frontiers
RESEARCH TOPICS

VALUE AND REWARD BASED LEARNING IN NEUROROBOTS

Topic Editors
Jeffrey L. Krichmar and Florian Röhrbein

frontiers in
NEUROROBOTICS

Cover image provided by Ibbl sarl, Lausanne CH

ISSN 1664-8714
ISBN 978-2-88919-431-5
DOI 10.3389/978-2-88919-431-5

ABOUT FRONTIERS

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

FRONTIERS JOURNAL SERIES

The Frontiers Journal Series is a multi-tier and interdisciplinary set of open-access, online journals, promising a paradigm shift from the current review, selection and dissemination processes in academic publishing.

All Frontiers journals are driven by researchers for researchers; therefore, they constitute a service to the scholarly community. At the same time, the Frontiers Journal Series operates on a revolutionary invention, the tiered publishing system, initially addressing specific communities of scholars, and gradually climbing up to broader public understanding, thus serving the interests of the lay society, too.

DEDICATION TO QUALITY

Each Frontiers article is a landmark of the highest quality, thanks to genuinely collaborative interactions between authors and review editors, who include some of the world's best academicians. Research must be certified by peers before entering a stream of knowledge that may eventually reach the public - and shape society; therefore, Frontiers only applies the most rigorous and unbiased reviews.

Frontiers revolutionizes research publishing by freely delivering the most outstanding research, evaluated with no bias from both the academic and social point of view.

By applying the most advanced information technologies, Frontiers is catapulting scholarly publishing into a new generation.

WHAT ARE FRONTIERS RESEARCH TOPICS?

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

Find out more on how to host your own Frontiers Research Topic or contribute to one as an author by contacting the Frontiers Editorial Office: researchtopics@frontiersin.org

VALUE AND REWARD BASED LEARNING IN NEUROROBOTS

Topic Editors:
Jeffrey L. Krichmar, University of California, Irvine, USA
Florian Röhrbein, Technische Universität München, Garching, Germany

Organisms are equipped with value systems that signal the salience of environmental cues to their nervous system, causing a change in the nervous system that results in modification of their behavior. These systems are necessary for an organism to adapt its behavior when an important environmental event occurs. A value system constitutes a basic assumption of what is good and bad for an agent. These value systems have been effectively used in robotic systems to shape behavior. For example, many robots have used models of the dopaminergic system to reinforce behavior that leads to rewards. Other modulatory systems that shape behavior are acetylcholine's effect on attention, norepinephrine's effect on vigilance, and serotonin's effect on impulsiveness, mood, and risk. Moreover, hormonal systems such as oxytocin and its effect on trust constitute as a value system. This book presents current research involving neurobiologically inspired robots whose behavior is: 1) Shaped by value and reward learning, 2) adapted through interaction with the environment, and 3) shaped by extracting value from the environment.

Table of Contents

frontiers in
NEUROROBOTICS

EDITORIAL
published: 13 September 2013
doi: 10.3389/fnbot.2013.00013

Value and reward based learning in neurorobots

Jeffrey L. Krichmar[1] and Florian Röhrbein[2]*

[1] Department of Cognitive Sciences, Department of Computer Science, University of California, Irvine, CA, USA
[2] Department of Informatics VI, Technische Universität München, Garching, Germany
*Correspondence: florian.roehrbein@in.tum.de

Edited by:
Alois C. Knoll, Technische Universität München, Germany

Keywords: value system, neuromodulation, reinforcement learning, action selection, neurorobotics, reward-based learning, basal ganglia, embodied cognition

Organisms are equipped with value systems that signal the salience of environmental cues to their nervous system, causing a change in the nervous system that results in modification of their behavior. These systems are necessary for an organism to adapt its behavior when an important environmental event occurs. A value system constitutes a basic assumption of what is good and bad for an agent. These value systems have been effectively used in robotic systems to shape behavior. For example, many robots have used models of the dopaminergic system to reinforce behavior that leads to rewards. Other modulatory systems that shape behavior are acetylcholine's effect on attention, norepinephrine's effect on vigilance, and serotonin's effect on impulsiveness, mood, and risk. Moreover, hormonal systems such as oxytocin and its effect on trust constitute as a value system. A recent Research Topic in Frontiers of Neurorobotics explored value and reward based learning. The topic comprised of nine papers on research involving neurobiologically inspired robots whose behavior was shaped by value and reward learning, adapted through interaction with the environment, or shaped by extracting value from the environment.

Value systems are often linked to reward systems in neurobiology and in modeling. For example, Jayet Bray and her colleagues developed a neurorobotic system that learned to categorize the valence of speech through positive verbal encouragement, much like a baby would (Jayet Bray et al., 2013). Their virtual robot, which interacted with a human partner, was controlled by a large-scale spiking neuron model of the visual cortex, premotor cortex, and reward system. An important issue in both biological and artificial reward systems is the credit assignment problem that is, how can a distal cue be linked to a reward. In other words, how can you extract the stimulus that predicts a future reward from all the noisy stimuli that you are faced with? Soltoggio and colleagues introduce the principle of rare correlations to resolve this issue (Soltosggio et al., 2013). By using Rarely Correlating Hebbian Plasticity, they demonstrated classical and operant conditioning in a set of human-robot experiments with the iCub robot.

The notion of value and reward has often been formalized in reinforcement learning systems. For example, Li and colleagues show that reinforcement learning, in the form of a dynamic actor-critic model, can be used to tune central pattern generators in a humanoid robot (Li et al., 2013). Through interaction with the environment, this dynamical system developed biped locomotion on a NAO robot that could adapt its gaits to different conditions. Elfwing and colleagues introduced a scaled version of free-energy reinforcement learning (FERL) and applied it to visual recognition and navigation tasks (Elfwing et al., 2013). This novel algorithm was shown to be significantly better than standard FERL and feedforward neural network RL. Another related method, Linearly solvable Markov Decision Process (LMDP) has been shown to have advantages over RL in optimal control policy (Kinjo et al., 2013). Kinjo and colleagues demonstrated the power of LMDP for robot control by applying the method to a pole balancing task, and a visually guided navigation problem using their Spring Dog robot which has six degrees-of-freedom.

Value does need not be reward-based; curiosity, harm, novelty, and uncertainty can all carry a value signal. For example, in a biomimetic model of the cortex, basal ganglia and phasic dopamine, Bolado-Gomez and colleagues (Bolado-Gomez and Gurney, 2013) showed that intrinsically motivated operant learning (i.e., action discovery) could replicate rodent experiments, in a virtual robot. In this case, phasic dopaminergic neuromodulation carried a novelty salience signal, rather than the more conventional reward signal. In a model called CURIOUSity-DRiven, Modular, Incremental Slow Feature Analysis (Curious Dr. MISFA), Luciw and colleagues showed that curiosity could shape the behavior of an iCub robot in a multi-context environment (Luciw et al., 2013). Their model was inspired by cortical regions of the brain involved in unsupervised learning, as well as neuromodulatory systems responsible for providing intrinsic rewards through dopamine and regulating levels of attention through norepinephrine. Different neuromodulatory systems in the brain may be related to different aspects of value (Krichmar, 2013). In a model of multiple neuromodulatory systems, Krichmar showed that interactions between the dopaminergic (reward), serotoninergic (harm aversion), and the cholinergic/noradrenergic (novelty) systems could lead to interesting behavioral control in an autonomous robot. Finally, in an interesting position paper, Friston, Adams, and Montague suggest that *value is evidence*, specifically log Bayesian evidence (Friston et al., 2012). They propose that reward or cost functions that underlie value in conventional models of optimal control can be cast as prior beliefs about future states, which is simply accumulation of evidence through Bayesian updating of posterior beliefs.

As can be gleaned from reading the papers in the Research Topic, as well as the empirical evidence and studies they are built on, *Value and Reward Based Learning* is an active and broad area of research. The application to neurorobotics is important for several reasons: (1) It provides an embodied platform for testing hypotheses regarding the neural correlates of value and reward, (2) it provides a means to test more theoretical hypotheses on the acquisition of value and its function for biological and artificial systems, and (3) it may lead to the development of improved learning systems in robots and other autonomous agents.

REFERENCES

Bolado-Gomez, R., and Gurney, K. (2013). A biologically plausible embodied model of action discovery. *Front. Neurorobot.* 7:4. doi: 10.3389/fnbot.2013.00004

Elfwing, S., Uchibe, E., and Doya, K. (2013). Scaled free-energy based reinforcement learning for robust and efficient learning in high-dimensional state spaces. *Front. Neurorobot.* 7:3. doi: 10.3389/fnbot.2013.00003

Friston, K., Adams, R., and Montague, R. (2012). What is value-accumulated reward or evidence? *Front. Neurorobot.* 6:11. doi: 10.3389/fnbot.2012.00011

Jayet Bray, L. C., Ferneyhough, G. B., Barker, E. R., Thibeault, C. M., and Harris, F. C. Jr., (2013). Reward-based learning for virtual neurorobotics through emotional speech processing. *Front. Neurorobot.* 7:8. doi: 10.3389/fnbot.2013.00008

Kinjo, K., Uchibe, E., and Doya, K. (2013). Evaluation of linearly solvable Markov decision process with dynamic model learning in a mobile robot navigation task. *Front. Neurorobot.* 7:7. doi: 10.3389/fnbot.2013.00007

Krichmar, J. L. (2013). A neurorobotic platform to test the influence of neuromodulatory signaling on anxious, and curious behavior. *Front. Neurorobot.* 7:1. doi: 10.3389/fnbot.2013.00001

Li, C., Lowe, R., and Ziemke, T. (2013). Humanoids learning to walk: a natural CPG-actor-critic architecture. *Front. Neurorobot.* 7:5. doi: 10.3389/fnbot.2013.00005

Luciw, M., Kompella, V., Kazerounian, S., and Schmidhuber, J. (2013). An intrinsic value system for developing multiple invariant representations with incremental slowness learning. *Front. Neurorobot.* 7:9. doi: 10.3389/fnbot.2013.00009

Soltosggio, A., Lemme, A., Reinhart, F., and Steil, J. J. (2013). Rare neural correlations implement robotic conditioning with delayed rewards and disturbances. *Front. Neurorobot.* 7:6. doi: 10.3389/fnbot.2013.00006

Received: 07 August 2013; accepted: 25 August 2013; published online: 13 September 2013.

Citation: Krichmar JL and Röhrbein F (2013) Value and reward based learning in neurorobots. Front. Neurorobot. ***7****:13. doi: 10.3389/fnbot.2013.00013*

This article was submitted to the journal Frontiers in Neurorobotics.

frontiers in
NEUROROBOTICS

ORIGINAL RESEARCH ARTICLE
published: 12 March 2013
doi: 10.3389/fnbot.2013.00004

A biologically plausible embodied model of action discovery

Rufino Bolado-Gomez and Kevin Gurney*

Department of Psychology, Adaptive Behaviour Research Group, University of Sheffield, Sheffield, UK

Edited by:
Florian Röhrbein, Technische Universität München, Germany

Reviewed by:
Suguru N. Kudoh, Kwansei Gakuin University, Japan
Mehdi Khamassi, Centre National de la Recherche Scientifique, France

****Correspondence:***
Kevin Gurney, Department of Psychology, Adaptive Behaviour Research Group, University of Sheffield, Western Bank, Sheffield S10 2TP, UK.
e-mail: k.gurney@sheffield.ac.uk

During development, animals can spontaneously discover action-outcome pairings enabling subsequent achievement of their goals. We present a biologically plausible embodied model addressing key aspects of this process. The biomimetic model core comprises the basal ganglia and its loops through cortex and thalamus. We incorporate reinforcement learning (RL) with phasic dopamine supplying a sensory prediction error, signalling "surprising" outcomes. Phasic dopamine is used in a cortico-striatal learning rule which is consistent with recent data. We also hypothesized that objects associated with surprising outcomes acquire "novelty salience" contingent on the predicability of the outcome. To test this idea we used a simple model of prediction governing the dynamics of novelty salience and phasic dopamine. The task of the virtual robotic agent mimicked an *in vivo* counterpart (Gancarz et al., 2011) and involved interaction with a target object which caused a light flash, or a control object which did not. Learning took place according to two schedules. In one, the phasic outcome was delivered after interaction with the target in an unpredictable way which emulated the *in vivo* protocol. Without novelty salience, the model was unable to account for the experimental data. In the other schedule, the phasic outcome was reliably delivered and the agent showed a rapid increase in the number of interactions with the target which then decreased over subsequent sessions. We argue this is precisely the kind of change in behavior required to repeatedly present representations of context, action and outcome, to neural networks responsible for learning action-outcome contingency. The model also showed cortico-striatal plasticity consistent with learning a new action in basal ganglia. We conclude that action learning is underpinned by a complex interplay of plasticity and stimulus salience, and that our model contains many of the elements for biological action discovery to take place.

Keywords: phasic dopamine, basal ganglia, reinforcement learning, synaptic plasticity, intrinsic motivation, action selection, operant behavior

1. INTRODUCTION

How can animals acquire knowledge of their potential *agency* in the world—that is, a repertoire of actions enabling the achievement of their goals? Moreover, how can this be done spontaneously without the animal being instructed, or without having some overt, primary reward assigned to successful learning? In this case we talk of *action discovery*, and call the learning *intrinsically motivated* (Oudeyer and Kaplan, 2007). It is typical of the kind of action learning found in the young as they discover their ability to influence their environment (Ryan and Deci, 2000). We argue that an understanding of the biological solution to these problems will lay foundations for a robust and extensible solution to skill acquisition in artificial agents like robots. We now outline the theoretical, behavioral and neuroscientific background to the paper.

The relation between actions and outcomes is not a given—the animal must use reinforcement learning (RL) to acquire *internal models* of action-outcome contingencies associating context, action and outcome, and be able to deploy the relevant action given a context and a desired outcome or goal. Consider, for example, the act of switching on a particular room light. There is a *forward, prediction model*: "if I am in front of this switch and I press it, the light in the corner will come on." There is also an *inverse model*: "if I need the light in the corner to come on, I need to press this switch here" (Gurney et al., 2013). The framework for action-outcome acquisition we propose is shown in **Figure 1A**.

We suppose that the internal models of action-outcome are encoded in associative neural networks. In order for these associations to be learned (possibly via some kind of Hebbian plasticity), representations of the motor action, sensory context, and the sensory outcome must be repeatedly activated in the relevant neural systems. This requires a transient change in the action selection propensities of the agent—its so-called selection *policy*—so that the to-be-learned action occurs more often than other competing actions. The repeated presentation of the representation of outcome is taken care of by physics; if the switch is pressed the agent doesn't have to do any more work to make the light come on.

The process of *repetition bias* in policy must continue until the new action-outcome has been learned, and then cease. We therefore require that the agent's policy is modulated by the predictions being developed in the forward model; as the outcome is predicted, the repetition bias must be reduced and, ultimately,

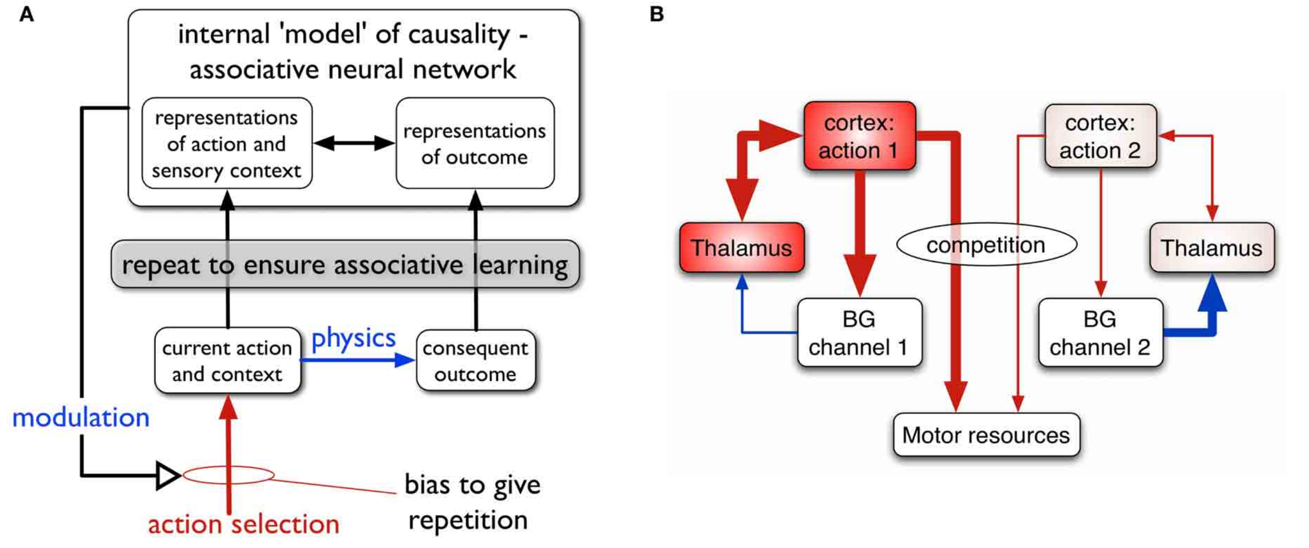

A
internal 'model' of causality - associative neural network
representations of action and sensory context
representations of outcome
repeat to ensure associative learning
current action and context
physics
consequent outcome
modulation
bias to give repetition
action selection
B
cortex: action 1
cortex: action 2
Thalamus
competition
Thalamus
BG channel 1
BG channel 2
Motor resources

FIGURE 1 | (A) Scheme for learning action-outcome associations—see text for details. **(B)** Loops through basal ganglia, thalamus, and cortex performing action selection in the animal brain. Two competing action channels are shown. The channel on the left encoding action 1 has a higher salience than that for channel 2. It has "won" the competition for behavioral expression in basal ganglia which has therefore released inhibition on its thalamic channel target, thereby allowing the corresponding thalamo-cortical loop to build up activity. Blue/red lines show inhibition/excitation, respectively and the width of lines encodes signal strength.

removed. In general, we propose that the intrinsically motivated behavior is driven by novelty—the agent engages with the situation because the target object (e.g., the switch) is novel or that the "surprise" of the outcome on first encountering the light cause some plastic change in the policy engine.

In this paper, one of our aims is to understand the dynamics of repetition bias. To proceed, we therefore turn to the machinery for solving the problem of action selection, and policy encoding in the animal brain. We and others (Mink and Thach, 1993; Doya, 1999; Redgrave et al., 1999; Houk et al., 2007) have argued that a set of subcortical nuclei—the basal ganglia—are well placed to help solve this problem, and act as the policy engine or "actor" in the vertebrate brain.

The basal ganglia are connected in closed looped circuits with cortex, via thalamus (**Figure 1B**). Their outputs are tonically active and inhibitory, and selection is achieved by selectively releasing inhibition on cortico-thalamic targets that encode specific actions (Deniau and Chevalier, 1985). We refer to the neural representation of an action, and its anatomical instantiation, as it runs through these loops as an action *channel* (Redgrave et al., 1999). Release of inhibition on a thalamic channel allows activity in its corresponding thalamo-cortical loop to build up and eventually reach a threshold which allows behavioral expression of the action. More details of this architecture are given in the section 2.

Within this framework we can identify two components of a successfully established action encoding. First, within cortex, there must be the correct specific patterning of contextual (sensory, cognitive, and possibly homeostatic) and preparatory motor features. We refer to this as the *action request* and the overall level of activity in the action request is supposed to signal its urgency or *salience*. Channels within basal ganglia are subject to competitive processes therein and action requests with the highest salience are those that are selected. Clearly, one mechanism then for inducing repetition bias would be to enhance the salience of requests for the action to be discovered (Redgrave et al., 2011). A second component of action encoding occurs at the level of the main basal ganglia input nucleus—the striatum. Here, the cortical action request must selectively activate a subset of the striatal projection neurons, or so-called medium spiny neurons (MSNs). In this way, a striatal channel is established which can "listen" to the action request (Redgrave et al., 2011). For a neuron computing a weighted sum of inputs, this occurs by a process of matching the pattern of synaptic efficiencies to the strengths of action request components, resulting in a proportional encoding of salience. Evidence for such an encoding of salience in striatum has recently been provided by human fMRI studies (Zink et al., 2006).

To establish channel selectivity in MSNs requires cortico-striatal plasticity whose dynamics depend on the animal's behavior and resulting environmental feedback. The theory of RL encompasses exactly this scenario (Sutton and Barto, 1998) and so it is not surprising that cortico-striatal plasticity has been the subject of study using the classic algorithms of RL (such as temporal difference learning) with reinforcement contingent on biological reward. The reinforcement signal in this scenario is supposed to be supplied by short-latency phasic dopamine bursts which encode a *reward* prediction error (Schultz et al., 1997).

In contrast to this, we have recently argued that such signals are unlikely to be associated with primary reward as such, because they occur too soon to be the result of a relatively lengthy process of explicit evaluation in which the stimulus is assigned rewarding, neutral or aversive status. Instead, we propose that phasic dopamine primarily encodes a *sensory* prediction error which may be used to guide acquisition of goal-directed actions (Redgrave and Gurney, 2006; Redgrave et al., 2008).

This interpretation does not preclude a role for reward in modulating the phasic dopamine signal, and these issues are explored further in section 4.3 in the "Discussion." However, under the sensory prediction error hypothesis, action acquisition is supposed to take place with the following sequence of events. An animal performs an action which results in an unexpected outcome. The phasic component of the outcome (not requiring computation of value) causes midbrain dopamine neurons to fire (Comoli et al., 2003) eliciting a phasic release of dopamine in striatum (the mechanistic substrate for this is described in more detail in section 2.5.4). This then acts to induce cortico-striatal plasticity associated with recently active action-based representations in cortex, and corresponding striatal responses. If repetition bias is operative, this sequence of events is repeated and MSNs in striatum can become selectively responsive to the action request which is required to elicit the environmental event. It is also possible that this plasticity can itself contribute to repetition bias, as each increment in the match between the patterns of synaptic strengths and action request should make the selection of the action more likely. However, one of the questions we address here is the extent to which this can be wholly responsible for transient policy changes seen *in vivo*. Fortunately there is a recent behavioral study (Gancarz et al., 2011) which provides data we can use to constrain the possibilities here.

At the neuronal level, electrophysiological data from studies in cortico-striatal plasticity have provided a complex and often confusing picture. Both long term depression (LTD) and long term potentiation (LTP) have been observed at glutamatergic (excitatory) cortical synapses on MSNs, and their expression is dependent on dopamine (Reynolds and Wickens, 2002; Calabresi et al., 2007). Further, this dependence is linked to specific dopamine receptor types in different populations of MSNs (Pawlak and Kerr, 2008) and has spike timing dependent characteristics (Fino et al., 2005; Pawlak and Kerr, 2008). This phenomenological complexity has hampered the development of a quantitative functional understanding of cortico-striatal plasticity. In particular, given the limitations of much *in vitro* data with regards to the class of MSNs based on their dopamine receptors, we would expect this data to display *mean* characteristics rather than those of one class alone. This is then necessarily reflected in models (Thivierge et al., 2007) which may account for spike timing and dopaminergic effects, but rely on data which is agnostic about the MSN classification.

Recently this impasse has been overcome in a study in striatal slices by Shen et al. (2008), in which the different classes of MSN could be reliably identified. In addition, this study deployed a variety of techniques to investigate the effects of dopamine depletion, thereby providing data at different levels of intrinsic dopamine. This study formed the basis of our recent spiking model of cortico-striatal plasticity (Gurney et al., 2009) which we adapt here for rate-coded neurons.

Within the framework described above, we seek to address in this study, the following questions about action discovery. Having proposed that action-outcome discovery depends on a repetition bias in selection policy, what are the mechanisms responsible for this? In particular what are the relative roles for enhanced cortical salience ("louder action request"), and better cortico-striatal transmission ("listening to the request") induced by dopamine modulated cortico-striatal plasticity? If increased cortical salience is required, what is its origin? How should salience and plasticity be moderated by the development of the prediction model? Is any cortico-striatal plasticity observed in the model consistent with the requirements of long term afferent/synaptic-strength pattern matching? To ensure a biologically plausible solution, we take advantage of recent behavioral data (Gancarz et al., 2011), made possible with our embodied (robotic) approach, and recent *in vitro* data (Shen et al., 2008) on cortico-striatal plasticity.

2. MATERIALS AND METHODS

2.1. *In vivo* EXPERIMENTAL COMPARISON

The robot task mimics an *in vivo* counterpart (Gancarz et al., 2011) in which rats spontaneously poke their snouts into one of two poke-holes in a small operant chamber (**Figure 2A**). Each experiment was conducted over 16 days with the rat exposed to a single 30 min session in the operant chamber each day. Critically, the animals were not food or liquid deprived, and were therefore not motivated by any extrinsic reward. The ambient light condition was complete darkness, and the rats were free to move around the chamber. In a first *habituation phase* (the first 6 days), there were no consequences to the animal making a snout entry into either poke hole. In a second *response contingent phase* (subsequent 10 days) one of the snout holes was designated the "active hole" and a snout entry here could cause a phasic light stimulus to flash briefly (mechanistically, this was achieved with two lights, one near the snout holes and one at the back of the chamber). This light flash was the only source of behavioral reinforcement and its occurrence was under control of a variable interval (VI) schedule with mean of 2 min. That is, there was a random interval (with mean 2 min) between potential snout-entry/light-flash pairings; premature snout entry into the active hole before completion of this interval caused no light flash. Snout entry into the active hole was designated an *active response* (with or without any consequent light flash) and entry into the other hole, an *inactive response*. The labeling of the snout holes in the response contingent phase is carried across to the habituation phase, although here it constitutes an arbitrary distinction. Thus, "active responses" in the habituation phase are simply those responses directed to the snout hole which becomes active during response contingency.

Relevant results of this experiment are shown in **Figure 3**. In that experiment, animals were divided in "low and high responders" according to a pre-experimental assay of overall levels of motoric activity (Gancarz et al., 2011). Here, we have averaged the data across the two groups. **Figure 3A** shows that there is no significant difference in responding to the two snout holes during the habituation phase. However, there is a clear difference during the response contingent phase; the animals spent more time engaging with the active snout hole. Other trends indicated are a gradual development, and subsequent decline, in the preference for the active hole during the response contingent phase. **Figure 3B** shows the mean behavior with each session during the response continent phase. There is a clear initial high number of

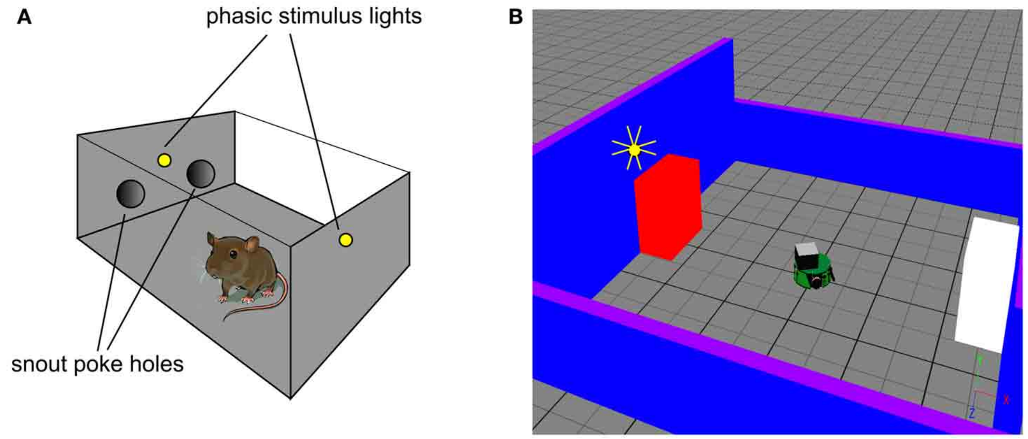


FIGURE 2 | *In vivo* experimental paradigm of Gancarz et al. (2011) (panel A) and our embodied in silico counterpart (panel B). **(A)** Shows the small test chamber used with rats undergoing instrumental learning. One side of the chamber has two poke holes with a light above them. Rat snout entry into the "active" poke hole may cause the two lights to flash and the active hole may be either one (for a particular rat). **(B)** Shows the virtual world created as a counterpart to that in **(A)**. A simulated Khepera I robot replaces the rat, and snout holes are replaced by colored blocks. Only the red block is ever designated the active one, and the white block corresponds to the inactive poke hole. There is a point-light located at the top of the red block which may flash if the robot bumps into the red block.

active and inactive responses, and a subsequent decline in both during the session.

2.1.1. Fixed-ratio variant

While the VI schedule provides valuable data to constrain the model, the action discovery paradigm, as encountered ethologically, is likely to be governed by less random reinforcement. In particular, if reinforcement is reliably given at every successful interaction with the target object, we have *fixed-ratio* (FR) schedule with ratio one (FR1). We therefore also ran simulations with this schedule.

At the time of completing this work, the corresponding biological data was not yet available and so the behavioral outcomes of the simulated agent became predictions for a similar *in vivo* experiment. However, during revision of this paper, we became aware that the laboratory responsible for the study described above had just published a followup which used an FR1 schedule (Lloyd et al., 2012). Our predictions were therefore immediately put to the test. The relevant data for the FR1 schedule from the study in (Lloyd et al., 2012) are shown in **Figures 3C,D**. Only active responses are shown in order to facilitate a comparison with the VI data described above (inactive responses are similar to that for the VI case). For FR1 training, the peak number of responses in the response contingent phase occurs in the first day of that phase, and shows a rapid decline thereafter (**Figure 3C**). In contrast, the peak response for VI training occurs after the first day of response contingency and shows a more gradual decline. Within a session, the FR1 schedule shows a steeper decline than its VI counterpart (**Figure 3D**).

2.2. SIMULATED ROBOT WORLD

We used simulation of a small autonomous robot in an arena with stimulus objects to mimic the *in vivo* experiment of Gancarz et al. (2011)—see **Figure 2B**. The robot was the K-Team Khepera (Mondada et al., 1999) and simulation used the Webots (v6.3.2) software environment (Cyberbotics, 2010a,b). The arena consisted of a tiled ground-plane (60 cm × 60 cm) with blue walls (two each of 10 cm and 20 cm height). The stimuli comprised two static blocks (5.9 cm by 9.8 cm by 10 cm) colored red and white, that played the role of the poke holes. Unlike the snout holes in the experiment with rats, the blocks were spatially well separated (opposite sides of the arena). For the rats, their use of local tactile (whisker-based), rather than wide-field visual information, means that the snout holes are well separated in the sensory space of the animal. This is what we achieve in the visual modality using the arrangement in **Figure 2B**. A light source that can flash briefly was located above the red block (there was no need for additional, rear-mounted lighting to cause sensor response in the Khepera). This light is triggered by the robot bumping into the red block (albeit possibly under VI-schedule control). The red block is therefore a surrogate for the active snout hole in the *in vivo* experiment of Gancarz et al. (2011).

The robot has a cylindrical body shape with height 3 cm and diameter 5.6 cm. Each wheel can be separately controlled to go forwards or backwards. There are eight infrared sensors in a radial configuration that were used for proximity detection in an "exploratory" behavior which also required avoiding contact with objects. The two front sensors were also used to detect the light flash. We used an RGB camera with 64(wide) × 1(high) pixel array mounted on top of the Khepera's central body to detect the colored blocks, and a binary tactile sensor at the front to detect bumping into objects. The supplementary material contains a short video showing the actions available to the virtual robot.

2.3. THE VIRTUAL ROBOT CONTROL ARCHITECTURE: OVERVIEW

The complete virtual embedded robot model is shown in **Figure 4**. It comprises three principal components: the *virtual*

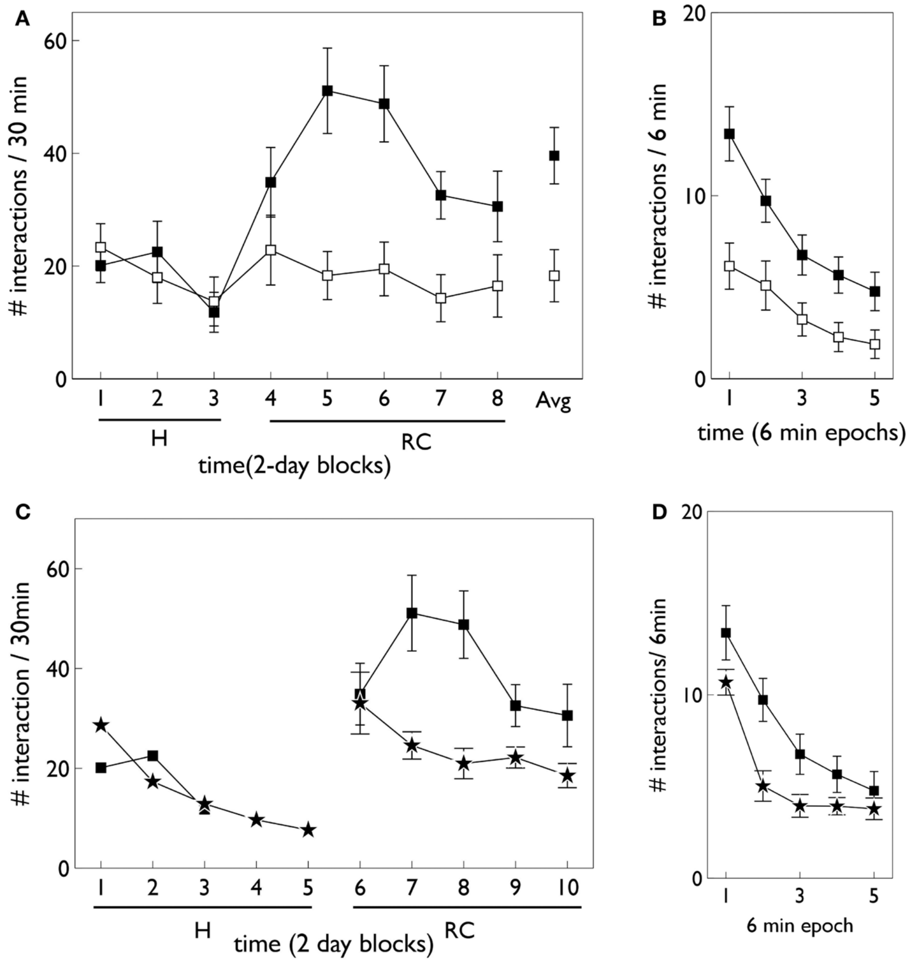


FIGURE 3 | Behavioral data adapted from the *in vivo* studies of Gancarz et al. (2011) (study 1) and Lloyd et al. (2012) (study 2). (A,B) For variable interval (VI) training from study 1. **(A)** Shows the number of inactive and active responses in each 2-day period (averaged over the two 30 min sessions therein) with white and black symbols, respectively. The habituation and response contingent phases (see text) are designated "H" and "RC," respectively, and the average response during the response contingent phase is shown on the extreme right as "Avg." **(B)** Shows the within-session behavior during the response contingent phase. Results are averaged over all 10 days of this phase and means are reported for each epoch of 6 min duration during the 30 min sessions. Error bars in both panels are the mean of the standard errors for the low and high responding animals (as originally reported in study 1). **(C)** Shows active responses (star-shaped data points) from a fixed-ratio (FR1) schedule reported in study 2. Also shown for comparison are the active response in **(A)** (black squares). Note, there were more days in the habituation phase of study 2, and error bars in the habituation phase are not shown. **(D)** Is a counterpart to **(B)** with FR1 data shown by stars, and the VI data from **(B)**, replicated for comparison (black squares).

robot—referencing its hardware, motor plant and peripheral sensors; an *embedding architecture*, or *engineered surround*, and the *biomimetic core* model. This partitioning scheme has been described in our previous work (Prescott et al., 2006; Gurney, 2009; Gurney and Humphries, 2012). The idea is to separate off the biomimetic model which is the primary subject of study, from less biologically realistic, and somewhat "engineered" components which are, nevertheless, required to produce a complete, behaving agent. In this way, we package together those elements of the architecture which are part of the model proper, and which encapsulate our hypotheses about brain function, and separate them from elements which are predicated on our hypothesis set. Thus, if we identify the cause of deficiencies in behavioral outcome with issues in the embedding architecture, we can be sure we are not falsifying hypotheses embodied in the biomimetic core. It is not necessary for a part of the biomimetic core to be a neural network; algorithmic elements are also candidates if they implement key model functions.

