folded matrices

We turn to utilizing the folded matrices as a step toward making it possible for robots to maximize U (by minimizing the swarm's CO). To do this, we re-examine the sequence of normal-form games making up the history of agent decisions.

We follow Kaminka et al. (2010); Douchan and Kaminka (2016) in making a Markovian assumption that for every collision, the outcomes of the robots' method selection depend only on the current joint action performed and not on the history of all joint actions performed. This means that the outcome of any collision, given a collision-avoidance action s, depends only on the action and not the history of previous collisions. Under this assumption, variables $h^j \in S^j$ that depend on the history of joint actions played until collision j, depend only on the joint action that was played in time j, $s^j \in S$. We therefore denote the avoidance time as $A_i(s^j)$. The same applies for P_i, g_i, c_i, u_i and U.

One consequence of this assumption is that instead of the task run being a sequence of different folded-game matrices depending on the history of play, it is now a single game matrix, which is the same for every collision in the task run. In game theory, such a sequence is termed *repeating games*. As the number of games is not known in advance, these settings are formally known as *infinite-horizon* repeating games.

Minimizing CO maximizes the global utility. Since T is the sum of all cycle length of any of the robots' task run, we can write $T = \sum_{j=1}^{C}(A_i(h^j) + P_i(h^j))$ for any robot i. Therefore, CO can also be written as

$$CO\left(h^C\right) = \sum_{i \in N} \frac{\sum_{j=1}^{C} A_i\left(s^j\right)}{\sum_{j=1}^{C} \left[A_i\left(s^j\right) + P_i\left(s^j\right) \right]}. \tag{4}$$

Given the above, it makes sense for swarm agents to attempt to *individually* increase their own $P_i()$ and minimize their own $A_i()$ by selecting appropriate individual actions. Indeed, Kaminka et al. (2010)—*predating the introduction of the coordination overhead*—proposed using the ratio of avoidance time to total cycle duration (since the last conflict) as a substitute for the robot's estimate of the swarm's utility from taking an action s. They refer to this ratio as the EI:

$$\mathrm{EI}(i, s) \coloneqq \frac{A_i(s)}{A_i(s) + P_i(s)}. \tag{5}$$

Kaminka et al. (2010) conjectured that individually minimizing EI is equivalent to maximizing the robot's utility and, thus, the swarm's utility. However, *this conjectured connection is generally incorrect*: maximizing the individual robot's *EI* may mean selecting an action that increases the costs to others, resulting in overall degraded performance for the swarm. To intuitively see why this happens, imagine some foraging robots are attempting to enter the collection area to drop foraged items, while others are attempting to leave, to search for new items. Those attempting to enter should ideally back off, allowing those inside the nest to go out. However, backing off adds to the duration of the avoidance mode and reduces from the duration of the program mode. Thus, those robots are motivated to push forward. This hinders the swarm from collecting items.

Despite its lacking, the structural similarity between the individual EI (Equation 5) and the coordination overhead in its rewritten form (Equation 4) has led us to define a related measure, $\mathbb{EI}_{tot}(s)$, the total sum of the effectiveness indices of all robots:

$$\begin{aligned}\mathbb{EI}_{tot}(s) &\coloneqq \sum_{i\in N} \mathrm{EI}(i,s)\\ &= \sum_{i\in N} \frac{A_i(s)}{A_i(s)+P_i(s)}.\end{aligned} \tag{6}$$

We draw a connection between $\mathbb{EI}_{tot}$ (Equation 6) and CO (Equations 3, 4). Let s^* be the joint action that minimizes $\mathbb{EI}_{tot}$:

$$s^* \coloneqq \arg\min_{s} \left(\mathbb{EI}_{tot}(s)\right).$$

Let the swarm play the joint action s^* repeatedly, generating the history $h^* = (s^*, s^*, \ldots, s^*)$. Then, $CO(h^*)$ will be equal to $\mathbb{EI}_{tot}(s^*)$: $CO(h^*) = \sum_{i\in N} \frac{C\cdot A_i(s^*)}{C\cdot[A_i(s^*)+P_i(s^*)]} = \sum_{i\in N} \frac{A_i(s^*)}{A_i(s^*)+P_i(s^*)} = \mathbb{EI}_{tot}(s^*)$.

Building on this, we show that for every sequence of joint actions, CO will be greater or equal to $\mathbb{EI}_{tot}(s^*)$. This means that in order to minimize CO, the system always needs to select s^* as the joint action.

Theorem 2: For any number of collisions C and histories of play h^C, $CO(h^C) \geq \mathbb{EI}_{tot}(s^*)$.

Proof.

$$\begin{aligned}CO\left(h^C\right) &= \sum_{i\in N} \frac{\sum_{j=1}^{C} A_i\left(s^j\right)}{\sum_{j=1}^{C}\left(A_i\left(s^j\right)+P_i\left(s^j\right)\right)}\\ &= \sum_{i\in N} \frac{\sum_{j=1}^{C} A_i\left(s^j\right)}{T}\\ &= \frac{1}{T}\sum_{i\in N}\sum_{j=1}^{C} A_i\left(s^j\right).\end{aligned}$$

We re-order the summations:

$$= \frac{1}{T}\sum_{j=1}^{C}\sum_{i\in N} A_i\left(s^j\right).$$

Defining $l_i(s)$ as the cycle length of robot i, given joint action s: $l_i(s) = A_i(s) + P_i(s)$, we rewrite

$$\begin{aligned}&= \frac{1}{T}\sum_{j=1}^{C}\left[l\left(s^j\right)\sum_{i\in N}\frac{A_i\left(s^j\right)}{l\left(s^j\right)}\right]\\ &= \frac{1}{T}\sum_{j=1}^{C}\left[l\left(s^j\right)\mathbb{EI}_{tot}\left(s^j\right)\right].\end{aligned}$$

Replacing s^j with the optimal joint action s^*, necessarily:

$$\begin{aligned}&\geq \frac{1}{T}\sum_{j=1}^{C}\left[l\left(s^j\right)\mathbb{EI}_{tot}(s^*)\right]\\ &= \mathbb{EI}_{tot}(s^*)\,\frac{1}{T}\sum_{j=1}^{C} l\left(s^j\right)\\ &= \mathbb{EI}_{tot}(s^*)\,\frac{1}{T}T\\ &= \mathbb{EI}_{tot}(s^*).\end{aligned}$$

This completes the proof. □

The step taken in Theorem 1 allows robots, in theory, to use measurements of time instead of global count of items picked (which in a swarm, they cannot possibly achieve). The step taken in Theorem 2 shows that under some assumption, the sequence of collisions can be treated as a repeating game with an infinite-horizon, where each stage is an identical normal-form game. Thus, determining s^*, the optimal joint action in a collision, leads to optimal results for the swarm.

However, robots cannot know s^* as it requires knowledge of the actions of other robots. We need to find a way to make the robots converge to s^* by using internal measurements only, without requiring knowledge of coordination methods selected and utilities obtained, by other robots.

3.4 Optimal joint actions

We approach the challenge by finding a potential function that turns the normal-form game into a *potential game* [Monderer and Shapley (1996)]. A potential game is a normal-form game, where, for every player i, the difference in the payoff of every unilateral deviation of player i's action s_i is related to the difference of a single potential function $\psi(s)$ mapping each joint action to a scalar. $\psi(s)$ can be seen as a global signal (not necessarily visible to the players) which depends on the joint action.

Potential games hold several important characteristics: First, they always have at least one pure-strategy Nash equilibrium. Furthermore, when players use pure strategies, an improvement in one player's individual payoff due to changing its individual action will necessarily improve the potential function, i.e., the individual payoff and potential function are *aligned*. When players choose to maximize their individual payoffs, the system will converge to a pure-strategy Nash equilibrium, which would be at least a local optimum of the potential function.

If the robots play a potential game with potential function $\mathbb{EI}_{tot}$, the swarm will converge to an optimal joint action in terms of $\mathbb{EI}_{tot}$. However, $\mathbb{EI}_{tot}$ is a global function: it is not accessible to the robots. We therefore seek a reward function for each robot that is both accessible to each robot and aligned with $\mathbb{EI}_{tot}$.

To derive a local aligned reward from $\mathbb{EI}_{tot}$, we use the formulation of *Wonderful Life Utility (WLU)* (Wolpert and Tumer, 1999; Tumer et al., 2002), later renamed the *difference reward* (Agogino and Tumer, 2008; Tumer et al., 2008; Devlin et al., 2014). Given a global function F, the difference reward for robot i using joint action s is the difference between the F value resulting from the action with i participating and the counterfactual F value when robot i is hypothetically absent. We denote the absence of robot i as the robot choosing a *null* action denoted by ϕ_i. We denote by s_{-i} the joint action of all robots excluding i. Then, $F(s_i, s_{-i})$ is the value resulting from the complete joint action (including all robots' actions), and $F(\phi_i, s_{-i})$ is the value of the swarm when robot i is absent. Then, the difference reward of F is given by $\Delta_i^F \coloneqq F(s_i, s_{-i}) - F(\phi_i, s_{-i})$. For reinforcement learning, it is a reward that is both accessible and aligned, and highly effective (Tumer et al., 2008; Arslan et al., 2007; Marden and Wierman, 2009).

Using $\mathbb{EI}_{tot}$ as the global potential function, we define the *Aligned Effectiveness Index* (AEI) as the difference reward of $\mathbb{EI}_{tot}$, i.e., $\Delta_i^{\mathbb{EI}_{tot}}(s)$:

Definition 2: Given a joint action $s = (s_i, s_{-i})$, the AEI reward of robot i is defined by

$$\mathrm{AEI}(i,s) \coloneqq \mathbb{EI}_{tot}(s_i, s_{-i}) - \mathbb{EI}_{tot}(\phi_i, s_{-i}).$$

AEI is a measurement of robot i's marginal contribution to $\mathbb{EI}_{tot}$, given the action s. If the robots individually select actions that optimize it, they will converge to a joint action that will, at least, be a local minimum of $\mathbb{EI}_{tot}$ due to the properties of potential games (Tumer et al., 2008; Marden and Wierman, 2009).

We derive a bounded closed-form expression of $\mathrm{AEI}_i(s)$:

$$\begin{aligned}
\mathrm{AEI}(i,s) &= \mathbb{EI}_{tot}(s_i, s_{-i}) - \mathbb{EI}_{tot}(\phi_i, s_{-i}), && (\text{From Def. 2})\\
&= \mathbb{EI}_{tot}(s) - \mathbb{EI}_{tot}(\phi_i, s_{-i}), && (\text{Notation:} s = (s_i, s_{-i}))\\
&= \sum_{j\in N} \frac{A_j(s)}{A_j(s)+P_j(s)} - \sum_{j\in N} \frac{A_j(\phi_i,s_{-i})}{A_j(\phi_i,s_{-i})+P_j(\phi_i,s_{-i})} && (\text{From Eq. 6})\\
&= \frac{A_i(s)}{A_i(s)+P_i(s)} + \sum_{j\in N\setminus\{i\}} \frac{A_j(s)}{A_j(s)+P_j(s)} - \sum_{j\in N} \frac{A_j(\phi_i,s_{-i})}{A_j(\phi_i,s_{-i})+P_j(\phi_i,s_{-i})}.
\end{aligned}$$

We observe that one of the components here is actually EI Kaminka et al. (2010) (see Equation 6).

$$= \mathrm{EI}(i,s) + \sum_{j\in N\setminus\{i\}} \frac{A_j(s)}{A_j(s)+P_j(s)} - \sum_{j\in N} \frac{A_j(\phi_i,s_{-i})}{A_j(\phi_i,s_{-i})+P_j(\phi_i,s_{-i})} \; (\text{From Eq. 5}).$$

Once again, let $l_i(s)$ be the cycle length of robot i given a joint action s: $l_i(s) = A_i(s) + P_i(s)$:

$$\begin{aligned}
&= \mathrm{EI}(i,s) + \sum_{j\in N\setminus\{i\}} \frac{A_j(s)}{l_j(s)} - \sum_{j\in N} \frac{A_j(\phi_i,s_{-i})}{l_j(\phi_i,s_{-i})}\\
&= \mathrm{EI}(i,s) + \sum_{j\in N\setminus\{i\}} \frac{A_j(s)}{l_j(s)} - \sum_{j\in N\setminus\{i\}} \frac{A_j(\phi_i,s_{-i})}{l_j(\phi_i,s_{-i})} - \frac{A_i(\phi_i,s_{-i})}{l_i(\phi_i,s_{-i})}.
\end{aligned}$$