The key for this approach to work is the signal interface between surround and core. Thus, just as in modular software, the signals must have the same interpretation for both components either side of this interface. In our context, the embedding architecture must supply signals to a "sensory cortical" area in the biomimetic core that can interpret them as saliences for action requests, as well as any internal state variables required to modulate them. Sensory indication of phasic events must be made available to the dopamine system, and the motoric output of the

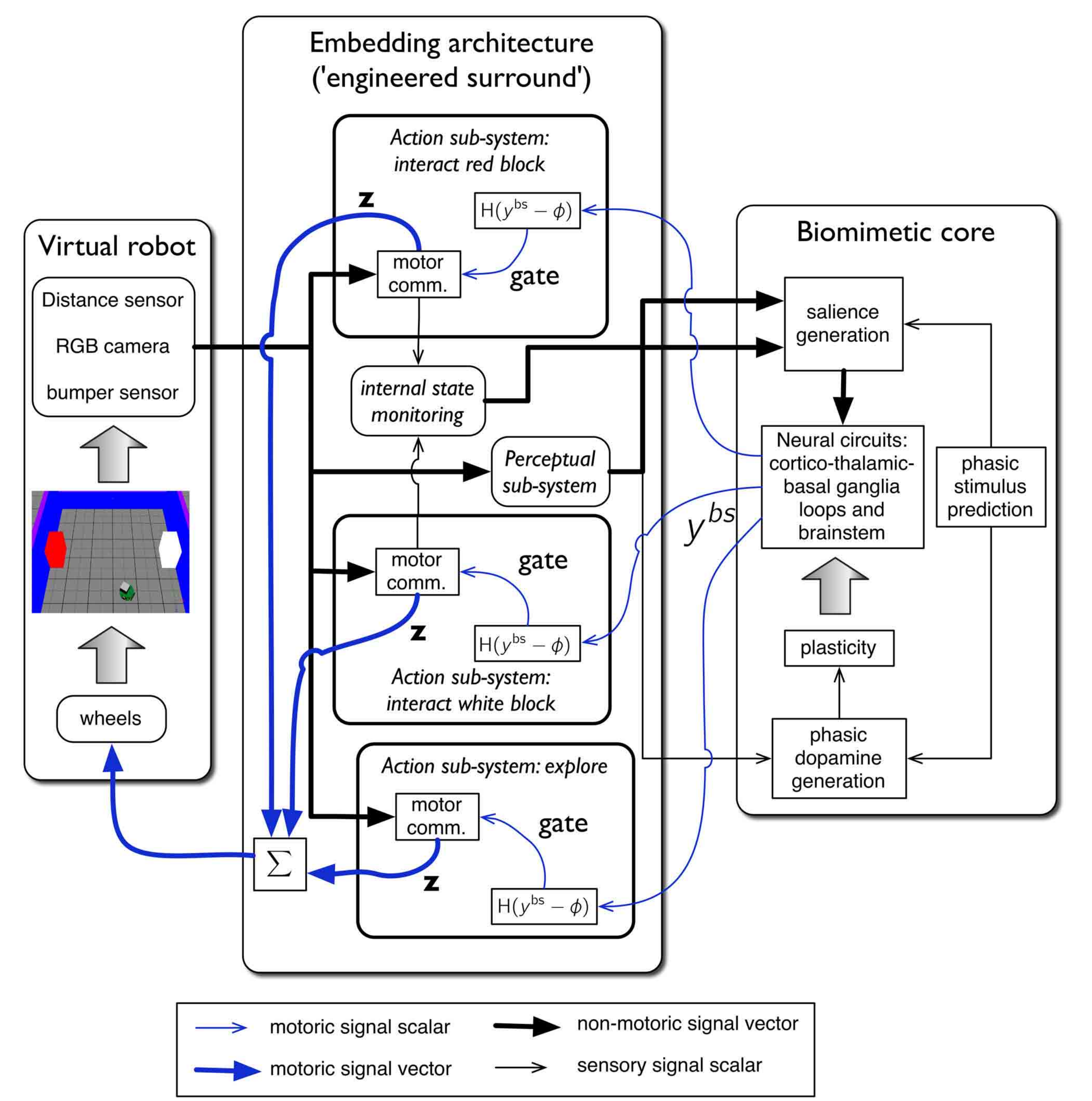


FIGURE 4 | The virtual robot control architecture, and its interaction with the robot and environment. The virtual Khepera robot is endowed a range of sensors and the motor output is locomotion via a pair of wheels. The architecture is split into embedding, and biomimetic core, components. The embedding architecture contains three action-subsystems: two for approaching-and-bumping into each of the red and white blocks ("interact red block," "interact white block"), and one ("explore") for randomly roaming the arena while avoiding object contact. Within each action subsystem the motor command units are designated "motor comm." The biomimetic core contains a biologically plausible circuit (representing basal ganglia, and its connectivity with cortex, thalamus, and brainstem), a phasic stimulus prediction mechanism, a source of phasic dopamine, and the new learning rules for basal ganglia plasticity. Other symbols and components are labeled as in the main text.

biomimetic core must comprise a "selection signal" that can be used to gate actions. This signal interface is precisely that shown in **Figure 4**. We now go on to describe each major system in more detail.

2.4. EMBEDDING ARCHITECTURE

The embedding architecture is based on that described by Prescott et al. (2006). The agent is supposed to have a fixed repertoire of behaviors or action-sequences, and the enactment of each one is encapsulated in an action subsystem. In the current model there are three such behaviors:

> **Explore:** move around the arena and avoid obstacles (blocks and walls).
>
> **Interact with the red block:** orient to the red block, approach it, and perform a controlled "bump" into it. This latter comprises, in turn, the following sub-actions: bump once against the red block, move backwards, stop, and then slowly approach the red block again.
>
> **Interact with the white block:** this is identical to its counterpart for the red block, except actions are directed to the white cube.

The block-interaction behaviors are surrogates for the snout hole poking in the *in vivo* experiment of Gancarz et al. (2011). The key difference in outcome between the two behaviors is that interaction with the red block causes the light flash—it comprises the active response—whereas interaction with the white block has no consequences and comprises the inactive response.

The granularity of behavior encoded in each action sub-system is clearly quite coarse; we have already noted that they each comprise small action sequences. Thus, they have similarities with the *fixed action patterns* (FAPs) of the ethologists (Lorenz, 1935; Tinbergen, 1951) and the *options* used in hierarchical RL (Barto et al., 2004). This is not a drawback in the current model as we are primarily interested in the basic principles of adaptive aspects of behavior with novel stimuli, and any consequent plasticity; the precise semantics of each action are not important. Further, the behaviors we encode are not as rigid as FAPs or options, as our method of behavioral maintenance—an excitatory recurrent connection within the motor cortex (see "Appendix")—allows the behaviors to be interrupted by "exploration" if this has sufficiently high salience. We will revisit the issues surrounding action granularity in the section 4.

Within each action-subsystem, the sequencing of primitive actions into behaviors is accomplished in a *motor command unit*. These units make use of sensory information to trigger various events in the sequence. The "explore" behavior is governed by the infra-red sensors which detect distance to objects in the robot's path, thereby allowing locomotion while avoiding objects. The block-interaction behaviors use camera information to identify, and orient to the blocks, and the bumper sensor to know when contact has been made.

The motor output of each motor command unit is 2-vector $\mathbf{z} = (z_l, z_r)$ whose components indicate the desired speed for each robot wheel (left and right) to enact the current segment of behavior. The motor command units are not neural networks but conventional procedural code which use sensor information to trigger the next action component in the sequence, and update $\mathbf{z}$ at each time step. If the behavior in the action subsystem has been selected by the biomimetic core, then the corresponding speed-output vector is sent forward to be averaged with output vectors from any other selected sub-systems. In this way, multiple selected actions are blended together to produce a final behavior. This forces a strong test of the action selection capability of the biomimetic core model which must prevent over-expression of such multiple action selection.

The selection criterion for an action subsystem i, is that the corresponding brainstem output signal from the biomimetic core, y_i^{bs} should exceed some threshold ϕ. That is, $H(y_i^{bs} - \phi) = 1$, where $H()$ is the Heaviside function. In our simulations $\phi = 0.5$.

The *perceptual sub-system* supplies sensory information for generation of the salience of the action requests for the block interaction behaviors. In the first instance, this is quite simple; the perceptual subsystem detects the presence of the red/white block in the visual field and triggers a salience for the red/white block-interaction behavior. However, the salience of the blocks is subject to a variety of additional processes driven by sensory habituation and perceived novelty of the stimulus. These processes are based on biological notions and so we reserve them for the biomimetic core. They also depend on the status of the block-interaction behaviors (completion of a block interaction cause an habituation increment). Therefore, these two command units also provide signals to an *internal state monitoring* unit that indicate if their respective sequences have recently been completed. This unit also provides a representation of the motivation to explore the arena, governing the selection of the "explore" action sub-system. Finally, the perceptual subsystem also provides a signal to the dopamine system about phasic events such as the light flash.

2.5. THE BIOMIMETIC CORE

The biomimetic core comprises several functional blocks (see **Figure 4**)—we now deal with each in turn.

2.5.1. Prediction of phasic stimuli

A key component in our model is the idea that the phasic outcome of the interaction with the blocks (the light flash) is subject to prediction via an internal model. This prediction is then used to modify the salience of objects in the visual field at the time of the light flash (the blocks) and also to form a sensory prediction error which forms the basis for the phasic dopamine signal.

Prediction is believed to be a fundamental process at the heart of perception and cognition (Bar, 2007; Bubic et al., 2010; Friston, 2010; Gurney et al., 2013) and is, in general, a complex neural process requiring substantial model resources. However, formalizing a *phenomenological* model of prediction of the phasic light flash is straightforward if we assume that the latter is represented by a single scalar feature $y_f(t)$ whose value is binary: a 1 signals the detection of a light, and 0 its absence (no light flash). The prediction is then a real-valued scalar between 0 and 1, where values close to 1 or 0 are strong predictions that the light will flash on or be absent, respectively.

To proceed further, consider the set of times $\{t_i\}$ when the light flash *might* occur (during block-interactions), where i indexes the block-interactions over the entire (multi-day) experiment. We distinguish between the phasic *manifestation* of the prediction $y_f^*(t_i)$, at discrete time t_i, and the internal *latent* representation of the prediction $y_f^{(*)}(t)$ which exists at all times t. The phasic prediction is supposed to correspond to phasic neural activity, whereas its latent counterpart is encoded in the structure (synaptic weights) of the internal model of prediction.

The model we use is phenomenological and we use a similar approach, based on exponential rise and decay as is used with habituation (Marsland, 2009). Thus, if a phasic event (light flash) occurs at t_i, the prediction is increased according to the recursive relation

$$y_f^{(*)}(t_i + \delta t) = 1 - k(1 - y_f^{(*)}(t_i)) \quad \text{where } 0 < k < 1 \tag{1}$$

This occurs within days and across day boundaries, because we assume no day-to-day unlearning of the internal model for prediction of phasic outcome. The definition is completed by defining the effect of the first reinforcing event: $y_f^{(*)}(t_1 + \delta t) = 0.2$. If, after a block-interaction, there is a non-zero prediction of a phasic outcome which was not delivered (no light flash), then the prediction is updated according to

$$y_f^{(*)}(t_i + \delta t) = k y_f^{(*)}(t_i) \tag{2}$$

(both within and between days). Thus, latent prediction is constant for the intervals $t_i < t \leq t_{i+1}$. Then, when activated by sensory cues, the model delivers the phasic prediction $y_f^*(t_i) = y_f^{(*)}(t_i)$. For all our simulations, $k = 0.95$. **Figure 5A** shows a cartoon of a typical sequence of events and the resulting predictions.

2.5.2. Salience generation

Salience for the block interaction behaviors is initiated by the perceptual subsystem being activated by the presence of a colored block in the field of view. This generates a nominal salience value which is then subject to habituation, dishabituation, and possibly a sensitization due to novelty. We refer to the nominal salience of the colored blocks modulated by (dis)habituation as the *intrinsic* salience of the blocks. This may be augmented by a separate *novelty salience*; both contributions are detailed below.

Habituation is defined as "a behavioral response decrement that results from repeated stimulation and that does not involve sensory adaptation/sensory fatigue or motor fatigue" (Rankin et al., 2009). Evidence for habituation in the *in vivo* experiment of

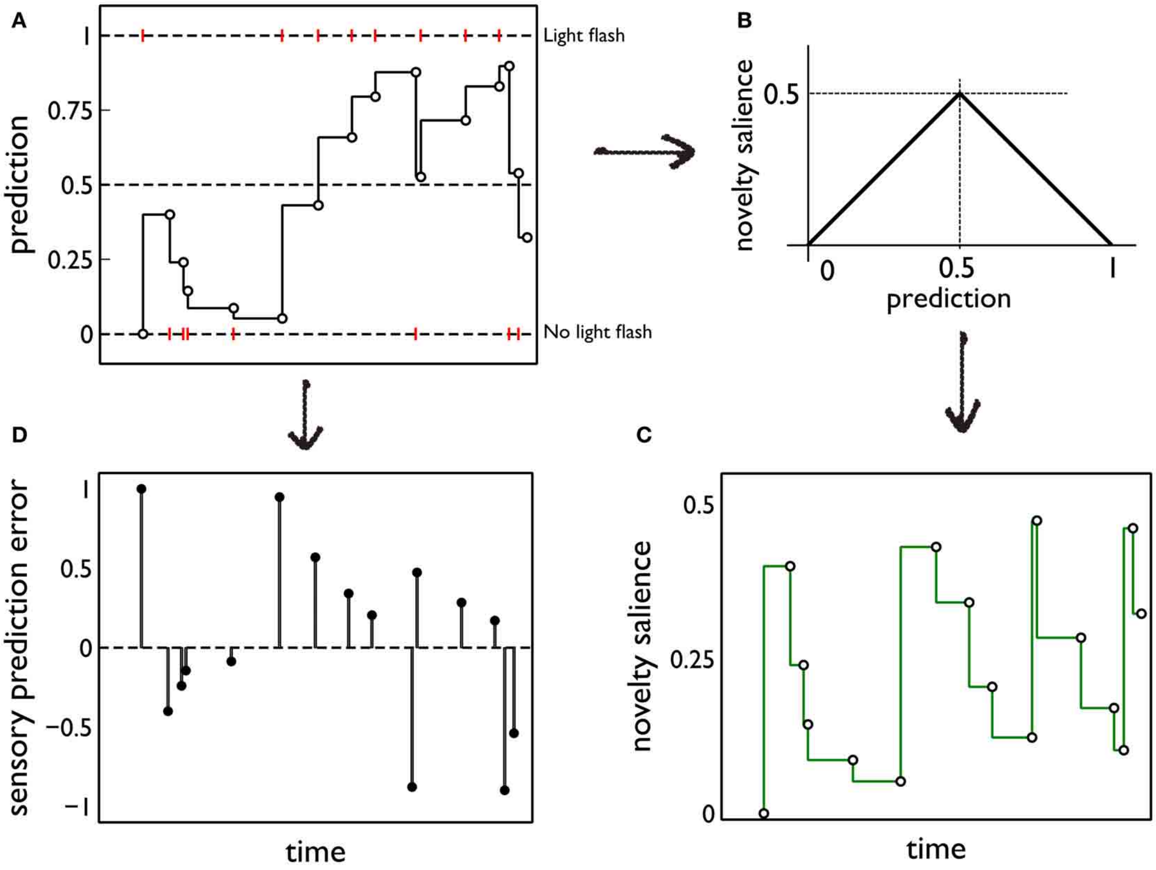


FIGURE 5 | Prediction and its deployment for novelty salience and sensory prediction error under a simple phenomenological model. **(A)** The red markers indicate the presence or absence of phasic outcome (light flash) during each interaction with the red (active) block. The latent prediction, $y_f^{(*)}(t)$, is shown as the solid line and the phasic prediction, $y_f^*(t_i)$, by the open markers. **(B)** The translation of prediction into novelty salience. **(C)** The time course of novelty salience corresponding to the prediction in **(A)**, obtained via the mapping in **(B)**. Open circles represent the salience perceived at each block interaction, when the block is in view. These bouts of block-perception are longer than the observation of the light flash, but we identify each interaction with a point-time marker for simplicity. The continuous line is a formal mapping of the latent prediction using Equation (3). **(D)** The sensory error signal derived from **(A)**.

Gancarz et al. (2011) comes from close examination of the data in **Figure 3**. There is clear evidence of a decline of inactive responses within each session (day) of the response contingent phase. There is also some indication of similar trends across days with in each phase of the experiment. Thus, linear fits to the means of inactive responses have a negative slope within each phase and, for the habituation phase, this was a significant trend (Gancarz et al., 2011). The inactive responses are least likely to be subject to any contribution from novelty and represent (as far as possible) a control stimulus. We therefore assume any behavioral changes in inactive responses are a consequence of the dynamics of the intrinsic salience of the stimuli. Thus, we incorporated salience habituation processes, both across, and within days, resulting in the decline of the intrinsic salience of both blocks on these two time scales.

It might be thought that the decline within a session could be due to a general "fatigue." However, this can be ruled out for several reasons. First, there is little effort in a snout poke response, and it is part of the normal behavioral repertoire of the rat. Second, there is ample use in behavioral studies of testing rats for much longer than the 30 min sessions used here. Third, in the study by Lloyd et al. (2012), animals confronted with a more difficult (VI) learning schedule, showed more responses within a session than those under a less demanding, fixed-ratio schedule. We therefore conclude that decrements in response are due to genuine adaptive neural processes.

Habituation is usually accompanied by a dishabituation process whereby, presentation of alternative stimuli, or a "rest period," allows habituated behavioral responses to recover to previously observed levels (Groves and Thompson, 1970; Rankin et al., 2009). These complementary processes may be modeled using simple exponential forms (Marsland, 2009), and we used this general approach in the following way. Thus, let $S^{i,j}_{\text{int}}$ be the intrinsic salience *during* the jth block-interaction on day i, given the associated block is in the visual field. Within a session, we do not update salience from moment to moment, but rather after each complete interaction with the block. This is consistent with recent ideas about habituation that include reference to response rate change in operant tasks (McSweeney and Murphy, 2009; Rankin et al., 2009). Therefore at the start of the $(j+1)$th interaction, $S^{i,j+1}_{\text{int}} = \gamma_b S^{i,j}_{\text{int}}$, with $\gamma_b < 1$. At the start of the next day, there is a re-initialization $S^{i+1,1}_{\text{int}} = \gamma_a S^{i,1}_{\text{int}}$, where $\gamma_a < 1$. Typically, as a result of this, there is dishabituation between days (so that, if $\hat{j}$ is the last interaction on day i, $S^{i,\hat{j}}_{\text{int}} < S^{i+1,1}_{\text{int}}$). Parameters were $S^{1,1}_{\text{int}} = 0.45$, $\gamma_a = \gamma_b = 0.95$.

We now suppose there may be an additional salience contribution to the target block interaction associated with the surprising phasic outcome (light flash). Thus, we make the hypothesis that objects or features in the perceptual field when a surprising phasic event occurs, acquire *novelty salience* by a process of "inheritance" or generalization from the surprise of the simple phasic outcome (e.g., light). This is an extension to neutral stimuli of the observation that sensitization usually occurs during the first few presentations of a (non-neutral) rewarding stimulus (McSweeney and Murphy, 2009). It is also consistent with the fact that habituation (the counterpart of sensitization) can engender generalization to other stimuli (Rankin et al., 2009).

To quantify this idea we assume that the novelty salience is maximum when the outcome of the interaction is least predictable or most uncertain; that is, when y^*_f is at its intermediate value of 0.5. For, at this point, there is no bias in the prediction of the phasic stimulus occurring or being absent. We then assign a novelty salience of zero to the "firm predictions" corresponding to $y^*_f = \pm 1$, and assume piecewise linearity elsewhere. This mapping is shown in **Figure 5B**. Formally, if $S^{i,j}_{\text{nov}}$ is the novelty salience for interaction j on day i, at time $t_{i,j}$,

$$S^{i,j}_{\text{nov}} = 0.5 - |y^*_f(t_{i,j}) - 0.5| \tag{3}$$

The ensuing novelty salience from the events in **Figure 5A** is shown in **Figure 5C**. The total salience is given by

$$S^{i,j}_{\text{tot}} = S^{i,j}_{\text{int}} + S^{i,j}_{\text{nov}} \tag{4}$$

Salience only occurs when the stimuli are perceived (at the points indicated by the open circles in **Figure 5C**). However, it is useful to indicate the causality of changes in novelty salience by formally transforming the latent prediction using Equation (3) so into novelty salience after each interaction is the salience that *would* be seen if the stimulus comes into view.

The salience for the exploratory action is assumed to be driven by an internal motivational process (like fear or foraging for food) which is notionally a component of "internal state monitoring." It manifests itself in a salience for exploration drawn from a uniform distribution with constant mean of 0.4, and standard deviation of 0.23.

2.5.3. Basal ganglia and loops through cortex

The main neural circuit in the biomimetic core is based on our previous work with models of basal ganglia (Gurney et al., 2001a,b) and loops through cortex (Humphries and Gurney, 2002). Key concepts were outlined in the Introduction; details of the particular form used here are shown in **Figure 6**. The model uses discrete processing channels for each action so that, within each nucleus, there is a localist representation of each channel as a population of neurons instantiated in a leaky integrator neural unit. Formally, each neural unit has an activation variable a governed by a first order ODE

$$\tau \frac{da}{dt} = -a(t) + I(t) \tag{5}$$

where τ is the characteristic membrane time constant (here, $\tau = 40\,\text{ms}$) and I is the summed, weighted input. The normalized firing rate y, of the neural unit is given by a piecewise linear squashing function

$$y(a) = L(a, \epsilon) = \begin{cases} 0 & a \leq \epsilon \\ a - \epsilon & \epsilon < a < 1 + \epsilon \\ 1 & a > 1 + \epsilon \end{cases} \tag{6}$$

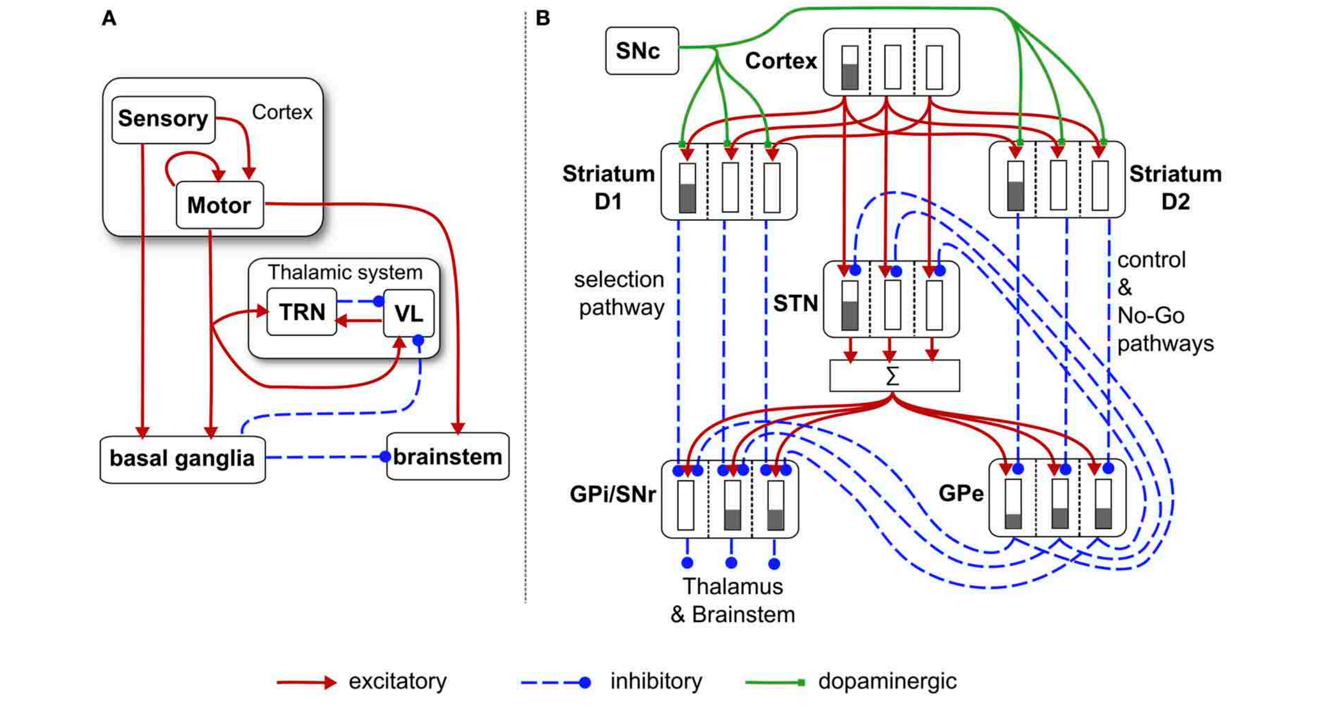


FIGURE 6 | Schematic diagram of the basal ganglia neural network component of the biomimetic core. (A) Cortex, basal ganglia, brainstem, and thalamic complex. The latter is comprised of the thalamic reticular nucleus (TRN) and ventrolateral thalamus (VL). Note that action channels are present but not explicitly shown here. **(B)** The basal ganglia circuit consisting of: striatal projection neurons expressing D1 or D2 dopamine receptors; subthalamic nucleus (STN); output nuclei—globus pallidus internal segment (GPi) and substantia nigra pars reticulata (SNr); globus pallidus external segment (GPe), and substantia nigra pars compacta (SNc). The three action channels are shown in this panel, and a typical set of activities indicated in cartoon form by the gray bars (the channel on the left is highly salient causing suppression of basal ganglia output on that channel). The summation box below STN is not anatomically present—it is graphic device to indicate that each target of STN sums its inputs across channels from STN.

where, ϵ is a threshold below which $y = 0$, immediately above which y depends linearly on a with unit slope, and there is saturation at $y = 1$.

There are three channels in the current model—one for each of the action-subsystems. The sensory cortex (**Figure 6A**) receives input from the salience generators, and initiates activity in motor cortex. This activity can potentially undergo amplification in the recurrent loop with the thalamic system, but this is under basal ganglia control. The motor cortex and the basal ganglia output nuclei project directly to the reticular formation and pendunculopontine nucleus brainstem areas a (Takakusaki et al., 2004; Jenkinson et al., 2009). If the increased drive from motor cortical channel i to its corresponding brainstem population, as well as the direct release of inhibition from that population, cause its activity y_i^{bs} to exceed the threshold ϕ, then the channel is selected for behavioral expression (see **Figure 4**).

Within the basal ganglia, there are two interdigitated populations of projection neurons in the main input nucleus—the striatum. These so-called *MSNs* are differentiated according to their preferential expression of dopamine receptor type—D1 or D2. We refer henceforth to these populations as *D1-striatum* and *D2-striatum*. The subthalamic nucleus (STN) is the only source of excitation in basal ganglia. The output nuclei of the basal ganglia are the globus pallidus internal segment (GPi) and substantia nigra pars reticulata (SNr). The circuit comprising D1-striatum, STN and GPi/SNr form a feedforward, off-center, on surround network implementing an inter-channel competition; hence it is dubbed the *selection pathway* . The "winning" channel in basal ganglia competitive processes is that which has the lowest output in GPi/SNr (inhibition to targets is released). This channel will have received the largest inhibitory input fron D1-striatum, which, in turn, will have been subject to the highest salience input. The circuit comprising the globus pallidus external segment (GPe), STN and D2-striatum exercise a *control* function acting on the selection pathway to ensure a good match between overall excitation from STN, and striatal inhibition of the output nuclei (Gurney et al., 2001a,b). The circuit through D2-striatum, GPe and SNr also implements a NO–GO function, actively preventing action selection (Frank et al., 2004). Parametric details of the application of Eqs. (5) and (6) to the circuits in **Figure 6** are given in the "Appendix."

The cortico-striatal synapses receive modulatory input from dopamine axons which branch profusely throughout striatum (Beckstead et al., 1979; Gauthier et al., 1999; Matsuda et al., 2009). Dopamine terminals also seem to innervate striatum in a dense, non-focal way within the neuropil of striatum (Moss and Bolam, 2008), and dopamine also acts extra-synaptically via volume transmission (Cragg and Rice, 2004). These data would indicate a diffuse innervation of striatum by dopamine neurons that cuts across channel boundaries.

Tonic (background) dopamine levels are thought to influence cortico-striatal transmission at D1 and D2 MSNs in opposite ways with D1/D2 receptors facilitating/attenuating cortico-striatal transmission (West and Grace, 2002). This is incorporated into our model by including a constant tonic dopamine level λ, which increases cortico-striatal D1-MSN weights by a multiplicative factor $1 + \lambda$, and decreases corresponding D2-MSN weights by $1 - \lambda$. More significantly for the current study are the dynamics of phasic (transient) dopamine, which are critical for cortico-striatal plasticity (Reynolds and Wickens, 2002), and to which we now turn.

2.5.4. Phasic dopamine and sensory prediction error

The starting point for this component of the model is our hypothesis that phasic dopamine signals a sensory prediction error (Redgrave and Gurney, 2006; Redgrave et al., 2008). Using the notation developed in section 2.5.1, the sensory prediction error $e(t_i)$ is given by $e(t_i) = y_f(t_i) - y^*_f(t_i)$. The error resulting from the sequence of events in **Figure 5A** is shown in **Figure 5D**. In the rest of this section, we drop the temporal argument and its indexing as it assumes a single block interaction.

However, we also wish to relate this form for e to its biological generation and realization in phasic dopamine. In particular, we invoke the evidence that phasic dopamine is released in response to neutral phasic stimuli and that this occurs via the recently discovered tecto-nigral pathway (Coizet et al., 2003; Comoli et al., 2003; Dommett et al., 2005). This is a direct (mono-synaptic) pathway between the superior colliculus (SC) (optic tectum in non-mammals) and midbrain dopamine neurons in substantia nigra pars compacta (SNc). The SC plays a key role in gaze shifting and orienting responses (Wurtz and Goldberg, 1972; Wurtz and Albano, 1980) and is believed to act as a detector of novel, phasic stimuli (Dean et al., 1989). In our terminology it detects y_f. Phasic responses in SC then excite SNc neurons and therefore potentially cause phasic bursts of activity therein. However, as the stimulus becomes predictable, this response in SNc disappears and, significantly, if the predicted reward is omitted, there is a phasic "dip" in the dopamine response below tonic level (Schultz et al., 1997; Schultz, 2006). Taking these pieces of evidence together, suggest that the null response in SNc under stimulus prediction is a result of the excitatory influence of SC, and a similarly timed inhibitory signal from another nucleus which we will call the "canceling signal." The lateral habenula may be a candidate for such signals in dopamine neurons (Matsumoto and Hikosaka, 2007).

To model the SC, we assume that its response is not only contingent on y_f but also on any phasic prediction y^*_f. This extends the temporally adaptive response of colliculus at long time scales under habituation (Drager and Hubel, 1975) to include phasic prediction at shorter time scales. Thus, if y^{SC}_f is the response of SC to phasic feature f, we put $y^{\mathrm{SC}}_f = [y_f - y^*_f]^+$, where $[x]^+ = \max(0, x)$. Then, the canceling signal y^{C}_f takes the form $y^{\mathrm{C}}_f = [y^*_f - y_f]^+$ and the sensory prediction error is given by

$$e = y^{\mathrm{SC}}_f - y^{\mathrm{C}}_f = [y_f - y^*_f]^+ - [y^*_f - y_f]^+ = y_f - y^*_f \qquad (7)$$

Since the collicular and canceling signals are not derived from prior inputs, we modeled their dynamics phenomenologically so that each of y^{SC}_f, y^{C}_f are triangular pulses of width 0.2 s.

In translating this into dopamine activity in our model there are several issues to contend with. First, we don't know the relation between positive and negative excursions of e and phasic dopamine bursts and dips—it could be that an error of +1 is signalled by a dopamine level many times that of tonic, but that an error of −1 is signalled by sufficiently prolonged dip with minimum of zero. We are therefore free to include parameters a^+, a^- in forming the effective input to a dopamine neuron, I^{SNc}, which encodes prediction error

$$I^{\mathrm{SNc}} = a^+ y^{\mathrm{SC}}_f - a^- y^{\mathrm{C}}_f \qquad (8)$$

These parameters were chosen for best model fit to the data of Gancarz et al. (2011) giving $a^+ = 2$, $a^- = 1$. Further, we don't know *a priori* the relationship between the magnitude of e (which lies in the interval $[-1, 1]$) and the corresponding level of simulated dopamine, d, expressed in our plasticity rules. We therefore use I^{SNc}, to determine an *effective* SNc output, y^{SNc}, which we can then equate with d. Thus, we form the SNc activation a^{SNc} in a first order ODE like that in Equation (5) and use this, in turn, to generate $y^{\mathrm{SNc}} \equiv d$ via the function

$$y^{\mathrm{SNc}} = \begin{cases} 0, & a^{\mathrm{SNc}} \leq -0.2 \\ a + 0.2, & a > 0.2 \end{cases} \qquad (9)$$

The lack of normalization is a requirement for interpreting y^{SNc} as the simulated dopamine level d, used in the next section.

2.5.5. Cortico-striatal plasticity: the learning rule

The learning rule is based on our recent work on cortico-striatal plasticity at the level of spikes (Gurney et al., 2009) which is, in turn, grounded in a comprehensive *in vitro* study (Shen et al., 2008). The latter was able to distinguish recordings between D1 and D2-type MSNs, and yielded responses at different levels of dopamine. The resulting learning rules are complex and reflect the unavoidable complexity in the data. However, the rules do provide an account of plasticity consistent with action discovery and so we sought to incorporate them in the current model. Fortunately Pfister and Gerstner (2006) have shown how to relate spike timing dependent plasticity (STDP) to the Bienenstock, Cooper, and Munro (BCM) rule for rate-coded neurons (Bienenstock et al., 1982; Cooper et al., 2004) which therefore allows us to proceed with this programme.

The work of Pfister and Gerstner (2006) dealt with STDP for spike pairs and triplets. The transition to firing rates is done by calculating the expected weight change $\langle dw/dt \rangle$. Let the pre- and post-synaptic firing rates be x and y, respectively. If $\Delta t = t_{\mathrm{post}} - t_{\mathrm{pre}}$ is the time interval between post- and pre-synaptic spike pairs then let τ^+, τ^- be time constants associated with processes for $\Delta t > 0$, $\Delta t < 0$, respectively. The rate coded rule takes the form

$$\left\langle \frac{dw}{dt} \right\rangle = A_3 \tau^+ \tau^y y(y - \theta_{\mathrm{BCM}})x \qquad (10)$$

$$\theta_{\mathrm{BCM}} = \langle y^2 \rangle C_{\mathrm{BCM}}$$

$$C_{\mathrm{BCM}} = \frac{-(A^- \tau^- + A^+ \tau^+)}{A_3 \tau^+ \tau^y}$$

Here, τ^y is a time constant associated with spike triplets, and A_3 is a factor for the plasticity from triplet timing. This has no direct counterpart in our spiking level model but we assume a positive value.

More importantly, the terms A^+, A^- are derived from the contributions to plasticity from positive and negative spike pair timing [here they are signed quantities; in (Pfister and Gerstner, 2006) they are absolute magnitudes]. Further, we endow them with dopamine dependence and specificity under the D1/D2 MSN dichotomy. Thus, following (Gurney et al., 2009) we use the data of Shen et al. (2008) to determine these terms for D1-MSNs at high levels of dopamine $A_+^{\mathrm{D1(hi)}}, A_-^{\mathrm{D1(hi)}}$, at low levels of dopamine $A_+^{\mathrm{D1(lo)}}, A_-^{\mathrm{D1(lo)}}$, and for corresponding quantities for D2-MSNs; we refer to these eight quantities as *plasticity coefficients*. For example, with positive spike-pair timing in D1-MSNs at high levels of dopamine, the data imply strong LTP, and for negative spike-pair timing, weak LTD (Shen et al., 2008). This led to the assignment shown in **Figure 7A** (see "D1(hi)" bar grouping). Other coefficient assignments are shown in **Figure 7A** and compared with the "classic" finding for STDP in hippocampus and cortex, in which with LTP/LTD is associated with positive/negative Δt (Song et al., 2000). Notice that several of the coefficient pairs give LTP/LTD assignments which are "non-classical"; for example, D2-MSNs at low dopamine have uniform LTP for both timings.

At levels of dopamine, d, intermediate between the "low" and "high" extremes, we define $A_\pm^{\mathrm{D1/D2}}(d)$ as a function of dopamine by "blending" the relevant plasticity coefficients together using a monotonic, saturating function $\alpha(d)$ (see **Figure 7B**)

$$\alpha(d) = \frac{4d}{1 + 4d} \tag{11}$$

For example, for D1-MSNs, A_+^{D1} is given by

$$A_+^{\mathrm{D1}}(d) = \alpha(d) A_+^{\mathrm{D1(hi)}} + (1 - \alpha(d)) A_+^{\mathrm{D1(lo)}} \tag{12}$$

with similar relations for $A_-^{\mathrm{D1}}(d), A_+^{\mathrm{D2}}(d), A_-^{\mathrm{D2}}(d)$. This gives, in turn, functional forms $C_{\mathrm{BCM}}(d)$ derived from scalar factors C_{BCM} in Equation (10) (see **Figures 7C,D**).

Weights from both motor cortex and sensory cortex to striatum ("motor weights" and "sensory weights," respectively) are subject to the learning rule described above. The motor weights are supposed to endow the agent with the ability to perform the three actions expressed in the action-subsystems. They are initialized in such a way as to allow this to occur in the presence of the

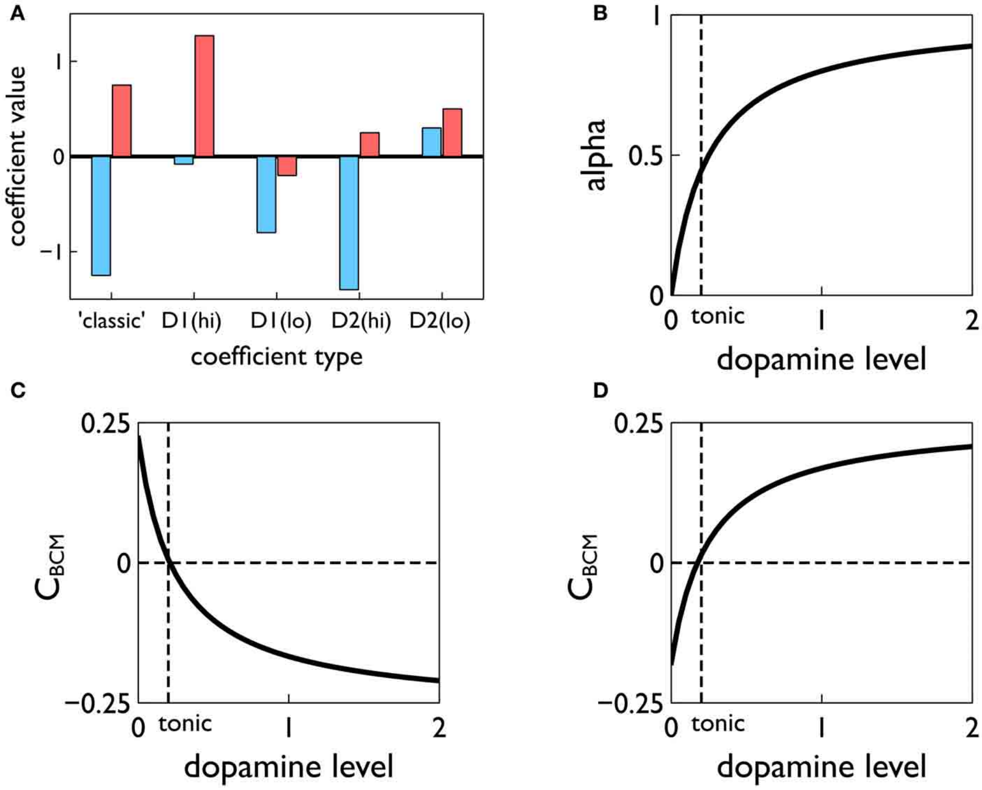


FIGURE 7 | Construction of the learning rule. (A) The plasticity coefficients consistent with the data of Shen et al. (2008). **(B)** The dopamine mixing function $\alpha(d)$ defined in Equation (11). **(C,D)** The dopamine-dependent versions of the factors C_{BCM} in Equation (10) for D1 and D2-MSNs, respectively.

exploration action, during an initial "weight calibration" learning session. In contrast, the sensory weights are initialized to zero, and any positive increments therein are thought of as supplying new "biases" in the selection of the three given actions, derived from contextual information. However, the uniform treatment of both motor and sensory weights means their trajectories will mirror each other in form (see for example, **Figure 10**).

3. RESULTS

3.1. CORTICO-STRIATAL PLASTICITY ALONE IS NOT SUFFICIENT TO ACCOUNT FOR VARIABLE INTERVAL TRAINING DATA

Figure 8 shows the behavioral outcome for an agent with no novelty salience (or its associated internal prediction model), undergoing VI training in the block-bumping task. Results are averaged over 10 repetitions with different initial random number seed, and the two panels show outcomes with and without phasic dopamine enabled. This dichotomy will be a recurring theme as we wish to explore the relative contributions of novelty salience and phasic dopamine during learning. We will refer to models with and without phasic dopamine enabled as "pDA," and "no-pDA" models, respectively. In the presence of phasic dopamine, there is a statistically significant difference between the number of interactions with the control (white) and target (red) blocks. However, this difference is nowhere near as substantial as that shown in the data of Gancarz et al. (2011). We conclude that other mechanisms must be at work and therefore invoked the notion of *novelty salience* as described in the section 2.

3.2. NOVELTY SALIENCE CAN ACCOUNT FOR BEHAVIORAL TRENDS IN VARIABLE INTERVAL LEARNING

Figure 9 shows the behavior for an agent in the presence of novelty salience and an internal prediction model (see section 2) undergoing VI learning (results are averaged over 10 repetitions). Both pDA and no-pDA models show qualitatively similar behavior to that from the *in vivo* experiment in **Figure 3**. That is, they show a substantial increase in active responses during the response contingent phase which declines toward the end of the experiment. In addition, the peak response does not occur on the first day of training in the response contingent phase. However, the no-pDA model shows markedly more active responses during the response contingent phase than its pDA counterpart. To quantify this, let r_{peak}, be the ratio (rounded to nearest integer) of the peak number of active responses during response contingency to the mean inactive response over this time. Note that, while absolute numbers of responses in the model are not directly comparable with those *in vivo*, we might expect ratios of responses under different regimes to be more so. For the *in vivo* experiment $r_{\text{peak}} = 3$, while for pDA and no-pDA models $r_{\text{peak}} = 7, 12$, respectively. This feature is therefore more realistically captured with the inclusion of phasic dopamine.

The role of phasic dopamine in explaining these differences in active responses is made apparent by reference to **Figure 10**, which shows the dynamics of the cortico-striatal weights in the active response (red-block-interaction) channel as learning progresses. For the no-pDA model there is (unsurprisingly) little change in the weights in the response contingent phase (for both D1 and D2-MSNs, and motor and sensory cortical inputs). However, for the pDA model, there is a decrease in D1-MSN weights and an increase in D2-MSN weights. This is consistent with a decrease in the ability of the selection pathway in basal ganglia to facilitate an active response, and an increase in the potential of the NO–GO pathway to suppress it (Frank et al., 2004) (see section 2.5.3). Phasic dopamine, and the biologically plausible learning rule, are therefore directly

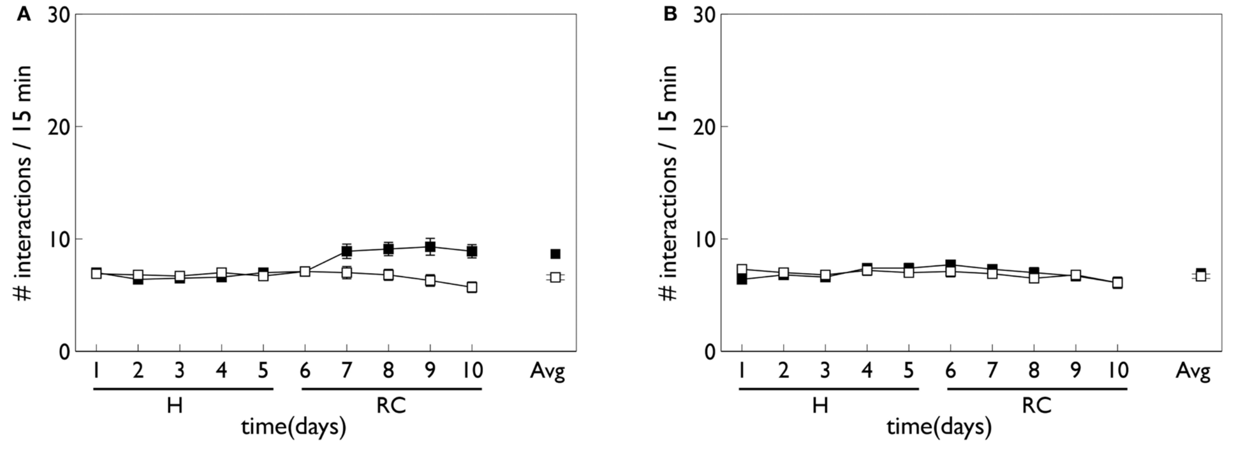


FIGURE 8 | Behavior of an agent with no novelty salience or internal prediction model, performing the block-bumping experiment. (A,B) For models with, and without, phasic dopamine, respectively (pDA, no-pDA), and each plot is an average over 10 runs. These plots are based on those of the *in vivo* data in **Figure 3**. Thus, each panel shows the number of interactions with the block stimuli in each 15 min session comprising a "virtual day" of learning, plotted against such days. Error bars are 1 standard error of the mean. Open symbols are for the white (control) block while solid symbols are for the red block, which elicits a phasic outcome in the response contingent phase (labeled "RC"). The habituation phase (when there is no environmental phasic outcome) is designated "H." The average of the interactions for each block over the entire response contingent phase is shown in the pair of data points on the extreme right of each panel.

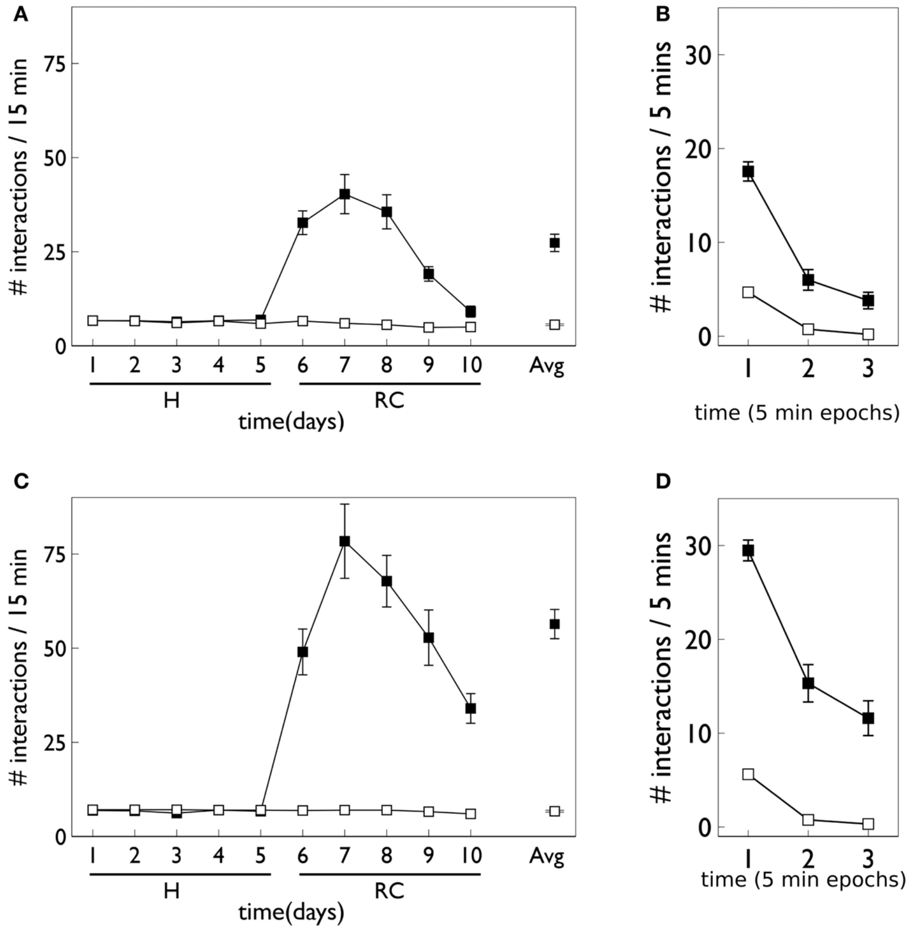


FIGURE 9 | Behavior of an agent with novelty salience and feature prediction performing the block-bumping experiment with variable interval training. (A,C) Have a similar interpretation to counterparts in **Figure 8** and are for pDA and no-pDA models, respectively. **(B,D)** Show the behavior within a virtual "day" (considered as three, 5 min epochs), averaged over the response contingent phase; **(B,D)** are for pDA and no-pDA, respectively.

responsible for the relative, overall difference in active responses in the pDA model, compared to its no-pDA counterpart (**Figure 9**).

We can see, mechanistically, the reason for the weight changes by examining the dynamics of the reinforcement signal (light flash), the prediction model, and resulting dopamine signal. These signals are shown in **Figure 11**. It is apparent that there are many more dopamine "dips" (negative prediction errors) than "bursts" (positive prediction errors) and so the factors C_{BCM} in the learning rule (Equation 10) are dominated by their low dopamine values. For D1/D2-MSNs this is positive/negative, respectively (**Figure 7**), which is also reflected in θ_{BCM}. In addition, the high novelty salience in cortex causes high activity $\langle y^2 \rangle$ in the MSNs, thereby amplifying θ_{BCM} and any consequent effects on learning. These signs and magnitudes of θ_{BCM} lead to LTD/LTP for D1/D2-MSNs being likely (as θ_{BCM} appears in the factor $(y - \theta_{\text{BCM}})$ in the learning rule). This pattern of learning has computational and ethological consequences taken up in the section 4.

3.3. PHASIC DOPAMINE PROMOTES PLASTICITY IN FIXED-RATIO TRAINING CONSISTENT WITH ACTION LEARNING IN STRIATUM

Figure 12 shows the behavioral responses of the robot in the fixed-ratio (FR) experiments. The results are qualitatively similar to those for VI training but there are fewer active responses and, unlike the VI behavior, the peak response occurs on the first day of the response contingent phase. This prediction was borne out by the study of Lloyd et al. (2012)—see **Figure 3C**. Within a session, the number of active responses declines more steeply than the corresponding VI data. This is similar to the *in vivo* data (**Figure 3C**) although the latter does not show such a tight clustering in the first epoch, with some residual responding at the end of the session.

The pDA and no-pDA models have similar behavior but the former shows somewhat more active responses (especially on the first response contingent day). This is quantified in the (rounded) ratios r_{peak} which are 6 and 4, respectively. These are both smaller than the values for the VI experiment, and have a different rank order (that for pDA is larger for FR, but is smaller for VI).

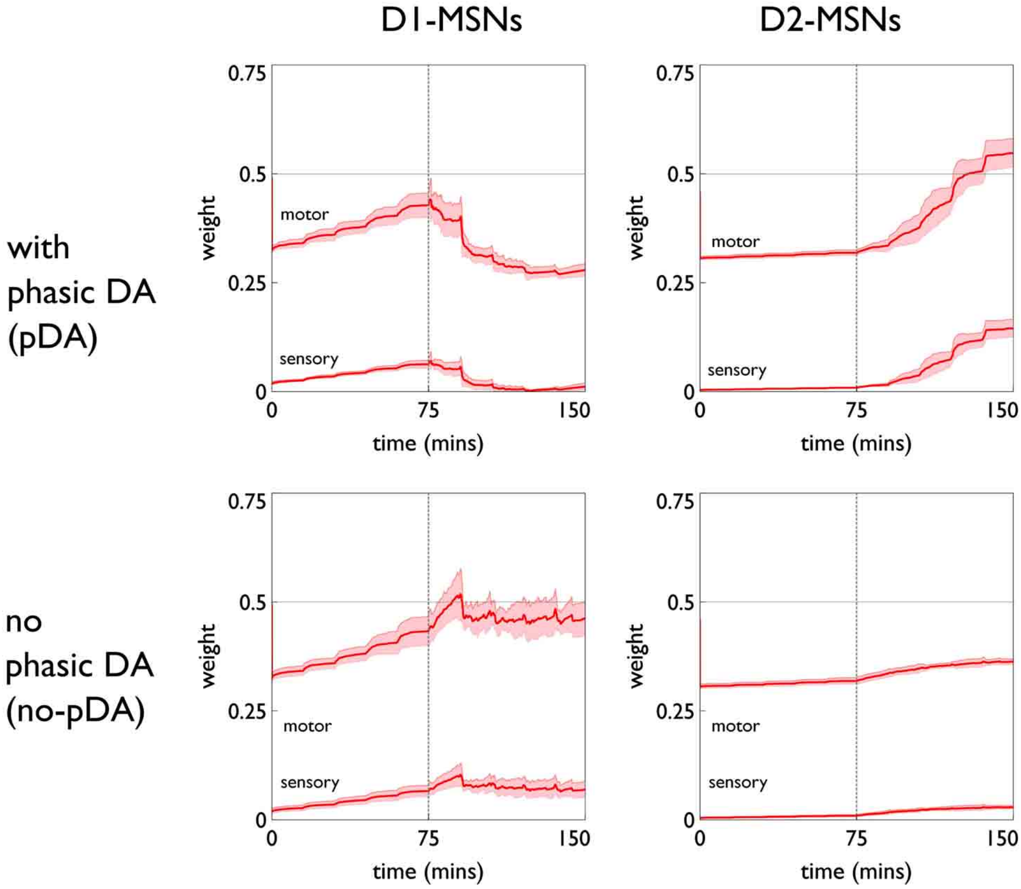


FIGURE 10 | Weight trajectories $w^*(t)$ for the active response channel, in models with novelty salience and prediction, undergoing variable interval training. Rows are for pDA and no-pDA models, columns for D1- and D2-type MSNs. Weights from motor cortex and sensory cortex are labeled "motor" and "sensory." The trajectories are expressed as continuous functions of time to show both within-day, and between-day dynamics, and the onset of the response contingent phase (at the start of day 6) is indicated at 75 min. These plots capture the statistics of the weights over a group of 10 models; the dark red line is the mean, and the red-shaded region encompasses ±1 std dev.

The similarity in behavioral response over the pDA, no-pDA variants is in stark contrast to the difference in weight trajectory (**Figure 13**).

The pDA model shows a very large transient change in the D1-MSN weights (both motor and sensory) with a substantial final change compared to initial baseline. This plasticity is clearly responsible for the extra activity in the response contingent phase compared to that for no-pDA models. None of the other weight trajectories show significant variation.

The clustering of active response in day 6 and the transient weight change associated with this are explained by reference to the prediction, novelty salience and dopamine signals shown in **Figure 14**. Thus, there is a large increase in novelty salience in the first part of the response contingent phase (panel **A**) but this is short lived as the prediction becomes reliable. This is made possible, of course, by the reliable delivery of the reinforcement. The phasic dopamine reflects this, and is almost always signalling positive reinforcement errors (the very few occasions for which this is not the case, are caused by failure of the robot to bump properly against the block). High levels of (phasic) dopamine occurring during these events is associated with negative values of C_{BCM} for D1-MSNs in the learning rule [Equation (10), and **Figure 7**]. This implies $\theta_{\mathrm{BCM}} < 0$ too, so that there is a likelihood of LTP as observed.

4. DISCUSSION

4.1. MAIN RESULTS AND THEIR INTERPRETATION

We have used the embodiment of a biologically plausible model of intrinsically motivated operant learning (action discovery) to explore the possible roles of cortical salience, cortico-striatal plasticity in basal ganglia, and phasic dopamine therein. The embodiment allowed us to use behavioral data (Gancarz et al., 2011) to constrain the model, and our core model component was sufficiently biologically plausible to take advantage of a new framework for dopamine-dependent cortico-striatal plasticity constrained by a comprehensive suite of physiological data (Shen et al., 2008; Gurney et al., 2009).

In seeking an understanding of action discovery, we are primarily interested in the ethological situation in which the required action reliably produces the desired outcome; in the

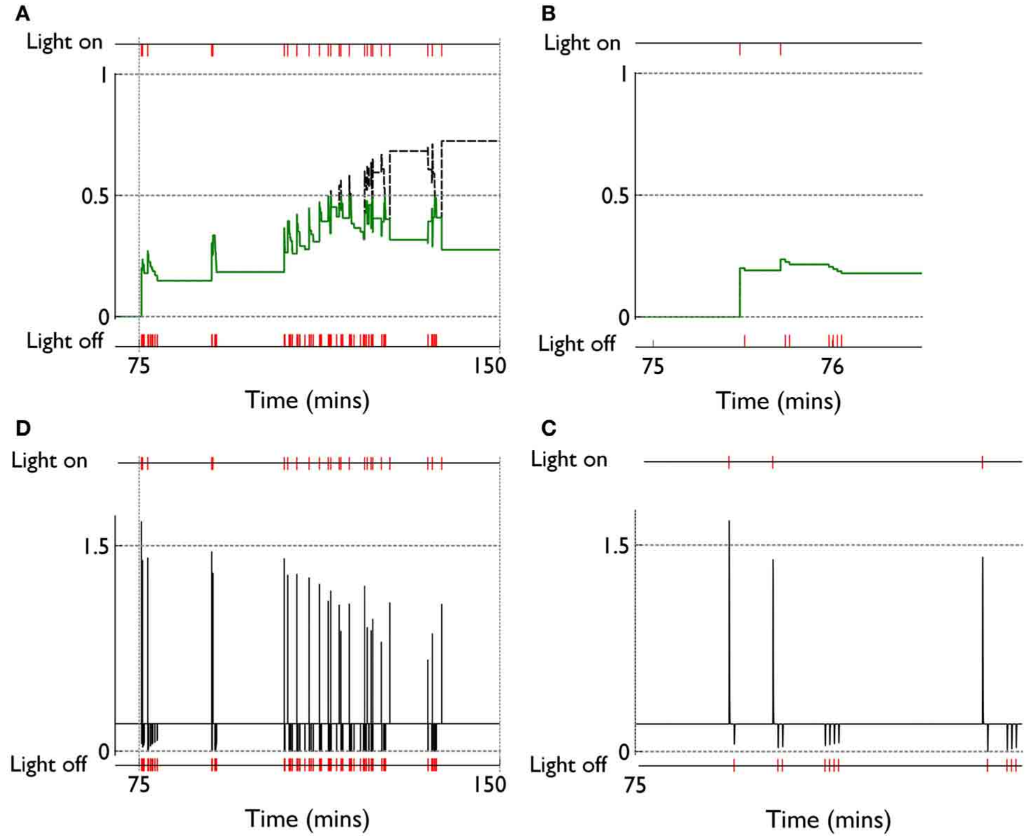


FIGURE 11 | Signals governing learning in pDA models with novelty salience and prediction, undergoing variable interval training. (A) Shows the novelty salience (solid green line) and prediction signal (dashed black line) during the response contingent (RC) phase in a similar way to that used in **Figure 5**, but here the symbols have been omitted. **(B)** Is similar to **(A)**, but for a smaller temporal window immediately after the onset of the RC phase. **(D)** Shows the phasic dopamine signal corresponding to the events in **(A)**. **(C)** Is similar to **(D)** and relates to events in panel **(B)**.

current context this is what has been referred to as the FR1 schedule. However, the data we have access to (Gancarz et al., 2011) concern a VI schedule. We have shown that cortico-striatal plasticity alone is insufficient to account for the increased active response in this data. In order to successfully model the behavioral data, we were therefore forced to consider the other possible contribution to more prolific action selection—an increase in the salience of the action request. Thus, we proposed that the sensory contribution to the action request for block interaction is enhanced by inheriting the novelty of any surprising phasic outcome associated with the target block. To incorporate this "novelty salience" we deployed a simple phenomenological model of prediction of the phasic outcome and its influence on the salience. We also used the prediction model to describe the dynamics of the sensory prediction error signal manifest in phasic dopamine.

With these components in place, the main trends in the behavioral data of the *in vivo* experiment could be replicated. Moreover, there was a somewhat counterintuitive result that there were fewer active responses with phasic dopamine than without. Further, the relative number of responses (active/inactive) in the data was better approximated by the inclusion of phasic dopamine. This difference could be explained by noting the preponderance of phasic dopamine dips in the VI schedule, the consequent weight dynamics, and their interpretation in the context of selection (GO) and NO–GO pathways in basal ganglia.

The attenuation of activity by dopamine mediated plasticity in the VI schedule is ethologically rational. The outcome in VI training is highly unpredictable and it is therefore fruitless for an intrinsically motivated agent to waste resources in attempting to build a model of agency. This notion has been formalized by Schmidhuber (2009) who argues that agents seek to compress information about their world (equivalent to our internal model building) and failure to see progress in this regard will cause them to disengage with the situation. Attempts to persist in doing so could lead to irrelevant and "superstitious" behavior (Pear, 1985). The dopamine mediate plasticity appears to prevent just this scenario. In addition, the failure of the D1-MSNs to show strong LTP would mitigate against the possibility that these neurons could learn to encode a match between their synapses and cortical representations of the new action request.

We carried over the notion of novelty salience to the FR1 schedule; there is no reason to suppose that the mechanisms for prediction and novelty salience generation suddenly become

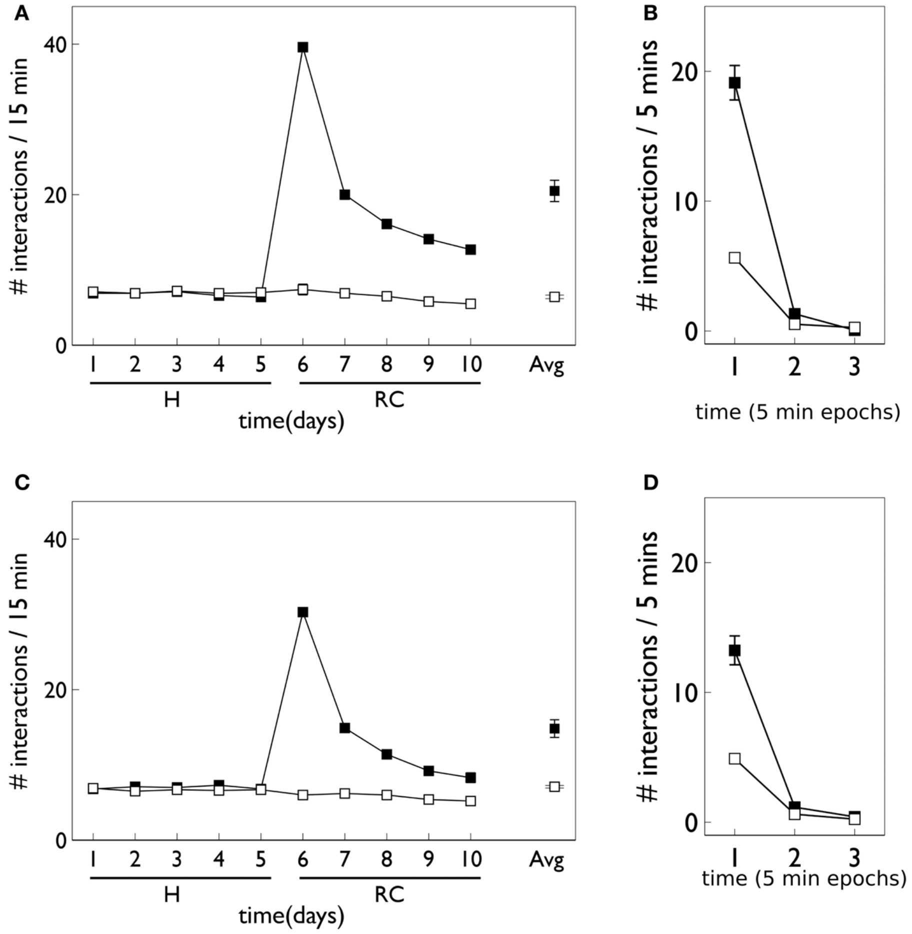


FIGURE 12 | Behavior of an agent with novelty salience and feature prediction performing the block-bumping experiment with fixed-ratio training. All panels have the same significance as their counterparts for variable interval training in **Figure 9**. Thus, **(A,B)** are for pDA models, whereas **(C,D)** are for no-pDA models.

muted because the statistics of the stimulus are changed. The result was a strong increase in active responses on the first day of the response contingent phase. Overall activity during this time was, however, less than that for the VI schedule. Both these predicted features were shown in a recent *in vivo* study (Lloyd et al., 2012).

In contrast with the simulated VI result, phasic dopamine in FR learning enhanced the activity level with respect to the no-dopamine control. Further, much of the interaction occurred early in the session (also broadly in line with the *in vivo* data) and subsequent epochs within a session showed little interaction with the blocks. Activity is refreshed somewhat at the start of each day, which can be attributed to the dishabituation of block salience between days.

The rapid increase in, and subsequent decline of, responding with the novel situation is exactly what we would require with our repetition bias hypothesis. The results suggest that, while the behavioral repetition is due to a combination of novelty salience *and* plasticity (there is more responding with phasic dopamine) the bulk of this effect is caused by the novelty salience. We therefore predict that lesioning systems that may be responsible for developing novelty salience should severely compromise action-outcome learning (see discussion of novelty below).

We also predict a residual, persistent elevation of the number of active responses at the end of the response contingent phase, compared to that at the end of the habituation phase. There is some indication of this in the study of Lloyd et al. (2012) but further experiments would help confirm or falsify this outcome. In the event that it is true, this may be interpreted as the "bumping-into-the-red-block" action having acquired the status of a preferred action or *affordance* (Gibson, 1986; McGrenere and Ho, 2000). Thus, we suppose, along with Cisek (2007), that affordances become what we have dubbed "action requests," subject to competitive selection by basal ganglia.

The weights in FR learning show strong LTP in D1-MSNs consistent with the encoding of the action in basal ganglia via synaptic-afferent matching. There is a marked peak during the early sessions of the response contingent phase (promoting repetition bias) before a decline to an equilibrium level which is elevated with respect to the initial value. It is only in the FR schedule with phasic dopamine that we see such a substantial weight

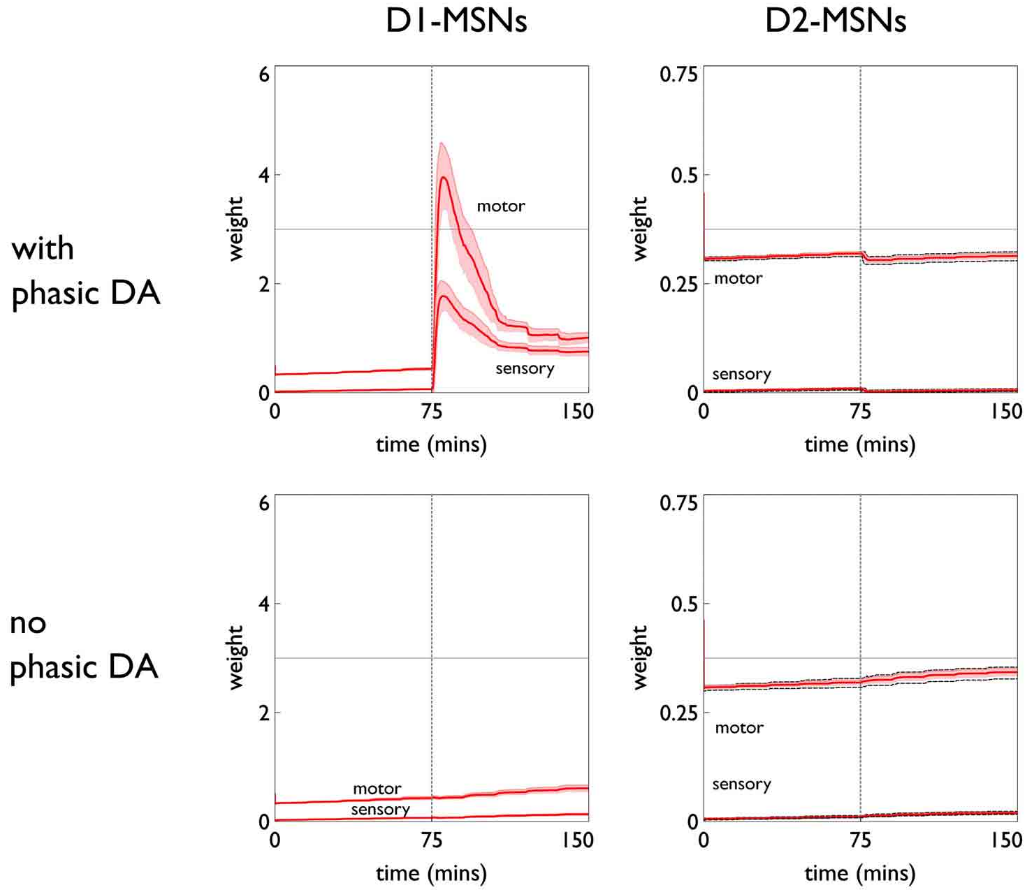


FIGURE 13 | Weight trajectories $w^*(t)$ for the active response channel, in models with novelty salience and prediction, undergoing fixed-ratio (FR-1) training. All panels have the same significance as their counterparts for variable interval training in **Figure 10**. Note the different scale for D1 and D2-MSNs.

increase and so we deem these conditions to be necessary for action learning.

4.2. RELATION TO OTHER WORK

There have been many attempts in disembodied models to describe the role of phasic dopamine in animal learning. Most of these use some kind of RL technique and, typically something like the temporal difference (TD) algorithm (Sutton and Barto, 1998) or variants therein—for a recent review see Samson et al. (2010). These machine learning algorithms require an explicit representation of *value* as the expected sum of rewards over some predefined trial or epoch. However, no such representation prevails in our model. Further, in the TD-like schemes, there is usually a fine-grained representation of time supporting a correspondingly rich state-based description of the environment; we have no recourse to such a description. Like TD, our model uses a prediction error. However, this error has a quite different form from that in TD, is used in a quite different way to update the weights, and the update rule for the prediction is different.

Another hallmark of the general RL models is their emphasis on obtaining optimal behavior driven by explicit biological reward. In contrast we have emphasized the concept of novelty and sensory prediction as a primary source of reinforcement in the learning rule. Novelty has been used in TD-learning models of learning under phasic dopamine, appearing in the guise of "novelty bonuses." Kakade and Dayan (2002) show how such a model may be used to enhance the explanatory power of the basic TD-learning approach, but the very term "bonus," is used advisedly here to imply that novelty is an "add on," and that optimality of reward acquisition is the primary feature of the algorithms. We revisit the issue of whether dopamine encodes reward or sensory prediction errors in section 4.3 where we give a possible resolution of this apparent dichotomy. The model of Kakade and Dayan (2002) is also unable to supply an explanation (even at an algorithmic level) of the intrinsically motivated learning seen in the study of Gancarz et al. (2011) because it does not address the issues of novelty salience that we have found necessary in our model.

In more biologically plausible (but still disembodied) approaches, many models of RL in basal ganglia use the actor-critic framework (Barto, 1995; Suri and Schultz, 1998, 1999). However, the applicability of this framework to the study of learning in basal ganglia has been questioned on the basis of its biological plausibility (Joel et al., 2002). In contrast, our approach

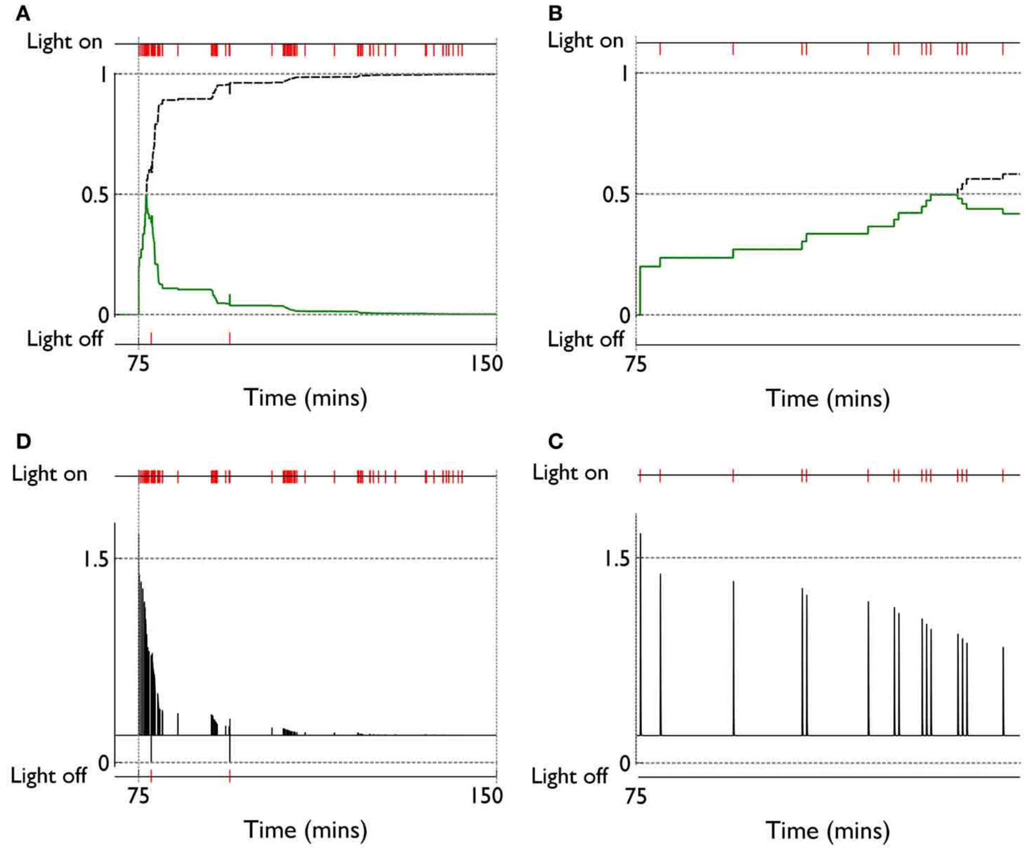


FIGURE 14 | Internal variables and phasic dopamine signals for a model with novelty salience and prediction undergoing fixed interval training. (A,B) Show novelty salience and the prediction signal, and are counterparts to **Figures 11B,C**. **(C,D)** Show phasic dopamine and are counterparts to **Figures 11A,D**.

does not rely on the actor-critic scheme. Further, many of the RL models that attempt to explain dopamine dynamics and learning in basal ganglia use the TD algorithm (Suri, 2002) which was noted above to be quite different from our approach. In a recent review, Frank (2011) notes several biologically plausible models of dopamine modulated learning in basal ganglia (Brown et al., 2004; Frank, 2005, 2006). However, these models do not address the problems surrounding intrinsically motivated learning and will therefore not seek to understand the automatically shaped, phasic period of repetition bias under the control of surprise or novelty, signalled by phasic dopamine. One recent model (Hazy et al., 2010) does note the possible utility of encoding "novelty value" in the phasic dopamine signal as well as reward, but this model is at a somewhat abstract level without explicit reference to basal ganglia components.

There are very few *robotic* models of operant learning that seek to explain the role of phasic dopamine. The model by Baldassarre et al. (2013) explores several of the issues in our general framework but at higher level of abstraction. It has a less physiologically constrained learning rule, several *ad hoc* mechanisms in place to test general computational hypotheses (such as repetition bias), the basal ganglia component is less well detailed, no mention is made of novelty salience, and there is no behavioral data against which it is constrained. Nevertheless, this model does integrate many of the features in the general scheme outlined in the Introduction (**Figure 1A**) and show how they may be deployed in concert with each other to achieve intrinsically motivated learning of actions.

The model of Sporns and Alexander (2002) (see also Alexander and Sporns, 2002) uses properties ascribed to the animal dopaminergic system in its learning, but the model architecture is rather abstract and has no reference to basal ganglia and cortico-striatal connectivity. In contrast to our own, this model also emphasizes the precise temporal representation of reward prediction reminiscent of the TD learning algorithm. An explicit use of TD learning was invoked by Pérez-Uribe (2001) but again, this model used a somewhat abstract actor-critic architecture. The model by Thompson et al. (2010) emphasizes limbic loops through the basal ganglia which deal with genuine reward-related behavior rather than intrinsically motivated behavior (hence no mention of novelty salience) and, again, it uses a different approach to learning. Khamassi et al. (2011) have recently described a robot model of learning with dopamine signalling prediction errors based on salient phasic events but their emphasis is on plasticity in cortico-cortical rather than cortico-striatal connections, with the aim of storing action values in anterior cingulate cortex (ACC).

4.3. NOVELTY, DOPAMINE, AND REWARD

One of the key ideas in our general framework is that intrinsically motivated action discovery is tightly bound up with the notion of novelty; new and unexpected objects or situations cause an agent to investigate them and discover operant contingencies. We have invoked two kinds of novelty in the present model: stimulus (object) novelty and surprise (phasic outcome). We have identified the detection of the latter with the SC and have noted the intimate link between the detection of surprise and release of phasic dopamine (Comoli et al., 2003; Dommett et al., 2005). However, the detection of novelty salience remains unresolved. Several brain areas have implicated in the detection of novelty and are candidates for this process including: lateral prefrontal cortex, anterior insular and anterior temporal cortex, parahippocampal cortices, and the hippocampal formation itself (Ranganath and Rainer, 2003). In regards to the latter, Kumaran and Maguire (2007) have proposed that the hippocampus acts as a comparator between prediction and perception, while Lisman and Grace (2005) have noted the link between hippocampus and midbrain dopamine systems in novelty detection. Using fMRI studies in humans, Bunzeck and Düzel (2006) have also demonstrated how stimulus novelty can drive the activation of dopamine neurons. However, when elicited by object novelty (rather than the surprise of an outcome) phasic dopamine may be more potent in facilitating learning in the structures which may encode the prediction models—namely areas like the hippocampal complex and prefrontal cortex (Lisman and Grace, 2005; Bunzeck and Düzel, 2006)—rather than motor and associative territories of striatum.