Multiplying the first sum by $\frac{l_i(s)}{l_i(s)} = 1$ and the second sum by $\frac{l_i(\phi_i,s_{-i})}{l_i(\phi_i,s_{-i})}$ yields

$$= \mathrm{EI}(i,s) + \sum_{j\in N\setminus\{i\}} \frac{A_j(s)\frac{l_i(s)}{l_j(s)}}{l_i(s)} - \sum_{j\in N\setminus\{i\}} \frac{A_j(\phi_i,s_{-i})\frac{l_i(\phi_i,s_{-i})}{l_j(\phi_i,s_{-i})}}{l_i(\phi_i,s_{-i})} - \frac{A_i(\phi_i,s_{-i})}{l_i(\phi_i,s_{-i})}.$$

We remind the reader that we had assumed earlier that collisions are synchronous and involve all agents. This means that the cycle length depends only on the joint action selected and not on any specific robot. Therefore, for all pairs of robots $i, j \in N$ and all joint actions s, $\frac{l_i(s)}{l_j(s)} = \frac{l_i(\phi_i,s_{-i})}{l_j(\phi_i,s_{-i})} = 1$. This yields

$$\mathrm{AEI}(i,s) = \mathrm{EI}(i,s) + \sum_{j\in N\setminus\{i\}} \frac{A_j(s)}{l_i(s)} - \sum_{j\in N\setminus\{i\}} \frac{A_j(\phi_i,s_{-i})}{l_i(\phi_i,s_{-i})} - \frac{A_i(\phi_i,s_{-i})}{l_i(\phi_i,s_{-i})}. \tag{7}$$

We note that all the denominators $l_i()$ are durations measured or estimated (as counterfactuals) by robot i. However, to compute its reward, robot i must seemingly require knowledge of the duration $A_j(s)$, which it does not know. We, therefore, seek a simplification of the above. This is done in two steps. First, we bound the value of AEI from below and above (Theorem 3 below). Then, we argue that as the number of robots increases, the bounds become tight, and thus, a simpler formula emerges.

Theorem 3:

$$\mathrm{EI}(i,s) + \frac{A_i^{\phi}(s)}{A_i(\phi_i,s_{-i}) + P_i(\phi_i,s_{-i})} \geq \mathrm{AEI}(i,s) \geq \mathrm{EI}(i,s) + \frac{A_i^{\phi}(s)}{A_i(s)+P_i(s)} - 1,$$

where $A_i^{\phi}(s) = \sum_{j\in N\setminus\{i\}}(A_j(s_i,s_{-i}) - A_j(\phi_i,s_{-i}))$.

Proof. Note that terms using ϕ are *counterfactuals*: they are hypothetical values, modeling the effects of robot i on others, when it is hypothetically logically absent from the collision, and unable to contribute. Two potential ways to model this assumption are (i) either that robot i was removed from the set N for the collision (Case 1 below), or that it is present but remained in collision avoidance during the entire cycle length and thus did not contribute (Case 2). In both cases, we begin with the closed form formulation for AEI(i,s), as found in Equation 7.

Case 1: Robot i hypothetically removed from N in the collision.

$$\frac{A_i(\phi_i,s_{-i})}{l_i(\phi_i,s_{-i})} = 0,$$

and we may also assume that its absence shortens the swarm's avoidance period (the collision resolution was shorter), and thus,

$$l_i(\phi_i,s_{-i}) \leq l_i(s).$$

Therefore, continuing from step 7 above,

$$\begin{aligned}
\mathrm{AEI}(i,s) &= \mathrm{EI}(i,s) + \sum_{j\in N\setminus\{i\}} \frac{A_j(s)}{l_i(s)} - \sum_{j\in N\setminus\{i\}} \frac{A_j(\phi_i,s_{-i})}{l_i(\phi_i,s_{-i})} - \frac{A_i(\phi_i,s_{-i})}{l_i(\phi_i,s_{-i})}\\
&= \mathrm{EI}(i,s) + \sum_{j\in N\setminus\{i\}} \frac{A_j(s)}{l_i(s)} - \sum_{j\in N\setminus\{i\}} \frac{A_j(\phi_i,s_{-i})}{l_i(\phi_i,s_{-i})}\\
&\leq \mathrm{EI}(i,s) + \sum_{j\in N\setminus\{i\}} \frac{A_j(s)}{l_i(\phi_i,s_{-i})} - \sum_{j\in N\setminus\{i\}} \frac{A_j(\phi_i,s_{-i})}{l_i(\phi_i,s_{-i})}.
\end{aligned}$$

Adding using the common denominator and using $A_i^{\phi}(s)$ to denote $\sum_{j\in N\setminus\{i\}}\left[A_j(s) - A_j(\phi_i,s_{-i})\right]$,

$$\begin{aligned}
&= \mathrm{EI}(i,s) + \frac{A_i^{\phi}(s)}{l_i(\phi_i,s_{-i})}\\
&= \mathrm{EI}(i,s) + \frac{A_i^{\phi}(s)}{A_i(\phi_i,s_{-i}) + P_i(\phi_i,s_{-i})} \; (\text{Transforming back from} l_i(\phi_i,s_{-i})).
\end{aligned}$$

This yields the left-hand inequality of the theorem. A_i^{ϕ} is the counterfactual change in the total avoidance time of the swarm when robot i is hypothetically not involved.

$$\mathrm{EI}(i,s) + \frac{A_i^{\phi}(s)}{A_i(\phi_i,s_{-i}) + P_i(\phi_i,s_{-i})} \geq \mathrm{AEI}(i,s). \tag{8}$$

Case 2: Robot i's absence—inability to contribute—is modeled as being in collision avoidance for the entire duration of the cycle. In this case, $A_i(\phi_i,s_{-i}) = l_i(\phi_i,s_{-i})$, and therefore,

$$\frac{A_i(\phi_i,s_{-i})}{l_i(\phi_i,s_{-i})} = 1.$$

As the agent is still hypothetically present, the counterfactual cycle length does not change:

$$l_i(\phi_i, s_{-i}) = l_i(s).$$

Then, continuing from Equation 7 yields

$$\begin{aligned}
\mathrm{AEI}(i,s) &= \mathrm{EI}(i,s) + \sum_{j\in N\setminus\{i\}} \frac{A_j(s)}{l_i(s)} - \sum_{j\in N\setminus\{i\}} \frac{A_j(\phi_i,s_{-i})}{l_i(\phi_i,s_{-i})} - \frac{A_i(\phi_i,s_{-i})}{l_i(\phi_i,s_{-i})} \\
&\geq \mathrm{EI}(i,s) + \sum_{j\in N\setminus\{i\}} \frac{A_j(s)}{l_i(s)} - \sum_{j\in N\setminus\{i\}} \frac{A_j(\phi_i,s_{-i})}{l_i(s)} - 1 \\
&= \mathrm{EI}(i,s) + \sum_{j\in N\setminus\{i\}} \frac{A_j(s)}{l_i(s)} - \sum_{j\in N\setminus\{i\}} \frac{A_j(\phi_i,s_{-i})}{l_i(s)} - 1 \\
&= \mathrm{EI}(i,s) + \frac{\sum_{j\in N\setminus\{i\}} \left[A_j(s) - A_j(\phi_i,s_{-i})\right]}{l_i(s)} - 1 \\
&= \mathrm{EI}(i,s) + \frac{A_i^{\phi}(s)}{l_i(s)} - 1 \qquad A_i^{\phi}(s)\,\text{as above.} \\
&= \mathrm{EI}(i,s) + \frac{A_i^{\phi}(s)}{A_i(s)+P_i(s)} - 1\,(\text{Transforming back from} l_i(s)).
\end{aligned}$$

This yields the right-hand inequality of the theorem. Putting it together with the left-hand inequality (Equation 8 above) yields

$$\mathrm{EI}(i,s) + \frac{A_i^{\phi}(s)}{A_i(\phi_i,s_{-i}) + P_i(\phi_i,s_{-i})} \geq \mathrm{AEI}(i,s) \geq \mathrm{EI}(i,s) + \frac{A_i^{\phi}(s)}{A_i(s)+P_i(s)} - 1,$$

thus completing the proof. □

We make the following observation with respect to the above derivation of bounds on $\mathrm{AEI}(i,s)$ in Theorem 3. As the swarm tends to grow in size ($|N|$), the difference between the two bounds will tend towards 1 as the counterfactual removal of robot i from the collision will not affect the cycle length, under the assumption of synchronous collisions.

Formally, we conjecture that

$$\lim_{|N|\to\infty} \left[A_i(\phi_i,s_{-i}) + P_i(\phi_i,s_{-i})\right] - \left[A_i(s)+P_i(s)\right] = 0.$$

Proving this conjecture depends on a formal model of the counterfactual removal of a robot from a swarm collision, which is outside the scope of this paper. Lacking such a model, we use $[A_i(s)+P_i(s)]$ as the cycle length for $[A_i(\phi_i,s_{-i}) + P_i(\phi_i,s_{-i})]$. This implies

$$\mathrm{EI}(i,s) + \frac{A_i^{\phi}(s)}{A_i(s)+P_i(s)} \geq \mathrm{AEI}(i,s) \geq \mathrm{EI}(i,s) + \frac{A_i^{\phi}(s)}{A_i(s)+P_i(s)} - 1. \quad (9)$$

The goal, of course, is for each robot i to minimize $\mathrm{AEI}(i,s)$ (henceforth, AEI for short) as it is an *aligned* reward function, to be used in a reinforcement learning algorithm. Robots minimizing it will necessarily minimize also the swarm's $\mathbb{EI}_{tot}$ and, thus, the swarm's *CO* (Theorem 2). This will maximize the swarm's utility, as shown in Theorem 1. The assumptions made in the development of the model are summarized in Table 1. We remind the reader that a nomenclature is given in Table A1. In practice, to minimize $\mathrm{AEI}(i,s)$, the robot can attempt to minimize $A_i(s)$ and/or improve $P_i(s)$. Minimizing the counterfactual $A_i^{\phi}(s)$ requires the robots to estimate the impact of the agent on others, as detailed in the next section.

4 Swarming in practice, through learning

We now turn to using the derived reward AEI in practice. Having no ability by the robot to estimate the cycle length when it is hypothetically absent, there are several gaps between the theory and practice: (i) computing $\mathrm{AEI}(i,s)$ requires knowledge about other robots' measurements (A_i^{ϕ}); (ii) collisions in practice are not necessarily synchronous, or even mutual; and finally, (iii) avoidance and program times (A, P) vary for the same method, breaking the Markov assumption. These gaps are discussed below.

4.1 Approximating AEI(i,s)

This practical approximation of $\mathrm{AEI}(i,s)$ in Equation 11 is composed of three elements: A_i, P_i (which are internal to the robot, thus known) and A_i^{ϕ}. The latter requires the robot to know the avoidance times of other robots and predict their change when i is hypothetically absent. This is impractical as the effects of a robot on other robots often impossible to perceive accurately by swarm robots. It is, therefore, necessary to use an estimate $\widehat{A_i^{\phi}}(s)$ instead, yielding

$$\widehat{\mathrm{AEI}(i,s)} \approx \mathrm{EI}(i,s) + \frac{\widehat{A_i^{\phi}}(s)}{A_i(s)+P_i(s)},$$

where $\widehat{A_i^{\phi}}(s) \approx \sum_{j\in N\setminus\{i\}} [A_j(s_i,s_{-i}) - A_j(\phi_i,s_{-i})]$.

As a first step, we impose a structure on the approximation, setting $\widehat{A_i^{\phi}}(s) \coloneqq n_a \cdot A_0$, where n_a is the number of robots affected by this robot and A_0 is an approximation of how much avoidance time was added or removed to each robot due to the presence of robot i. One way of measuring n_a is by the number of robots in the vicinity of the robot i as the collision occurs.

Next, we propose a number of potential values for A_0. These will be evaluated empirically (see Section 5 for results). Three immediate estimators are

- $A_0 = 0$. Setting $A_0 = 0$ yields $\widehat{\mathrm{AEI}(i,s)} = \mathrm{EI}(i,s)$, demonstrating that EI is a special case of AEI, where the avoidance times of other robots are unaffected.
- *Same for all*, $A_0 = A_i(s)$. This assumes each robot's change to avoidance time is worsened, by $A_i(s)$.
- *Average over time*, $A_0 = \frac{1}{C}\sum_{j=1}^{C} A_i(s^j)$. The addition in avoidance time to other robots is this robot's average avoidance time measured in its history, for any action.

The last estimate raises the opportunity to utilize the robot's own experience with the specific action selected as the basis for estimating the effect of the collision on others. Given a history of play h^C and a joint action $s \in S$, we define $R(s) \subseteq \{1,\ldots,C\}$ as the subset of collision indices where joint action s was played. In the same manner, we define $R(s_i)$ as the subset of collision indices where the robot i chose individual action $s_i \in S_i$, regardless of the actions chosen by other robots. Using this notation, we additionally propose the following possible approximations for A_0:

TABLE 1 Assumptions made in the development of the theoretical model and the motivation for creating them.

Assumption	Motivation
Gains in avoidance time are zero	Robots cannot directly contribute to the task when they focus on conflict resolution and avoid collisions
Gains are proportional to program time	The more a robot works uninterrupted, the higher its gains will be; assumption for theoretical derivation
Costs are proportional to time	Robots spend resources (e.g., power) when operating. Longer operations lead to more spending; assumption for theoretical derivation
Outcomes of robots' actions do not depend on the history of method choices	Without learning, the success of collision avoidance in past collisions does not impact its success in the current collision; for theoretical derivation
Cycle length is the same for all robots for a joint action	The theoretical model is synchronous, for all N agents
When a robot is hypothetically absent, its gains are zero	By definition, it cannot contribute

- *Average over actions*, $A_0 = \frac{1}{|S_i|}\sum_{s' \in S_i} \frac{1}{|R(s')|}\sum_{j \in R(s')} A_i(s^j)$. The addition in avoidance time to other robots is by measuring this robot's average avoidance time for each type of method it selected $s' \in S_i$ and then averaging over those averages.
- *Minimum over actions*, $A_0 = \min_{s' \in S_i}(\frac{1}{|R(s')|}\sum_{j \in R(s')} A_i(s^j))$. The addition in avoidance time to other robots is by finding the individual action $s' \in S_i$ that has the lowest average avoidance time.
- *Maximum over actions*,$A_0 = \max_{s' \in S_i}(\frac{1}{|R(s')|}\sum_{j \in R(s')} A_i(s^j))$. The addition in avoidance time to other robots is by finding the individual action $s' \in S_i$ that has the highest average avoidance time.

4.2 Dealing with asynchronous and non-mutual collisions

An important assumption made in the derivation of the theoretical model is that collisions are synchronous to the swarm: all robots are assumed to be involved in every collision. In reality, as swarms grow in size, collisions between robots are asynchronous and may even be non-mutual (some robots physically involved in a collision may not recognize the collision state).

As it turns out, the effects of breaking this assumption in practice are mild. First, when a collision occurs and a robot cannot recognize it, there is nothing this robot can do but continue in its task, in which case it will not learn from the collision. This is compatible with the expectation that if the robot does not recognize the collision, then its effect on it is negligible. If, however, it does recognize a collision, its learning from it depends only on its own estimates of $\widehat{\text{AEI}}$, which do not require any cooperation from the other robot. Thus once again, we expect the learning robot to be able to learn effectively from the collision.

A potential complication in practice may occur, when a robot taking a collision-resolution action may find itself colliding again with the same or other robots. Once again, however, this is addressed easily. Compatibility with the theoretical model is maintained in such cases by preempting the first collision (essentially, treating the entire cycle leading from the first collision to the new collision as a period of collision avoidance, with A_i being set to the entire cycle duration). Then, a new collision is declared, with the robot having the opportunity to once again choose a coordination action (s) and learn from its application.

4.3 Varying avoidance and program times

An assumption made in theory is that the outcome of a collision, given a joint action selected, remains the same. However, in practice, this assumption breaks from the point of view of the learning robot. First, the robot does not know the joint action played but only its own individual action, which is only a component in the joint action. Thus, as it chooses the same individual action, it may measure different avoidance and program durations, due to other robots varying their own individual actions, synthesizing different joint actions without its knowledge. Second, the cycle length may vary even for the same joint action due to latent environment variables, which states that the robot is unable to sense directly.

To address this, we propose to use an averaging procedure on A_i,P_i and A_i^{ϕ} and then calculate a $\widehat{\text{AEI}}$ approximation, which is averaging over a number of collisions $\widehat{\text{AEI}} = \frac{\overline{A_i}+\overline{A_i}^{\phi}}{\overline{A_i}+\overline{P_i}}$. This can cause inaccuracies in learning because the cycle length $A_i + P_i$ is a real-valued signal in continuous time while sampling of A_i,P_i and A_i^{ϕ} is discrete.

We treat the learning problem as reinforcement learning in *semi-Markov decision processes (SMDPs)* (Bradtke and Duff, 1994), rather than discrete MDPs. We use the SMDPs to represent discrete sampling of a continuous-time reward and also introduce a Q-Learning variant for SMDPs, called the *continuous-time Q-Learning*. It differs from Q-Learning in the update step: first, the learning rate α is now a function of interval length: the longer the interval, the closer it will be to 1, thus giving more weight to cycles with longer intervals. Second, A_i,P_i and A_i^{ϕ} are now also scaled, according to the interval length. Algorithm 1 shows the update step. Note that due to the game-theory conventions, s denotes actions, not states, while we use x here to denote the state perceived by the robot. We experiment with this algorithm in comparison with the familiar Q-learning (see next section).

```
procedure CTQL-UPDATE($A_i, P_i, A_o, \tau, \gamma, x_i, x'_i, s_i$)
    $\alpha \leftarrow 1 - e^{-\frac{\Delta t}{\tau}}$
    $A'_i \leftarrow (1 - e^{-\frac{A_i}{\tau}})$
    $P'_i \leftarrow e^{-\frac{A_i}{\tau}}(1 - e^{-\frac{P_i}{\tau}}) \cdot P_i$
    $A_i^{\phi'} \leftarrow (1 - e^{-\frac{A_i^{\phi'}}{\tau}}) \cdot A_o$
    $Q(x_i, s_i) \leftarrow (1 - \alpha)Q(x_i, s_i) + \alpha(-\frac{A'_i + A_i^{\phi'}}{A'_i + P'_i} + \gamma \cdot max_{s'}(Q(x'_i, s')))$
end procedure
```

Algorithm 1. Continuous-time Q-Learning.

5 Experiments

We report below on experiments that evaluate swarms utilizing the reinforcement learning using the AEI reward function. The results show that these swarms perform *better*, and more so, that these improvements come from the learning, leading to specialization in the robots: they become *heterogeneous* in their reactions to collisions.

Section 5.1 explains the experiment environments (simulation and robots). Then, we present results from experiments utilizing adaption (Section 5.2, and from experiments using learning (Section 5.3). In all sections, we emphasize the role of heterogeneity.

5.1 Experimentation environments

We conducted experiments in two environments: the *Alphabet Soup* order picking simulator (Hazard and Wurman, 2006) created by Kiva Systems engineers, and the Krembot swarm robots, built by Robotican[3]. Videos showing the physical in simulated robots and an overview of the evolution of the EI reward are available online on the project's web page[4].

The Alphabet Soup simulator simulates 2D continuous-area order picking by considering the items as letters and orders (combinations of items) as words. Several *word stations* are positioned in the area, each with a list of words to be composed. *Buckets* which contain letters, *letter stations* that are used to re-fill buckets with letters and robots. The robots have three main tasks: the first is to take a bucket to a word station in order to put one letter in this station. The second is to return a bucket to its original position, and the third is to take a bucket to a letter station. Figure 4 shows a screenshot of the simulator in action.

The simulator comes with a centralized task allocation mechanism, which we do not modify. The original collision avoidance mechanism in place is run individually by each robot. It is a reactive heuristic which is a combination of dynamic window (moving towards most vacant direction) and waiting in place for a random amount of time. This mechanism was replaced by an algorithm-selection mechanism, which can choose between various reactive collision-avoidance algorithms, including the original. This choice would be governed by a learning algorithm (as described above) or a different method.

The main measurement of performance for this simulator is the amount of letters placed in word stations in a given amount of time. Unless stated otherwise, each simulation is 10 min long with the last 30 s used for measuring performance and other statistics.

The Krembot robots were used in a variant of multi-robot foraging, where the objective of the robots is to find as many items in a given time. They have relatively limited sensing and processing capabilities. They are cylindrical-shaped robots with a height of 10.5 cm and a diameter of 6.5 cm. Despite their limited sensing capabilities, those robots can detect collisions and also distinguish between a robot and a static object.

The behavior of the robot was controlled by three behavioral states and a few transitions, triggered by specific perceived events. The three states are as follows:

- *Wander*: Search for a station by randomly wandering over the field. Whenever the robot is in this state, its LED light will be magenta (both red and blue simultaneously).
- *Go to homebase*: Go to the homebase to retrieve the item after a station was found. When the robot is in this state, its LED light will be blue.
- *Resolve conflict*: The robot enters this state when it detects an imminent collision with another robot (not a static obstacle). In this state, the robot learns and chooses a reactive coordination method. When the robot is in this state, its LED light will be red.

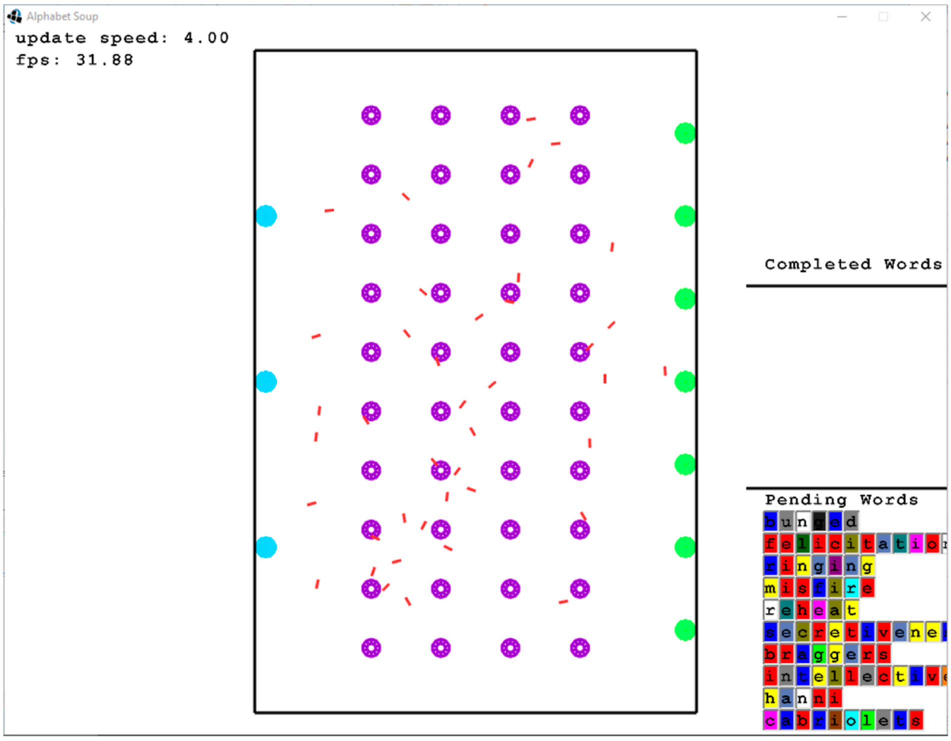
If a robot detects an imminent collision with a static obstacle, it executes a fixed behavior, unlike with a robot where it executes a coordination method by reactive method arbitration. For each of the three states, there are several transitions from it to other states:

- *Wander* → *Go to homebase*: The robot found a station.
- *Go to homebase* → *Wander*: The robot retrieved an item to the homebase.
- *Wander/Go to homebase* → *Resolve Conflict*: The robot detected an imminent collision with another robot.
- *Resolve Conflict* → *Wander/Go to homebase*: The robot finished executing the reactive coordination method and goes back to its previous state.