The preceding discussion has highlighted the ubiquity of phasic dopamine as an encoder of novelty and, consistent with this, is a recurrent theme in our work that dopamine is a *sensory* prediction error. However, there is a substantial literature arguing for its role in encoding reward (for recent review see Schultz, 2010). Thus, several studies (Fiorillo et al., 2003; Tobler et al., 2005; Morris et al., 2006; Roesch et al., 2007) have shown that, with well trained animals, size of reward or its probability of delivery reward associated with unpredictable phasic cues produced phasic dopamine responses which reflected the expected amount of reward. This is often cited as strong evidence that phasic dopamine is signalling *reward*-prediction error. However, one possible resolution of this apparent conflict is to suppose that dopamine encodes a sensory prediction error which may be *modulated* by reward value. This can occur because repeated delivery of reward is known to sensitise primary sensory areas including: visual cortex (Weil et al., 2010), somatosensory cortex (Pleger et al., 2008), and SC (Ikeda and Hikosaka, 2003). Thus, using an abbreviated form of our prior notation, let y_f and y_f^* be representations of a sensory feature and its prediction, respectively, and let S_R be a *reward sensitization* of y_f under extensive training (as typically deployed experimentally). We now hypothesise (Gurney et al., 2013) that phasic dopamine encodes

$$e = S_R(y_f - y_f^*) \qquad (13)$$

Notice that e can still be thought of as a *sensory* prediction error—there is no mention of a difference between *observed* or its prediction, as such. The stimulus feature has been "tagged" with additional value but the difference is fundamentally one between sensory features and their prediction. This idea can accomodate a recent theory by Bromberg-Martin et al. (2010) in which two classes of dopamine neuron are identified. In one class, dopamine neurons encode *motivational value*—the conventional idea that dopamine signals prediction errors of rewarding/aversive stimuli with positive/negative-going responses, respectively. A second class of neuron encode *motivational salience* with positive responses irrespective of the rewarding/aversive significance of the predicted stimulus. However, both classes of dopamine neuron signal "alerting" or unpredicted sensory cues. This classification is consistent with Equation (13) if we allow two cases in which S_R is either a signed quantity, encoding rewarding/aversive value, or simply the absolute magnitude of this quantity.

4.4. FUTURE DIRECTIONS

The action discovery used in our model is of the simplest kind; a given "atomic" movement (bump a block) has been paired with a context (the red block in this arena) to facilitate the prediction of the outcome (light flash above the block). However, in general we can imagine more complex combinations of action components may need to be assembled with the context. For example, the agent may not know how to perform a bumping sequence (move forward, then back and slow down), in which case it has to explore possible combinations of atomic movements at a lower level of granularity and chunk them together to make the new action. These lower level action components may also have to occur simultaneously rather than sequentially (e.g., bumping may require extending an effector as well as moving forward). Modeling the discovery of these more complex action assemblies is an important next step.

One of the requirements of a multi-component action model would be a true distributed representation of motoric commands. Even with a single atomic movement this is most likely encoded in a more plausible way a vector of command components. Further work would test the learning rule with these higher dimensional vector inputs. This was the approach taken in our spiking model of plasticity (Gurney et al., 2009) and, indeed, one possible progression of the model would be to embed the spiking model of MSNs into the larger basal ganglia model used here. This multiscale model would enable a closer examination of the finer details of the learning rule as originally conceived. Finally, we aim to test experimentally, predictions about the expected behavior of animals in an FR learning schedule with dopamine lesions.

ACKNOWLEDGMENTS

This work was supported by EU grant FP7-ICT-IP-231722 (IM-CLeVeR), and the Mexican Government programme of CONACYT (Consejo Nacional de Ciencia y Tecnologia—the National Council of Science and Technology).

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: http://www.frontiersin.org/Neurorobotics/10.3389/fnbot.2013.00004/abstract

REFERENCES

Alexander, W. H., and Sporns, O. (2002). An embodied model of learning, plasticity, and reward. *Adaptive Behav.* 10, 143–159.

Baldassarre, G., Mannella, F., Fiore, V. G., Redgrave, P., Gurney, K., and Mirolli, M. (2013). Intrinsically motivated action-outcome learning and goal-based action recall: a system-level bio-constrained computational model. *Neural Netw.* doi: 10.1016/j.neunet.2012.09.015. [Epub ahead of print].

Bar, M. (2007). The proactive brain: using analogies and associations to generate predictions. *Trends Cogn. Sci.* 11, 280–289.

Barto, A. G. (1995). "Adaptive critics and the basal ganglia," in *Models of Information Processing in the Basal Ganglia*, eds J. C. Houk, J. Davis, and D. Beiser (Cambridge, MA: MIT Press), 215–232.

Barto, A. G., Singh, S., and Chantanez, N. (2004). "Intrinsically motivated reinforcement learning," in *18th Annual Conference on Neural Information Processing Systems (NIPS)* (Vancouver, BC).

Beckstead, R. M., Domesick, V. B., and Nauta, W. J. (1979). Efferent connections of the substantia nigra and ventral tegmental area in the rat. *Brain Res.* 175, 191–217.

Bienenstock, E. L., Cooper, L. N., and Munro, P. W. (1982). Theory for the development of neuron selectivity: orientation specificity and binocular interaction in visual cortex. *J. Neurosci.* 2, 32–48.

Bromberg-Martin, E. S., Matsumoto, M., and Hikosaka, O. (2010). Dopamine in motivational control: rewarding, aversive, and alerting. *Neuron* 68, 815–834. (PMID: 21144997 PMCID: PMC3032992).

Brown, J. W., Bullock, D., and Grossberg, S. (2004). How laminar frontal cortex and basal ganglia circuits interact to control planned and reactive saccades. *Neural Netw.* 17, 471–510.

Bubic, A., von Cramon, D. Y., and Schubotz, R. I. (2010). Prediction, cognition and the brain. *Front. Hum. Neurosci.* 4:25. doi: 10.3389/fnhum.2010.00025

Bunzeck, N., and Düzel, E. (2006). Absolute coding of stimulus novelty in the human substantia nigra/VTA. *Neuron* 51, 369–379.

Calabresi, P., Picconi, B., Tozzi, A., and Di Filippo, M. (2007). Dopamine-mediated regulation of corticostriatal synaptic plasticity. *Trends Neurosci.* 30, 211–219.

Cisek, P. (2007). Cortical mechanisms of action selection: the affordance competition hypothesis. *Philos. Trans. R. Soc. Lond. B Biol. Sci.* 362, 1585–1599.

Coizet, V., Comoli, E., Westby, G. W. M., and Redgrave, P. (2003). Phasic activation of substantia nigra and the ventral tegmental area by chemical stimulation of the superior colliculus: an electrophysiological investigation in the rat. *Eur. J. Neurosci.* 17, 28–40.

Comoli, E., Coizet, V., Boyes, J., Bolam, J. P., Canteras, N. S., Quirk, R. H., et al. (2003). A direct projection from superior colliculus to substantia nigra for detecting salient visual events. *Nat. Neurosci.* 6, 974–980.

Cooper, L. N., Intrator, N., Blais, S. B., and Shouval, Z. H. (2004). *Theory of Cortical Plasticity*. Hackensack, NJ: World Scientific Publishing.

Cragg, S., and Rice, M. (2004). DAncing past the DAT at a DA synapse. *Trends Neurosci.* 27, 270–277.

Cyberbotics. (2010a). Webots Reference Manual. Retrieved from http://www.cyberbotics.com/reference.pdf

Cyberbotics. (2010b). Webots User Guide. Retrieved from http://www.cyberbotics.com/guide.pdf

Dean, P., Redgrave, P., and Westby, G. (1989). Event or emergency – 2 response systems in the mammalian superior colliculus. *Trends Neurosci.* 12, 137–147.

Deniau, J., and Chevalier, G. (1985). Disinhibition as a basic process in the expression of striatal functions II. The striato-nigral influence on thalamocortical cells of the ventromedial thalamic nucleus. *Brain Res.* 334, 227–233.

Dommett, E., Coizet, V., Blaha, C. D., Martindale, J., Lefebvre, V., Walton, N., et al. (2005). How visual stimuli activate dopaminergic neurons at short latency. *Science* 307, 1476–1479.

Doya, K. (1999). What are the computations of the cerebellum, the basal ganglia and the cerebral cortex? *Neural Netw.* 12, 961–974.

Drager, U. C., and Hubel, D. H. (1975). Responses to visual stimulation and relationship between visual, auditory, and somatosensory inputs in mouse superior colliculus. *J. Neurophysiol.* 38, 690–713.

Fino, E., Glowinski, J., and Venance, L. (2005). Bidirectional activity-dependent plasticity at corticostriatal synapses. *J. Neurosci.* 25, 11279–11287.

Fiorillo, C. D., Tobler, P. N., and Schultz, W. (2003). Discrete coding of reward probability and uncertainty by dopamine neurons. *Science* 299, 1898.

Frank, M. J. (2005). Dynamic dopamine modulation in the basal ganglia: a neurocomputational account of cognitive deficits in medicated and nonmedicated parkinsonism. *J. Cogn. Neurosci.* 17, 51–72.

Frank, M. J. (2006). Hold your horses: a dynamic computational role for the subthalamic nucleus in decision making. *Neural Netw.* 19, 1120–1136.

Frank, M. J. (2011). Computational models of motivated action selection in corticostriatal circuits. *Curr. Opin. Neurobiol.* 21, 381–386.

Frank, M. J., Seeberger, L. C., and O'Reilly, R. C. (2004). By carrot or by stick: cognitive reinforcement learning in parkinsonism. *Science* 306, 1940–1943.

Friston, K. (2010). The free-energy principle: a unified brain theory? *Nat. Rev. Neurosci.* 11, 127–138.

Gancarz, A. M., San George, M. A., Ashrafioun, L., and Richards, J. B. (2011). Locomotor activity in a novel environment predicts both responding for a visual stimulus and self-administration of a low dose of methamphetamine in rats. *Behav. Processes* 86, 295–304.

Gauthier, J., Parent, M., Lvesque, M., and Parent, A. (1999). The axonal arborization of single nigrostriatal neurons in rats. *Brain Res.* 83, 228–232.

Gibson, J. J. (1986). *The Ecological Approach to Visual Perception*. Oxford, UK: Lawrence Erlbaum Associates.

Groves, P. M., and Thompson, R. F. (1970). Habituation: a dual-process theory. *Psychol. Rev.* 77, 419–450.

Gurney, K., and Humphries, M. (2012). "Methodological issues in modelling at multiple levels of description," in *Computational Systems Neurobiology*. (Netherlands: Springer), 259–281.

Gurney, K., Prescott, T. J., and Redgrave, P. (2001a). A computational model of action selection in the basal ganglia. I. A new functional anatomy. *Biol. Cybern.* 84, 401–410.

Gurney, K., Prescott, T. J., and Redgrave, P. (2001b). A computational model of action selection in the basal ganglia. II. Analysis and simulation of behaviour. *Biol. Cybern.* 84, 411–423.

Gurney, K. N. (2009). Reverse engineering the vertebrate brain: Methodological principles for a biologically grounded programme of cognitive modelling. *Cogn. Comput.* 1, 29–41.

Gurney, K. N., Humphries, M. D., and Redgrave, P. (2009). Cortico-striatal plasticity for action-outcome learning using spike timing dependent eligibility. *BMC Neurosci.* 10(Suppl. 1):P135. doi: 10.1186/1471-2202-10-S1-P135

Gurney, K. N., Lepora, N., Shah, A., Koene, A., and Redgrave, P. (2013). *Action Discovery and Intrinsic Motivation: A Biologically Constrained Formalisation*. Berlin: Verlag, Springer.

Hazy, T. E., Frank, M. J., and O'Reilly, R. C. (2010). Neural mechanisms of acquired phasic dopamine responses in learning. *Neurosci. Biobehav. Rev.* 34, 701–720.

Houk, J. C., Bastianen, C., Fansler, D., Fishbach, A., Fraser, D., Reber, P. J., et al. (2007). Action selection and refinement in subcortical loops through basal ganglia and cerebellum. *Philos. Trans R. Soc. Lond. B Biol. Sci.* 362, 1573–1583.

Humphries, M. D., and Gurney, K. N. (2002). The role of intra-thalamic and thalamocortical circuits in action selection. *Network* 13, 131–156.

Ikeda, T., and Hikosaka, O. (2003). Reward-dependent gain and bias of visual responses in primate superior colliculus. *Neuron* 39, 693–700.

Jenkinson, N., Nandi, D., Muthusamy, K., Ray, N. J., Gregory, R., Stein, J. F., et al. (2009). Anatomy, physiology, and pathophysiology of the pedunculopontine nucleus. *Mov. Disord.* 24, 319–328.

Joel, D., Niv, Y., and Ruppin, E. (2002). Actor-critic models of the basal ganglia: new anatomical and computational perspectives. *Neural Netw.* 15, 535–547.

Kakade, S., and Dayan, P. (2002). Dopamine: generalization and bonuses. *Neural Netw.* 15, 549–559.

Khamassi, M., Lallée, S., Enel, P., Procyk, E., and Dominey, P. F. (2011). Robot cognitive control with a neurophysiologically inspired reinforcement learning model. *Front. Neurorobot.* 5:1. doi: 10.3389/fnbot.2011.00001

Kumaran, D., and Maguire, E. A. (2007). Which computational mechanisms operate in the hippocampus during novelty detection? *Hippocampus* 17, 735–748.

Lisman, J. E., and Grace, A. A. (2005). The hippocampal-vta loop: controlling the entry of information into long-term memory. *Neuron* 46, 703–713.

Lloyd, D. R., Gancarz, A. M., Ashrafioun, L., Kausch, M. A., and Richards, J. B. (2012). Habituation and the reinforcing effectiveness of visual stimuli. *Behav. Processes* 91, 184–191.

Lorenz, K. (1935). Der kumpan in der umwelt des vogels. *J. Ornithol.* 83, 137–213; 289–413.

Marsland, S. (2009). Using habituation in machine learning. *Neurobiol. Learn. Mem.* 92, 260–266.

Matsuda, W., Furuta, T., Nakamura, K. C., Hioki, H., Fujiyama, F., Arai, R., et al. (2009). Single nigrostriatal dopaminergic neurons form widely spread and highly dense axonal arborizations in the neostriatum. *J. Neurosci.* 29, 444–453.

Matsumoto, M., and Hikosaka, O. (2007). Lateral habenula as a source of negative reward signals in dopamine neurons. *Nature* 447, 1111–1115.

McGrenere, J, and Ho, W. (2000). "Affordances: clarifying and evolving a concept," in *Graphics Interface 2000: Proceedings* (Montreal, QC), 179. Available online at: http://www.interaction-design.org/references/conferences/proceedings_of_graphics_interface_2000.html

McSweeney, F. K., and Murphy, E. S. (2009). Sensitization and habituation regulate reinforcer effectiveness. *Neurobiol. Learn. Mem.* 92, 189–198.

Mink, J. W., and Thach, W. T. (1993). Basal ganglia intrinsic circuits and their role in behavior. *Curr. Opin. Neurobiol.* 3, 950–957.

Mondada, F., Franzi, E., and Guignard, A. (1999). "The development of Khepera," in *Experiments with the Mini-Robot Khepera, Proceedings of the First International Khepera Workshop*, HNI-Verlagsschriftenreihe (Heinz Nixdorf Institut), 7–14. Available online at: http://infoscience.epfl.ch/record/89709

Morris, G., Nevet, A., Arkadir, D., Vaadia, E., and Bergman, H. (2006). Midbrain dopamine neurons encode decisions for future action. *Nat. Neurosci.* 9, 1057–1063.

Moss, J., and Bolam, J. P. (2008). A dopaminergic axon lattice in the striatum and its relationship with cortical and thalamic terminals. *J. Neurosci.* 28, 11221–11230.

Oudeyer, P.-Y., and Kaplan, F. (2007). What is intrinsic motivation? a typology of computational approaches. *Front. Neurorobot.* 1:6. doi: 10.3389/neuro.12.006.2007

Pawlak, V., and Kerr, J. N. (2008). Dopamine receptor activation is required for corticostriatal spike-timing-dependent plasticity. *J. Neurosci.* 28, 2435.

Pear, J. J. (1985). Spatiotemporal patterns of behavior produced by variable-interval schedules of reinforcement. *J. Exp. Anal. Behav.* 44, 217–231. (PMID: 16812432.)

Pfister, J.-P., and Gerstner, W. (2006). Triplets of spikes in a model of spike timing-dependent plasticity. *J. Neurosci.* 26, 9673–9682.

Pleger, B., Blankenburg, F., Ruff, C. C., Driver, J., and Dolan, R. J. (2008). Reward facilitates tactile judgments and modulates hemodynamic responses in human primary somatosensory cortex. *J. Neurosci.* 28, 8161–8168.

Prescott, T. J., Montes Gonzalez, F. M., Gurney, K., Humphries, M. D., and Redgrave, P. (2006). A robot model of the basal ganglia: behavior and intrinsic processing. *Neural Netw.* 19, 31–61.

Pérez-Uribe, A. (2001). "Using a time-delay actor-critic neural architecture with dopamine-like reinforcement signal for learning in autonomous robots," in *Emergent Neural Computational Architectures Based on Neuroscience, Vol. 2036*, eds S. Wermter, J. Austin, and D. Willshaw (Berlin, Heidelberg: Springer), 522–533.

Ranganath, C., and Rainer, G. (2003). Neural mechanisms for detecting and remembering novel events. *Nat. Rev. Neurosci.* 4, 193–202.

Rankin, C. H., Abrams, T., Barry, R. J., Bhatnagar, S., Clayton, D. F., Colombo, J., et al. (2009). Habituation revisited: an updated and revised description of the behavioral characteristics of habituation. *Neurobiol. Learn. Mem.* 92, 135–138.

Redgrave, P., and Gurney, K. (2006). The short-latency dopamine signal: a role in discovering novel actions? *Nat. Rev. Neurosci.* 7, 967–975.

Redgrave, P., Gurney, K., and Reynolds, J. (2008). What is reinforced by phasic dopamine signals? *Brain Res. Rev.* 58, 322–339.

Redgrave, P., Prescott, T. J., and Gurney, K. (1999). The basal ganglia: a vertebrate solution to the selection problem? *Neuroscience* 89, 1009–1023.

Redgrave, P., Vautrelle, N., and Reynolds, J. N. J. (2011). Functional properties of the basal ganglia's re-entrant loop architecture: selection and reinforcement. *Neuroscience* 198, 138–151.

Reynolds, J. N. J., and Wickens, J. R. (2002). Dopamine-dependent plasticity of corticostriatal synapses. *Neural Netw.* 15, 507–521.

Roesch, M. R., Calu, D. J., and Schoenbaum, G. (2007). Dopamine neurons encode the better option in rats deciding between differently delayed or sized rewards. *Nat. Neurosci.* 10, 1615–1624.

Ryan, R. M., and Deci, E. L. (2000). Intrinsic and extrinsic motivations: classic definitions and new directions. *Contemp. Educ. Psychol.* 25, 54–67.

Samson, R. D., Frank, M. J., and Fellous, J. M. (2010). Computational models of reinforcement learning: the role of dopamine as a reward signal. *Cogn. Neurodyn.* 4, 91–105.

Schmidhuber, J. (2009). "Driven by compression progress: a simple principle explains essential aspects of subjective beauty, novelty, surprise, interestingness, attention, curiosity, creativity, art, science, music, jokes," in *Anticipatory Behavior in Adaptive Learning Systems, volume 5499 of Lecture Notes in Computer Science*, 48–76. EU Funded Projects. 4th Workshop on Anticipatory Behavior in Adaptive Learning Systems, Munich, Germany, Jun 26–27, 2008.

Schultz, W. (2006). Behavioral theories and the neurophysiology of reward. *Annu. Rev. Psychol.* 57, 87–115.

Schultz, W. (2010). Dopamine signals for reward value and risk: basic and recent data. *Behav. Brain Funct.* 6, 24.

Schultz, W., Dayan, P., and Montague, P. R. (1997). A neural substrate of prediction and reward. *Science* 275, 1593–1599.

Shen, W., Flajolet, M., Greengard, P., and Surmeier, D. J. (2008). Dichotomous dopaminergic control of striatal synaptic plasticity. *Science* 321, 848–851.

Song, S., Miller, K. D., and Abbott, L. F. (2000). Competitive hebbian learning through spike-timing-dependent synaptic plasticity. *Nat. Neurosci.* 3, 919–926.

Sporns, O., and Alexander, W. H. (2002). Neuromodulation and plasticity in an autonomous robot. *Neural Netw.* 15, 761–774.

Suri, R. (2002). TD models of reward predictive responses in dopamine neurons. *Neural Netw.* 15, 523–533.

Suri, R. E., and Schultz, W. (1998). Learning of sequential movements by neural network model with dopamine-like reinforcement signal. *Exp. Brain Res.* 121, 350–354.

Suri, R. E., and Schultz, W. (1999). A neural network model with dopamine-like reinforcement signal that learns a spatial delayed response task. *Neuroscience* 91, 871–890.

Sutton, R. S., and Barto, A. G. (1998). *Reinforcement Learning: An Introduction*. Cambridge, MA: MIT Press.

Takakusaki, K., Saitoh, K., Harada, H., and Kashiwayanagi, M. (2004). Role of basal ganglia-brainstem pathways in the control of motor behaviors. *Neurosci. Res.* 50, 137–151.

Thivierge, J.-P., Rivest, F., and Monchi, O. (2007). Spiking neurons, dopamine, and plasticity: timing is everything, but concentration also matters. *Synapse* 61, 375–390.

Thompson, A. M., Porr, B., and Woergoetter, F. (2010). Learning and reversal learning in the subcortical limbic system: a computational model. *Adaptive Behav.* 18, 211–236.

Tinbergen, N. (1951). *The Study of Instinct*. Oxford, UK: Oxford University Press.

Tobler, P., Fiorillo, C., and Schultz, W. (2005). Adaptive coding of reward value by dopamine neurons. *Science* 307, 1642.

Weil, R. S., Furl, N., Ruff, C. C., Symmonds, M., Flandin, G., Dolan, R. J., et al. (2010). Rewarding feedback after correct visual discriminations has both general and specific influences on visual cortex. *J. Neurophysiol.* 104, 1746–1757.

West, A. R., and Grace, A. A. (2002). Opposite influences of endogenous dopamine d1 and d2 receptor activation on activity states and electrophysiological properties of striatal neurons: studies combining *in vivo* intracellular recordings and reverse microdialysis. *J. Neurosci.* 22, 294–304.

Wurtz, R. H., and Albano, J. E. (1980). Visual-motor function of the primate superior colliculus. *Annu. Rev. Neurosci.* 3, 189–226.

Wurtz, R. H., and Goldberg, M. E. (1972). The primate superior colliculus and the shift of visual attention. *Invest. Ophthalmol.* 11, 441–450.

Zink, C. F., Pagnoni, G., Chappelow, J., Martin-Skurski, M., and Berns, G. S. (2006). Human striatal activation reflects degree of stimulus saliency. *Neuroimage* 29, 977–983.

Conflict of Interest Statement: 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.

Received: 03 December 2012; paper pending published: 30 December 2012; accepted: 20 February 2013; published online: 12 March 2013.

Citation: Bolado-Gomez R and Gurney K (2013) A biologically plausible embodied model of action discovery. Front. Neurorobot. 7:4. doi: 10.3389/fnbot.2013.00004

APPENDIX

DETAILS OF BIOMIMETIC CORE MODEL

We give details here of the equations defining the biomimetic core model which were not given in the main text. In most cases this amounts to identifying the form of the net input I in Equation (5), and parameterizing the output function (Equation 6). In what follows, indices refer to action channels.

Basal ganglia

Sensory, and motor cortical output are denoted by y_i^{S}, y_i^{M}, respectively. The tonic dopamine level $\lambda = 0.2$.

$$\begin{aligned}
\text{Striatum D1:}\quad & I_i^{\mathrm{D1}} = (w_i^{\mathrm{S,D1}} y_i^{\mathrm{S}} + w_i^{\mathrm{M,D1}} y_i^{\mathrm{M}})(1+\lambda) \\
& \text{with initial weight values;}\quad w_i^{\mathrm{S,D1}} = 0, \\
& w_i^{\mathrm{M,D1}} = 0.45 \\
& y_i^{\mathrm{D1}} = L(I_i^{\mathrm{D1}}, 0.1) \\
\text{Striatum D2:}\quad & I_i^{\mathrm{D2}} = (w_i^{\mathrm{S,D2}} y_i^{\mathrm{S}} + w_i^{\mathrm{M,D2}} y_i^{\mathrm{M}})(1-\lambda) \\
& \text{with initial weight values;}\quad w_i^{\mathrm{S,D2}} = 0, \\
& w_i^{\mathrm{M,D2}} = 0.45 \\
& y_i^{\mathrm{D2}} = L(I_i^{\mathrm{D2}}, 0.1) \\
\text{STN:}\quad & I_i^{\mathrm{STN}} = 0.4(y_i^{\mathrm{S}} + y_i^{\mathrm{M}}) - 0.2 y_i^{\mathrm{GPE}} \\
& y_i^{\mathrm{STN}} = L(I_i^{\mathrm{STN}}, -0.25) \\
\text{GPe:}\quad & I_i^{GPe} = 0.3 \sum_{i=1}^{3} y_i^{\mathrm{STN}} - 0.9 y_i^{\mathrm{D2}} \\
& y_i^{\mathrm{GPe}} = L(I_i^{\mathrm{GPe}}, -0.2) \\
\text{GPi/SNr:}\quad & I_i^{\mathrm{GPi}} = 0.3 \sum_{i=1}^{3} y_i^{\mathrm{STN}} - 0.7 y_i^{\mathrm{D1}} - 0.4 y_i^{\mathrm{GPe}} \\
& y_i^{\mathrm{GPi}} = L(I_i^{\mathrm{GPi}}, -0.12)
\end{aligned}$$

Thalamus and brainstem

$$\begin{aligned}
\text{TRN :}\quad & I_i^{\mathrm{TRN}} = y_i^{\mathrm{M}} + y_i^{\mathrm{VL}} \\
& y_i^{\mathrm{TRN}} = L(I_i^{\mathrm{TRN}}, 0) \\
\text{VL Thalamus :}\quad & I_i^{\mathrm{VL}} = 0.9 y_i^{\mathrm{M}} - y_i^{\mathrm{GPi}} \\
& \qquad - 0.01 y_i^{\mathrm{TRN}} \left(1 - 0.11 \sum_{j \neq i} y_j^{\mathrm{TRN}}\right) \\
& y^{\mathrm{VL}} = L(I_i^{\mathrm{VL}}, 0) \\
\text{Brainstem:}\quad & I_i^{\mathrm{BS}} = y_i^{\mathrm{M}}(1 - 1.5 y_i^{\mathrm{GPi}}) \\
& y_i^{\mathrm{BS}} = L(I_i^{\mathrm{BS}}, 0)
\end{aligned}$$

The action is behaviorally enacted if $y_i^{\mathrm{BS}} > \phi$ (recall $\phi = 0.5$).

Cortex

For the sensory cortex, the input c_i is provided by the salience generation process (section 2.5.2)

$$\begin{aligned}
I_i^{\mathrm{S}} &= c_i \\
y_i^{\mathrm{S}} &= L(I_i^{\mathrm{S}}, 0)
\end{aligned}$$

For motor cortex, we consider two classes of action representation. For the "explore" action, arbitrarily assigned as channel 1

$$\begin{aligned}
I_1^{\mathrm{M}} &= 0.75 y_1^{\mathrm{S}} + 0.89 y_1^{\mathrm{VL}} \\
y_i^{\mathrm{M}} &= L(I_i^{\mathrm{M}}, 0)
\end{aligned}$$

For the block-interaction channels ($i = 2, 3$), we incorporated a recurrent, self reinforcing connection if the action is currently selected.

$$\begin{aligned}
I_i^{\mathrm{M}} &= 0.75 y_i^{\mathrm{S}} + 0.89 y_1^{\mathrm{VL}} + 0.005 y_i^{\mathrm{M}} H(y_i^{\mathrm{BS}} - \phi) \\
y_i^{\mathrm{M}} &= L(I_i^{\mathrm{M}}, 0)
\end{aligned}$$

where $H()$ is the Heaviside step function and ϕ is the same threshold used in selecting behavior in brainstem (see "Thalamus and Brainstem," above). The self-recurrence here plays a similar role to the "busy signal" used by Prescott et al. (2006) to ensure correct execution of fixed action patters (FAPs) which should not time-out before their completion. This signal was driven explicitly by an internal clock and knowledge of the FAP duration. In contrast, we have taken a slightly different approach, which is more neurally plausible and does allow for interruption of the action by a very highly salient competitor. In this way we have something more akin to a soft-action pattern (SAP) process.

frontiers in
NEUROROBOTICS

ORIGINAL RESEARCH ARTICLE
published: 28 February 2013
doi: 10.3389/fnbot.2013.00003

Scaled free-energy based reinforcement learning for robust and efficient learning in high-dimensional state spaces

Stefan Elfwing*, Eiji Uchibe and Kenji Doya

Neural Computation Unit, Okinawa Institute of Science and Technology, Graduate University, Okinawa, Japan

Edited by:
Jeffrey L. Krichmar, University of California, Irvine, USA

Reviewed by:
Mehdi Khamassi, Centre National de la Recherche Scientifique, France
Brandon Rohrer, Sandia National Laboratories, USA

***Correspondence:**
Stefan Elfwing, Neural Computation Unit, Okinawa Institute of Science and Technology, Graduate University, 1919-1 Tancha, Onna-son, Okinawa 904-0412, Japan.
e-mail: elfwing@oist.jp

Free-energy based reinforcement learning (FERL) was proposed for learning in high-dimensional state- and action spaces, which cannot be handled by standard function approximation methods. In this study, we propose a scaled version of free-energy based reinforcement learning to achieve more robust and more efficient learning performance. The action-value function is approximated by the negative free-energy of a restricted Boltzmann machine, divided by a constant scaling factor that is related to the size of the Boltzmann machine (the square root of the number of state nodes in this study). Our first task is a digit floor gridworld task, where the states are represented by images of handwritten digits from the MNIST data set. The purpose of the task is to investigate the proposed method's ability, through the extraction of task-relevant features in the hidden layer, to cluster images of the same digit and to cluster images of different digits that corresponds to states with the same optimal action. We also test the method's robustness with respect to different exploration schedules, i.e., different settings of the initial temperature and the temperature discount rate in softmax action selection. Our second task is a robot visual navigation task, where the robot can learn its position by the different colors of the lower part of four landmarks and it can infer the correct corner goal area by the color of the upper part of the landmarks. The state space consists of binarized camera images with, at most, nine different colors, which is equal to 6642 binary states. For both tasks, the learning performance is compared with standard FERL and with function approximation where the action-value function is approximated by a two-layered feedforward neural network.

Keywords: reinforcement learning, free-energy, restricted Boltzmann machine, robot navigation, function approximation

1. INTRODUCTION

Reinforcement learning (Sutton and Barto, 1998) has been proven to be effective for a wide variety of delayed reward problems. However, standard reinforcement learning algorithms cannot handle high-dimensional state spaces. For standard action-value function approximators, such as tile coding and radial basis function networks, the number of features of the function approximator grows exponentially with the dimension of the state- and action spaces.

Sallans and Hinton (2004) proposed free-energy based reinforcement learning (FERL) to handle high-dimensional state- and action spaces. In their method, the action-value function, Q, is approximated as the negative free-energy of a restricted Boltzmann machine (Smolensky, 1986; Freund and Haussler, 1992; Hinton, 2002). In this study, we propose a scaled version of FERL to achieve more robust and more efficient learning. The action-value function is approximated as the negative free-energy, divided with a constant scaling factor that is related to the size of the Boltzmann machine (the square root of the number of state nodes in this study). The initialization of the network weights and, thereby the initial Q-values, is a difficult problem in FERL. Even if the network weights are randomly initialized using a distribution with zero mean, the magnitude of the initial free-energy grows with the size of the network. The introduction of a scaling factor can, therefore, reduce this problem by initializing the Q-values to a more appropriate range. In addition, the scaling of the free-energy reduces the effect of a change in the weight values (i.e., a learning update) on the approximated Q-values. This makes it less likely that the learning diverges or get trapped in suboptimal solutions.

To validate the scaled version of FERL, we compare the learning performance with standard FERL and learning with a two-layered feedforward neural network. Our first experiment is a digit floor gridworld task, where the states are represented by images of handwritten digits from the MNIST data set. The purpose of the task is to investigate our proposed method's ability to extract task-relevant features in the hidden layer, i.e., to cluster images of the same digit and to cluster images of different digits that correspond to states with the same optimal action. We also test the method's robustness with respect to different exploration schedules, i.e., different settings of the initial temperature and the temperature discount rate in softmax action selection. Our second experiment is a robot visual navigation task, where the goal is to reach the correct goal area, which can be inferred by the color of the upper part of four landmarks. The color of the lower part of each landmark is unique and identifies the landmark's position, and can therefore be used for localization.

Apart from Sallans' and Hinton's (Sallans and Hinton, 2004) pioneering work, there have been few studies using a free-energy approach to function approximation in reinforcement learning. In our earlier study (Elfwing et al., 2010), we demonstrated the feasibility to use FERL for on-line control with high-dimensional state inputs in a visual navigation and battery capturing task with similar experimental setup as the visual navigation task in this study. We also demonstrated successful on-line learning in a real robot for a simpler battery capturing task. In this study, we compare the performance of scaled FERL with the standard FERL approach that was used in our earlier study. Otsuka et al. (2010) extended the FERL method to handle partially observable Markov decision processes (POMDPs), by incorporating a recurrent neural network that learns a memory representation that is sufficient for predicting future observations and rewards. The incorporation of memory capability does not improve the learning performance of standard FERL for the MDP tasks considered in this study.

2. METHOD

2.1. GRADIENT-DESCENT SARSA(λ)

The FERL method that we propose here is based on the on-policy reinforcement learning algorithm (Sutton and Barto, 1998) Sarsa(λ) (Rummery and Niranjan, 1994; Sutton, 1996), which learns an estimate of the action-value function, Q^π, while the agent follows policy π. If the approximated action value function, $Q_t \approx Q^\pi$, is parameterized by the parameter vector $\boldsymbol{\theta}_t$, then the gradient-descent update of the parameters is

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + \alpha\delta_t \boldsymbol{e}_t, \tag{1}$$

where the TD-error, δ_t is

$$\delta_t = r_t + \gamma Q_t(s_{t+1}, a_{t+1}) - Q_t(s_t, a_t), \tag{2}$$

and the eligibility trace vector, $\boldsymbol{e}_t$, is

$$\boldsymbol{e}_t = \gamma\lambda e_{t-1} + \nabla_{\boldsymbol{\theta}_t} Q_t(s_t, a_t), \quad \boldsymbol{e}_0 = \boldsymbol{0}. \tag{3}$$

Here, s_t is the state at time t, a_t is the action selected at time t, r_t is the reward for taking action a_t in state s_t, α is the learning rate, and γ is the discount factor of future rewards, λ is the trace-decay rate, and $\nabla_{\boldsymbol{\theta}_t} Q_t$ is the vector of partial derivatives of the function approximator with respect to each component of $\boldsymbol{\theta}_t$. In this study, the action-value function is approximated by the negative free-energy of a restricted Boltzmann machine.

2.2. FREE-ENERGY BASED FUNCTION APPROXIMATION

The use of a restricted Boltzmann machine (Smolensky, 1986; Freund and Haussler, 1992; Hinton, 2002) as a function approximator for reinforcement learning was proposed by Sallans and Hinton (2004). A restricted Boltzmann machine (**Figure 1**) is a bi-directional neural network which consists of binary state nodes, $\boldsymbol{s}$, binary action nodes $\boldsymbol{a}$, and hidden nodes, $\boldsymbol{h}$. The ith state node, s_i, is connected to hidden node h_k by the weight w_{ik}, and the jth action node, a_j, is connected to hidden node h_k by the weight u_{jk}. In addition, the state nodes, the action nodes, and the hidden nodes are all connected to a constant bias input with a value of 1, with connection weights b_i, b_j, and b_k, respectively. The free-energy, F, of the restricted Boltzmann machine is given as

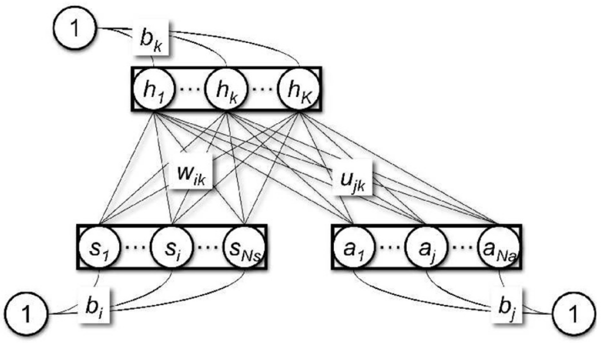


FIGURE 1 | Restricted Boltzmann machine.

$$\begin{aligned} F(\boldsymbol{s}, \boldsymbol{a}) = &-\sum_{k=1}^{K}\left(\sum_{i=1}^{N_s} w_{ik}s_ih_k + \sum_{j=1}^{N_a} u_{jk}a_jh_k\right) - \sum_{i=1}^{N_s} b_is_i \\ &- \sum_{j=1}^{N_a} b_ja_j - \sum_{k=1}^{K} b_kh_k + \\ &+ \sum_{k=1}^{K}\left(h_k \log h_k + (1-h_k)\log(1-h_k)\right). \end{aligned} \tag{4}$$

Here, K is the number of hidden nodes, N_s is the number of state nodes, and N_a is the number of action nodes. The free-energy of each action j is computed by setting the corresponding action node, a_j, to 1 and the rest of the action nodes to 0. h_k is the activation of the kth hidden node, given as

$$h_k = \sigma\left(\sum_{i=1}^{N_s} w_{ik}s_i + \sum_{j=1}^{N_a} u_{jk}a_j + b_k\right), \tag{5}$$

where

$$\sigma(x) = \frac{1}{1+e^{-x}}. \tag{6}$$

In Sallans' and Hinton's (Sallans and Hinton, 2004) original proposal, the action-value function was approximated by the negative free-energy, i.e., $Q_t = -F_t$. In this study, we propose that the performance and the robustness of free-energy based function approximation can be improved by scaling the free-energy by a constant scaling factor, Z, that is related to the size of the Boltzmann machine, i.e., $Q_t = -F_t/Z$. The update of the learning parameters (Equations 1–3) then becomes

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + \alpha\delta_t \boldsymbol{e}_t, \tag{7}$$

$$\delta_t = r_t - \gamma\frac{F_t(\boldsymbol{s}_{t+1}, \boldsymbol{a}_{t+1})}{Z} + \frac{F_t(\boldsymbol{s}_t, \boldsymbol{a}_t)}{Z}, \tag{8}$$

$$\boldsymbol{e}_t = \gamma\lambda\boldsymbol{e}_{t-1} + \frac{1}{Z}\nabla_{\boldsymbol{\theta}_t}(-F_t(\boldsymbol{s}_t, \boldsymbol{a}_t)). \tag{9}$$

The derivatives of the negative free-energy, with respect to the function approximator parameters (w_{ik}, u_{jk}, b_i, b_j, and b_k), can be computed as

$$\begin{aligned}\nabla_{w_{ik}}(-F(\boldsymbol{s}, \boldsymbol{a})) &= s_i h_k,\\ \nabla_{u_{jk}}(-F(\boldsymbol{s}, \boldsymbol{a})) &= a_j h_k,\\ \nabla_{b_i}(-F(\boldsymbol{s}, \boldsymbol{a})) &= s_i,\\ \nabla_{b_j}(-F(\boldsymbol{s}, \boldsymbol{a})) &= a_j,\\ \nabla_{b_k}(-F(\boldsymbol{s}, \boldsymbol{a})) &= h_k.\end{aligned} \tag{10}$$

Since

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + \alpha\left(r_t - \gamma\frac{F_t(\boldsymbol{s}_{t+1}, \boldsymbol{a}_{t+1})}{Z} + \frac{F_t(\boldsymbol{s}_t, \boldsymbol{a}_t)}{Z}\right)$$

$$\sum_{i=1}^{t}\frac{\gamma^{t-i}\lambda^{t-i}}{Z}\nabla_{\boldsymbol{\theta}_i}(-F_i(\boldsymbol{s}_i, \boldsymbol{a}_i)), \tag{11}$$

$$= \boldsymbol{\theta}_t + \frac{\alpha}{Z^2}(Zr_t - \gamma F_t(\boldsymbol{s}_{t+1}, \boldsymbol{a}_{t+1}) + F_t(\boldsymbol{s}_t, \boldsymbol{a}_t))$$

$$\sum_{i=1}^{t}\gamma^{t-i}\lambda^{t-i}\nabla_{\boldsymbol{\theta}_i}(-F_i(\boldsymbol{s}_i, \boldsymbol{a}_i)), \tag{12}$$

the scaled version of FERL can be transformed to the original formulation by re-scaling the learning rate ($\alpha' = \alpha/Z^2$) and the magnitude of the reward function ($r_t' = Zr_t$).

2.3. ACTION SELECTION

In this study, we use softmax action selection with a Boltzmann distribution, where the probability to select action a in state s is defined as

$$P(a|s) = \frac{\exp(Q(s, a)/\tau)}{\sum_b \exp(Q(s, b)/\tau)}. \tag{13}$$

Here, τ is the temperature that controls the trade-off between exploration and exploitation. In this study, we used hyperbolic discounting of the temperature and the temperature was decreased every episode i:

$$\tau(i) = \frac{\tau_0}{1 + \tau_k i}. \tag{14}$$

Here, τ_0 is the initial temperature and τ_k controls the rate of discounting.

To transform the scaled version to the original formulation when using softmax action selection, the temperature has also to be re-scaled ($\tau' = Z\tau$).

2.4. DIGIT FLOOR GRIDWORLD TASK

Figure 2 shows the digit floor gridworld task. The thick purple lines indicate the outer walls and the wall between state "1" and state "4." The yellow lines indicate zero reward state transitions. The red lines indicate negative reward (-0.01) for premature state transitions to the absorbing goal state (state "5") from states "2," "6," and "8." The green line indicates positive reward (+1) for successful completion of the task, i.e., state transition from state "4" to state "5." There were four actions that moved the agent one step in the directions North, East, South, and West. If the agent moved into a wall, then the agent remained in the current state and received a zero reward. The agent started each episode at state "1" and the goal of the task was to reach state "5" by moving counterclockwise along a path through states "2," "3," "6," "9," "8," "7," and "4." Each state consisted of an image of a handwritten digit from the MNIST data set (LeCun et al., 1998). The 28 × 28 pixels grayscale images were binarized by setting pixels with grayscale values larger than or equal to 128 to 1 and pixels with values smaller than or equal to 127 to 0. For each state, we used 20 different digit images that were randomly selected from the first 1000 images in the MNIST data set. At the start of each episode, the image for each state was randomly selected among the 20 possible images. An episode ended either when the agent moved to the absorbing state (state "5") or after a maximum number of steps (set to 1000).

2.5. ROBOT VISUAL NAVIGATION TASK

For the robot navigation task we used a simulation environment that was developed in MATLAB (2010) to mimic the properties of the Cyber Rodent robot (Doya and Uchibe, 2005). The Cyber Rodent is a small mobile robot, 22 cm in length and 1.75 kg in weight. The robot has a variety of sensors, including an

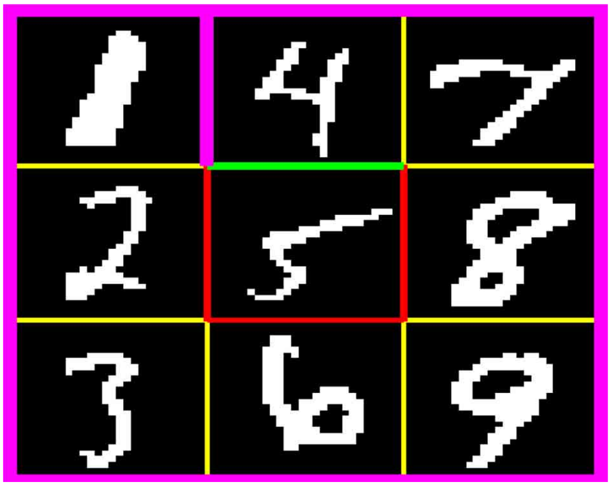


FIGURE 2 | Digit floor gridworld task. The thick purple lines indicate the outer walls and the wall between states "1" and "4." The yellow lines indicate zero reward state transitions. The red lines indicate negative reward (−0.01) for premature state transitions to the absorbing goal state (state "5") from states "2," "6," and "8." The green line indicates positive reward (+1) for successful completion of the task, i.e., state transition from state "4" to "5."

omnidirectional C-MOS camera, an infrared range sensor, seven infrared proximity sensors, gyros, and accelerometers. It has two wheels and a maximum speed of $1.3\,\mathrm{ms}^{-1}$. In addition to an on-board CPU (SH-4), it has an FPGA for real-time color blob detection.

The goal of the robot task (**Figure 3**) was to navigate to one of the four goal areas in the corners of the 2.5 × 2.5 m experimental area [dashed quarter circles in **Figure 3** (left panel)], by learning to infer the correct goal area by the color of the upper part of four landmarks (cyan color in **Figure 3**). The landmarks were located outside the corners of the experimental area. The color of the lower part of each landmark was unique and non-changing (red, green, blue, and black colors in **Figure 3**), and could therefore be used for localization. At the start of each episode, the correct goal area was randomly changed, and, thereby, also the corresponding color of the upper part of the landmarks. The robot was randomly placed in one of the four starting areas [dotted rectangles in **Figure 3** (left panel)]. The initial position within the starting area and the robot's initial heading angle were also randomly selected. We performed experiments with one goal area (southwest), two goal areas (southwest and northeast), three goal areas (southwest, southwest, and northeast), and all four goal areas.

The robot's simulated camera had a resolution of 738 (41 × 18) pixels covering a horizontal field of view of ±75°, with a 3.75° distance between the pixels. It could detect up to nine different colored objects: obstacles (purple in **Figure 3**), the lower part of the four landmarks (red, green, blue, and black in **Figure 3**), and one to four colors of the upper part of the landmarks (cyan in **Figure 3**), depending on the number of goals in the experiment. Within the field of view, the landmarks were visible from all distances and the obstacles were visible up to 2 m. The size of an object in the camera image increased with the inverse of the distance to the object. The state vector was constructed by creating a binary image of equal size to the original image for each color the robot could detect. The pixels that detected a colored object was extracted from the original image and the same pixels in the corresponding binary image was set to 1. All other pixels were set to 0. In addition, the state vector consisted of three normalized real-valued distance measures from the robot's front proximity sensors, located at −30°, 0°, and +30° in relation to the robot's heading direction. The distance information was normalized to the interval [0, 1] and higher values corresponded to shorter distances. The total length of the state vector in the experiment with four goals was 6645 (41 × 18 × 9 + 3). The robot could execute five actions, pairs of velocities (cm/s) of the left and the right wheels: rotate right (20, −20), curve right (40, 20), go straight (30, 30), curve left (20, 40), and rotate left (−20, 20). Gaussian noise was added to each wheel velocity, with zero mean and a standard deviation equal to 1% of the amplitude of the velocity. An episode ended either when the robot moved its head inside the correct goal area or when the length of the episode exceeded a fixed threshold of 2000 time steps. The robot received a +1 reward if it reached the correct goal area, otherwise the reward was set to 0.

3. RESULTS

To evaluate the proposed scaled version of FERL, we compared the performance with standard FERL and with function approximation using a two-layered feedforward neural network (hereafter NNRL). The state nodes s_i of the neural network were connected to K hidden nodes by weights w_{ik}. The hidden nodes had sigmoid activation functions (Equation 6), $\delta_k = \sigma(\sum_i w_{ik} s_i)$. The hidden nodes were connected to Q-value output nodes with linear activation by weights w_{ka}. The approximated Q-values were computed as the linear combination of the output weights and the hidden activation

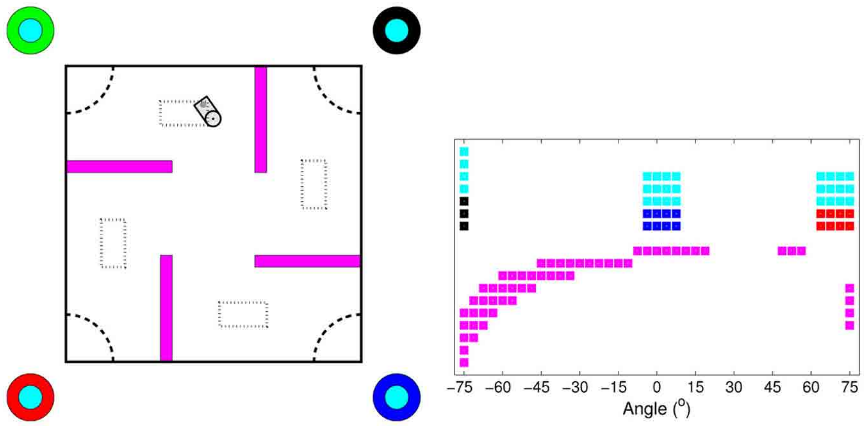


FIGURE 3 | Overview of the experimental area for the visual navigation tasks (left panel) and the camera image corresponding to the robot's position in the environment (right panel). In the left panel, the dashed quarter circles at the corners indicate the four goal areas and the dotted rectangles indicate the starting areas. The circles outside the experimental area indicate the four landmarks. The color of the lower part of each landmark was unique and non-changing. The color of the upper part of all landmarks corresponded to the correct goal area and was randomly changed at the start of each episode. Note that the difference in radius between the lower and the upper part of the landmarks is only for illustrative purposes. In the experiments, both parts of the landmarks had the same radius.

$[Q(\mathbf{s}, a) = \sum_k w_{ka}\delta_k]$, with derivatives with respect to the weight parameters computed as

$$\nabla_{w_{ik}}\left(Q(\mathbf{s}, a)\right) = \delta_k(1 - \delta_k)w_{ka}s_i,$$
$$\nabla_{w_{ka}}\left(Q(\mathbf{s}, a)\right) = \delta_k. \tag{15}$$

For the scaled FERL, we concluded after a trial and error process that a scaling factor equal to the square root of the number of state nodes ($Z = \sqrt{N_s}$) was an appropriate value for the experiments conducted in this study. For both tasks, the number of hidden nodes (K) was set to 20 for all three methods. In the gridworld task, we tested the robustness of the methods with the respect to different exploration schedules by comparing the learning performance for action selection with τ_0 set to 0.5, 1, and 2 and τ_k set to 0.01, 0.001, and 0.0005 (Equation 14). In the robot navigation task, τ_0 and τ_k were determined by searching for appropriate values in the experiment setting with two goal targets. τ_0 was set to 0.5 for all three methods. τ_k was set to 0.01 for scaled FERL and 0.002 for FERL and NNRL. **Table 1** shows the settings of α, γ, and λ in the experiments. For all three methods, the weights were randomly initialized using a Gaussian distribution with zero mean. For the weights connecting the state nodes and the hidden nodes the variance was equal to 0.001 and for weights connecting the hidden nodes and the action nodes the variance was equal to 1.

Table 1 | Meta-parameter settings for the experiments.

	Gridworld task			Robot task		
	Scaled FERL	**FERL**	**NNRL**	**Scaled FERL**	**FERL**	**NNRL**
α	0.01 × Z	0.001	0.001	0.01 × Z	0.001	0.001
γ	0.96	0.96	0.96	0.98	0.98	0.98
λ	0.8	0.8	0.8	0.8	0.8	0.8

3.1. DIGIT FLOOR GRIDWORLD TASK

For the gridworld task, we performed 20 simulations runs for each method and each setting of τ_0 and τ_k. **Figure 4** shows the average rewards computed over every 100 episodes. The result clearly shows better and more robust learning performance for scaled FERL (left panel in **Figure 4**). The learning converged to average reward values exactly equal to, or close to equal to, the maximum reward of 1 for 8 out of the 9 different settings of τ_0 and τ_k. The only exception was the experiment with the largest initial temperature ($\tau_0 = 2$) and lowest discount rate ($\tau_k = 0.0005$) where the average reward was still increasing at the end of learning (dotted blue line in the left panel in **Figure 4**). The learning speed was, not surprisingly, determined by the exploration schedule. Experiments with smaller initial temperatures and higher discount rates converged faster. In the experiment with the smallest initial temperature ($\tau_0 = 0.5$) and highest discount rate ($\tau_k = 0.01$), the average learning performance reached close to 1 after about 2500 episodes (solid red line in the left panel in **Figure 4**). The learning then converged after about 5250 episodes with the average reward exactly equal to 1 with 0 variance. If we define successful learning as a simulation run where, at the end of learning, the greedy action [$\mathrm{argmax}_a\, Q(s, a)$] was equal to the optimal action for all 20 digit images for all states, then

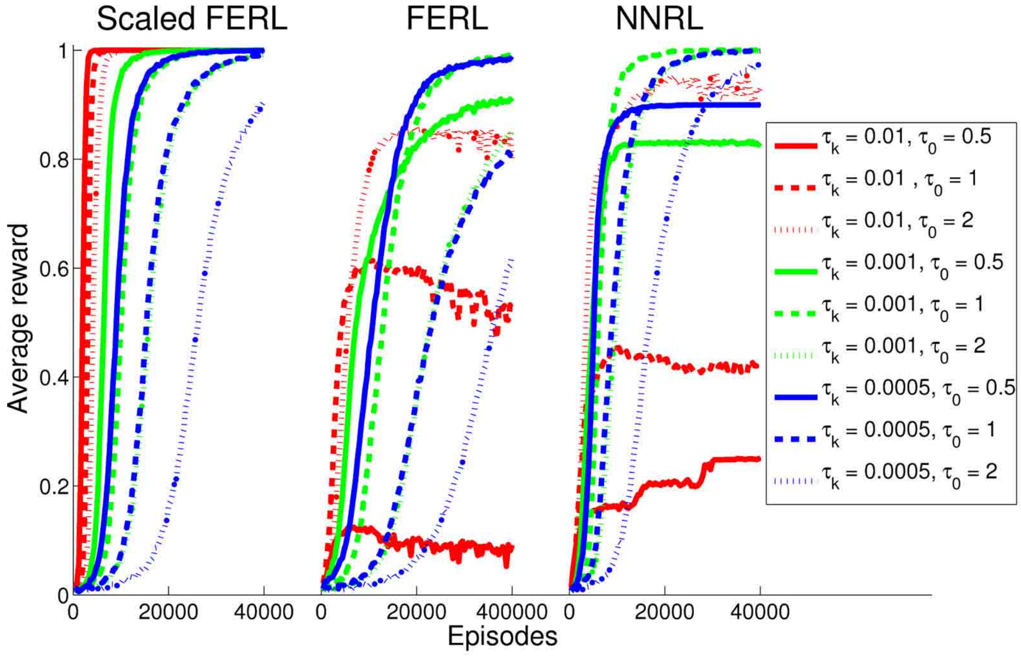


FIGURE 4 | The average reward computed over every 100 episodes and 20 simulation runs, for scaled FERL (left panel), FERL (middle panel), and NNRL (right panel). The line colors correspond to the settings of τ_k (red: 0.01, green: 0.001, and blue: 0.0005) and the line types correspond to the setting of τ_0 (solid: 0.5, dashed: 1, and dotted: 2).

scaled FERL was successful in 100% (20) of the simulation runs for eight settings of τ_0 and τ_k. The only exception was, again, the experiment with $\tau_0 = 2$ and $\tau_k = 0.0005$, where 90% (18) of the simulation runs were successful.

The exploration schedule did also effect the learning of the Q-values for optimal and non-optimal actions. **Figure 5** shows the average learned Q-values (circles) with standard deviations (bars) in the experiments with $\tau_0 = 0.01$ and $\tau_k = 0.01$ (left panel), $\tau_0 = 1$ and $\tau_k = 0.001$ (middle panel), and $\tau_0 = 2$ and $\tau_k = 0.0005$ (right panel), computed over all state images for all states in all 20 simulation runs for scaled FERL. The different colors show the values of the four different types of actions for the states along path from the initial state "1" to the goal state "5": (1) red for optimal actions; (2) blue for actions that moved the agent into a wall; (3) purple for actions that moved the agent away from the goal; and (4) black for negative rewarded actions. Since the goal reward was set to +1, the optimal Q-values (dashed red lines) were equal to γ^{t-1}, where t is the number of steps to the goal. A move into a wall increased the steps to the goal by one (optimal Q-values equal to γ^t, see dashed blue lines) and actions that moved agent away from the goal increased the steps to the goal by 2 (optimal Q-values equal to γ^{t+1}, see dashed purple lines). In the experiment with the smallest initial temperature and highest discount rate (left panel in **Figure 5**), scaled FERL learned almost perfect Q-values for the optimal actions. For the non-optimal actions, the average Q-values differed significantly from the optimal Q-values and for several states the learned values were in the wrong order. This is explained by the fast convergence of the learning. The average number of steps to goal converged close to the optimal number of steps of 8 after about 5000 episodes and to exactly 8 steps after about 25,000 episodes. After the initial learning phase, there was almost no exploration to improve the estimates of the Q-values of the non-optimal actions, only exploitation of the already learned optimal actions. In the experiments with larger initial temperatures and lower discount rates (middle and right panels in **Figure 5**), scaled FERL not only learned estimates of the Q-values for the optimal actions, but of the full action-value function. In both experiments, there were clear separations between the average Q-values for all actions in all states. At the end of learning, there was still considerable exploration of the environment, even if the greedy actions were equal to the optimal actions for all, or almost all, state images. The average number of steps to the goal were 10.2 steps ($\tau_0 = 1$ and $\tau_k = 0.001$) and 19.8 steps ($\tau_0 = 2$ and $\tau_k = 0.0005$). The results showed a trade-off between fast learning convergence, which required fast decay of the temperature, and learning of the full action-value function, which required slower decay of the temperature and much longer learning time.

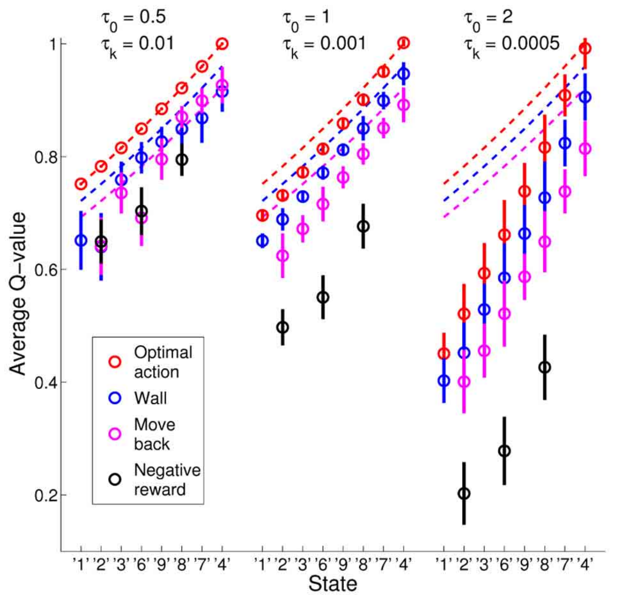


FIGURE 5 | Average Q-values (circles) with standard deviation (bars) computed over all 20 simulation runs using scaled FERL in the experiments with $\tau_0 = 0.01$ and $\tau_k = 0.01$ (left panel), $\tau_0 = 1$ and $\tau_k = 0.001$ (middle panel), and $\tau_0 = 2$ and $\tau_k = 0.0005$ (right panel). The figure shows the average learned Q-values, along the optimal path from the initial state "1" to the goal state "5," of the four types of actions: (1) optimal actions (red); (2) actions that moved the agent into a wall (blue); (3) actions that moved the agent away from the goal (purple); and (4) negative rewarded actions (black).

FERL and NNRL (middle and right panels in **Figure 4**) required careful tuning of both τ_0 and τ_k to converge to average reward values close to the maximum reward of 1 within the learning time. FERL achieved this for only two settings of τ_0 and τ_k (dashed green and solid blue lines in **Figure 4**) and NNRL achieved this for three settings (dashed green, dotted green, and dashed blue lines in **Figure 4**). The low average learning performance for many settings of τ_0 and τ_k was caused by that the learning completely failed in some simulation runs. The agent either moved prematurely to the goal state (−0.01 reward), or the agent remained in the gridworld until the maximum number of steps (1000) had passed. In general, NNRL learned faster and had a higher rate of successful learning, compared with FERL. For NNRL, the highest rate of successful learning was 100% of the simulations runs ($\tau_0 = 1$ and $\tau_k = 0.001$) and the average success rate, computed over all nine settings of τ_0 and τ_k, was 76%. For FERL, the highest success rate was 70% ($\tau_0 = 1$ and $\tau_k = 0.001$) and the average success rate was only 30%.

To try to explain the difference in performance between the scaled version of FERL and standard FERL, we looked at the patterns of activation in the hidden nodes. **Figure 6** shows typical hidden activation patterns after successful learning, for all 20 digit images for all states. The displayed activation patterns are grouped according to state and optimal action, i.e., South for states "1," "2," and "4," East for states "3" and "6," North for states "8" and "9," and West for state "7." The difference in hidden activation patterns between the two methods is quite remarkable. FERL learned a very sparse and strong action-coding with minimal separation between images of the same digit and between states with the same optimal action. The action-coding was achieved with a few active hidden nodes and the majority of the nodes were silent for all state inputs. In the hidden activation pattern shown in **Figure 6**, the action-coding was achieved using almost only hidden node 17. The actions were separated by differences in the node's activation level: 0.16 ± 0.04 for South, 0.44 ± 0.06

for East, 0.92 ± 0.03 for North, and 0.65 ± 0.06 for West. In contrast, the coding learned by scaled FERL was much more complex with no silent hidden nodes. The pattern of hidden activation did not only separate states according to optimal action, there was also clear differentiation between states and even individual state images.

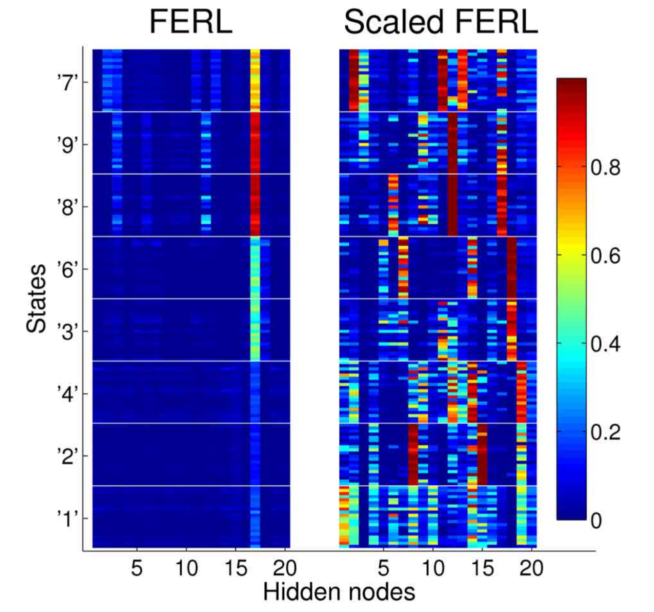


FIGURE 6 | Typical hidden activation patterns after successful learning for FERL (left panel) and scaled FERL (right panel). The figures show the activation of all 20 hidden nodes for all 20 digit images for all states.

3.2. ROBOT VISUAL NAVIGATION TASK

The result of robot navigation task is summarized in **Figure 7**. The left panel shows the average number of steps to goal, computed over every 100 episodes and 10 simulation runs for each experiment. The right panel shows the average number of steps to goal with standard deviation in the final 100 episodes. Scaled FERL converged to similar average number of time steps to goal, with low variance, in all simulation runs in each of the four experiments. The learning converged faster and the final learning performance was significantly better ($p < 0.001$) in all four experiments. The only exception was NNRL in the one goal experiment, which performed very similar to scaled FERL, both with respect to convergence speed and final learning performance. For experiments with 2 and 3 goals, NNRL performed almost as well as scaled FERL. The learning performance decreased significantly in the experiment with four goals. NNRL failed to learn to navigate to the goal for at least one starting area and one goal area in 7 (out of 10) simulation runs. The final learning performance of FERL was reasonably good in the experiments with one and two goals. The learning only failed in one simulation run, in the experiment with two goals. However, the convergence speed was slow compared to the other two methods. In the experiments with 3 and 4 goals, the learning performance decreased significantly and the learning failed in 4 and 5 simulation runs, respectively.

To try to explain the difference in learning performance between standard FERL and scaled FERL, we looked at learned trajectories and the corresponding hidden activation patterns. **Figure 8** shows typical trajectories learned by FERL (left panel) and scaled FERL (right panel) for navigating to the northeast (NE) goal and the southwest (SW) goal in the experiment with two goals, starting from the center of the south starting area and the north starting area, respectively, and facing the outer wall.

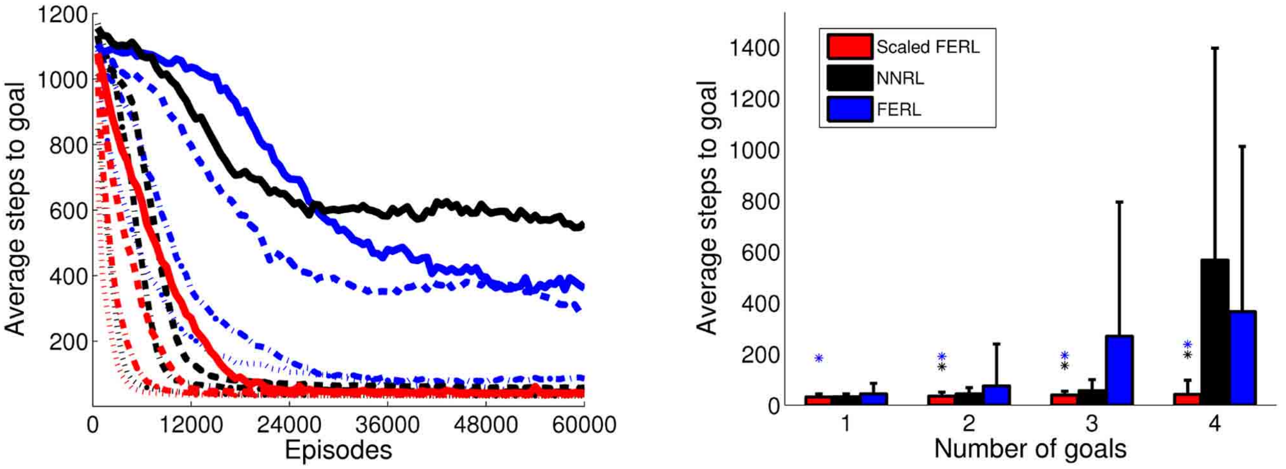


FIGURE 7 | The average number of time steps to goal for the whole learning process (left panel) and in the final 100 episodes (right panel), for the four experiments with 1, 2, 3, and 4 goal areas. The average values were computed over every 100 episodes and 10 simulation runs in each experiment. In the left panel, the line type indicates the number of goals: dotted lines for 1 goal, dash-dotted lines for 2 goals, dashed lines for 3 goals, and solid lines for 4 goals. The colored asterisks in the right panel indicate experiments in which the final average performance of scaled FERL was significantly better ($p < 0.001$) than NNRL (black) or FERL (blue).

The color coding indicates the selected actions: red for rotate right (action 1), green for curve right (action 2), blue for go straight (action 3), yellow for curve left (action 4), and cyan for rotate left (action 5). **Figure 9** shows the activation of the hidden nodes and the selected actions along the learned trajectories.

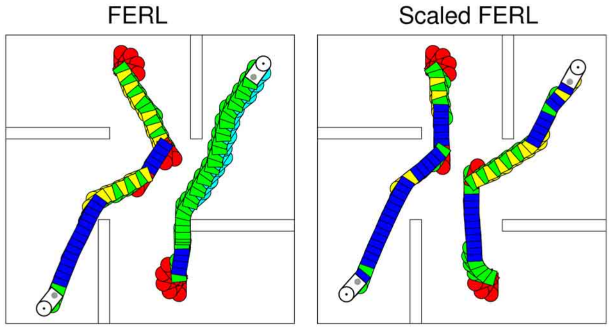


FIGURE 8 | Typical trajectories learned by FERL (left panel) and scaled FERL (right panel) for navigating to the northeast goal and the southwest goal in the experiment with two goals. The color coding indicates the selected actions: red for rotate right (action 1), green for curve right (action 2), blue for go straight (action 3), yellow for curve left (action 4), and cyan for rotate left (action 5).

The learned policies and the hidden activation patterns were very different between the two methods. FERL learned a policy which selected separate combinations of actions for navigation to different goal areas. In the example shown in the left panel in **Figure 8**, the robot only executed the curve right and the rotate left actions to reach the NE goal, after the initial part of the trajectory. To reach the SW goal, the robot executed either the curve right and the curve left actions to pass obstacles, or the go straight action to move toward the goal and the rotate right action for course corrections. FERL learned, as in the gridworld task, a very sparse and strong action-coding with little separation between states corresponding to the same action (left panels in **Figure 9**). Each action corresponded to the activation of one or few hidden nodes, e.g., hidden node 6 coded action 5 (rotate left) and hidden node 7 coded action 4 (curve left). Scaled FERL learned a policy which selected similar actions in corresponding positions along the trajectories to different goals, as shown in the right panel in **Figure 8**. In contrast to FERL, there was clear differentiation in the hidden activation patterns for different states (right panels in **Figure 9**).

4. DISCUSSION

In this study, we proposed a scaled version of FERL, where the action-value function is approximated as the negative free-energy of a restricted Boltzmann machine, divided by a constant scaling

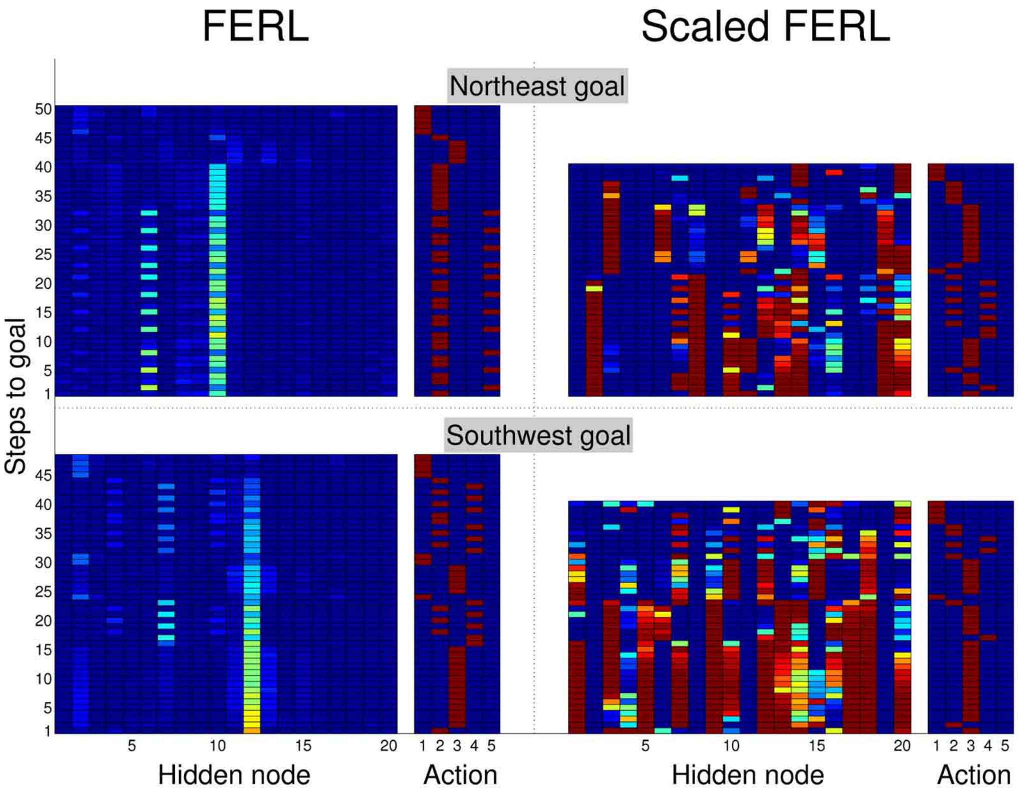


FIGURE 9 | Hidden activation patterns and selected actions for the states along the trajectories to the northeast goal (top panels) and southwest goal (bottom panels) shown in Figure 8, for FERL (left panels) and scaled FERL (right panels).

factor. The scaling factor was set to the square root of the number of state nodes. To validate our proposed method, we compared the learning performance with standard FERL and with NNRL (function approximation using a two-layered feedforward neural network), for a digit floor gridworld task and a robot visual navigation task. The learning with scaled FERL performed significantly better than the other two methods for both tasks. In the gridworld task, we also compared the robustness with respect to different exploration schedules (i.e., settings of initial temperature and temperature discount rate in softmax action selection). The learning with scaled FERL was very robust and the results showed a trade-off between fast learning convergence, which required fast decay of the temperature, and learning of the full action-value function, which required slower decay of the temperature and much longer learning time. In contrast, the learning with FERL and NNRL could only converge to average reward values close to the maximum reward for a narrow range of initial temperatures and discount rates. Analysis of activation patterns in the hidden nodes showed big differences between FERL and scaled FERL. FERL learned a very sparse action-coding with little separation between different states corresponding to the same action. In contrast, scaled FERL learned a much richer neural encoding with no silent hidden nodes and clear separation between different states corresponding to the same action.

Although quite arbitrary, the setting of the scaling factor to the square root of the number of state nodes worked very well for the tasks considered in this study. One reason was probably that we used the same number of hidden nodes (20) in all experiments. A more general setting of the scaling factor should probably also include the number of hidden nodes, because the magnitude of the initial negative free-energy increases with the number of hidden nodes of the Boltzmann machine. For example, in the gridworld task, the magnitude of the initial negative free-energy is about 16 with 20 hidden nodes, about 80 with 100 hidden nodes, and about 160 with 200 hidden nodes. An alternative approach would be to include the scaling factor as a parameter of the function approximator. The scaling factor, Z, would then be updated according to $\nabla_Z Q_t = F_t/Z^2$. We plan to investigate the setting of the scaling factor more thoroughly in future work.

The introduction of the scaling factor can ensure that the Q-values are initialized within a more appropriate range, e.g., between zero and one in the episodic delayed reward tasks with a goal reward of $+1$ considered in this study. This could partly explain why the learning with scaled FERL was more stable than learning with FERL. However, it does not explain the much faster convergence speed of scaled FERL and the remarkable difference in activation patterns of the hidden nodes. These issues will also be explored in future work.

In our earlier research, we have developed methods such as multiple model-based reinforcement learning (MMRL) (Doya et al., 2002) and competitive-cooperative-concurrent reinforcement learning with importance sampling (CLIS) (Uchibe and Doya, 2004) to improve the learning performance and the learning speed of reinforcement learning. FERL and such methods are complementary and suitable for different types of learning tasks. Restricted Boltzmann machines are global function approximators. They grow linearly with number of nodes and they are, therefore, well suited for tasks with very high-dimensional binary state inputs, such as binarized images. FERL offers few, if any, benefits in tasks with low-dimensional state spaces and real-valued state input. MMRL has proven to work well for low-dimensional non-linear control problems, but would, in our opinion, not scale well to tasks with very high-dimensional state input. In addition, MMRL requires a continuous reward function, because each module learns its policy in separate parts of the state space and there is no sharing of values between modules. In the two task in this study, it would therefore be impossible for a module to learn a policy for a part of the trajectory to the goal, since the reward is zero for all state transitions except transitions to the absorbing goal state. CLIS was developed for tasks with real-valued state input. CLIS selects an appropriate policy out of a set of heterogeneous modules with different levels of resolution in the state representation (i.e., simpler modules with coarse discretization of the state input and more complex modules with fine discretization of the state input). The CLIS framework, therefore, offers no benefit for tasks with binary state inputs. A common alternative approach to use an advanced function approximator, such as FERL, is to use a hybrid approach with a separate state abstraction module combined with a simple reinforcement learning algorithm. In our experience, a hybrid approach makes concurrent learning difficult, because it in most cases requires pre-training of the state abstraction module to achieve efficient learning. The experimental results in this study show that scaled FERL can achieve both fast learning convergence (with appropriate settings of τ_0 and τ_k) and generalization of the state space in the neural encoding in the hidden layer.