Figure 5 shows the environment where experiments with the Krembots were conducted. On the table, the wooden cylinders are the stations where robots gather items from. The arena consisted of a 150 × 80 cm^2 tabletop, where we evenly spread 11 stations, fixed in position. Once a robot reaches one of those stations, an item is considered taken, and the robot needs to simulate transporting it, by moving back to the a small area devoted to be a home. It should be noted that since the Krembot robots have no localization capabilities, they are unable to either remember or plan a path to one of the stations not home. Therefore, they do it by randomly searching for a station. The home is lighted (bottom left corner in Figure 5) for identification.

[3] https://www.robotican.net/kremebot

[4] https://www.cs.biu.ac.il/~galk/research/swarms/


Alphabet Soup

```
update speed: 4.00
fps: 31.88
```

Completed Words

Pending Words

```
bunged
felicitatio
ringing
misfire
reheat
secretivene
braggers
intellectiv
hanni
cabriolets
```

FIGURE 4
Alphabet Soup simulator. Red lines are the robots, purple circles are the buckets, green circles are the word stations, and cyan circles are the letter stations.

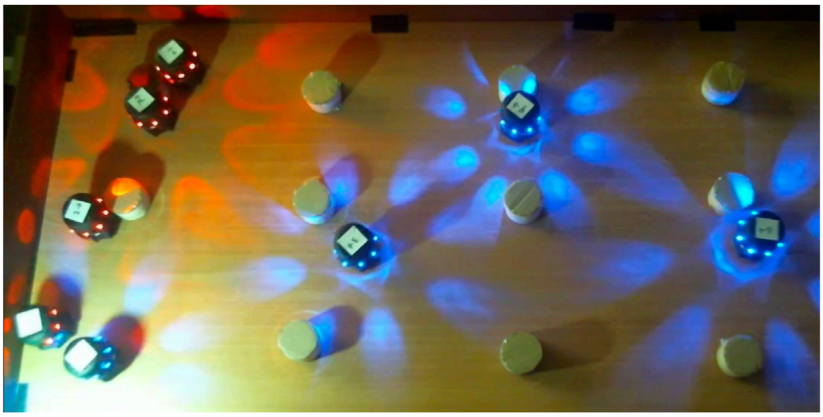


FIGURE 5
Krembot swarm robots.

5.2 Heterogeneity in adaptation

We distinguish between *learning* and *adaptation*. Learning focuses on *converging to a policy* which consistently chooses the best action for each state. On the other hand, adaptation focuses on *rapidly changing between policies*, according to what is best now.

Previous work by Kaminka et al. (2010) focused on adaptation. To do this, they used stateless Q-Learning with a very high learning rate (as high as 0.8). This allows the robots to rapidly switch between policies; the robots do not typically converge to a particular preferred choice. The reward function used was the original EI (which is a special case of the $\widehat{\text{AEI}}$ we discuss).

We began by evaluating the use of *EI*-based adaptation (learning rate $\alpha = 0.5$) versus convergent learning ($\alpha \leq 0.1$), in the two environments: Alphabet Soup and Krembot foraging. The goal is to examine whether heterogeneous swarms do better and whether adaptation or learning leads to performance increases.

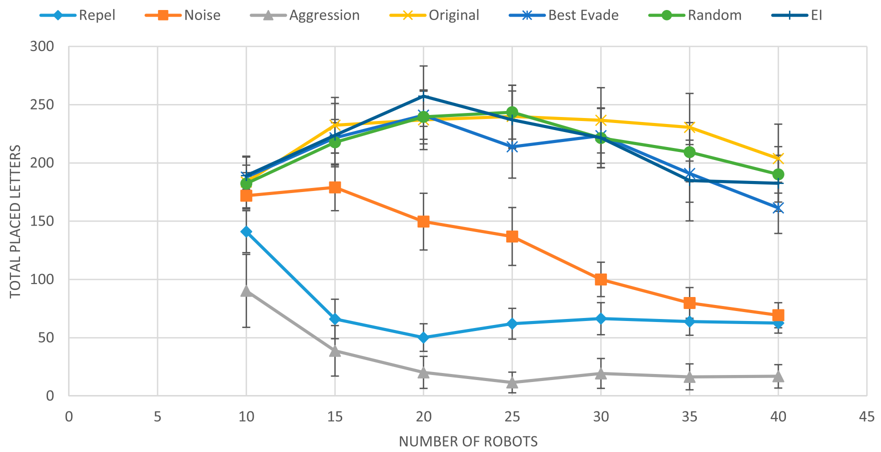


FIGURE 6
Results obtained by Douchan and Kaminka (2016). The horizontal axis marks the group size, and the vertical axis represents the group performance in terms of total placed letters.

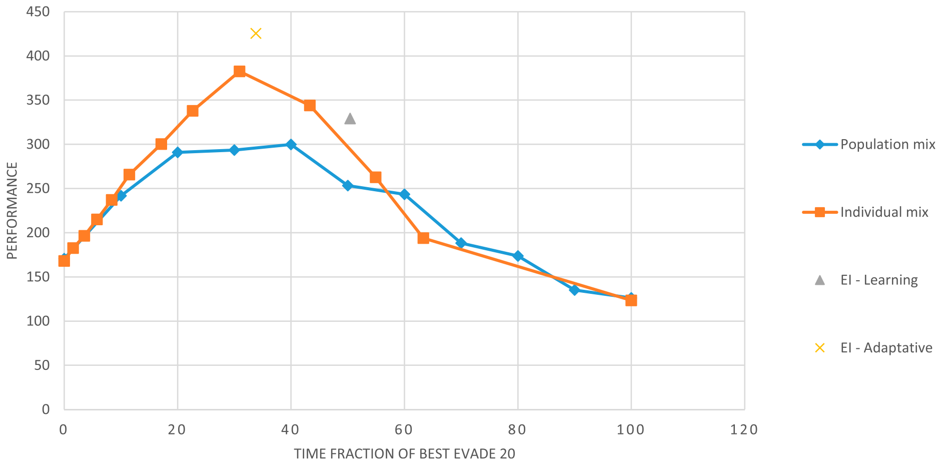


FIGURE 7
Swarm performance as a function of the heterogeneity of the swarm (time fraction of BE20, *p*).

5.2.1 Adaptation is better in Alphabet Soup

As a first step, we briefly summarize early results evaluating the use of EI-based adaptation in the Alphabet Soup simulator, published by Douchan and Kaminka (2016). They tested the performance of five reactive methods alone (i.e., each used by a homogeneous swarm, where all robots use the same collision avoidance method). Three of these have been used by Rosenfeld et al. (2008) and Kaminka et al. (2010): *Repel* [moving back for 20 ms, as introduced by Rosenfeld et al. (2008)], *Noise* [moving toward a random direction for 20 ms, as introduced by Balch and Arkin (1998)], and *Aggression* [randomly choose between backing off like in repel, or staying put until the robot has moved, as introduced by Vaughan et al. (2000)]. Two additional methods were provided by the Alphabet Soup simulator: *Best Evade* (always go to most vacant direction for a given amount of time) and the default method (termed *Original*), a stochastic combination of *best evade* and noise.

Douchan and Kaminka (2016) compared the performance of these five basic methods with random selection of methods by each robot, in each collision (a *Random* selection method), and with an adaptive use of stateless Q-learning (learning rate $\alpha = 0.5$ and exploration rate $\epsilon = 0.1$). All settings tested in group sizes vary between 10 and 40 and were repeated 60 times.

Figure 6 shows the results from these experiments. The figure shows that three of the fixed collision avoidance methods (repel, noise, and aggression) are inferior to the others. These three are behaviorally homogeneous swarms: all robots use the same collision-avoidance methods. Of the four top performers (not necessarily statistically distinguishable), three are behaviorally heterogeneous: the *Random* method, by definition, has every robot change its selected collision-avoidance method with every collision, independently of other robots; the *Original* method stochastically switched between *best-evade* and *noise*; the *EI* method is the adaptive

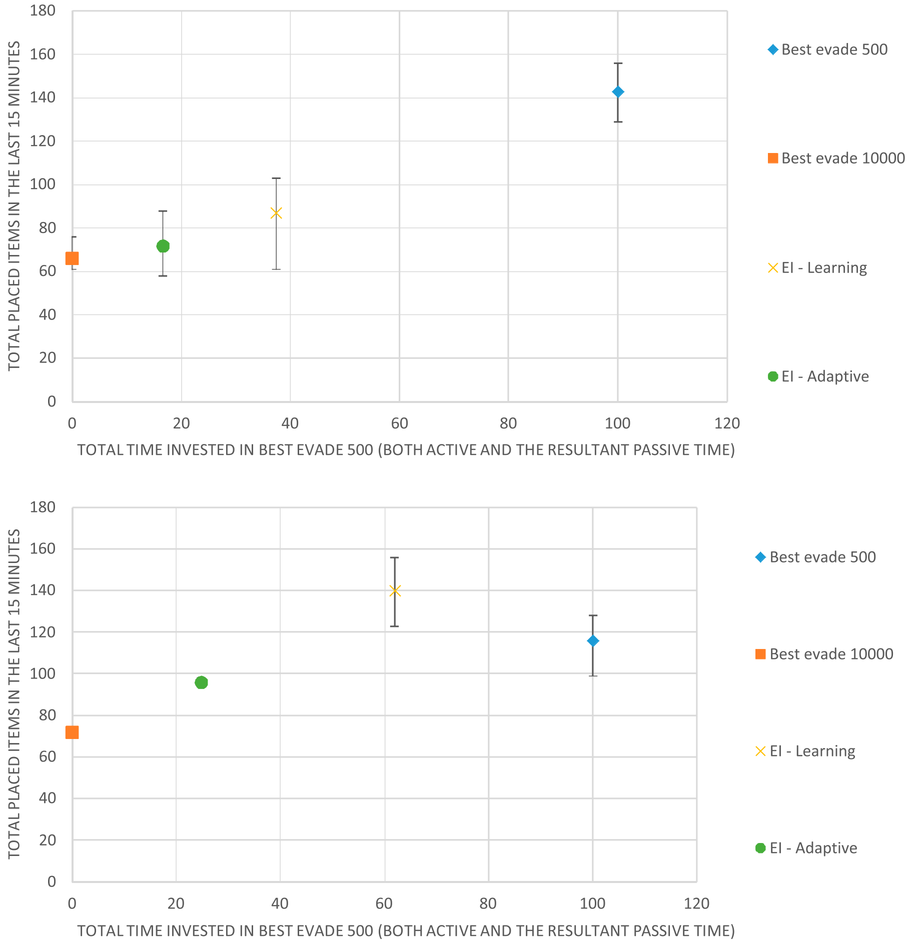


FIGURE 8
Results for Best Evade 500 (BE500) and Best Evade 10,000 with four (top) and eight (bottom) Krembot robots, respectively. The horizontal axis marks the fraction of BE500 used. The vertical axis marks the group performance in terms of total retrieved items.

method using the original EI as a reward function, i.e., AEI with $A_0 = 0$.

Intrigued by these results, we used the Alphabet simulator to directly evaluate the level of heterogeneity of the swarm and its effect on performance, especially in relation to the use of the EI reward. Fixing the group size to 20, we focused on the only homogeneous method that proved to perform well in the experiments reported above: *best evade*. We allowed each robot to select between two variants of this method: *Best evade* for 20 ms (*BE20*) and *best evade* for 2000 ms (*BE 2000*).

We then evaluated four configurations of the robots selections: in the *individual mix* configuration, each of the robots, when entering a collision, chooses BE20 with probability $p/100$ and BE2000 with probability $(1-p)/100$, independently of others. These robots make heterogeneous choice that vary over time. In the *population mix* configuration, p percent of the robots always chooses BE20, and the rest always chooses BE 2000; their choices do not vary with time. These configurations allow systematic evaluation of the level of heterogeneity: when $p = 0$, the swarm is homogeneous, and all robots select BE 2000; when $p = 100$, all robots use BE20. In between these two extremes, the swarm is heterogeneous, more-so in the individual mix configuration than in the population mix. We emphasize these are not learning methods: p is controlled and fixed.

Figure 7 shows the performance of the two configurations as a function of the fraction of BE20 in Alphabet Soup as p is controlled and varied between 0 and 100. We interpolate linearly between the sampled experiment points. The figure shows that both controlled-mix configurations reach their maxima points at p between approximately 30 and 50, i.e., where only between 30% and 50% of the robots select BE20. Both homogeneous swarms shown (at fraction = 0 and at fraction = 100) have the lowest performance.

The figure also shows two specific performance points, resulting from the application of reinforcement learning with the EI reward. The *EI-Adaptation* point marks the result of using learning with a learning rate $\alpha = 0.5$ and exploration rate of 0.1, both encouraging rapid changes in the learned policy, just as the individual mix changes selection by the same robot over time. The *EI-Learning* point marks the result of using learning rate $\alpha = 0.05$ and exploration rate of 0.02, to encourage convergence

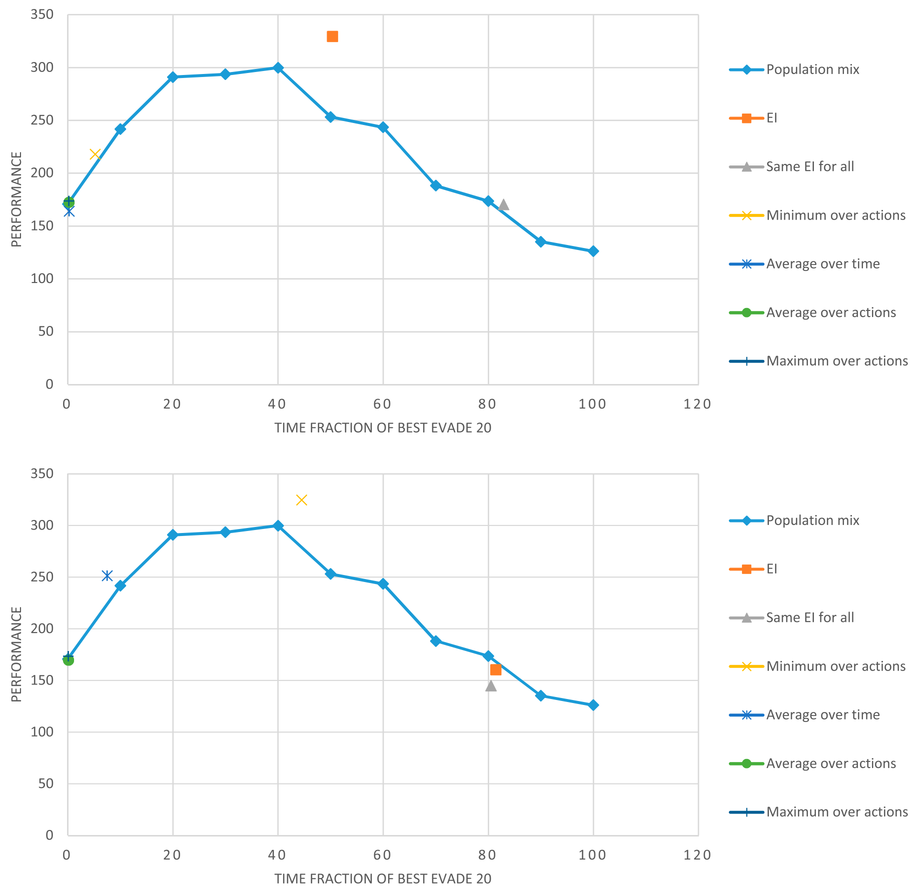


FIGURE 9
Performance of different WLU approximations with regular Q-learning (upper chart) vs. with continuous time Q-learning (lower chart) and where they are relative to the population mix.

to a single selection (just as the population mix enforces). We note that the adaptive method outperforms the individual mix and the learning method outperforms the population mix, and both points are reached when the swarm is behaviorally heterogeneous.

5.2.2 Adaptation is not always better in Krembots

We now turn to testing the role of adaptation and learning with real robots, hoping to draw lessons as to the role of heterogeneity in these different settings. We test two coordination methods of the same type but with different time parameters (the speed of the robots is different, and so these were empirically determined): Best Evade for 500 ms (*BE500*) and Best Evade for 10,000 ms (*BE10000*). We first test each method separately and then perform test selection using *EI-Adaptation* (learning rate $\alpha = 0.5$ and an exploration rate of 0.1) and *EI-learning* (learning rate $\alpha = 0.05$ and an exploration rate of 0.02).

We tested the performance of the different configurations in four robots and eight robots. We measure the performance of each configuration and the time fraction the robots spent on *BE500*. The duration of each run is 1 h long. For each hour-long run, we logged each event, such as a collision or an item that was retrieved. From this log, we extracted statistics on the number of items retrieved and the coordination method choices of the robots. We extracted statistics based only on the last 15 min of the run since we want the learning to stabilize. As before, this allows controlling the heterogeneity of the swarm by fixing the fraction of *BE500* or assessing it from the logs.

Figure 8 shows the results for 4 Krembot robots (top) and for Figure 8 (bottom). Like the previous figure, the horizontal axis measures the fraction of the time, in which the robots spent using BE500, i.e., the behavioral heterogeneity of the swarm: The 0 point on this axis marks a homogeneous swarm that never uses BE500 and instead always uses *BE10000*. The point marked 100 on this axis shows the results for another homogeneous swarm, where all robots use BE500.

Figure 8 shows that in the case of 4 robots (top), the best-performing swarm is a homogeneous swarm (all robots choose BE500 collision-avoidance). Both EI-Adaptive and EI-Learning fail to achieve equivalent performance. However, in the case of 8 robots, a heterogeneous swarm is the best method, and it is achieved using EI-Learning (where about 50% of the robots choose BE500). Here, while heterogeneity proves superior, it is achieved by learning using regular Q-Learning, rather than the adaptive method proposed by Kaminka et al. (2010).

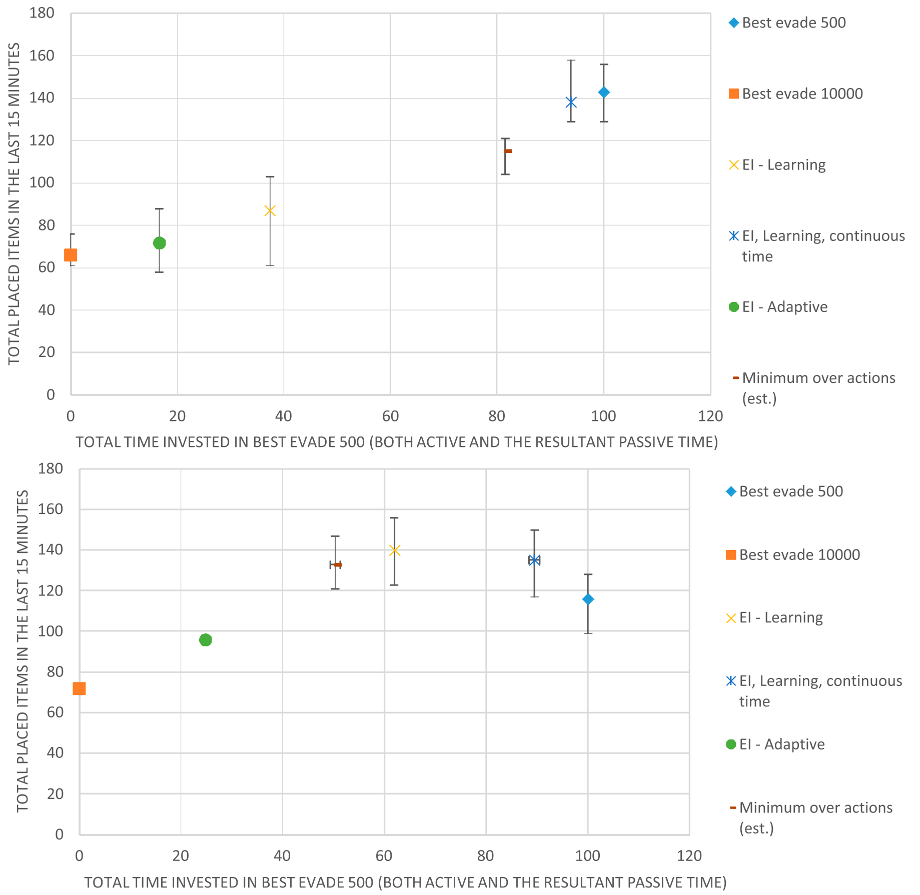


FIGURE 10
Learning in Krembots for four robots (upper chart) and eight robots (bottom chart).

5.3 Heterogeneity in learning

As the use of learning does not seem to work stably, we explore it further. In learning, robots converge to a fixed policy. We compare regular Q-Learning to continuous time Q-Learning. We do so by measuring the performance of different WLU approximations (Section 4.1), each with regular Q-Learning and continuous time Q-Learning (Algorithm 1, Section 4.3). The parameters of regular Q-Learning were set as follows: the learning rate was 0.05, and the exploration rate was 0.02. The parameters of continuous time Q-Learning are as follows: $\tau = 10$ seconds and the exploration rate was 0.02.

We begin again with the Alphabet Soup simulator, with the action set containing two actions as before: *BE20* and *BE 2000*. Figure 9 shows the results when using regular Q-Learning (top) and continuous-time Q-Learning (bottom). The line shows the population-mix, as before. The top figure shows EI learning being superior to all others (as in Figure 7). The bottom figure shows the *Minimum over actions* being superior. We draw two lessons from these results. First, regardless of the learning method and assumptions, the top performing swarm is always a heterogeneous swarm. Second, the algorithm used is sensitive to the selected $\widehat{\mathrm{AEI}}$ approximation.

Finally, we go back to the Krembot robots to evaluate the use of the learning algorithms, with different rewards and both adaptive and learning parameters. We tested BE500 and BE10000 with EI (Kaminka et al., 2010) using the same Q-Learning parameters for learning (EI-learning) and adaptation (EI-Adpative). We also evaluate the use of EI with the continuous-time Q-Learning algorithm (Algorithm 1) and, alternatively, the use of the minimum-over-actions approximation with the same algorithm. Its parameters were set to $\tau = 10$ seconds and an exploration rate of 0.02.

Figure 10 shows the results. For four robots, as before, a homogeneous swarm (everyone uses BE500) is the best. It is good to see, however, that the use of Algorithm 1 with EI comes very close to its performance. Indeed, it results in a heterogeneous swarm where 95% of robots select BE500. Given the exploration rate and the fact that there are only two methods, this corresponds to exactly the 5% of the time where the exploration rate forces the robot to choose BE10000. The bottom figure shows that all best swarms are heterogeneous.

In a different publication, Douchan et al. (2019) reported on additional experiments utilizing the learning methods we presented, contrasting their results with those achieved by testing directly with the true swarm utility U, WLU of the utility U, and several WLU alternatives. We refer the reader there, for further details.

6 Conclusion

This paper explores the role of behavioral heterogeneity in robot swarms engaged in foraging. It presents an abstract theoretical model of this swarm task, showing a mathematical connection between the *Coordination Overhead* (*CO*) of the robots in foraging—defined by the portion of time spent coordinating—to the global utility of the swarm. We then connected between the swarm *CO* of the whole lifetime of the swarm, to the decisions of individual robots in a single collision. This allows us to show that in principle, swarm robots can maximize an individual reward for each collision that will yield good global utility in the long run.

Specifically, we presented the *Aligned Effectiveness Index* (AEI), a reward function that ties the global *CO* of the swarm with individual estimates. This reward function allows individuals within the swarm to make decisions that improve the swarm's performance while adapting to changing collision conditions. It is a generalization of the EI reward proposed in earlier work Kaminka et al. (2010), which is not aligned, and for which the bounds we present are not known.

We focused on *swarm foraging*, a canonical swarm task of great interest both scientifically and commercially (e,g., in order picking, search and rescue, and agriculture; see applications discussed in Section 2). We have shown several solutions to challenges that may rise in practice when applying the theoretical model. First, we discussed several possible approximations for the AEI reward function that stands at the basis of utilizing learning in this task. Second, we developed a continuous time variant of Q-Learning in order to address possible inaccuracies of regular Q-Learning that may rise in continuous-time settings, in which robots operate. The utilization of learning by agents, in a completely distributed manner, often leads to specialization of behavioral roles, and thus to more heterogeneous swarms.

The results of the experiments clearly support the hypothesis that *diversity in decision-making* can play an important role in the performance of a swarm. This conclusion agrees with studies of swarms, whose members evolve their decision-making controllers using evolutionary computation (Montague et al., 2023), and studies of behavioral diversity in models of human pedestrians [e.g., in mixed culture (Kaminka and Fridman, 2018)]. Surprisingly, perhaps, Balch (1999) has investigated the use of machine learning in simulated foraging robots (up to 8) and reached conclusions opposite from ours, which states that foraging robots seemed to benefit from being homogeneous. We believe that this seeming contradiction in conclusions is a result of the previous study utilizing robots that were able to communicate information about the location of items and home. We also note that the results demonstrate that diversity *can* be extremely important to the success of the swarm but is not always needed. For instance, in the experiments we conducted, homogeneous decision-making seems to do well in smaller swarm sizes (see Figure 10).