In this study, we used a machine learning approach to visual navigation in neurorobotics, where the neural encoding is an emergent property of the function approximation used in the learning algorithm. An alternative approach is to use biologically-inspired computational modeling of the brain circuits involved in navigation in real animals (Arleo and Gerstner, 2000; Krichmar et al., 2005; Fleischer et al., 2007; Barrera and Weitzenfeld, 2008; Giovannangeli and Gaussier, 2008; Milford and Wyeth, 2010; Caluwaerts et al., 2012). Currently, the two approaches are mostly complementary. In the former approach, the main focus is to develop efficient and robust learning algorithms that works well for a wide variety of learning tasks. In the latter approach, the main focus is to increase our understanding of the underlying brain mechanisms of animal behavior. The most important test is whether the robot's behavior and the activity of the simulated nervous system match empirical data from experiments with real animals. A natural long-term goal of neurorobotics would be to merge the two approaches to achieve both efficient learning and biologically plausible neural encoding.

ACKNOWLEDGMENTS

We would like to thank Makoto Otsuka for helpful discussions. We would also like to thank the two reviewers for their valuable comments and suggestions to improve the quality of the paper.

REFERENCES

Arleo, A., and Gerstner, W. (2000). Spatial cognition and neuromimetic navigation: a model of hippocampal place cell activity. *Biol. Cybern.* 83, 287–299.

Barrera, A., and Weitzenfeld, A. (2008). Biologically-inspired robot spatial cognition based on rat neurophysiological studies. *Auton. Robots* 25, 147–169.

Caluwaerts, K., Staffa, M., N'Guyen, S., Grand, C., Dollé, L., Favre-Félix, A., et al. (2012). A biologically inspired meta-control navigation system for the psikharpax rat robot. *Bioinspir. Biomim.* 7, 1–29.

Doya, K., Samejima, K., Katagiri, K., and Kawato, M. (2002). Multiple model-based reinforcement learning. *Neural Comput.* 14, 1347–1369.

Doya, K., and Uchibe, E. (2005). The cyber rodent project: exploration of adaptive mechanisms for self-preservation and self-reproduction. *Adapt. Behav.* 13, 149–160.

Elfwing, S., Otsuka, M., Uchibe, E., and Doya, K. (2010). "Free-energy based reinforcement learning for vision-based navigation with high-dimensional sensory inputs," in *Proceedings of the International Conference on Neural Information Processing (ICONIP2010)* (Berlin, Heidelberg), 215–222.

Fleischer, J. G., Gally, J. A., Edelman, G. M., and Krichmar, J. L. (2007). Retrospective and prospective responses arising in a modeled hippocampus during maze navigation by a brain-based device. *Proc. Natl. Acad. Sci. U.S.A.* 104, 3556–3561.

Freund, Y., and Haussler, D. (1992). "Unsupervised learning of distributions on binary vectors using two layer networks," in *Advances in Neural Information Processing Systems 4*, eds J. E. Moody, S. J. Hanson, and R. P. Lippmann (Denver, CO: Morgan Kaufmann), 912–919.

Giovannangeli, C., and Gaussier, P. (2008). "Autonomous vision-based navigation: goal-oriented action planning by transient states prediction, cognitive map building, and sensory-motor learning," in *Proceedings of the International Conference on Intelligent Robots and Systems (IROS2008)* (Nice, France), 676–683.

Hinton, G. E. (2002). Training products of experts by minimizing contrastive divergence. *Neural Comput.* 12, 1771–1800.

Krichmar, J. L., Seth, A. K., Nitz, D. A., Fleischer, J., and Edelman, G. M. (2005). Spatial navigation and causal analysis in a brain-based device modeling cortical-hippocampal interactions. *Neuroinformatics* 3, 197–221.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proc. IEEE* 86, 2278–2324.

MATLAB. (2010). *version 7.14.0 (R2012a).* Natick, MA: The MathWorks Inc.

Milford, M., and Wyeth, G. (2010). Persistent navigation and mapping using a biologically inspired slam system. *Int. J. Rob. Res.* 29, 1131–1153.

Otsuka, M., Yoshimoto, J., and Doya, K. (2010). "Free-energy-based reinforcement learning in a partially observable environments," in *Proceedings of the European Symposium on Artificial Neural Networks (ESANN2010)* (Bruges, Belgium), 541–545.

Rummery, G. A., and Niranjan, M. (1994). *On-line Q-learning Using Connectionist Systems.* Technical Report CUED/F-INFENG/TR 166, Cambridge University Engineering Department.

Sallans, B., and Hinton, G. E. (2004). Reinforcement learning with factored states and actions. *J. Mach. Learn. Res.* 5, 1063–1088.

Smolensky, P. (1986). "Information processing in dynamical systems: foundations of harmony theory," in *Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Volume 1: Foundations*, eds D. E. Rumelhart and J. L. McClelland (Cambridge, MA: MIT Press), 194–281.

Sutton, R. S. (1996). "Generalization in reinforcement learning: successful examples using sparse coarse coding," in *Advances in Neural Information Processing Systems 8*, eds D. S. Touretzky, M. C. Mozer, and M. E. Hasselmo (Denver, CO: MIT Press), 1038–1044.

Sutton, R. S., and Barto, A. (1998). *Reinforcement Learning: An Introduction*. Cambridge, MA: MIT Press.

Uchibe, E., and Doya, K. (2004). "Competitive-cooperative-concurrent reinforcement learning with importance sampling," in *Proceedings of the International Conference on Simulation of Adaptive Behavior: From Animals and Animats (SAB2004)* (Santa Monica, CA: MIT Press), 287–296.

Conflict of Interest Statement: 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.

Received: 01 December 2012; accepted: 12 February 2013; published online: 28 February 2013.

Citation: Elfwing S, Uchibe E and Doya K (2013) Scaled free-energy based reinforcement learning for robust and efficient learning in high-dimensional state spaces. Front. Neurorobot. ***7****:3. doi:* ***10.3389/fnbot.2013.00003***

frontiers in
NEURОROBOTICS

HYPOTHESIS AND THEORY ARTICLE
published: 02 November 2012
doi: 10.3389/fnbot.2012.00011

What is value—accumulated reward or evidence?

***Karl Friston*[1]*, *Rick Adams*[1] *and Read Montague*[1,2]**

[1] Wellcome Trust Centre for Neuroimaging, University College London, London, UK
[2] Department of Physics, Virginia Tech Carilion Research Institute, Virginia Tech, Roanoke, VA, USA

Edited by:
Florian Röhrbein, Technische Universität München, Germany

Reviewed by:
Jun Tani, RIKEN Brain Science Institute, Japan
Subramanian Ramamoorthy, The University of Edinburgh, UK

***Correspondence:**
Karl Friston, Wellcome Trust Centre for Neuroimaging, Institute of Neurology, 12 Queen Square, London, WC1N 3BG, UK.
e-mail: k.friston@ucl.ac.uk

Why are you reading this abstract? In some sense, your answer will cast the exercise as valuable—but what is value? In what follows, we suggest that *value is evidence* or, more exactly, log Bayesian evidence. This implies that a sufficient explanation for valuable behavior is the accumulation of evidence for internal models of our world. This contrasts with normative models of optimal control and reinforcement learning, which assume the existence of a value function that explains behavior, where (somewhat tautologically) behavior maximizes value. In this paper, we consider an alternative formulation—active inference—that replaces policies in normative models with prior beliefs about the (future) states agents should occupy. This enables optimal behavior to be cast purely in terms of inference: where agents sample their sensorium to maximize the evidence for their generative model of hidden states in the world, and minimize their uncertainty about those states. Crucially, this formulation resolves the tautology inherent in normative models and allows one to consider how prior beliefs are themselves optimized in a hierarchical setting. We illustrate these points by showing that any optimal policy can be specified with prior beliefs in the context of Bayesian inference. We then show how these prior beliefs are themselves prescribed by an imperative to minimize uncertainty. This formulation explains the saccadic eye movements required to read this text and defines the value of the visual sensations you are soliciting.

Keywords: free energy, active inference, value, evidence, surprise, self-organization, selection, Bayesian

INTRODUCTION

So, why are you reading this paper? According to what follows, the answer is fairly simple: you are compelled to selectively sample sensory input that conforms to your predictions and—*a priori*—you believe that reading this text will reduce your uncertainty about what we are going to say (you are going to see) next. This may sound a rather trite explanation but it contains two fundamental premises. Both of these premises can be motivated from the basic principles of self-organization: namely, the imperative to minimize surprise (maximize evidence) associated with sensory states—by actively sampling the environment—and the imperative to minimize uncertainty about the inferred causes of that input—by making inferences about future or fictive states. Together, these provide a complete account of optimal behavior, in which value becomes log-evidence or negative surprise. This paper tries to unpack these assertions using formal arguments and simulations. In fact, the final simulation reproduces a simple form of reading, in which an agent garners evidence for its beliefs using saccadic eye movements (Rayner, 1978).

Implicit in this account of optimal behavior is a hierarchical perspective on optimization, in which behavior is cast as active Bayesian inference that is constrained by prior beliefs. Crucially, these prior beliefs are themselves optimized at a higher hierarchal level. This is important because it resolves the tautology inherent in normative schemes based upon optimal control theory and cost or reward functions. The tautology here is almost self-evident: if behavior is optimal, then it maximizes value. But what is value—other than an objective function that describes optimal behavior. It is this descriptive (circular) aspect of conventional formulations we associate with normative schemes. Put simply, adopting a normative model subverts questions about the origin and optimization of value functions *per se*. For example, it would be difficult to specify a reward or value function that explains why you are reading this text.

In the context of active inference, this issue is resolved by appeal to hierarchical Bayesian inference, in which optimization at one level is constrained by *empirical* priors from a higher level. Optimization in this setting refers to maximizing Bayesian model evidence (or minimizing surprise). In most real-world examples—for example the Bayesian brain (Yuille and Kersten, 2006)—a hierarchical aspect to inference emerges naturally from a separation of temporal scales. For example, inference about the causes of some data is constrained by the parameters of a generative model that are learned after all the data have been seen. Similarly, the form of the model itself can be optimized through model selection, after the parameters of competing models have been optimized. Neurobiologically, these optimization or inference processes may be associated with synaptic activity, synaptic plasticity and synaptic regression—each operating at successively slower timescales. Although the optimization processes may differ (e.g., neuronal dynamics, associative learning, and neurodevelopment), they are all fulfilling the same objective; namely, to maximize the Bayesian model evidence averaged over time. Clearly, one can develop this hierarchical perspective to an evolutionary level, where natural selection may play the role of Bayesian model selection. In short, contextualizing optimization processes at different temporal scales allows one to examine the process theories (putative implementation) at each level and consider them in relation to the level above. We will see an example of this later, in terms of empirical prior beliefs that are updated slowly after fast eye movements. Furthermore, formulating optimal behavior in terms of active inference means that one can associate value in normative schemes with probabilistic attributes of sensory states. This is important because it provides a link between normative models of optimal control and normative models based upon information theory (Barlow, 1961; Linsker, 1990; Bialek et al., 2001; Zetzsche and Röhrbein, 2001)—such as the principle of least action, the principle of maximum entropy, the principle of minimum redundancy and the principle of maximum information transfer. This link rests on replacing reward or cost functions in optimal control theory with prior beliefs in the context of Bayes-optimal inference.

OVERVIEW

This paper comprises six sections. The first three focus on conventional optimal control and reinforcement learning schemes and their formulation in terms of active inference. In particular, they show how cost functions can be replaced by prior beliefs under active inference. These sections use discrete time formulations and summarises the material in Friston et al. (2012b). The final three sections consider where prior beliefs come and move from the abstract formulations of normative models to biophysically realistic formulations. These sections use continuous time and summarises the material in Friston et al. (2012a).

The first section reviews the role of cost and value functions in Markov decision processes (MDPs) and their extensions to partially observable Markov decision processes (POMDPs). We then revisit these formulations from the point of view of active inference and demonstrate their formal relationships. In brief, active inference separates *inference* about hidden states causing observations from *action*. The motivation for this is pragmatic; in that real agents cannot know how their action affects hidden states (because hidden states have to be inferred). This means that action must be based on a function of observed states, as opposed to hidden states. Active inference assumes that this function is the same variational free energy used in approximate Bayesian inference (Hinton and van Camp, 1993; Dayan et al., 1995; MacKay, 1995; Neal and Hinton, 1998). In other words, active inference extends the minimization of variational free energy that underlies approximate Bayesian inference to *include action* (Friston et al., 2010b). However, requiring action to minimize variational free energy appears to contradict optimal control theory, which requires action to minimize expected cost. The purpose of the second section is to resolve this conflict. We will see that the cost functions that are used to guide action in optimal control can be absorbed into prior beliefs in active inference. Effectively, this means that agents expect their state transitions to minimize cost, while action realizes these prior beliefs by maximizing the marginal likelihood of observations. This means one can use standard Bayesian inference schemes to solve optimal control problems—see also McKinstry et al. (2006). The third section illustrates this by showing how optimal policies can be inferred under prior beliefs about future (terminal) states using standard variational Bayesian procedures (Beal, 2003). This section concludes with an example (the mountain car problem) that illustrates how active inference furnishes online nonlinear optimal control, with partially observed (hidden) states.

The fourth section turns to the nature and origin of prior beliefs and shows how they can be derived from the basic imperatives of self-organization (Ashby, 1947; Tschacher and Haken, 2007). This section uses a general but rather abstract formulation of agents—in terms of the states they can occupy—that enables us to explain action, perception and control as corollaries of variational free energy minimization. The focus here is on prior beliefs about control and how they relate to the principle of maximum mutual information and specific treatments of visual attention such as Bayesian surprise (Itti and Baldi, 2009). Having established the underlying theory, the fifth section considers neurobiological implementations in terms of predictive coding and recurrent message passing in the brain. This section reprises a neural architecture we have described in previous publications and extends it to include the encoding of prior beliefs in terms of (place coded) saliency maps. The final section provides an illustration of the basic ideas, using neuronally plausible simulations of visual search and the control of saccadic eye movements. This illustration allows us to understand Bayes-optimal searches in terms of the accumulation of evidence during perceptual synthesis.

MARKOVIAN FORMULATIONS OF VALUE AND OPTIMAL CONTROL

In the following sections, we apply variational free energy minimization to a well-studied problem in optimal decision theory, psychology and machine learning; namely MDPs. In brief, we show that free energy minimization (active inference) and optimal decision theory provide the same solutions when the *policies* from optimal decision theory are replaced by *prior beliefs* about transitions from one state to another. This is important because specifying behavior in terms of prior beliefs finesses the difficult problem of optimizing behavior to access distal rewards. Furthermore, it enables one to consider optimality in terms of accessing particular states in the future. Bayes-optimal behavior then depends upon a representation of future behaviors that necessarily entails a model of agency.

This section considers discrete time (Markov) decision processes of the sort found in optimal control theory, models of behavior and decision making (Bellman, 1952; Watkins and Dayan, 1992; Camerer, 2003; Daw and Doya, 2006; Todorov, 2006; Dayan and Daw, 2008). Its aim is to establish a link between classical approaches to optimizing decisions, in terms of policy optimization, and the variational free energy minimization that underlies active inference (Beal, 2003; Friston et al., 2009). Here, classical schemes are taken to imply that actions (and beliefs about hidden states of the world) are chosen to maximize the expected reward of *future states*. Conversely, in active inference, actions and beliefs minimize a variational free energy bound on the (negative log) marginal likelihood of *observed states*—that is, they maximize the marginal likelihood or Bayesian model evidence. Linking the two formulations necessarily requires us to formulate free energy minimization in discrete time and think about how reward or cost functions are accommodated.

The key distinction between optimal control and active inference is that in optimal control, action optimizes the expected cost associated with the hidden states a system or agent visits. In contrast, active inference requires action to optimize the marginal likelihood (Bayesian model evidence) of observed states, under a generative model. This introduces a distinction between cost-based optimal control and Bayes-optimal control that eschews cost. The two approaches are easily reconciled by ensuring the generative model embodies prior beliefs about state transitions that minimize expected cost. Our purpose is therefore not to propose an alternative implementation of optimal control but accommodate optimal control within the larger framework of active inference.

MARKOV DECISION PROCESSES

First, we briefly consider Markov decision problems and their solutions based upon cost or reward functions that are an integral part of optimal control theory and reinforcement learning.

Notation and set up: We will use X for a finite set of states and $x \in X$ for particular values. A probability distribution will be denoted by $P(x) = \Pr\{X = x\}$ using the usual conventions. The tilde notation $\tilde{x} = (x_0, \ldots, x_T)$ denotes a sequence of values at time points $t = 0, \ldots, T$.

Definition: A Markov decision process is the tuple (X, A, T, r), where

- *Hidden states* X—a finite set of states.
- *Action* A—a finite set of actions.
- *Transition probability* $T(x'|x, a) = \Pr(\{x_{t+1} = x'|x_t = x, a_t = a\})$—the probability that the state $x' \in X$ at time $t + 1$ follows action $a \in A$ in state $x \in X$ at time t.
- *Reward* $r(x) \in \mathbb{R}$—some reward received at state $x' \in X$.

Problem: The goal is to find a *policy* $\pi : X \to A$ that maximizes cumulative rewards. This can be expressed in terms of the sequence of actions $\tilde{a} := (a_0, \ldots, a_T)$ that maximizes *value* or negative *cost-to-go*:

$$V(x) = \max_{\tilde{a}} \left\{ r(x) + \sum_{i=1}^{T} \sum_{x'} \Pr(\{x_i = x'|x_0 = x, a_0, \ldots, a_i\}) r(x') \right\} \tag{1}$$

The solution to this equation is a policy or sequence of optimal actions $a_t := \pi(x_t)$ that maximizes expected reward in the future, given a probabilistic model of state transitions. In this setting, (T, r) constitutes a model that comprises a transition matrix and rewards defined on states. Equation (1) can be expressed as the *Bellman optimality equation* by exploiting the Markovian nature of the problem using recursive substitution (Bellman, 1952):

$$V(x) = \max_{a} \left\{ r(x) + \sum_{s'} T(x'|x, a) V(x') \right\} \tag{2}$$

For simplicity, we have assumed a *finite horizon* problem, in which the reward is maximized from $t = 0$ to $t = T$. This allows us to eschew notions of discounting required in infinite horizon problems. Solutions to MDPs can be divided into *reinforcement learning* schemes that compute the value function explicitly and *direct policy searches* that find the optimal policy directly.

In direct policy searches (Williams, 1992; Baxter et al., 2001; Gomez and Miikkulainen, 2001), a policy is optimized by mapping each state directly to an action, without reference to the value of the state. Direct policy searches are useful when the value function is hard to learn but the policy is easy to find. In reinforcement learning there are two general approaches: The first *model based* schemes compute the value function using a model of state transitions and is usually considered when the state space is sufficiently small. This is also known as *dynamic programming* and involves iterating the following two steps (Bellman, 1952):

$$\begin{aligned} \pi(x) &= \arg\max_{a} \left\{ r(x) + \sum_{s'} T(x'|x, a) V(x') \right\} \\ V(x) &= r(x) + \sum_{s'} T(x'|x, \pi(x)) V(x') \end{aligned} \tag{3}$$

This scheme is guaranteed to find the optimal solution, provided all states are visited. In *value iteration* or *backwards induction*, the policy is only calculated when needed. This gives the combined step in (1). In *policy iteration* (Howard, 1960), the first step is repeated until convergence, thereby providing a definite stopping condition. If the transition probabilities or rewards are unknown or the state space is large (precluding a visit to every state), the problem is usually solved with *model free* reinforcement learning. In these schemes the value function is itself learnt (Rescorla and Wagner, 1972; Sutton and Barto, 1981; Watkins and Dayan, 1992; Friston et al., 1994): This enables one to solve Markov decision problems without learning the transition probabilities, because the value function acts as a guidance function for action.

PARTIALLY OBSERVABLE MARKOV DECISION PROCESSES

The formulation above assumes that the agent knows what state it is in. This is often unrealistic because an agent cannot know the exact state of the world, given noisy or partial observations (Rao, 2010). This leads to an extension of the MDP framework to accommodate partially observed states (Kaelbling et al., 1998); namely a POMDP. Although it is possible to solve POMDPs using direct policy searches (Gomez et al., 2009), one cannot perform value iteration or reinforcement learning directly, as they require the hidden states. However, a POMDP can be converted to a MDP using beliefs about the current state that can be computed recursively from the observations and actions using Bayes rule. This enables one to convert the partially observed process to a (Belief) MDP by treating the beliefs as states and replacing reward with its expected value under the current belief state.

In summary, conventional approaches to MDPs rest on the optimization of future rewards and specify an optimal policy in terms of an action from any given state. Partially observed MDPs make inference explicit by introducing a probabilistic mapping between hidden states of the world and observations. In this setting, the beliefs that the agent forms (by observing histories of actions and states) can be exploited to optimize behavior.

OPTIMAL CONTROL AS INFERENCE

Our focus is on optimal decision making or control as an inference process: see Filatov and Unbehauen (2004) for a review of early work in this area. Initial approaches were based on converting control problems into inference problems—by replacing reward with an auxiliary random variable conditioned on desired observations. This makes maximizing reward equivalent to maximizing the likelihood of desired observations (Cooper, 1988; Shachter, 1988). Subsequent work focused on efficient methods to solve the ensuing inference problem (Jensen et al., 1994; Zhang, 1998). Later, Dayan and Hinton (1997) proposed an Expectation Maximization algorithm for reinforcement learning with immediate rewards, while Toussaint and Storkey (2006) cast the problem of computing optimal policies as a likelihood maximization problem. This generalized the work of Cooper (1988) and Shachter (1988) to the case of infinite horizons and cost functions over future states. More recently, this approach has been pursued by applying Bayesian procedures to problems of optimal decision making in MDPs (Botvinick and An, 2008; Toussaint et al., 2008; Hoffman et al., 2009).

Related work on stochastic optimal control (Kappen, 2005a,b; van den Broek et al., 2008; Rawlik et al., 2010) exploits the reduction of control problems to inference problems by appealing to variational techniques to provide efficient and computationally tractable solutions. In particular, formulating the problem in terms of Kullback–Leibler minimization (Kappen, 2005a,b) and path integrals of cost functions (Theodorou et al., 2010; Braun et al., 2011).

The variational formalism has also found a powerful application in the setting of optimal control and the construction of adaptive agents. For example, Ortega and Braun (2010), consider the problem of optimizing active agents, where past actions need to be treated as causal interventions. They show that that the solution to this variational problem is given by a stochastic controller called the Bayesian control rule, which implements adaptive behavior as a mixture of experts. This work illustrates the close connections between minimizing (relative) entropy and the ensuing active Bayesian inference that we will appeal to the later.

SUMMARY

In summary, current approaches to partially observed MDPs and stochastic optimal control minimize cumulative cost using the same procedures employed by maximum likelihood and approximate Bayesian inference schemes. Indeed, the formal equivalence between optimal control and estimation was acknowledged by Kalman at the inception of Bayesian filtering schemes (Todorov, 2008). In the next section, we revisit this equivalence and show that any optimal control problem can be formulated as a Bayesian inference problem, within the active inference framework. The key aspect of this formulation is that action does not minimize cumulative cost but maximizes the marginal likelihood of observations, under a generative model that entails an optimal policy.

ACTIVE INFERENCE

This section introduces active inference, in which the optimization of action and beliefs about hidden states are treated as two separate processes that both maximize Bayesian model evidence or the marginal likelihood of observations. In active inference, action elicits *observations* that are the most plausible under beliefs about (future) states. This is in contrast to conventional formulations, in which actions are chosen to elicit (valuable) states. We will see that active inference can implement any optimal policy; however, it does not solve the optimal control problem explicitly, because active inference does not minimize cost-to-go but minimizes the surprise of observations (maximizes their marginal likelihood). This follows from the fact that active inference is a corollary of the free energy principle:

THE FREE-ENERGY PRINCIPLE

The free-energy principle (Friston et al., 2006) tries to explain how agents occupy a small number of attracting states by minimizing the Shannon entropy of the probability distribution over their sensory states. Under ergodic assumptions, this entropy is (almost surely) the long-term time average of self-information or surprise (Birkhoff, 1931). Surprise, or more precisely *surprisal*, is a (probability) measure $-\ln P(s_t|m)$ on the states that are sampled by an agent.

Minimizing the long-term average $E_t[-\ln P(s_t|m)]$ is assured when agents minimize surprise at each time point. Crucially, surprise is just the negative marginal likelihood or Bayesian model evidence, which means minimizing surprise maximizes Bayesian model evidence. Surprise is minimized—approximately or exactly—if agents minimize a variational free energy bound on surprise (Feynman, 1972; Hinton and van Camp, 1993), given a generative model m of state transitions (Dayan et al., 1995; Friston, 2010). We will return to the relationship between entropy, surprise and Bayesian model evidence in Section "Bayes-optimal control without cost functions," when we examine the motivation for free energy minimization in more detail. Here, we consider the nature of active inference in terms of free energy minimization, where free energy is defined in relation to the following definitions:

Definition: Active inference rests on the tuple $(X, A, \vartheta, P, Q, R, S)$ comprising:

- A finite set of *hidden states* X
- Real valued *hidden parameters* $\vartheta \in \mathbb{R}^d$
- A finite set of *sensory states* S
- A finite set of *actions* A
- Real valued *internal states* $\mu \in \mathbb{R}^d$ that parameterize a conditional density

- A *sampling probability* $R(s'|s, a) = \Pr(\{s_{t+1} = s'|s_t = s, a_t = a\})$ that observation $s' \in S$ at time $t+1$ follows action $a \in A$, given observation $s \in S$ at time t
- A *generative probability* $P(\tilde{s}, \tilde{x}, \theta|m) = \Pr(\{s_0, \ldots, s_t\} = \tilde{s}, \{x_0, \ldots, x_T\} = \tilde{x}, \vartheta = \theta)$ over observations to time t, states at all times and parameters
- A *conditional probability* $Q(\tilde{x}, \theta|\mu) = \Pr(\{x_0, \ldots, x_T\} = \tilde{x}, \vartheta = \theta)$ over a sequence of states and parameters, with sufficient statistics $\mu \in \mathbb{R}^d$

Remarks: Here, m denotes the form of a generative model or probability distribution over sensory and hidden states and parameters: $P_m(\tilde{s}, \tilde{x}, \theta) := P(\tilde{s}, \tilde{x}, \theta|m)$. For clarity, we will omit the conditioning on m for all but prior terms in the generative probability. The sufficient statistics of the conditional probability $Q_\mu(\tilde{x}, \theta) := Q(\tilde{x}, \theta|\mu)$ encode a probability distribution over a sequence of hidden states $\tilde{x} = \{x_0, \ldots, x_T\}$ and the parameters of the model $\theta \in \vartheta$. Crucially, the conditional probability and its sufficient statistics encode hidden states in the future and past, which themselves can change with time: for example, $\mu_k = \{\mu_0^k, \ldots, \mu_T^k\}$, where μ_t^k is the probability over hidden states at time t in the future or past, under the conditional probability at the present time k.

The probabilities above (P, Q, R) underwrite the action and perception of the agent—they correspond to its formal beliefs about the sensory consequences of action (sampling probability) and the hidden states causing observations (generative probability). Because the true states generating observations are unknown and unknowable from the point of view of the agent, they can only be inferred in terms of an approximate posterior probability (conditional probability).

There are three important distinctions between this setup and that used by MDPs. As in partially observed MDPs, there is a distinction between states and observations. However, the transition probability over hidden states no longer depends on action. In other words, the agent does not need to know the effect of its actions on the (hidden) state of the world. It is instead equipped with a probabilistic mapping between its actions and their direct sensory consequences—this is the sampling probability. This is a central tenet of active inference, which separates knowledge about the sensory consequences of action from beliefs about the causes of those consequences. In other words, the agent knows that if it moves it will sense movement (c.f. proprioception); however, beliefs about hidden states in the world causing movement have to be inferred. These hidden states may or may not include its own action: the key distinction between the *agency free* and *agency based* schemes considered below depends on whether the agent represents its own action or not.

The second distinction is that hidden states include future and past states. In other words, the agent represents a sequence or trajectory over states. This enables inference about a particular state in the future to change with time. This will become important when we consider planning and agency. Finally, there are no reward or cost functions. This reflects the fact that active inference does not call upon the notion of reward to optimize behavior—optimal behavior minimizes variational free energy, which is a functional of observations and the conditional probability distribution or its sufficient statistics. As we will see below, cost functions are replaced by priors over hidden states and transitions, such that costly states are surprising and are avoided by action.

PERCEPTION AND ACTION

The free energy principle states that the sufficient statistics of the conditional probability and action minimize free energy

$$\begin{aligned} \mu_t &= \arg\min_{\mu} F(\{s_0, \ldots, s_t\}, \mu) \\ a_t &= \arg\min_{a} \sum_{S} R(s_{t+1}|s_t, a) F(\{s_0, \ldots, s_{t+1}\}, \mu_t) \end{aligned} \tag{4}$$

This dual optimization is usually portrayed in terms of perception and action, by associating the sufficient statistics with internal states of the agent (such as neuronal activity) and associating action with the state of effectors or the motor plant. Equation (4) just says that internal states minimize the free energy of currently observed states, while action selects the next observation that, on average, has the smallest free energy. By factorizing the generative probability $P(\tilde{s}, \tilde{x}, \theta|m) = P(\tilde{s}|\tilde{x}, \theta)P(\tilde{x}, \theta|m)$ into likelihood and prior probabilities, one can express the free energy as follows:

$$\begin{aligned} F(\tilde{s}, \mu) &= E_Q[-\ln P(\tilde{s}, \tilde{x}, \theta|m)] - E_Q[-\ln Q(\tilde{x}, \theta|\mu)] \\ &= D_{KL}[Q(\tilde{x}, \theta|\mu)||P(\tilde{x}, \theta|\tilde{s})] - \ln P(\tilde{s}|m) \end{aligned} \tag{5}$$

The first equality in Equation (5) expresses free energy as a Gibbs energy (expected under the conditional distribution) minus the entropy of the conditional distribution. The second shows that free energy is an upper bound on surprise, because the first (Kullback–Leibler divergence) term is nonnegative by Gibbs inequality (Beal, 2003). This means that when free energy is minimized, the conditional distribution approximates the posterior distribution $Q(\tilde{x}, \theta|\mu) \approx P(\tilde{x}, \theta|\tilde{s})$ over hidden states and parameters. This formalizes the notion of unconscious inference in perception (Helmholtz, 1866/1962; Dayan et al., 1995; Dayan and Hinton, 1997) and, under some simplifying assumptions, corresponds to predictive coding (Rao and Ballard, 1999).

This formulation highlights the fact that action selects observable states (not hidden states) that are the least surprising or have the smallest free energy. The free energy is determined by the sufficient statistics of the conditional distribution. The optimization of these sufficient statistics or internal states—the first equality in Equation (4)—rests upon the generative model and therefore depends on prior beliefs. It is these beliefs that specify what is surprising and reproduces the optimal policies considered above. There are clearly many ways to specify the generative probability. We will consider two forms, both of which respect the Markov property of decision processes. The first reproduces the behavior under the optimal policy for Markov decision problems and can be regarded as the corresponding free energy formulation:

AN AGENCY FREE FORMULATION OF OPTIMAL POLICIES

The natural generative model for a partially observable Markov decision process can be expressed in terms of a likelihood plus priors over states and parameters, with the following forms:

$$\begin{aligned} P(\tilde{s}, \tilde{x}, \theta|m) &= P(\tilde{s}|\tilde{x}, \theta)P(\tilde{x}|\theta)P(\theta|m) \\ P(\{s_0, \ldots, s_t\}|\tilde{x}, \theta) &= P(s_0|x_0)P(s_1|x_1)\ldots P(s_t|x_t) \\ P(\tilde{x}|\theta) &= P(x_0|m)\prod_{t=0}^{T-1} P(x_{t+1}|x_t, \theta) \end{aligned} \tag{6}$$

This implies that the current observation depends only on the current hidden state (like a belief MDP), where the hidden states are a Markov process, whose transition probabilities depend upon the parameters (unlike a belief MDP). We will assume that the priors over the parameters $P(\theta|m) = \delta(\theta - \theta_\pi)$ make the priors over state transitions equivalent to the optimal policy of the previous section. In other words, we assume the priors have a point mass over values that render the transition probabilities $P(x_{t+1}|x_t, \theta_\pi) = T(x_{t+1}|x_t, \pi(x_t))$ optimal in the conventional sense. The second equality in Equation (5) shows that minimizing the free-energy, with respect to the sufficient statistics of the conditional distribution, renders it the posterior over hidden states and parameters. This means that the conditional distribution becomes the posterior distribution, where (noting that the posterior and prior over parameters are the same Dirac delta function)

$$Q(\tilde{x}, \theta|\mu_t) \approx P(\tilde{x}|\{s_0, \ldots, s_t\}, \theta)\delta(\theta - \theta_\pi) \tag{7}$$

We have used an approximate equality here because we are assuming approximate Bayesian inference. In this context, free-energy minimization with respect to action becomes, from Equations (4) and (5):

$$\begin{aligned} a_t &= \arg\min_a \sum_S R(s_{t+1}|s_t, a)F(\{s_0, \ldots, s_{t+1}\}, \mu_t) \\ &= \arg\max_a \sum_S R(s_{t+1}|s_t, a)\mathbf{E}_{Q(x_{t+1})}[\ln P(s_{t+1}|x_{t+1})] \\ Q(x_{t+1}) &\approx \sum_X P(x_{t+1}|x_t, \pi(x_t))P(x_t|\{s_0, \ldots, s_t\}) \end{aligned} \tag{8}$$

Note that the free energy of the new observation is just its improbability, expected under posterior beliefs about the hidden states that cause it—these posterior beliefs correspond to the marginal conditional distribution $Q(s_{t+1})$, over the next hidden state.

It can be seen from Equation (8) that action under active inference is exactly the same as action under the optimal policy. This is because action selects the observation that is most likely under the (approximate) posterior distribution. In turn, this is the hidden state that follows the currently inferred state, under the optimal policy. This means that active inference can be considered as a generalization of optimal control. This is because there are prior beliefs that can reproduce an optimal policy to minimize expected cost. However, there are prior beliefs that specify Bayes-optimal control that cannot be expressed as minimizing value (Friston and Ao, 2012). Put simply, although prior beliefs about a particular trajectory through state space may be the solution to an optimal control problem, there may be prior beliefs that are not. These prior beliefs are particularly relevant in robotics and the continuous time formulations considered later. In brief, any trajectory specified by a prior belief can be decomposed into divergence and curl free components (by the fundamental theorem of vector calculus or the Helmholtz decomposition). Crucially, only the curl free (irrotational) component can be specified by a value function. This is problematic because nearly every real-world movement trajectory has divergence free components; such as the rotational components of walking, reading and writing. These are relatively easy to specify and simulate using appropriate priors—see the handwriting simulations in Friston et al. (2011) or the animate behaviors in Tani (2003)—but cannot be specified in terms of a value function of states. See Friston and Ao (2012) for a technical discussion and Friston (2011) for a discussion in the setting of motor control.

SUMMARY

In summary, we have seen that is fairly straightforward to place optimal decision or Markovian control theory schemes in an active inference framework. This involves replacing optimal policies, defined by cost or reward functions, with prior beliefs about transitions among hidden states. The advantage of doing this is that we can formulate action and perception as jointly minimizing the same objective function that provides an upper bound on surprise or negative log Bayesian evidence. This enables optimal control to be cast as Bayesian inference, with a clear distinction between action and inference about partially observed or hidden states. We will see later that formulating the optimal control problem in terms of prior beliefs enables us to connect to other normative theories about perception and entertain questions about where these prior beliefs come from. For example, the prior beliefs above depend upon the parameters of the generative model (transition probabilities among hidden states) that can be learned in a Bayes-optimal sense. See Friston et al. (2009) for an example.

The fact that one can replace cost functions with priors to produce the same behavior is related to the complete class theorem (Brown, 1981). The complete class theorem states that any admissible decision rule (behavior) is Bayes-optimal for at least one pair of prior beliefs and cost function (Robert, 1992). However, this pair is not necessarily unique: in other words, the same decisions can be reproduced under different combinations of prior and cost functions. In one sense, this duality is resolved by replacing the cost functions of optimal control theory with prior beliefs about state transitions. Casting Bayes-optimal decisions in this way simply means that the agent believes it will sample state space in a way that minimizes future costs, while action fulfills these prior beliefs. In the next section, we consider what would happen if the agent inferred its own action:

BAYES-OPTIMAL CONTROL WITHOUT COST FUNCTIONS

In this section, we consider agency based optimization, in which the hidden states are extended to include hidden (control), states that model action. This is necessary, when inferring optimal state transitions, because transitions depend upon action in the future which is hidden from observation. In what follows, we focus on policies that are specified by prior beliefs about specific states that will be occupied at specific times in the future. This corresponds to a finite horizon control problem with terminal costs over states and intermediate control costs that are specified through prior beliefs about control.

AGENCY-BASED OPTIMIZATION

In what follows, we describe a scheme for POMDPs that optimizes action in relation to prior beliefs about future states. This scheme uses representations of hidden states in the future to optimize a sequence of fictive actions before they are enacted. This calls for a more sophisticated generative model—a model of agency or control. In other words, the agent must infer its future actions via Bayesian updates of posterior beliefs about the future. The heuristic benefit of introducing hidden control states is that future actions can be optimized, when choosing the best current action. The ensuing solutions are optimal in relation to prior beliefs about states that will be occupied. These are prior beliefs about the final (desired) hidden state and can be expressed in terms of the following generative model:

An agency based model: The generative probability used in this section introduces (a finite set of) hidden control states $u \in U$ and can be expressed in terms of the following likelihood and prior distributions:

$$\begin{aligned} P(\tilde{s}, \tilde{x}, \tilde{u}, \theta|m) &= P(\tilde{s}|\tilde{x}, \theta)P(\tilde{x}, \tilde{u}|\theta)P(\theta|m) \\ P(\{s_0, \ldots, s_t\}|\tilde{x}, \theta) &= P(s_0|x_0, \theta)P(s_1|x_1, \theta)\ldots P(s_t|x_t, \theta) \quad (9) \\ P(\tilde{x}, \tilde{u}|\theta) &= P(x_T|\theta)\prod_{t=1}^{T} P(x_{t-1}|x_t, u_t, \theta)P(u_t|\theta) \end{aligned}$$

Remarks: There are two important aspects of this generative model: First, control states are not action—they are an internal representation of action that may or may not be related to actions emitted by the agent. In the generative model, control states affect the transitions among hidden states; in other words, they only affect outcomes vicariously through hidden states. It is these control states that represent agency, which may or may not be a veridical representation of what the agent can actually do (or is doing)—in this sense, they can be regarded as fictive action that gives the generative model extra degrees of freedom to model state transitions under prior beliefs. Recall that action only changes observations and is selected on the basis of posterior beliefs about the next observable state. Conversely, control states are modeled as hidden states over time and are inferred. This means they only exist in the mind (posterior beliefs) of the agent.

Second, the priors on the hidden states $P(\tilde{x}, \tilde{u}|\theta)$ are formulated in a pullback sense; that is, they run backwards in time. This preserves the Markov dependencies but allows us to specify the prior over a sequence of states in terms of transition probabilities and a prior distribution over the final (terminal) state. Put simply, the parameters of the (transition) model encode where I came from, not where I am going. See **Figure 1**. This particular form of prior belief is chosen for convenience, because it accommodates beliefs about the desired final state—of the sort that would be specified with a terminal cost function, $r(x_T)$.

The generative model in Equation (9) is fairly general and makes no specific assumptions about the implicit cost of inferred control (it does not assume quadratic control costs) or allowable state transitions. In what follows, we illustrate inference or model inversion using a particular parameterization and variational inversion scheme. This example is used to illustrate agency-based inference, accepting that there are many different model parameterizations and inversion schemes that could have been used.

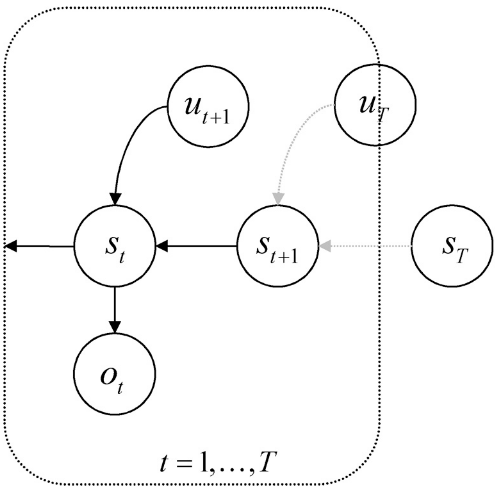


FIGURE 1 | Probabilistic graphical model illustrating the Markovian dependencies among hidden states generating sensory data. These hidden states (s_t, u_t) are represented explicitly, over all time points: $t = 0, \ldots, T$. This means there is a representation of the past and future that includes hidden states mediating control. Note that the dependency of this hidden Markov model runs backwards in time so that all preceding hidden states are conditioned recursively on the final or terminal (goal) state.

Generative probability: The generative model used below comprises the following likelihood and prior distributions:

$$\begin{aligned} P(s_t|x_t, \theta) &= \mathbf{A} \cdot x_t \\ P(x_{t-1}|x_t, u_t, \theta) &= \left(\prod_i \mathbf{B}_i^{u_{ti}}\right) \cdot x_t \\ P(x_T|\theta) &= \mathbf{c} \\ P(u_t|\theta) &= \prod_i \mathbf{d}_i^{u_{ti}} \end{aligned} \quad (10)$$

The parameters $\theta = \{\mathbf{A}, \mathbf{B}_1, \mathbf{B}_2, \ldots, \mathbf{c}, \mathbf{d}\}$ of this model are

$$\begin{aligned} \mathbf{A} &= \{a_{ij}\} : \sum_j a_{ij} = 1, \ \forall i \\ \mathbf{B}_k &= \{b_{ijk}\} : \sum_j b_{ijk} = 1, \ \forall i, k \\ \mathbf{c} &= \{c_i\} : \sum_i c_i = 1 \\ \mathbf{d} &= \{d_i\} : \sum_i d_i = 1 \end{aligned} \quad (11)$$

The parameters in the matrices $\mathbf{B}_k$ encode transition probabilities among hidden states that are engaged when the control state $u_k = 1$, where the control states have a multinomial distribution—only one can be "on" at any time. The hidden states cause observed states through the mapping encoded by **A**. The vectors **c** and **d** encode the prior distribution over the final hidden state and control states, respectively; these specify the goal and prior costs on control.

Conditional probability: To exploit the Markovian form of the generative model we will use an efficient approximate inference scheme afforded by variational Bayesian learning (Beal, 2003); for a tutorial see Fox and Roberts (2011). The efficiency rests on replacing posterior dependencies among hidden states (over time) with mean field effects on the marginal probabilities at each time point. This is achieved using the following *mean-field assumption* for the conditional distribution:

$$
\begin{aligned}
Q(s, u) &= \prod_{t=1}^{T} Q(s_t)Q(u_t) \\
Q(s_t|\alpha_t) &= \prod_i \alpha_{ti}^{s_i} : \sum_i \alpha_{ti} = 1 \\
Q(u_t|\beta_t) &= \prod_i \beta_{ti}^{u_i} : \sum_i \beta_{ti} = 1
\end{aligned}
\tag{12}
$$

Standard variational Bayesian learning now provides a recipe for optimizing the sufficient statistics (α_t, β_t) of the conditional probability over hidden and control states. The ensuing variational updates for the sufficient statistics $\mu_k = \{\alpha_0^k, \ldots, \alpha_T^k, \beta_0^k, \ldots, \beta_T^k\}$ at successive times k are Friston et al. (2012b):

```
for $k = 1$ to T
  until · convergence:
    for $t = (T-1)$ to $(k+1)$
      $\alpha'_t = \exp([\ln \mathbf{A}^T \cdot s_t] + \sum_j \beta^k_{(t+1)j} \ln \mathbf{B}_j \cdot\alpha^k_{(t+1)} + \sum_j \beta^k_{tj} \ln \mathbf{B}_j^T \cdot \alpha^k_{(t-1)})$
      $\alpha_t^{k+1} = \frac{\alpha'_t}{\sum_i \alpha'_{ti}}$
      $\beta'_{ti} = \exp(\alpha^{kT}_{t-1} \cdot \ln \mathbf{B}_i \cdot \alpha^k_t + \ln d_i)$
      $\beta_t^{k+1} = \frac{\beta'_t}{\sum_i \beta'_{ti}}$          (13)
```

The square brackets in $[\ln A^T \cdot s_t]$ indicate that this term is used only when observations are available. This speaks to an important aspect of these update schemes; namely, posterior beliefs about the hidden states at all points during the sequence are updated iteratively at each time point. At each time point, the variational updates cycle over representations of future states to update the sufficient statistics encoding posterior beliefs. These update cycles are themselves repeated as time progresses, so that there is convergence both within and between cycles. This means the sufficient statistics change over two timescales; a fast timescale that updates posterior beliefs about the future and a slow timescale that updates posterior beliefs in the future. Posterior beliefs about the trajectory, at both timescales, ensure that the trajectory convergences on the final (desired) location, where the anticipated trajectory is realized through action. It is interesting to speculate about neurophysiologic implementations of this sort of scheme, particularly in relation to nested electrophysiological oscillations (Canolty et al., 2006). The notion here is that the electrophysiological correlates of updating may show nested oscillations, with fast (gamma) oscillations reflecting updates in a fictive future and slower (theta) dynamics that reflect updates in real time; with timescales of 25 and 250 ms respect, respectively. To illustrate the nature of this optimal control, we now apply it to a well-known problem in optimal control theory that presents some special challenges.

THE MOUNTAIN CAR PROBLEM

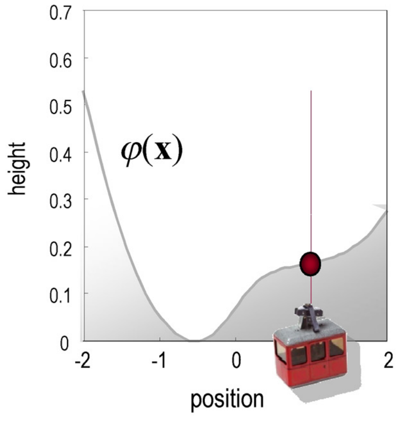
In the mountain car problem, one has to park a mountain car halfway up the side of a valley. However, the mountain car is not strong enough to climb directly to the parking place, which means the only way to assess the goal is to ascend the other side of the valley to acquire sufficient momentum during the return trip. This represents an interesting problem, when considered in the state space of position and velocity: the agent has to move away from its target location to attain the goal later. In other words, it has to execute a circuitous trajectory through state space (as in avoiding obstacles). We have used this problem previously to illustrate how Bayes-optimal control can be learned in terms of the parameters controlling prior beliefs about trajectories (Friston et al., 2009) and using heuristic policies (Gigerenzer and Gaissmaier, 2011) based on the destruction of costly fixed point attractors (Friston, 2010).

It should be noted that the mountain car problem is normally cast as a learning problem—in which an optimal policy has to be learned. However, here, we use it to illustrate optimal behavior in terms of inference. In other words, we assume the agent has already learned the constraints afforded by the world it operates in—and now has to infer an optimal policy within a single trial. In this setting, the mountain car problem provides a challenging inference problem, particularly when we include random fluctuations in both the states generating observations and the observations themselves. The mountain car problem can be specified with the equations of motion in **Figure 2**. Here, we consider a discrete state space and time formulation of this problem and use it to illustrate agency based control.

To create a discrete version, we ensured that expected changes in position and velocity match the equations of motion, when integrated over discrete time intervals (here $\Delta t = 2\text{s}$). The ensuing pullback probabilities for each level of control satisfy (subject to the constraint that only the states adjacent to the expected position and velocity are non-zero).

$$
\sum_i \mathbf{x}(x_i)B_{ijk} = \mathbf{x}(\tilde{x}_j) - f(\mathbf{x}(x_j), a(u_k))\Delta t \tag{14}
$$

$$\varphi(\mathbf{x}) = \begin{cases} -2\mathbf{x}-1 & :\mathbf{x} \le 0 \\ -\mathbf{x}^2(1+5\mathbf{x}^2)^{-3/2} - \frac{1}{8}\mathbf{x}^4 & :\mathbf{x} > 0 \end{cases}$$



Equations of motion

$$f(\mathbf{x}) = \begin{bmatrix} \dot{\mathbf{x}} \\ \dot{\mathbf{x}}' \end{bmatrix} = \begin{bmatrix} \mathbf{x}' \\ \sigma(a) - \partial_x \varphi - \frac{1}{8}\mathbf{x}' \end{bmatrix}$$

FIGURE 2 | Schematic of the mountain car problem. The **upper panel** (and associated equations) illustrate the landscape or potential energy function that defines the motion of the car. This has a minima at $x = (-0.5, 1)$. The mountain-car is shown at the desired parking position at the top of the hill on the right $x = (1, 0)$ (indicated with a red ball). The equations of motion in the **lower panel** describe the forces exerted on the car, which include $\sigma(a)$, a sigmoid (hyperbolic tangent) function of action, gravitational forces, and friction.

Here, $\mathbf{x}(x_i) \in \mathbb{R}^2$ returns the continuous position and velocity associated with the i-th hidden state. Similarly, $a(u_k) \in \mathbb{R}$ returns the real valued action associated with the k-th control state. In these simulations, we used five levels of control corresponding to $a(u_k) \in \{-2, -1, 0, 1, 2\}$. This means the agent assumes that strong or intermediate acceleration can be applied in a right or leftward direction. To simulate random fluctuations in the motion of the mountain car, we smoothed the parameter matrix **B** to augment the uncertainty about the previous state incurred by discretizing state space. The state space comprised 32 position (from −2 to 2) and velocity bins (from −3 to 3), giving $32 \times 23 = 1024$ discrete states. For simplicity, we assumed a one-to-one mapping between hidden and observed states; that is $\mathbf{A} = I$ and placed uniform prior costs over control. Prior beliefs about the final state specify the goal $\mathbf{x} = (1, 0)$—namely, to maintain a position at the parking location with zero velocity; see **Figure 2**. Finally, the action-dependent sampling probabilities $R(s_{t+1}|s_t, a_t)$ were the transposed versions of the pullback probabilities in Equation (14). These sampling probabilities were used to select action and to generate the next sensory input. Action used the same five levels as the control states—however, as noted above, there is no requirement that action and control be related in this way.

Figure 3 shows the results of a simulation using $T = 16$ time steps and a starting position of $\mathbf{x} = (0, 0)$. In these simulations the variational updates were repeated eight times and then an action was selected. The upper panel shows the trajectories (real and anticipated) through state space, while the lower panels show the inferred control states and selected action as a function of time. The darker line in the upper panel connects the states visited over the 16 time steps, while the gray lines report the anticipated trajectories from the beginning of the trial to the end. The inferred trajectories are shown as the expected position and velocity, based on posterior beliefs over discrete states. One can see that the actual trajectory fulfills, fairly faithfully, the anticipated sequences and that there has been relatively little updating during execution. As anticipated, the mountain car moves away from its target to acquire sufficient momentum to access the goal on the right. Note the similarity between the selected actions (right) and the inferred control states (left). The interesting thing here is that the agent was not always sure about which control state was currently engaged. However, the control state with the highest posterior probability, which corresponds to the action the agent believes it will emit next, is always selected by active inference. In other words, even under uncertainty about hidden and control states, there is sufficient confidence in the next sensory state to inform action.

SUMMARY

In summary, we have reviewed conventional approaches to (partially observable) Markov decision problems and have cast reward or cost functions in terms of prior beliefs about state transitions. This implicitly resolves the redundancy between cost functions and priors that underlies the complete class theorems. We then exploited this redundancy by specifying optimal policies in terms of prior beliefs about future (terminal) states. The ensuing scheme may provide a metaphor for model-based decision-making in real agents that has an explicit planning or anticipatory aspect. This solution was based upon approximate (variational) Bayesian inference that respects the Markov nature of decision processes.

The aim of this work was to unpack some of the implications of optimal control for its implementation in real-world agents. The most important is the representation of hidden control states that are required for accessing distal rewards in the future. This contrasts with the usual problem formulation of MDPs, which is to define a normative model and a corresponding notion of optimality. In optimal control theory, state transitions are specified in terms of value functions that are solutions to the appropriate Bellman optimality equations, given a cost function. The notion that the Bellman optimality principle "can be derived as a limit case" from the variational principles that underlie active inference also emerges in recent information theoretic formulations of bounded rationality (Braun et al., 2011): Braun

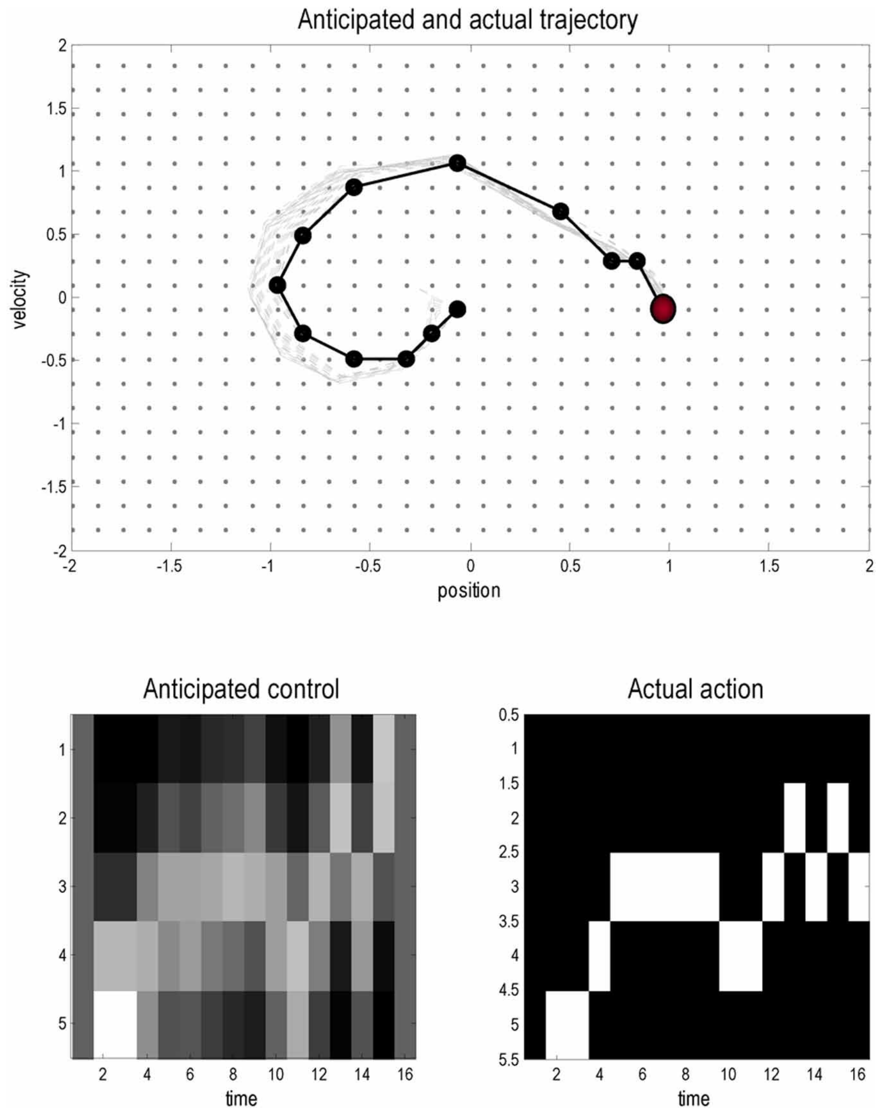


FIGURE 3 | This figure shows the results of a simulated (agency based) trajectory over $T = 16$ time steps starting at $x = (0, 0)$ and ending at the goal location $x = (1, 0)$ (red ball) using active inference and explicit representations of the future. The **upper panel** shows the trajectories in the state space of position and velocity. The grey lines represent anticipated trajectories accumulated during control, while the dark (dotted) lines show the actual trajectory through state space. The anticipated trajectories are the expected values based upon posterior expectations about past and future states. They are therefore continuous functions of position and velocity. In contrast, the actual trajectory is restricted to the 1024 discrete states that can be occupied; these are shown as light grey dots. The **lower panels** show the anticipated control and the actual actions selected under active inference (in image format where lighter colors mean a higher probability). Note that there is a high degree of correspondence; however, the posterior beliefs about control and not always absolutely certain: these are the beliefs at the times each action is selected.

et al. consider control costs in terms of the (cross) entropy of choice probabilities and augment expected utility to produce a free energy optimality criterion. This *free utility* captures bounded rationality by ensuring the divergence between optimal and prior choice probabilities is minimized. They show that minimizing free utility includes both discrete and continuous stochastic optimal control as special cases and can be derived "without invoking the Hamilton–Jacobi–Bellman equation or the Bellman optimality equations". See also Theodorou et al. (2010), who exploit a similar formalism but with a more classical motivation. The generalization of optimal control using free utility is compelling and unifies approximate optimal control methods in both the continuous and discrete domain. However, free utility is fundamentally different from variational free energy, because it is a functional of choice probabilities over hidden states. In contrast, variational free energy is a function of observed states. Crucially, free utility depends on a cost function, while free energy does not. This is because the free energy principle is based on the invariant or ergodic solution $P(s|m)$ to the *Kolmogorov forward equation*, which specifies the value of an observed state $V(s|m) = \ln P(s|m)$

directly, without reference to cost—see next section and Friston and Ao (2012). In other words, value is (log) evidence or negative surprise. Conversely, free utility is based on the *Kolmogorov backward equation*, which can only be solved given terminal costs.

In answer to the title of this paper, the value of an observed state is then prescribed by a generative model in terms of the probability a state will be occupied. It can be seen easily that minimizing the entropy of the invariant probability distribution over observations maximizes expected value:

$$\mathbf{E}_P[-\ln P(s|m)] = \mathbf{E}_P[V(s|m)] \tag{15}$$

Minimizing the entropy of observed states is the *raison d'être* for the free energy principle (see below), which invokes variational free energy to finesse the intractable problem of marginalizing over hidden states to evaluate value or negative surprise. This complements the use of free utility to finesse the intractable problem of solving Bellman optimality equations (Braun et al., 2011). It can be seen from Equation (5) that free energy $F(s, \mu) \geq -\ln P(s|m) = -V(s|m)$ bounds surprise and can therefore be minimized to maximize value.

In conclusion, we have described a variational free energy formulation of (partially observable) Markov decision problems in decision making under uncertainty. We have seen that optimal control can be cast as *active inference*, in which both *action and posterior beliefs* about hidden states minimize a free energy bound on the value (log Bayesian model evidence) of observed states, under a generative model. In this setting, reward or cost functions are absorbed into prior beliefs about state transitions and terminal states. This converts optimal control into a pure inference problem, enabling the application of standard Bayesian filtering techniques. Crucially, this entails modeling future states state that endows the generative model with a sense of agency. This leads to a distinction between models with and without inference on future states—namely, agency free and agency based models, respectively. In the next section, we ask: where do prior beliefs about future states come from?

ACTION, PERCEPTION, AND CONTROL

The previous section suggested that value is simply the log-evidence associated with sensory samples or evidence for an internal model or hypothesis about the world. In this setting, valuable behavior simply involves sampling the world to ensure model predictions are fulfilled, where these predictions rest upon (prior) beliefs about future states. In this section, we motivate the imperative to maximize log-evidence from the basic principles of self-organization. We go on to show that prior beliefs about future states have a relatively simple form; namely, we believe that our future states will minimize uncertainty about our current beliefs.

If perception corresponds to hypothesis testing (Gregory, 1980); then sensory sampling might be correspond to experiments that generate sensory data. In the next three sections, we explore the idea that eye movements are optimal experiments, in which data are gathered to test hypotheses or beliefs about how those data are caused. This provides a plausible model of visual search that can be motivated from the basic tenets of self-organized behavior: namely, the imperative to minimize the entropy of hidden states of the world and their sensory consequences. Simulations of the resulting active inference scheme reproduce sequential eye movements that are reminiscent of empirically observed saccades and provide some counterintuitive insights into the way that sensory evidence is accumulated or assimilated into beliefs about the world.

If variational free energy minimization is applied to both action and perception, action will fulfill predictions based upon conditional beliefs about the state of the world. However, the uncertainty associated with those conditional beliefs depends upon the way data are sampled: for example, where we direct our gaze or how we palpate a surface. The deployment of sensory epithelia is itself a hidden state that has to be inferred. However, these hidden states can be changed by action, which means there is a subset of hidden states over which we have control. These are the hidden control states of the previous section. Prior beliefs about these hidden control states dictate how we engage actively with the environment and lead to the notion of fictive or *counterfactual representations*; in other words, what we would infer about the world, if we sampled it in a particularly way. This leads naturally to the internal representation of prior beliefs about future sampling and the emergence of things like agency, intention, and salience. We will illustrate these points using visual search and the optimal control of saccadic eye movements (Grossberg et al., 1997; Itti and Baldi, 2009; Srihasam et al., 2009); noting that similar principles should apply to other sensory modalities. For example, they should apply to motor control when making inferences about objects causing somatosensory sensations (Gibson, 1979).

ACTIVE INFERENCE—A CONTINUOUS TIME FORMULATION

This section establishes the nature of Bayes-optimal inference in the context of controlled sensory searches. It starts with the basic premise that underlies free energy minimization; namely, the imperative to minimize the dispersion of sensory states and their hidden causes to ensure a homoeostasis of the external and internal milieu (Ashby, 1947). It rehearses briefly how action and perception follow from this imperative and highlights the important role of prior beliefs about the sampling of sensory states. At this point, we move away from the discrete formulations of MDPs and turned to continuous formulations, where probability distributions become densities and discrete time becomes continuous. This shift is deliberate and allows the discrete formulations of the previous sections to be compared and contrasted with the equivalent continuous time formulations that predominate in biologically realistic simulations.

Notation and set up: Here we use $X : \Omega \times \ldots \to \mathbb{R}$ for real valued random variables and $x \in X$ for particular values. A probability density will be denoted by $p(x) = \Pr\{X = x\}$ using the usual conventions and its entropy $H[p(x)]$ by $H(X)$. From now on, the tilde notation $\tilde{x} = (x, x', x'', \ldots)$ denotes variables in generalized coordinates of motion (Friston, 2008), where each prime denotes a temporal derivative (using Lagrange's notation). For simplicity, constant terms will be omitted from equalities.

Definition: Active inference rests on the tuple $(\Omega, \Psi, S, A, R, q, p)$ that comprises the following:

- *A sample space* Ω or non-empty set from which random fluctuations or outcomes $\omega \in \Omega$ are drawn
- *Hidden states* $\Psi : \Psi \times A \times \Omega \to \mathbb{R}$—states of the world that cause sensory states and depend on action
- *Sensory states* $S : \Psi \times A \times \Omega \to \mathbb{R}$—the agent's sensations that constitute a probabilistic mapping from action and hidden states
- *Action* $A : S \times R \to \mathbb{R}$—an agent's action that depends on its sensory and internal states
- *Internal states* $R : R \times S \times \Omega \to \mathbb{R}$—the states of the agent that cause action and depend on sensory states
- *Generative density* $p(\tilde{s}, \tilde{\psi}|m)$—a probability density function over sensory and hidden states under a generative model denoted by m
- *Conditional density* $q(\tilde{\psi}) := q(\tilde{\psi}|\tilde{\mu})$—an arbitrary probability density function over hidden states $\tilde{\psi} \in \Psi$ that is parameterized by internal states $\tilde{\mu} \in R$

We assume that the imperative for any biological system is to minimize the dispersion of its sensory and hidden states, with respect to action (Ashby, 1947; Nicolis and Prigogine, 1977; Friston and Ao, 2012). We will refer to the sensory and hidden states collectively as *external states* $S \times \Psi$. As noted above, the dispersion of external states corresponds to the (Shannon) entropy of their probability density that, under ergodic assumptions, equals (almost surely) the long-term time average of a Gibbs energy:

$$\begin{aligned} H(S, \Psi) &= E_t[G(\tilde{s}(t), \tilde{\psi}(t))] \\ G &= -\ln p(\tilde{s}(t), \tilde{\psi}(t)|m) \end{aligned} \tag{16}$$

Gibbs energy $G(\tilde{s}, \tilde{\psi})$ is defined in terms of the generative density or model. Clearly, agents cannot minimize this energy directly because the hidden states are unknown. However, we can decompose the entropy into the entropy of the sensory states (to which the system has access) and the conditional entropy of hidden states (to which the system does not have access). This second term is also called the *equivocation* of the hidden states about the sensory states:

$$\begin{aligned} H(S, \Psi) &= H(S) + H(\Psi|S) \\ &= E_t[-\ln p(\tilde{s}(t)|m) + H(\Psi|S = \tilde{s}(t))] \end{aligned} \tag{17}$$

This decomposition means that the entropy of the external states can be minimized through action to minimize sensory surprise $-\ln p(\tilde{s}(t)|m)$, under the assumption that the consequences of action minimize the equivocation or average uncertainty about hidden states:

$$\begin{aligned} a(t) &= \arg\min_{a \in A}\{-\ln p(\tilde{s}(t)|m)\} \\ \tilde{u}(t) &= \arg\min_{\tilde{u} \in U}\{H(\Psi|S = \tilde{s}(t))\} \end{aligned} \tag{18}$$

The consequences of action are expressed by changes in a subset of hidden states $U \subset \Psi$—the hidden control states or *hidden controls*. When Equation (18) is satisfied, the variation of entropy in Equation (16) with respect to action and its consequences are zero, which means the entropy has been minimized (at least locally). However, the hidden controls cannot be optimized explicitly because they are hidden from the agent. To resolve this problem, we first consider action and then return to optimizing hidden control states.

ACTION AND PERCEPTION

Action cannot minimize sensory surprise directly because this would involve an intractable marginalization over hidden states, so—as in the discrete formulation—surprise is replaced with an upper bound called variational free energy (Feynman, 1972). However, replacing surprise with free energy means that internal states also have to minimize free energy, because free energy is a function of internal states:

$$\begin{aligned} a(t) &= \arg\min_{a \in A}\{F(\tilde{s}(t), \tilde{\mu}(t))\} \\ \tilde{\mu}(t) &= \arg\min_{\tilde{\mu} \in R}\{F(\tilde{s}(t), \tilde{\mu})\} \\ F &= E_q[G(\tilde{s}, \tilde{\psi})] - H[q(\tilde{\psi}|\tilde{\mu})] \\ &= -\ln p(\tilde{s}|m) + D[q(\tilde{\psi})||p(\tilde{\psi}|\tilde{s}, m)] \\ &\geq -\ln p(\tilde{s}|m) \end{aligned} \tag{19}$$

This induces a dual minimization with respect to action and the internal states that parameterize the conditional density. These minimizations correspond to action and perception, respectively. In brief, the need for perception is induced by introducing free energy to finesse the evaluation of surprise; where free energy can be evaluated by an agent fairly easily, given a generative model. The last equality says that free energy is always greater than surprise because the second (Kullback–Leibler divergence) term is non-negative. As in the discrete formulation, when free energy is minimized with respect to the internal states, free energy approximates surprise and the conditional density approximates the posterior density over external states:

$$D[q(\tilde{\psi})||p(\tilde{\psi}|\tilde{s}, m)] \approx 0 \Rightarrow \begin{cases} q(\tilde{\psi}) \approx p(\tilde{\psi}|\tilde{s}, m) \\ H[q(\tilde{\psi})] \approx H(\Psi|S = \tilde{s}) \end{cases} \tag{20}$$

Minimizing free energy also means that the entropy of the conditional density approximates the equivocation of the hidden states. This allows us to revisit the optimization of hidden controls, provided we know how they affect the conditional density.

THE MAXIMUM ENTROPY PRINCIPLE AND THE LAPLACE ASSUMPTION

If we admit an encoding of the conditional density up to second order moments, then the maximum entropy principle (Jaynes, 1957) implicit in the definition of free energy (Equation 19) requires $q(\tilde{\psi}|\tilde{\mu}) = \mathcal{N}(\tilde{\mu}, \Sigma)$ to be Gaussian. This is because a

Gaussian density has the maximum entropy of all forms that can be specified with two moments. Adopting a Gaussian form is known as the Laplace assumption and enables us to express the entropy of the conditional density in terms of its first moment or expectation. This follows because we can minimize free energy with respect to the conditional covariance as follows:

$$\begin{aligned} F &= G(\tilde{s}, \tilde{\mu}) + \tfrac{1}{2} tr(\Sigma \cdot \partial_{\tilde{\mu}\tilde{\mu}} G) - \tfrac{1}{2} \ln |\Sigma| \\ &\Rightarrow \ \partial_{\Sigma} F = \tfrac{1}{2} \partial_{\tilde{\mu}\tilde{\mu}} G - \tfrac{1}{2} \Pi \\ \partial_{\Sigma} F = 0 &\Rightarrow \Pi = \partial_{\tilde{\mu}\tilde{\mu}} G \Rightarrow H(\Psi | S = \tilde{s}) \\ &\approx H[q(\tilde{\psi})] = -\tfrac{1}{2} \ln |\partial_{\tilde{\mu}\tilde{\mu}} G| \end{aligned} \tag{21}$$

Here, the conditional precision $\Pi(\tilde{s}, \tilde{\mu})$ is the inverse of the conditional covariance $\Sigma(\tilde{s}, \tilde{\mu})$. In short, the entropy of the conditional density and free energy are functions of the conditional expectations and sensory states. Now that we have (an approximation to) the equivocation, we can return to its minimization through prior beliefs.

BAYES-OPTIMAL CONTROL

We can now optimize the hidden controls vicariously through prior expectations that are fulfilled by action. This can be expressed in terms of prior expectations about hidden controls.

$$\tilde{\eta}_u(t) = \arg \min_{\tilde{\eta}_u \in U} \{H[q(\tilde{\psi} | \tilde{\mu}_x(t + \tau), \tilde{\eta}_u)]\} \tag{22}$$

This equation means the agent expects hidden control states to minimize uncertainty about hidden states in the future—this is the entropy of the conditional density in the future, which we will call a counterfactual density. Interestingly, Equations (19) and (22) say that conditional expectations (about hidden states) maximize conditional uncertainty, while prior expectations (about hidden controls) minimize conditional uncertainty. This means the posterior and prior beliefs are in opposition, trying to maximize and minimize uncertainty about hidden states, respectively. The latter represent prior beliefs that hidden states are sampled to maximize conditional confidence, while the former minimizes conditional confidence to ensure the explanation for sensory data does not depend on particular hidden states—in accord with the maximum entropy principle (or Laplace's principle of indifference). In what follows, we will refer to the negative entropy of the counterfactual density as *salience*; noting that salience is a measure of confidence about hidden states that depends on how they are sampled. This means that the agent believes, a priori, that salient features will be sampled.

SUMMARY AND RELATED PRINCIPLES

To recap, we started with the assumption that biological systems minimize the dispersion or entropy of states in their external milieu to ensure a sustainable and homoeostatic exchange with their environment (Ashby, 1947). Clearly, these states are hidden and therefore cannot be measured or changed directly. However, if agents know how their action changes sensations (for example, if they know contracting certain muscles will necessarily excite primary sensory afferents from stretch receptors), then they can minimize the dispersion of their sensory states by countering surprising deviations from expected values. However, reducing the dispersion of sensory states will only reduce the dispersion of hidden states, if the sensory states report the underlying hidden states faithfully. This faithful reporting requires agents to minimize their conditional uncertainty about hidden states, through prior beliefs about the way sensory organs are deployed. This imperative—to minimize conditional uncertainty—is remarkably consistent with a number of other constructs, such as Bayesian surprise (Itti and Baldi, 2009). It is fairly easy to show that maximizing salience is the same as maximizing Bayesian surprise (Friston et al., 2012a). This is important because it links salience in the context of active inference with salience in the theoretical (Humphreys et al., 2009) and empirical literature (Shen et al., 2011; Wardak et al., 2011). Here, we will focus on the principle of maximum mutual information.

Priors about hidden controls express the belief that conditional uncertainty will be minimal. The long-term average of this conditional uncertainty is the conditional entropy of hidden states, which can be expressed as the entropy over hidden states minus the mutual information between hidden and sensory states:

$$H(\Psi | S) = E_t[H(\Psi | S = \tilde{s}(t))] = H(\Psi) - I(\Psi; S) \tag{23}$$

In other words, minimizing conditional uncertainty is equivalent to maximizing the mutual information between external states and their sensory consequences. This is one instance of the Infomax principle (Linsker, 1990). Previously, we have considered the relationship between free energy minimization and the principle of maximum mutual information, or minimum redundancy (Barlow, 1961, 1974; Optican and Richmond, 1987; Oja, 1989; Olshausen and Field, 1996; Bialek et al., 2001) in terms of the mapping between hidden and internal states (Friston, 2010). In this setting, one can show that "the infomax principle is a special case of the free-energy principle that obtains when we discount uncertainty and represent sensory data with point estimates of their causes." Here, we consider the mapping between external and sensory states and find that prior beliefs about how sensory states are sampled further endorse the Infomax principle. In what follows, we consider the neurobiological implementation of these principles.

NEUROBIOLOGICAL IMPLEMENTATIONS OF ACTIVE INFERENCE

In this section, we take the general principles above and consider how they might be implemented in a (simulated) brain. The equations in this section may appear a bit complicated; however, they are based on just four assumptions.

- The brain minimizes the free energy of sensory inputs defined by a generative model.
- This model includes prior expectations about hidden controls that maximize salience.
- The generative model used by the brain is hierarchical, nonlinear, and dynamic.
- Neuronal firing rates encode the expected state of the world, under this model.

The first assumption is the free energy principle, which leads to active inference in the embodied context of action. The second assumption follows from the arguments of the previous section. The third assumption is motivated easily by noting that the world is both dynamic and non-linear and that hierarchical causal structure emerges inevitably from a separation of temporal scales (Ginzburg and Landau, 1950; Haken, 1983). Finally, the fourth assumption is the Laplace assumption that, in terms of neural codes, leads to the *Laplace code* that is arguably the simplest and most flexible of all neural codes (Friston, 2009).

Given these assumptions, one can simulate a whole variety of neuronal processes by specifying the particular equations that constitute the brain's generative model. The resulting perception and action are specified completely by the above assumptions and can be implemented in a biologically plausible way as described below (see **Table 1** for a list of previous applications of this scheme). In brief, these simulations use differential equations that minimize the free energy of sensory input using a generalized gradient descent (Friston et al., 2010a).

$$\begin{aligned}\dot{\tilde{\mu}}(t) &= \mathcal{D}\tilde{\mu}(t) - \partial_{\tilde{\mu}} F(\tilde{s}, \tilde{\mu}) \\ \dot{a}(t) &= -\partial_a F(\tilde{s}, \tilde{\mu})\end{aligned} \tag{24}$$

Table 1 | Processes and paradigms that have been modeled using the generalized Bayesian filtering scheme in this paper.