The conclusion we reach on behavioral diversity complements analogous conclusions as to the importance of *diversity in capabilities*, in related studies. For example, the *swarmanoid* project (Dorigo et al., 2012) explores the use of mechanically different robots in carrying out complex tasks. Berumen et al. (2023) demonstrated the impact of robots with diverse error models on foraging performance, and Adams et al. (2023) developed foraging swarms made of two types of robots: *searchers* and *beacons* that assist in communicating signals in-limited communication settings. Swarms of heterogeneous nanobots are able to carry out complex tasks, e.g., a form of Asimov's laws of robots (Kaminka et al., 2017). In the larger context of multi-robot systems, similar ideas about the importance of diversity have been presented in investigations of heterogeneous *teams*, where robots communicate globally and essentially without restrictions so as to coordinate how to bring their different capabilities to bear on the joint problem. Such investigations include those by Xu et al. (2005), Parker and Tang (2006), Tang and Parker (2007), and Liemhetcharat and Veloso (2013). Likewise, heterogeneity plays an important role in natural swarms as well, e.g., see (Ariel et al., 2022).

Although this study focused on *diversity* of the swarm, complementary studies examine the *optimality* of the individual robot's decision-making, when robots use aligned rewards. Recent investigations of alternative learning approaches and alternative formulations begin to explore the question of how individual self-interested rational reward maximization leads to collective utility maximization (Fatima et al., 2024; Katz, 2023; Kaminka, 2025).

Data availability statement

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

Author contributions

GK: writing–original draft and writing–review and editing. YD: writing–original draft and writing–review and editing.

Funding

The author(s) declare that financial support was received for the research, authorship, and/or publication of this article. The authors gratefully acknowledge partial funding support from Israel Science Foundation Grants #2306/18 and #1373/24.

Acknowledgments

GK thanks K. Ushi, as always.

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

Adams, S., Jarne Ornia, D., and Mazo, M. (2023). A self-guided approach for navigation in a minimalistic foraging robotic swarm. *Aut. Robots* 47, 905–920. doi:10.1007/s10514-023-10102-y

Adhikari, S. (2021). *Study of scalability in a robot swarm performance and demonstration of superlinear performance in conveyor bucket brigades and collaborative pulling*. Spain: Master's thesis, The University of Toledo.

Agmon, N., Hazon, N., and Kaminka, G. A. (2008a). The giving tree: constructing trees for efficient offline and online multi-robot coverage. *Ann. Math Artif. Intell.* 52, 143–168. doi:10.1007/s10472-009-9121-1

Agmon, N., Kraus, S., and Kaminka, G. A. (2008b). "Multi-robot perimeter patrol in adversarial settings," in Proceedings of IEEE International Conference on Robotics and Automation (ICRA-08), Pasadena, CA, USA, 19-23 May 2008, 2339–2345. doi:10.1109/robot.2008.4543563

Agogino, A., and Tumer, K. (2008). Analyzing and visualizing multiagent rewards in dynamic and stochastic domains. *J. Aut. Agents Multi-Agent Syst.* 17, 320–338. doi:10.1007/s10458-008-9046-9

Albiero, D., Pontin Garcia, A., Kiyoshi Umezu, C., and Leme De Paulo, R. (2022). Swarm robots in mechanized agricultural operations: a review about challenges for research. *Comput. Electron. Agric.* 193, 106608. doi:10.1016/j.compag.2021.106608

Alers, S., Tuyls, K., Ranjbar-Sahraei, B., Claes, D., and Weiss, G. (2014). "Insect-inspired robot coordination: foraging and coverage," in ALIFE 14: Proceedings of the Fourteenth International Conference on the Synthesis and Simulation of Living Systems, 761–768. doi:10.1162/978-0-262-32621-6-ch123

Aljalaud, F., and Kurdi, H. A. (2021). Autonomous task allocation for multi-UAV systems based on area-restricted search behavior in animals. *Procedia Comput. Sci.* 191, 246–253. doi:10.1016/j.procs.2021.07.031

Amanatiadis, A., Henschel, C., Birkicht, B., Andel, B., Charalampous, K., Kostavelis, I., et al. (2015). "Avert: an autonomous multi-robot system for vehicle extraction and transportation," in IEEE International Conference on Robotics and Automation (ICRA), Seattle, WA, USA, 26-30 May 2015, 1662–1669. doi:10.1109/icra.2015.7139411

Ariel, G., Ayali, A., Be'er, A., and Knebel, D. (2022). Variability and heterogeneity in natural swarms: experiments and modeling. *Act. Part.* 3, 1–33. doi:10.1007/978-3-030-93302-9_1

Arslan, G., Marden, J. R., and Shamma, J. S. (2007). Autonomous vehicle-target assignment: a game theoretical formulation. *ASME J. Dyn. Syst. Meas. Control* 129, 584–596. doi:10.1115/1.2766722

Balch, T. (1999). "The impact of diversity on performance in multi-robot foraging," in Proceedings of the third annual conference on Autonomous Agents (ACM), 92–99.

Balch, T., and Arkin, R. C. (1998). Behavior-based formation control for multirobot teams. *IEEE Trans. Robotics Automation* 14, 926–939. doi:10.1109/70.736776

Balch, T., and Hybinette, M. (2000). "Social potentials for scalable multirobot formations," in Proceedings of IEEE International Conference on robotics and automation (ICRA-00).

Basilico, N., Gatti, N., and Amigoni, F. (2009). "Leader-follower strategies for robotic patrolling in environments with arbitrary topologies," in Proceedings of the International Joint Conference on Autonomous Agents and Multi-Agent Systems, 57–64.

Bastien, R., and Romanczuk, P. (2020). A model of collective behavior based purely on vision. *Sci. Adv.* 6, 0792. doi:10.1126/sciadv.aay0792

Batalin, M., and Sukhatme, G. (2002). "Spreading out: a local approach to multi-robot coverage," in *International symposium on distributed autonomous robotic systems (DARS)*, 373–382.

Berumen, E. B., Alden, K. J., Pomfret, A., Timmis, J., and Tyrrell, A. (2023). A study of error diversity in robotic swarms for task partitioning in foraging tasks. *Front. Robotics AI.* doi:10.3389/frobt.2022.904341

Bouraine, S., Fraichard, T., Azouaoui, O., and Salhi, H. (2014). "Passively safe partial motion planning for mobile robots with limited field-of-views in unknown dynamic environments," in Proceedings of IEEE International Conference on Robotics and Automation, 3576–3582. doi:10.1109/icra.2014.6907375

Bradtke, S. J., and Duff, M. O. (1994). "Reinforcement learning methods for continuous-time markov decision problems," in *Advances in neural information processing systems* (MIT Press), 393–400.

Cheraghi, A. R., Peters, J., and Graffi, K. (2020). "Prevention of ant mills in pheromone-based search algorithm for robot swarms," in 2020 3rd International Conference on Intelligent Robotic and Control Engineering (IRCE), 23–30doi. doi:10.1109/IRCE50905.2020.9199239

Claus, C., and Boutilier, C. (1998). The dynamics of reinforcement learning in cooperative multiagent systems. *AAAI/IAAI* 1998, 746–752.

Danassis, P., and Faltings, B. (2018). A courteous learning rule for ad-hoc anti-coordination. *arXiv preprint arXiv:1801.07140*

Desai, J. P. (2002). A graph theoretic approach for modeling mobile robot team formations. *J. Robotic Syst.* 19, 511–525. doi:10.1002/rob.10057

Desai, J. P., Ostrowski, J., and Kumar, V. (1998). "Controlling formations of multiple mobile robots," in Proceedings of IEEE International Conference on Robotics and Automation, 2864–2869. doi:10.1109/robot.1998.680621

Devlin, S., Yliniemi, L., Kudenko, D., and Tumer, K. (2014). "Potential-based difference rewards for multiagent reinforcement learning," in Proceedings of the Thirteenth International Joint Conference on Autonomous Agents and Multiagent Systems (Paris, France).

Dias, M. B., and Stentz, A. T. (2000). "A free market architecture for distributed control of a multirobot system," in 6th International Conference on Intelligent Autonomous Systems (IAS-6), 115–122.

Dias, M. B., Zlot, R., Zinck, M., Gonzalez, J. P., and Stentz, A. (2004). "A versatile implementation of the traderbots approach for multirobot coordination," in Proceedings of the Eighth Conference on Intelligent Autonomous Systems (IAS-8).

Dorigo, M., Floreano, D., Gambardella, L., Mondada, F., Nolfi, S., Baaboura, T., et al. (2012). Swarmanoid: a novel concept for the study of heterogeneous robotic swarms. *IEEE Robotics and Automation Mag.* 20, 60–71. doi:10.1109/mra.2013.2252996

Dorigo, M., Theraulaz, G., and Trianni, V. (2021). Swarm robotics: past, present, and future [point of view]. *Proc. IEEE* 109, 1152–1165. doi:10.1109/JPROC.2021.3072740

Douchan, Y., and Kaminka, G. A. (2016). "The effectiveness index intrinsic reward for coordinating service robots," in *13th international symposium on distributed autonomous robotic systems (DARS-2016)*. Editors S. Berman, M. Gauci, E. Frazzoli, A. Kolling, R. Gross, A. Martinoli, et al. (Springer).

Douchan, Y., Wolf, R., and Kaminka, G. A. (2019). "Swarms can be rational," in Proceedings of the International Joint Conference on Autonomous Agents and Multi-Agent Systems.

Duncan, S., Estrada-Rodriguez, G., Stocek, J., Dragone, M., Vargas, P. A., and Gimperlein, H. (2022). Efficient quantitative assessment of robot swarms: coverage and targeting Lévy strategies. *Bioinspiration and Biomimetics* 17, 036006. doi:10.1088/1748-3190/ac57f0

Elmaliach, Y., Agmon, N., and Kaminka, G. A. (2007). "Multi-robot area patrol under frequency constraints," in Proceedings of IEEE International Conference on Robotics and Automation (ICRA-07), 385–390. doi:10.1109/robot.2007.363817

Elmaliach, Y., Agmon, N., and Kaminka, G. A. (2009). Multi-robot area patrol under frequency constraints. *Ann. Math Artif. Intell.* 57, 293–320. doi:10.1007/s10472-010-9193-y

Elmaliach, Y., Shiloni, A., and Kaminka, G. A. (2008). "A realistic model of frequency-based multi-robot fence patrolling," in Proceedings of the Seventh International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS-08), 63–70.1

Erusalimchik, D., and Kaminka, G. A. (2008). "Towards adaptive multi-robot coordination based on resource expenditure velocity," in Proceedings of the Tenth Conference on Intelligent Autonomous Systems (IAS-10) (Amsterdam, Netherlands: IOS Press).

Farinelli, A., Iocchi, L., and Nardi, D. (2004). Multirobot systems: a classification focused on coordination. *IEEE Trans. Syst. Man, Cybern. Part B Cybern.* 34, 2015–2028. doi:10.1109/TSMCB.2004.832155

Farinelli, A., Iocchi, L., Nardi, D., and Ziparo, V. A. (2006). Assignment of dynamically perceived tasks by token passing in multi-robot systems. *Proc. IEEE* 94, 1271–1288. Special issue on Multi-Robot Systems. doi:10.1109/jproc.2006.876937

Fatima, S., Jennings, N. R., and Wooldridge, M. (2024). Learning to resolve social dilemmas: a survey. *J. Artif. Intell. Res.* 79, 895–969. doi:10.1613/jair.1.15167

Ferrer, E. C., Hardjono, T., Pentland, A., and Dorigo, M. (2021). Secure and secret cooperation in robot swarms. *Sci. Robotics* 6, eabf1538. doi:10.1126/scirobotics.abf1538

Fox, D., Burgard, W., and Thrun, S. (1997). The dynamic window approach to collision avoidance. *IEEE Robot. Autom. Mag.* 4, 23–33. doi:10.1109/100.580977

Francisco, J., Gonzalez Herrera, F., and Lara-Alvarez, C. (2018). "A coordinated wilderness search and rescue technique inspired by bacterial foraging behavior," in In In proceedings of the IEEE International Conference on Robotics and Biomimetics (ROBIO), 318–324. doi:10.1109/ROBIO.2018.8665267

Fredslund, J., and Mataric, M. J. (2002). A general algorithm for robot formations using local sensing and minimal communication. *IEEE Trans. Robotics Automation* 18, 837–846. doi:10.1109/tra.2002.803458

Gage, D. W. (1992). "Command control for many-robot systems," in *The nineteenth annual AUVS Technical Symposium (AUVS-92)*.

Gerkey, B. P., and Matarić, M. (2004). A formal analysis and taxonomy of task allocation in multi-robot systems. *Int. J. Robotics Res.* 23, 939–954. doi:10.1177/0278364904045564

Gerkey, B. P., and Mataric, M. J. (2002). Sold!: auction methods for multi-robot coordination. *IEEE Trans. Robotics Automation* 18, 758–768. doi:10.1109/tra.2002.803462

Giuggioli, L., Arye, I., Robles, A. H., and Kaminka, G. A. (2016). "From ants to birds: a novel bio-inspired approach to online area coverage," in *International symposium on distributed autonomous robotic systems (DARS)*. Editors S. Berman, M. Gauci, E. Frazzoli, A. Kolling, R. Gross, A. Martinoli, et al. (Springer).

Godoy, J., Karamouzas, I., Guy, S. J., and Gini, M. (2015). "Adaptive learning for multi-agent navigation," in Proceedings of the 2015 International Conference on Autonomous Agents and Multi-Agent Systems (International Foundation for Autonomous Agents and Multi-Agent Systems), 1577–1585.

Goldberg, D., Cicirello, V., Dias, M. B., Simmons, R., Smith, S., and Stentz, A. T. (2003). "Market-based multi-robot planning in a distributed layered architecture," in *Multi-robot systems: from swarms to intelligent automata: proceedings from the 2003 international workshop on multi-robot systems* (Kluwer Academic Publishers), 2, 27–38.

Goldberg, D., and Matarić, M. (1997). "Interference as a tool for designing and evaluating multi-robot controllers," in *AAAI/IAAI*, 637–642.

Goldberg, D., and Mataric, M. J. (1997). "Interference as a tool for designing and evaluating multi-robot controllers," in Proceedings of the Fourteenth National Conference on Artificial Intelligence (AAAI-97) (Providence, RI: AAAI Press), 637–642.

Guy, S. J., Lin, M. C., and Manocha, D. (2010). "Modeling collision avoidance behavior for virtual humans," in Proceedings of the Ninth International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS-10), 575–582.

Hahn, C., Ritz, F., Wikidal, P., Phan, T., Gabor, T., and Linnhoff-Popien, C. (2020). "Foraging swarms using multi-agent reinforcement learning," in In Proceedings of the 2020 Conference on Artificial Life (MIT Press), 333–340. doi:10.1162/isal_a_00267

Hamann, H. (2018). *Swarm robotics: a formal approach*. Springer.

Hazard, C. J., and Wurman, P. R. (2006). "Alphabet soup: a testbed for studying resource allocation in multi-vehicle systems," in *In proceedings of the 2006 AAAI workshop on auction mechanisms for robot coordination*, 23–30.

Hazon, N., and Kaminka, G. (2008). On redundancy, efficiency, and robustness in coverage for multiple robots. *Robotics Aut. Syst.* 56, 1102–1114. doi:10.1016/j.robot.2008.01.006

Hénard, A., Rivière, J., Peillard, E., Kubicki, S., and Coppin, G. (2023). A unifying method-based classification of robot swarm spatial self-organisation behaviours. *Adapt. Behav.* 31, 577–599. doi:10.1177/10597123231163948

Hernandez-Leal, P., Kaisers, M., Baarslag, T., and de Cote, E. M. (2019). A survey of learning in multiagent environments: dealing with non-stationarity. *Tech. Rep.* 1707, 09183v2. [cs].

Hoff, N. R., Sagoff, A., Wood, R. J., and Nagpal, R. (2010). "Two foraging algorithms for robot swarms using only local communication," in IEEE International Conference on Robotics and Biomimetics (ROBIO IEEE), 123–130.

Jensen, E., Franklin, M., Lahr, S., and Gini, M. (2011). "Sustainable multi-robot patrol of an open polyline," in Proceedings of IEEE International Conference on Robotics and Automation (ICRA-11), 4792–4797. doi:10.1109/icra.2011.5980279

Jin, B., Liang, Y., Han, Z., and Ohkura, K. (2020). Generating collective foraging behavior for robotic swarm using deep reinforcement learning. *Artif. Life Robot.* 25, 588–595. doi:10.1007/s10015-020-00642-2

Kaminka, G. A. (2025). *Swarms can be rational*. Philosophical Transactions of the Royal Society A In press.

Kaminka, G. A., Erusalimchik, D., and Kraus, S. (2010). "Adaptive multi-robot coordination: a game-theoretic perspective," in Proceedings of IEEE International Conference on Robotics and Automation (ICRA-10), 328–334. doi:10.1109/robot.2010.5509316

Kaminka, G. A., and Frenkel, I. (2005). "Flexible teamwork in behavior-based robots," in Proceedings of the Twentieth National Conference on Artificial Intelligence (AAAI-05).

Kaminka, G. A., and Frenkel, I. (2007). "Integration of coordination mechanisms in the BITE multi-robot architecture," in Proceedings of IEEE International Conference on Robotics and Automation (ICRA-07), 2859–2866. doi:10.1109/robot.2007.363905

Kaminka, G. A., and Fridman, N. (2018). Simulating urban pedestrian crowds of different cultures. *ACM Trans. Intelligent Syst. Technol.* 9 (27), 1–27. doi:10.1145/3102302

Kaminka, G. A., and Glick, R. (2006). "Towards robust multi-robot formations," in Proceedings of IEEE International Conference on Robotics and Automation (ICRA-06).

Kaminka, G. A., Lupu, I., and Agmon, N. (2016). "Construction of optimal control graphs in multi-robot systems," in *13th international symposium on distributed autonomous robotic systems (DARS-2016)*. Editors S. Berman, M. Gauci, E. Frazzoli, A. Kolling, R. Gross, A. Martinoli, et al. (Springer).

Kaminka, G. A., Schechter-Glick, R., and Sadov, V. (2008). Using sensor morphology for multi-robot formations. *IEEE Trans. Robotics* 24, 271–282. doi:10.1109/tro.2008.918054

Kaminka, G. A., Spokoini-Stern, R., Amir, Y., Agmon, N., and Bachelet, I. (2017). Molecular robots obeying Asimov's three laws of robotics. *Artif. Life* 23, 343–350. doi:10.1162/ARTL_a_00235

Kaminka, G. A., Traub, M., Elmaliach, Y., Erusalimchik, D., and Fridman, A. (2013). "On the use of teamwork software for multi-robot formation control (an extended abstract)," in Proceedings of the Twelfth International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS-13).

Kaminka, G. A., Yakir, A., Erusalimchik, D., and Cohen-Nov, N. (2007). "Towards collaborative task and team maintenance," in Proceedings of the Sixth International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS-07).

Kapetanakis, S., and Kudenko, D. (2002). Reinforcement learning of coordination in cooperative multi-agent systems. *AAAI/IAAI* 2002, 326–331.

Katz, K. (2023). "Competitive multi-swarm systems,". Israel: Bar Ilan University. Master's thesis.

Kober, J., Bagnell, J. A. D., and Peters, J. (2013). Reinforcement learning in robotics: a survey. *Int. J. Robotics Res.* 32, 1238–1274. doi:10.1177/0278364913495721

Kuckling, J. (2023). Recent trends in robot learning and evolution for swarm robotics. *Front. Robotics AI* 10, 1134841. doi:10.3389/frobt.2023.1134841

Lee, D., Lu, Q., and Au, T.-C. (2022). "Dynamic robot chain networks for swarm foraging," in 2022 International Conference on Robotics and Automation (ICRA), 4965–4971. doi:10.1109/ICRA46639.2022.9811625

Lemay, M., Michaud, F., Létourneau, D., and Valin, J.-M. (2004). "Autonomous initialization of robot formations," in Proceedings of IEEE International Conference on Robotics and Automation (ICRA-04), 3018–3023 Vol.3. doi:10.1109/robot.2004.1307520

Lerman, K., and Galstyan, A. (2002). Mathematical model of foraging in a group of robots: effect of interference. *Aut. Robots* 13, 127–141. doi:10.1023/a:1019633424543

Liemhetcharat, S., and Veloso, M. M. (2013). "Synergy graphs for configuring robot team members," in Proceedings of the Twelfth International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS-13), 111–118.

Löffler, R. C., Panizon, E., and Bechinger, C. (2023). Collective foraging of active particles trained by reinforcement learning. *Sci. Rep.* 13, 17055. doi:10.1038/s41598-023-44268-3

Lu, Q., Fricke, G. M., Ericksen, J. C., and Moses, M. E. (2020). Swarm foraging review: closing the gap between proof and practice. *Curr. Robot. Rep.* 1, 215–225. doi:10.1007/s43154-020-00018-1

Marden, J. R., and Wierman, A. (2009). "Overcoming limitations of game-theoretic distributed control," in Proceedings of the 48h IEEE Conference on Decision and Control (CDC) (United States: IEEE Press), 6466–6471.

Marino, A., Parker, L. E., Antonelli, G., and Caccavale, F. (2009). "Behavioral control for multi-robot perimeter patrol: a finite state automata approach," in Proceedings of IEEE International Conference on Robotics and Automation (ICRA-09), 831–836. doi:10.1109/robot.2009.5152710

Mataric, M. J. (1994). *Interaction and intelligent behavior*. United States: Ph.D. thesis, Massachusetts Institute of Technology.

McGuigan, L., Sterritt, R., Wilkie, G., and Hawe, G. (2022). Decentralised autonomic self-adaptation in a foraging robot swarm. *Int. J. Adv. Intelligent Syst.* 15, 12–23.

Michael, N., Zavlanos, M. M., Kumar, V., and Pappas, G. J. (2008). "Distributed multi-robot task assignment and formation control," in Proceedings of IEEE International Conference on Robotics and Automation (ICRA-08), 128–133. doi:10.1109/robot.2008.4543197

Michaud, F., Létourneau, D., Gilbert, M., and Valin, J.-M. (2002). "Dynamic robot formations using directional visual perception," in Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems.

Monderer, D., and Shapley, L. (1996). Potential games. *Games Econ. Behav.* 14, 124–143. doi:10.1006/game.1996.0044

Montague, K., Hart, E., Nitschke, G., and Paechter, B. (2023). "A quality-diversity approach to evolving a repertoire of diverse behaviour-trees in robot swarms," in Proceedings of International Conference on the Applications of Evolutionary Computation (Brno, Czech Republic: Springer).

Moshtagh, N., Michael, N., Jadbabaie, A., and Daniilidis, K. (2009). Vision-based, distributed control laws for motion coordination of nonholonomic robots. *IEEE Trans. Robotics* 25, 851–860. doi:10.1109/TRO.2009.2022439

Nuzhin, E. E., Panov, M. E., and Brilliantov, N. V. (2021). Why animals swirl and how they group. *Sci. Rep.* 11, 20843. doi:10.1038/s41598-021-99982-7

Ordaz-Rivas, E., Rodriguez-Liñan, A., and Torres-Treviño, L. (2021). Autonomous foraging with a pack of robots based on repulsion, attraction and influence. *Aut. Robots* 45, 919–935. doi:10.1007/s10514-021-09994-5

Ordaz-Rivas, E., and Torres-Treviño, L. (2022). "Modeling and simulation of swarm of foraging robots for collecting resources using RAOI behavior policies," in *Advances in computational intelligence*. Editors O. Pichardo Lagunas, J. Martínez-Miranda, and B. Martínez Seis (Cham: Springer Nature Switzerland), 266–278. doi:10.1007/978-3-031-19496-2_20

Osherovich, E., Yanovski, V., I. A, W., and Bruckstein, A. M. (2007). "Robust and Efficient Covering of Unknown Continuous Domains with Simple, Ant-Like A(ge)nts," in *Tech. Rep. CIS-2007-04*. Israel: Technion Computer Science Department

Pacheco, A., Strobel, V., Reina, A., and Dorigo, M. (2022). "Real-time coordination of a foraging robot swarm using blockchain smart contracts," in *Swarm intelligence*. Editors M. Dorigo, H. Hamann, M. López-Ibáñez, J. García-Nieto, A. Engelbrecht, C. Pinciroli, et al. (Cham: Springer International Publishing), 196–208. doi:10.1007/978-3-031-20176-9_16

Parker, L. E. (1998). ALLIANCE: an architecture for fault tolerant multirobot cooperation. *IEEE Trans. Robotics Automation* 14, 220–240. doi:10.1109/70.681242

Parker, L. E. (2008). "Multiple mobile robot systems," in *Springer handbook of robotics* (Springer), 921–941.

Parker, L. E., and Tang, F. (2006). Building multi-robot coalitions through automated task solution synthesis. *Proc. IEEE* 94 (Special issue on Multi-Robot Systems), 1289–1305. doi:10.1109/jproc.2006.876933

Pham, N. H., and Nguyen, M. D. (2020). Bacterial foraging algorithm for optimal joint-force searching strategy of multi – SAR vessels at sea. *Tech. Rep. 2020030471, Prepr.* doi:10.20944/preprints202003.0471.v1

Pitonakova, L., Crowder, R., and Bullock, S. (2014). "Understanding the role of recruitment in collective robot foraging," in Artificial Life XIV: Proceedings of the Fourteenth International Conference on the Synthesis and Simulation of Living Systems. Editors H. Lipson, H. Sayama, J. Rieffel, S. Risi, and R. Doursat (Massachusetts Institute of Technology (MIT) Press), 264–271.

Portugal, D., and Rocha, R. P. (2013). Multi-robot patrolling algorithms: examining performance and scalability. *Adv. Robot.* 27, 325–336. doi:10.1080/01691864.2013.763722

Rekleitis, I., New, A. P., Rankin, E. S., and Choset, H. (2008). Efficient boustrophedon multi-robot coverage: an algorithmic approach. *Ann. Math. Artif. Intell.* 52, 109–142. doi:10.1007/s10472-009-9120-2

Rosenfeld, A., Kaminka, G. A., Kraus, S., and Shehory, O. (2008). A study of mechanisms for improving robotic group performance. *Artif. Intell.* 172, 633–655. doi:10.1016/j.artint.2007.09.008

Rybski, P., Larson, A., Lindahl, M., and Gini, M. (1998). "Performance evaluation of multiple robots in a search and retrieval task," in *Proceedings of the Workshop on artificial Intelligence and manufacturing (albuquerque, NM)*, 153–160.

Salman, M., Garzón Ramos, D., and Birattari, M. (2024). Automatic design of stigmergy-based behaviours for robot swarms. *Commun. Eng.* 3, 30–13. doi:10.1038/s44172-024-00175-7

Schloesser, D. S., Hollenbeck, D., and Kello, C. T. (2021). Individual and collective foraging in autonomous search agents with human intervention. *Sci. Rep.* 11, 8492. doi:10.1038/s41598-021-87717-7

Schneider-Fontan, M., and Matarić, M. (1998). Territorial multi-robot task division. *IEEE Trans. Robotics Automation* 14, 815–822. doi:10.1109/70.720357

Sempe, F., and Drogoul, A. (2003). "Adaptive patrol for a group of robots," in Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, 2865–2869. doi:10.1109/IROS.2003.1249305

Sharon, G., Stern, R., Felner, A., and Sturtevant, N. R. (2015). Conflict-based search for optimal multi-agent pathfinding. *Artif. Intell.* 219, 40–66. doi:10.1016/j.artint.2014.11.006

Simmons, R. G., Goodwin, R., Haigh, K. Z., Koenig, S., and O'Sullivan, J. (1997). "A layered architecture for office delivery robots," in Proceedings of the First International Conference on Autonomous Agents (Agents-97), 245–252. doi:10.1145/267658.267723

Snape, J., van den Berg, J. P., Guy, S. J., and Manocha, D. (2011). The hybrid reciprocal velocity obstacle. *IEEE Trans. Robotics* 27, 696–706. doi:10.1109/tro.2011.2120810

Song, Z., and Vaughan, R. T. (2013). "Sustainable robot foraging: adaptive fine-grained multi-robot task allocation for maximum sustainable yield of biological resources," in Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IEEE), 3309–3316. doi:10.1109/iros.2013.6696827

Stern, R., Sturtevant, N. R., Felner, A., Koenig, S., Ma, H., Walker, T. T., et al. (2019). "Multi-agent pathfinding: definitions, variants, and benchmarks," in *In proceedings of the annual symposium on combinatorial search*.

Suarez, J., and Murphy, R. (2011). "A survey of animal foraging for directed, persistent search by rescue robotics," in *2011 IEEE international symposium on safety, security, and rescue robotics*, 314–320. doi:10.1109/SSRR.2011.6106744

Sung, C., Ayanian, N., and Rus, D. (2013). "Improving the performance of multi-robot systems by task switching," in 2013 IEEE International Conference on Robotics and Automation, 2999–3006. doi:10.1109/ICRA.2013.6630993

Tang, F., and Parker, L. E. (2007). "A complete methodology for generating multi-robot task solutions using asymtre-d and market-based task allocation," in Proceedings of IEEE International Conference on Robotics and Automation (ICRA-07), 3351–3358.

Thrun, S., Beetz, M., Bennewitz, M., Burgard, W., Cremers, A. B., Dellaert, F., et al. (2000). Probabilistic algorithms and the interactive museum tour-guide robot minerva. *Int. J. Robotics Res.* 19, 972–999. doi:10.1177/02783640022067922

Tumer, K., Agogino, A. K., and Wolpert, D. H. (2002). "Learning sequences of actions in collectives of autonomous agents," in Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-02) (New York, NY, United States), 378–385. doi:10.1145/544741.544832

Tumer, K., Welch, Z., and Agogino, A. (2008). "Aligning social welfare and agent preferences to alleviate traffic congestion," in Proceedings of the Seventh International Joint Conference on Autonomous Agents and Multiagent Systems (Estoril, Portugal).

Van Calck, L., Pacheco, A., Strobel, V., Dorigo, M., and Reina, A. (2023). A blockchain-based information market to incentivise cooperation in swarms of self-interested robots. *Sci. Rep.* 13, 20417. doi:10.1038/s41598-023-46238-1

van den Berg, J., Guy, S., Lin, M., and Manocha, D. (2011). Reciprocal n-body collision avoidance. *Robotics Res.*, 3–19. doi:10.1007/978-3-642-19457-3_1

van den Berg, J., Lin, M. C., and Manocha, D. (2008). "Reciprocal velocity obstacles for real-time multi-agent navigation," in Proceedings of IEEE International Conference on Robotics and Automation (ICRA-08), 1928–1935.

Vaughan, R., Støy, K., Sukhatme, G., and Matarić, M. (2000). "Go ahead, make my day: robot conflict resolution by aggressive competition," in Proceedings of the 6th int. conf. on the Simulation of Adaptive Behavior (Paris, France), 491–500. doi:10.7551/mitpress/3120.003.0052

Vig, L., and Adams, J. A. (2006). "Market-based multi-robot coalition formation," in *International symposium on distributed autonomous robotic systems (DARS)*. Editors M. Gini, and R. Voyles (Springer Japan), 227–236.

Winfield, A. F. (2009). "Foraging robots," in *Encyclopedia of complexity and systems science*. Editor R. A. Meyers (New York, NY: Springer New York), 3682–3700.

Wolpert, D., Wheeler, K. R., and Tumer, K. (1999). Collective intelligence for control of distributed dynamical systems. *Tech. Rep. cs.LG/9908013, CoRR/arxiv*.

Wolpert, D. H., and Tumer, K. (1999). An introduction to collective intelligence. *Tech. Rep. NASA-ARC-IC-99-63, NASA. Also Corr. cs.LG/9908014*.

Wurman, P. R., Dándrea, R., and Mountz, M. (2008). Coordinating hundreds of cooperative, autonomous vehicles in warehouses. *AI Mag.* 29, 9–19. doi:10.1609/aimag.v29i1.2082

Xu, Y., Scerri, P., Yu, B., Okamoto, S., Lewis, M., and Sycara, K. (2005). "An integrated token-based approach to scalable coordination," in Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS-05).

Yan, C., and Zhang, T. (2016). Multi-robot patrol: a distributed algorithm based on expected idleness. *Int. J. Adv. Robotic Syst.* 13, 1729881416663666. doi:10.1177/1729881416663666

Yehoshua, R., Agmon, N., and Kaminka, G. A. (2016). Robotic adversarial coverage of known environments. *Int. J. Robotics Res.* 35, 1419–1444. doi:10.1177/0278364915625785

Yu, J., and Lavalle, S. M. (2015). Optimal multi-robot path planning on graphs: structure and computational complexity. *arXiv preprint arXiv:1507.03289*

Zedadra, O., Jouandeau, N., Seridi, H., and Fortino, G. (2017). Multi-agent foraging: state-of-the-art and research challenges. *Complex Adapt. Syst. Model.* 5, 3. doi:10.1186/s40294-016-0041-8

Zhang, K., Yang, Z., and Başar, T. (2021). "Multi-agent reinforcement learning: a selective overview of theories and algorithms," in *Handbook of reinforcement learning and control*, 321–384.

Zlot, R., and Stentz, A. (2006). Market-based multirobot coordination for complex tasks. *Int. J. Robotics Res.* 25, 73–101. doi:10.1177/0278364906061160

Zuluaga, M., and Vaughan, R. T. (2005). "Reducing spatial interference in robot teams by local-investment aggression," in Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, 2798–2805. doi:10.1109/iros.2005.1545099

Appendix A: Nomenclature

TABLE A1 List of symbols and notations used in this paper. Where appropriate, we also list equivalent notation used by Kaminka et al. (2010).

Symbol	Meaning	Equivalent in Kaminka et al. (2010)
$N = \{1, 2, \ldots, n\}$	Set of $n = \|N\|$ robots (players)	$A = \{a_1, \ldots, a_n\}$
S_i	Set of actions available for robot i	M
$s_i \in S_i$	Specific action taken by robot i	α_i or α
$S = S_1 \times \cdots \times S_n$	Set of all possible joint actions	
$s = (s_1, \ldots, s_n) \in S$	Joint action, a combination of the individual specific actions of all robots	
s_i^j	Specific action of the robot i, in collision j	
s^j	Joint action taken in collision j	
$h^j = (s^1, s^2, \ldots, s^j)$	History of joint actions played up until (and including) collision j	
S^j	Set of all possible joint action histories until (and including) collision j	
$g_i{:}S^j \mapsto \mathbb{R}$	Gain by robot i as a function of the joint actions played up until (and including) collision j	*gain*
$c_i{:}S^j \mapsto \mathbb{R}$	Cost incurred by robot i as a function of the joint actions played up until (and including) collision j	C_i^C, c
$u_i{:}S^j \mapsto \mathbb{R}$	Utility of player i, given the joint actions played up until (and including) collision j	$u_i(\alpha_i)$
$\mathbb{U} \coloneqq \bigcup_{i \in N} u_i$	Set of utilities of each player as a function of the joint actions played up until (and including) collision j	
$A_i{:}S^j \mapsto \mathbb{R}$	Active time of the player i as a function of the joint actions played up until (and including) collision j	I_i^a
$P_i{:}S^j \mapsto \mathbb{R}$	Passive time of the player $i$as a function of the joint actions played up until (and including) collision j	I_i^p
$l_i(s)$	Cycle length of robot i, given joint action s: $A_i(s) + P_i(s)$	
$\mathrm{EI}(i, s) \coloneqq \frac{A_i(s)}{A_i(s)+P_i(s)}$	Effectiveness index (Kaminka et al., 2010)	$EI_i(s)$
$\mathbb{EI}_{tot}(s)$	Sum of $EI_i(s)$ over all agents $i \in N$	
$\mathrm{AEI}(i, s)$	Aligned effectiveness index, defined as the difference reward (WLU) with respect to global function $\mathbb{EI}_{tot}$, i.e., $\Delta_i^{\mathbb{EI}_{tot}}(s)$	
$C \in \mathbb{N}$	Number of collisions during swarm lifetime	
$T \in \mathbb{R}$	Total swarm lifetime duration	T