Domain	Process or paradigm
Perception	Perceptual categorization (bird songs) (Friston and Kiebel, 2009a,b) Novelty and omission-related responses (Friston and Kiebel, 2009a,b) Perceptual inference (speech) (Kiebel et al., 2009)
Sensory learning	Perceptual learning (mismatch negativity) (Friston and Kiebel, 2009a,b)
Attention	Attention and the Posner paradigm (Feldman and Friston, 2010) Attention and biased competition (Feldman and Friston, 2010)
Motor control	Retinal stabilization and oculomotor reflexes (Friston et al., 2010b) Saccadic eye movements and cued reaching (Friston et al., 2010b) Motor trajectories and place cells (Friston et al., 2011)
Sensorimotor integration	Bayes-optimal sensorimotor integration (Friston et al., 2010b)
Behavior	Heuristics and dynamical systems theory (Friston and Ao, 2012) Goal-directed behavior (Friston et al., 2009)
Action observation	Action observation and mirror neurons (Friston et al., 2011)

These coupled differential equations describe perception and action, respectively, and just say that internal brain states and action change in the direction that reduces free energy. The first is known as generalized predictive coding and has the same form as Bayesian (e.g., Kalman–Bucy) filters used in time series analysis; see also Rao and Ballard (1999). The first term in Equation (24) is a prediction based upon a differential matrix operator $\mathcal{D}$ that returns the generalized motion of the expectation, such that $\mathcal{D}\tilde{\mu} = [\mu', \mu'', \mu''', \ldots]^T$. The second term is usually expressed as a mixture of prediction errors that ensures the changes in conditional expectations are Bayes-optimal predictions about hidden states of the world. The second differential equation says that action also minimizes free energy. The differential equations above are coupled because sensory input depends upon action, which depends upon perception through the conditional expectations. This circular dependency leads to a sampling of sensory input that is both predicted and predictable, thereby minimizing free energy and surprise.

To perform neuronal simulations under this scheme, it is only necessary to integrate or solve Equation (24) to simulate the neuronal dynamics that encode conditional expectations and ensuing action. Conditional expectations depend upon the brain's generative model of the world, which we assume has the following hierarchical form.

$$\begin{aligned} s &= g^{(1)}(x^{(1)}, v^{(1)}, u^{(i)}) + \omega_v^{(1)} \\ \dot{x}^{(1)} &= f^{(1)}(x^{(1)}, v^{(1)}, u^{(i)}) + \omega_x^{(1)} \\ &\vdots \\ v^{(i-1)} &= g^{(i)}(x^{(i)}, v^{(i)}, u^{(i)}) + \omega_v^{(i)} \\ \dot{x}^{(i)} &= f^{(i)}(x^{(i)}, v^{(i)}, u^{(i)}) + \omega_x^{(i)} \\ &\vdots \end{aligned} \tag{25}$$

This equation is just a way of writing down a model that specifies a probability density over the sensory and hidden states, where the hidden states $\Psi = X \times V \times U$ have been divided into hidden dynamic, causal, and control states. Here, $(g^{(i)}, f^{(i)})$ are non-linear functions of hidden states that generate sensory inputs at the first level. Hidden causes $V \subset \Psi$ can be regarded as functions of hidden dynamic states; hereafter, hidden states $X \subset \Psi$. Random fluctuations $(\omega_x^{(i)}, \omega_v^{(i)})$ on the motion of hidden states and causes are conditionally independent and enter each level of the hierarchy. It is these that make the model probabilistic: they play the role of sensory noise at the first level and induce uncertainty about states at higher levels. The inverse amplitudes of these random fluctuations are quantified by their precisions $(\Pi_x^{(i)}, \Pi_v^{(i)})$. Hidden causes link hierarchical levels, whereas hidden states link dynamics over time. Hidden states and causes are abstract quantities (like the motion of an object in the field of view) that the brain uses to explain or predict sensations. In hierarchical models of this sort, the output of one level acts as an input to the next. This input can produce complicated (generalized) convolutions with deep (hierarchical) structure.

PERCEPTION AND PREDICTIVE CODING

Given the form of the generative model (Equation 25) we can now write down the differential equations (Equation 24) describing neuronal dynamics in terms of (precision-weighted) prediction errors on the hidden causes and states. These errors represent the difference between conditional expectations and predicted values, under the generative model (using $A \cdot B := A^T B$ and omitting higher-order terms):

$$\begin{aligned}
\dot{\tilde{\mu}}_x^{(i)} &= \mathcal{D}\tilde{\mu}_x^{(i)} + \frac{\partial \tilde{g}^{(i)}}{\partial \tilde{\mu}_x^{(i)}} \cdot \xi_v^{(i)} + \frac{\partial \tilde{f}^{(i)}}{\partial \tilde{\mu}_x^{(i)}} \cdot \xi_x^{(i)} - \mathcal{D}^T \xi_x^{(i)} \\
\dot{\tilde{\mu}}_v^{(i)} &= \mathcal{D}\tilde{\mu}_v^{(i)} + \frac{\partial \tilde{g}^{(i)}}{\partial \tilde{\mu}_v^{(i)}} \cdot \xi_v^{(i)} + \frac{\partial \tilde{f}^{(i)}}{\partial \tilde{\mu}_v^{(i)}}^T \cdot \xi_x^{(i)} - \xi_v^{(i+1)} \\
\dot{\tilde{\mu}}_u^{(i)} &= \mathcal{D}\tilde{\mu}_u^{(i)} + \frac{\partial \tilde{g}^{(i)}}{\partial \tilde{\mu}_u^{(i)}} \cdot \xi_v^{(i)} + \frac{\partial \tilde{f}^{(i)}}{\partial \tilde{\mu}_u^{(i)}} \cdot \xi_x^{(i)} - \xi_u^{(i+1)} \\
\xi_x^{(i)} &= \Pi_x^{(i)}(\mathcal{D}\tilde{\mu}_x^{(i)} - \tilde{f}^{(i)}(\tilde{\mu}_x^{(i)}, \tilde{\mu}_v^{(i)}, \tilde{\mu}_u^{(i)})) \\
\xi_v^{(i)} &= \Pi_v^{(i)}(\tilde{\mu}_v^{(i-1)} - \tilde{g}^{(i)}(\tilde{\mu}_x^{(i)}, \tilde{\mu}_v^{(i)}, \tilde{\mu}_u^{(i)})) \\
\xi_u^{(i)} &= \Pi_u^{(i)}(\tilde{\mu}_u^{(i-1)} - \tilde{\eta}_u^{(i)})
\end{aligned} \tag{26}$$

Equation (26) can be derived fairly easily by computing the free energy for the hierarchical model in Equation (25) and inserting its gradients into Equation (24). This produces a relatively simple update scheme, in which conditional expectations are driven by a mixture of prediction errors, where prediction errors are defined by the equations of the generative model.

It is difficult to overstate the generality and importance of Equation (26): its solutions grandfather nearly every known statistical estimation scheme, under parametric assumptions about additive or multiplicative noise (Friston, 2008). These range from ordinary least squares to advanced variational deconvolution schemes. The resulting scheme is called *generalized filtering* or predictive coding (Friston et al., 2010a). In neural network terms, Equation (26) says that error-units receive predictions from the same level and the level above. Conversely, conditional expectations (encoded by the activity of state units) are driven by prediction errors from the same level and the level below. These constitute bottom–up and lateral messages that drive conditional expectations toward a better prediction to reduce the prediction error in the level below. This is the essence of recurrent message passing between hierarchical levels to optimize free energy or suppress prediction error: see Friston and Kiebel (2009a) for a more detailed discussion. In neurobiological implementations of this scheme, the sources of bottom–up prediction errors are thought to be superficial pyramidal cells that send forward connections to higher cortical areas. Conversely, predictions are conveyed from deep pyramidal cells, by backward connections, to target (polysynaptically) the superficial pyramidal cells encoding prediction error (Mumford, 1992; Friston and Kiebel, 2009a). **Figure 4** provides a schematic of the proposed message passing among hierarchically deployed cortical areas.

ACTION

In active inference, conditional expectations elicit behavior by sending top–down predictions down the hierarchy that are unpacked into proprioceptive predictions at the level of the cranial nerve nuclei and spinal-cord. These engage classical reflex arcs to suppress proprioceptive prediction errors and produce the predicted motor trajectory.

$$\dot{a} = -\frac{\partial}{\partial a} F = -\frac{\partial \tilde{s}}{\partial a} \cdot \xi_v^{(1)} \tag{27}$$

The reduction of action to classical reflexes follows because the only way that action can minimize free energy is to change sensory (proprioceptive) prediction errors by changing sensory signals; cf., the equilibrium point formulation of motor control (Feldman and Levin, 1995). In short, active inference can be regarded as equipping a generalized predictive coding scheme with classical reflex arcs: see Friston et al. (2009, 2010b) for details. The actual movements produced clearly depend upon top–down predictions that can have a rich and complex structure.

COUNTERFACTUAL PROCESSING

To optimize prior expectations about hidden controls it is necessary to identify those that maximize the salience. We will focus on visual searches and assume that competing (counterfactual) prior expectations are represented explicitly in a saliency map. In other words, we assume that salience is encoded on a grid corresponding to discrete values of competing prior expectations associated with different hidden control states. The maximum of this map defines the prior expectation with the greatest salience. This prior expectation enters the predictive coding in Equation (25). The salience of the *j*-th counterfactual prior expectation is, from Equations (21) and (22),

$$\begin{aligned}
\tilde{\eta}_u(t) &= \arg\max_{\tilde{\eta}_j} S(\tilde{\eta}_j) \\
S(\tilde{\eta}_j) &= \tfrac{1}{2} \ln |\partial_{\tilde{\mu}\tilde{\mu}} G(\tilde{\mu}_x(t+\tau), \tilde{\mu}_v(t+\tau), \tilde{\eta}_j)|
\end{aligned} \tag{28}$$

Given that we will be simulating visual searches with saccadic eye movements, we will consider the prior expectations to be updated at discrete times to simulate successive saccades, where the hidden control states correspond to locations in the visual scene that attract visual fixation.

SUMMARY

In summary, we have derived equations for the dynamics of perception and action using a free energy formulation of adaptive (Bayes-optimal) exchanges with the world and a generative model that is generic and biologically plausible. In what follows, we use Equations (26), (27), and (28) to simulate neuronal and behavioral responses. A technical treatment of the material above can be found in Friston et al. (2010a), which provides the details of the generalized Bayesian filtering scheme used to produce the simulations in the next section. The only addition to previous illustrations of this scheme is Equation (28), which maps conditional expectations about hidden states to prior expectations

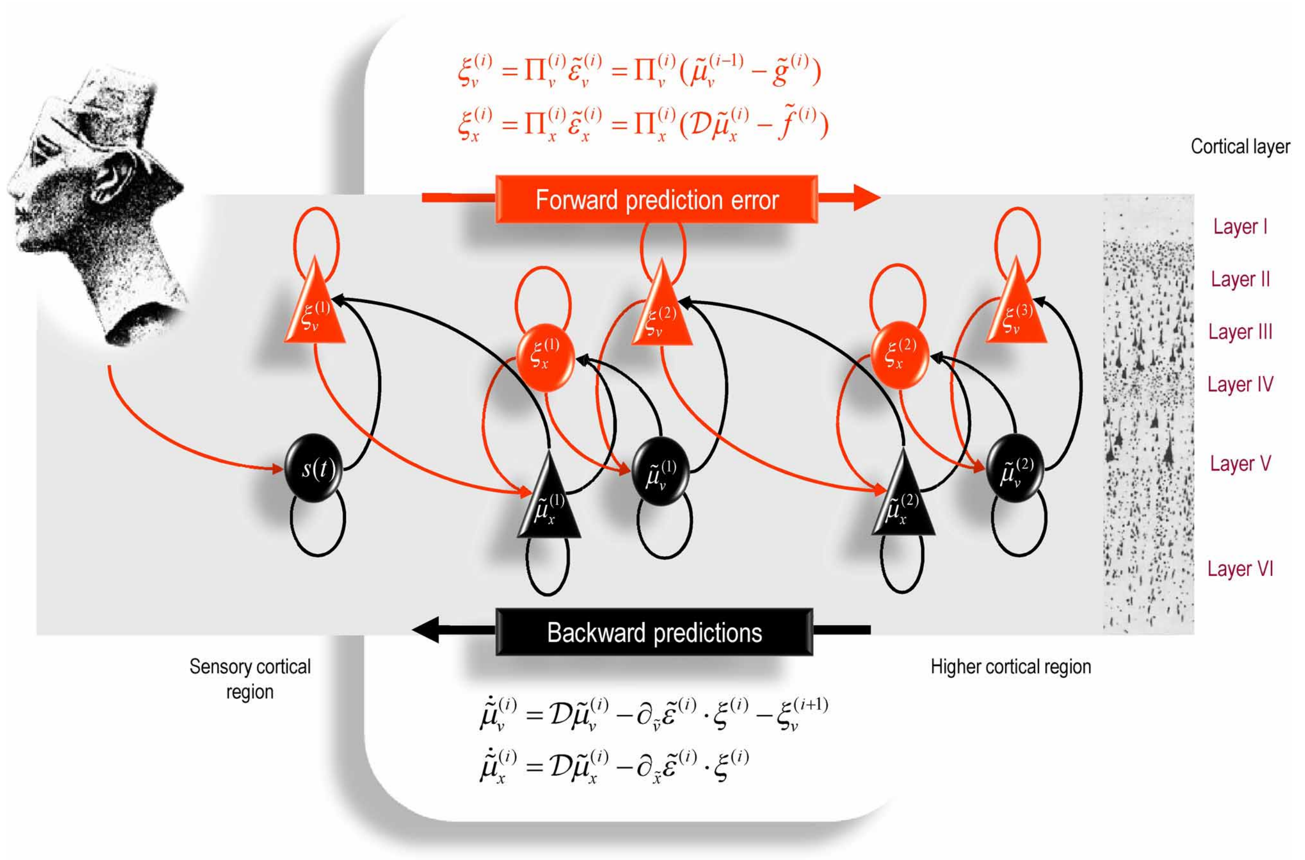

$\xi_v^{(i)} = \Pi_v^{(i)}\tilde{\varepsilon}_v^{(i)} = \Pi_v^{(i)}(\tilde{\mu}_v^{(i-1)} - \tilde{g}^{(i)})$
$\xi_x^{(i)} = \Pi_x^{(i)}\tilde{\varepsilon}_x^{(i)} = \Pi_x^{(i)}(\mathcal{D}\tilde{\mu}_x^{(i)} - \tilde{f}^{(i)})$
Cortical layer
Forward prediction error
Layer I
Layer II
Layer III
Layer IV
Layer V
Layer VI
$\xi_v^{(1)}$
$\xi_x^{(1)}$
$\xi_v^{(2)}$
$\xi_x^{(2)}$
$\xi_v^{(3)}$
$s(t)$
$\tilde{\mu}_x^{(1)}$
$\tilde{\mu}_v^{(1)}$
$\tilde{\mu}_x^{(2)}$
$\tilde{\mu}_v^{(2)}$
Backward predictions
Sensory cortical region
Higher cortical region
$\dot{\tilde{\mu}}_v^{(i)} = \mathcal{D}\tilde{\mu}_v^{(i)} - \partial_{\tilde{v}}\tilde{\varepsilon}^{(i)} \cdot \xi^{(i)} - \xi_v^{(i+1)}$
$\dot{\tilde{\mu}}_x^{(i)} = \mathcal{D}\tilde{\mu}_x^{(i)} - \partial_{\tilde{x}}\tilde{\varepsilon}^{(i)} \cdot \xi^{(i)}$

FIGURE 4 | Schematic detailing the neuronal architecture that might encode conditional expectations about the states of a hierarchical model. This shows the speculative cells of origin of forward driving connections that convey prediction error from a lower area to a higher area and the backward connections that construct predictions (Mumford, 1992). These predictions try to explain away prediction error in lower levels. In this scheme, the sources of forward and backward connections are superficial and deep pyramidal cells, respectively. The equations represent a generalized descent on free-energy under the hierarchical models described in the main text: see also Friston (2008). State-units are in black and error-units in red. Here, neuronal populations are deployed hierarchically within three cortical areas (or macro-columns). Within each area, the cells are shown in relation to cortical layers: supra-granular (I–III), granular (IV), and infra-granular (V and VI) layers. For simplicity, conditional expectations about control states had been absorbed into conditional expectations about hidden causes.

about hidden controls: it is this mapping that underwrites the sampling of salient features and appeals to the existence of hidden control states that action can change. Put simply, this formulation says that action fulfills predictions and we predict that the consequences of action (hidden control states) minimize our uncertainty about predictions.

MODELING SACCADIC EYE MOVEMENTS

This section illustrates the theory of the previous section, using simulations of sequential eye movements. Saccadic eye movements are a useful vehicle to illustrate active inference because they speak directly to visual search strategies and a wealth of psychophysical, neurobiological, and theoretical study (e.g., Grossberg et al., 1997; Ferreira et al., 2008; Srihasam et al., 2009; Bisley and Goldberg, 2010; Shires et al., 2010; Tatler et al., 2011; Wurtz et al., 2011). We will focus on a fairly simple paradigm—the categorization of faces—and therefore sidestep many of the deeper challenges of understanding visual searches.

THE GENERATIVE PROCESS

That first thing that we need to do is to define the processes generating sensory signals as a function of (hidden) states and action:

$$\begin{aligned}
s_p &= \mathbf{x}_p + \boldsymbol{\omega}_{v,p} \\
s_q &= g(I, \mathbf{x}_p) + \boldsymbol{\omega}_{v,q} \\
g_i &= I(d_{i,1} + \mathbf{x}_{p,1}, d_{i,2} + \mathbf{x}_{p,2}) \cdot h_i \qquad (29) \\
\dot{\mathbf{x}}_p &= a - \tfrac{1}{16}\mathbf{x}_p + \boldsymbol{\omega}_{x,p}
\end{aligned}$$

Note that these hidden states are true states that actually produce sensory signals. These have been written in boldface to distinguish them from the hidden states assumed by the generative model (see below). In these simulations, the world is actually very simple: sensory signals are generated in two modalities—proprioception and vision. Proprioception, $s_p \in \mathbb{R}^2$

reports the center of gaze or foveation as a displacement from the origin of some extrinsic frame of reference. Inputs in the visual modality comprise a list $s_q \in \mathbb{R}^{256}$ of values over an array of sensory channels sampling a two-dimensional image or visual scene $I : \mathbb{R}^2 \to \mathbb{R}$. This sampling uses a grid of 16 × 16 channels that samples a small part the image—representing a local high-resolution (foveal) sampling that constitutes an attentional focus. To make this sampling more biologically realistic, each channel was equipped with a center-surround receptive field that samples a local weighted average of the image. This provides an on-off center-surround sampling. Furthermore, the signals are modulated by a two-dimensional Hamming function—to model the loss of precise visual information from the periphery of the visual field.

The only hidden states in this generative process $\mathbf{x}_p \in \mathbb{R}^2$ are the center of oculomotor fixation, whose motion is driven by action and decays with a suitably long time constant of 16 time bins (were a time bin corresponds to 12 ms). In practice, the visual scene corresponds to a large grayscale image, where the i-th visual channel is sampled at location $d_i + \mathbf{x}_p \in \mathbb{R}^2$. Here, $d_i \in \mathbb{R}^2$ specifies the displacement of the i-th channel from the center of the sampling grid. The proprioceptive and visual signals were effectively noiseless, where there random fluctuations had a log-precision of 16. The motion of the fixation point was subject to low amplitude fluctuations with a log-precision of eight. This completes our description of the process generating proprioceptive and visual signals for any given action. We now turn to the model of this process that generates predictions and action.

THE GENERATIVE MODEL

The model of sensory signals used to specify variational free energy and consequent action (visual sampling) is slightly more complicated than the actual process generating data:

$$\begin{aligned} s_p &= x_p + \omega_{v,p} \\ s_q &= \sum_i \exp(x_{q,i}) g(I_i, x_p) + \omega_{v,q} \\ \dot{x}_p &= \tfrac{1}{4}(u - x_p) + \omega_{x,p} \\ \dot{x}_q &= 1 - \sum_i \exp(x_{q,i}) - \tfrac{1}{1024} x_q + \omega_{x,p} \end{aligned} \tag{30}$$

As above, proprioceptive signals are just a noisy mapping from hidden proprioceptive states encoding the direction of gaze. The visual input is modeled as a mixture of images sampled at a location specified by the proprioceptive hidden state. This hidden state decays with a time constant of four time bins (48 ms) toward a hidden control state. In other words, the hidden control determines the location that attracts gaze.

The visual input depends on a number of hypotheses or internal images $I_i : \mathbb{R}^2 \to \mathbb{R} : i \in \{1, \ldots N\}$ that constitute the agent's prior beliefs about what could cause its visual input. In this paper, we use $N = 3$ hypotheses. The input encountered at any particular time is a weighted mixture of these internal images, where the weights correspond to hidden perceptual states. The dynamics of these perceptual states (last equality above) implement a form of dynamic softmax—in the sense that the solution of their equations of motion ensures the weights sum (approximately) to one:

$$\dot{x}_q = 0 \Rightarrow \sum_i \exp(x_{q,i}) \approx 1 \tag{31}$$

This means we can interpret $\exp(x_{q,i})$ as the (softmax) probability that the i-th internal image or hypothesis is the cause of visual input. The decay term (with a time constant of 512 time bins) just ensures that perceptual states decay slowly to the same value, in the absence of perceptual fluctuations.

In summary, given hidden proprioceptive and perceptual states the agent can predict its proprioceptive and visual input. The generative model is specified by Equation (17) and the precision of the random fluctuations that determine the agent's prior certainty about sensory inputs and the motion of hidden states. In the examples below, we used a log-precision of eight for proprioceptive sensations and the motion of hidden states. We let the agent believe its visual input was fairly noisy, with a log-precision of four. In practice, this means it is more likely to change its (less precise) posterior beliefs about the causes of visual input to reduce prediction error, as opposing to adjusting its (precise) posterior beliefs about where it is looking.

PRIORS AND SALIENCY

To simulate saccadic eye movements, we integrated the active inference scheme for 16 time bins (196 ms) and then computed a map of salience to reset the prior expectations about the hidden control states that attract the center of gaze. Salience was computed for 1024 = 32 × 32 locations distributed uniformly over the visual image or scene. The prior expectation of the hidden control state was the location that maximized salience, according to Equation (28). The ensuing salience over the 32 × 32 locations constitutes a salience map that drives the next saccade. Notice that salience is a function of, and only of, fictive beliefs about the state of the world and essentially tells the agent where to look next.

Figure 5 provides a simple illustration of salience based upon the posterior beliefs or hypothesis that local (foveal) visual inputs are caused by an image of Nefertiti. The left panels summaries the classic results of the Yarbus (1967); in terms of a stimulus and the eye movements it elicits. The right panels depict visual input after sampling the image on the right with center-surround receptive fields and the associated saliency map based on a local sampling of 16 × 16 pixels, using Equation (21). Note how the receptive fields suppress absolute levels of luminance contrast and highlight edges. It is these edges that inform posterior beliefs about the content of the visual scene and where it is being sampled. This information reduces conditional uncertainty and is therefore salient. The salient features of the image include the ear, eye, and mouth. The location of these features and a number of other salient locations appear to be consistent with the locations that attract saccadic eye movements (as shown on the right). Crucially, the map of salience extends well beyond the field of view (circle on the picture). This reflects the fact that salience is

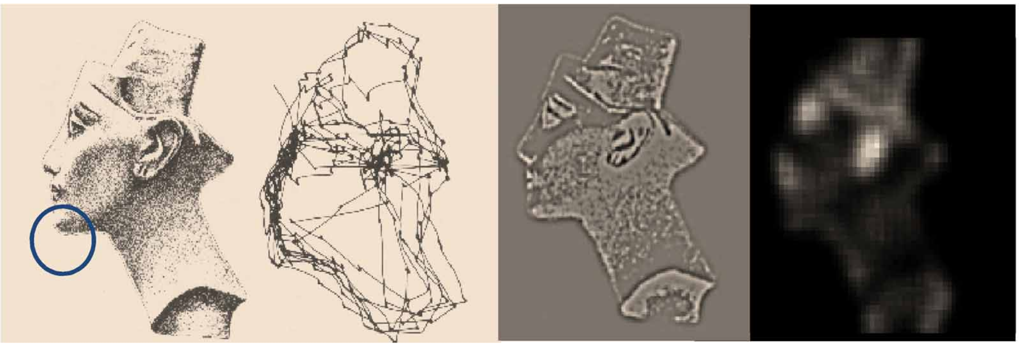


FIGURE 5 | This provides a simple illustration of salience based upon the posterior beliefs or hypothesis that local (foveal) visual inputs are caused by an image of Nefertiti. The **left panels** summaries the classic results of the Yarbus; in terms of a stimulus and the eye movements it elicits. The **right panels** depict visual input after sampling the image on the right (using conventional center surround receptive fields) and the associated saliency map based on a local sampling of 16 × 16 pixels, using the generative model described in the main text. The size of the resulting field of view, in relation to the visual scene, is indicated with the circle on the left image. The key thing to note here is that the salient features of the image include the ear, eye, and mouth. The location of these features and other salient locations appear to be consistent with the locations that attract saccadic eye movements (as shown on the right).

not an attribute of what is seen, but what might be seen under a particular hypothesis about the causes of sensations.

To make the simulations a bit more realistic, we added a further prior implementing inhibition of return (Itti and Koch, 2001; Wang and Klein, 2010). This involved suppressing the salience of locations that have been recently foveated, using the following scheme:

$$\begin{aligned}\mathbf{S}_k &= S_k - (S_k \times R_{k-1}) \\ R_k &= \rho(\mathbf{S}_k) + \tfrac{1}{2}R_{k-1}\end{aligned} \tag{32}$$

Here, $S_k = S(\tilde{\eta}_j) - \min(S(\tilde{\eta}_j))$ is the differential salience for the k-th saccade and R_k is an inhibition of return map that remembers recently foveated locations. This map reduces the salience of previous locations if they were visited recently. The function $\rho(\mathbf{S}_k) \in [0, 1]$ is a Gaussian function (with a standard deviation of 1/16 of the image size) of the distance from the location of maximum salience that attracts the k-th saccade. The addition of inhibition of return ensures that a new location is selected by each saccade and can be motivated ethologically by prior beliefs that the visual scene will change and that previous locations should be revisited.

FUNCTIONAL ANATOMY

Figure 6 provides an intuition as to how active inference under salience priors might be implemented in the brain. This schematic depicts a particular instance of the message passing scheme in **Figure 4**, based on the generative model above. This model prescribes a hierarchical form for generalized predictive coding; shown here in terms of state and error units (black and red, denoting deep and superficial pyramidal cell populations, respectively) that have been assigned to different cortical or subcortical regions. The insert on the left shows a visual scene (a picture of Nefertiti) that can be sampled locally by foveating a particular point—the true hidden state of the world. The resulting visual input arrives in primary visual cortex to elicit prediction errors that are passed forward to "what" and "where" streams (Ungerleider and Mishkin, 1982). State units in the "what" stream respond by adjusting their representations to provide better predictions based upon a discrete number of internal images or hypotheses. Crucially, the predictions of visual input depend upon posterior beliefs about the direction of gaze, encoded by the state units in the "where" stream (Bisley and Goldberg, 2010). These posterior expectations are themselves informed by top–down prior beliefs about the direction of gaze that maximizes salience. The salience map shown in the center is updated between saccades based upon conditional expectations about the content of the visual scene. Conditional beliefs about the direction of gaze provide proprioceptive predictions to the oculomotor system in the superior colliculus and pontine nuclei, to elaborate a proprioceptive prediction error (Grossberg et al., 1997; Shires et al., 2010; Shen et al., 2011). This prediction error drives the oculomotor system to fulfill posterior beliefs about where to look next. This can be regarded as an instance of the classical reflects arc, whose set point is determined by top–down proprioceptive predictions. The anatomical designations should not be taken seriously (for example, the salience map may be assembled in the pulvinar or frontal cortex and mapped to the deep layer of the superior colliculus). The important thing to take from this schematic is the functional logic implied by the anatomy that involves reciprocal message passing and nested loops in a hierarchical architecture that is not dissimilar to circuits in the real brain. In particular, note that representations of hidden perceptual states provide bilateral top–down projections to early visual system is (to predict visual input) and to the systems computing salience, which might involve the pulvinar of the thalamus (Wardak et al., 2011; Wurtz et al., 2011).

SIMULATING SACCADIC EYE MOVEMENTS

Figure 7 shows the results of a simulated visual search, in which the agent had three internal images or hypotheses about the scene it might sample (an upright face, an inverted face, and a rotated

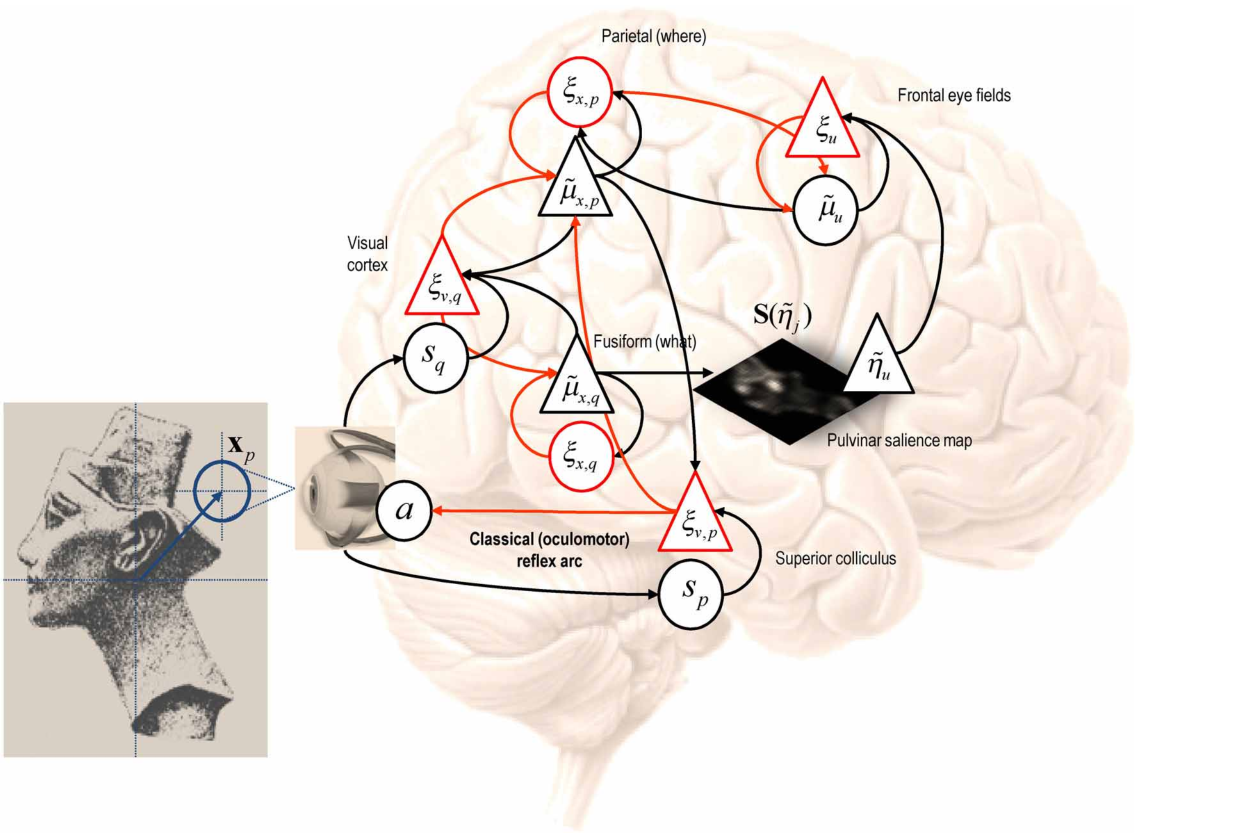


FIGURE 6 | This schematic depicts a particular instance of the message passing scheme in Figure 4. This example follows from the generative model of visual input described in the main text. The model prescribes a particular hierarchical form for generalized predictive coding; shown here in terms of state and error units (black and red, respectively) that have been assigned to different cortical or subcortical regions. The insert on the left shows a visual scene (a picture of Nefertiti) that can be sampled locally by foveating a particular point—the true hidden state of the world. The resulting visual input arrives in primary visual cortex to elicit prediction errors that are passed forward to what and where streams. State units in the "what" stream respond by adjusting their representations to provide better predictions based upon a discrete number of internal images or hypotheses. Crucially, the predictions of visual input depend upon posterior beliefs about the direction of gaze encoded by state units in the "where" stream. These conditional expectations are themselves informed by top–down prior beliefs about the direction of gaze that maximizes salience. The salience map shown in the center is updated between saccades based upon posterior beliefs about the content of the visual scene. Posterior beliefs about the content of the visual scene provide predictions of visual input and future hidden states subtending salience. Posterior beliefs about the direction of gaze are used to form predictions of visual input and provide proprioceptive predictions to the oculomotor system in the superior colliculus and pontine nuclei, to elaborate a proprioceptive prediction error. This prediction error drives the oculomotor system to fulfill posterior beliefs about where to look next. This can be regarded as an instance of the classical reflects arc, whose set point is determined by top–down proprioceptive predictions. The variables associated with each region are described in detail in the text, while the arrows connecting regions adopt same format as in **Figure 4** (forward prediction error afferents in red and backward predictions in black).

face). The agent was presented with an upright face and its posterior expectations were evaluated over 16 (12 ms) time bins, after which salience was evaluated. The agent then emitted a saccade by foveating the most salient location during the subsequent 16 time bins—from its starting location (the center of the visual field). This was repeated for eight saccades. The upper row shows the ensuing eye movements as red dots (in the extrinsic coordinates of the true scene) at the fixation point of each saccade. The corresponding sequence of eye movements are shown in the insert on the upper left, where the red circles correspond roughly to the agent's field of view. These saccades were driven by prior beliefs about the direction of gaze based upon the salience maps in the second row. Note that these maps change with successive saccades as posterior beliefs about the hidden perceptual states become progressively more confident. Note also that salience is depleted in locations that were foveated in the previous saccade—this reflects the inhibition of return. Posterior beliefs about hidden states provide visual and proprioceptive predictions that suppress visual prediction errors and drive eye movements, respectively. Oculomotor responses are shown in the third row in terms of the two hidden oculomotor states corresponding to vertical and horizontal displacements. The portions of the image sampled (at the end of each saccade) are shown in the fourth row (weighted by the Hamming function above). The final two rows show the posterior beliefs in terms of their sufficient statistics (penultimate row) and the perceptual categories (last row), respectively. The posterior beliefs are plotted here in terms of posterior expectations and 90% confidence interval about the true stimulus. The key thing

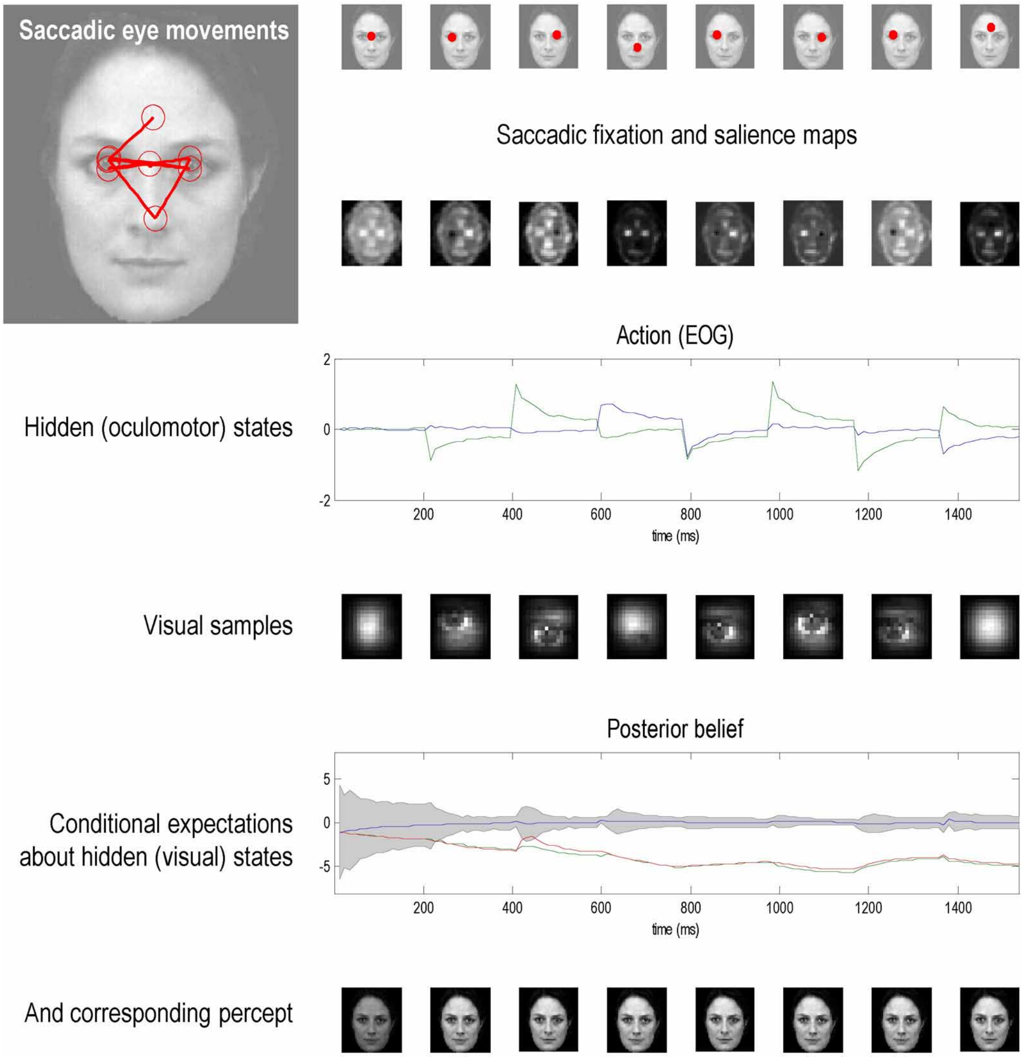


FIGURE 7 | This figure shows the results of a simulation, in which a face was presented to an agent, whose responses were simulated using the optimal inference scheme described in the main text. In this simulation, the agent had three internal images or hypotheses about the stimuli it might sample (an upright face, an inverted face, and a rotated face). The agent was presented with an upright face and its conditional expectations were evaluated over 16 (12 ms) time bins until the next saccade was emitted. This was repeated for eight saccades. The ensuing eye movements are shown as red dots at the location (in extrinsic coordinates) at the end of each saccade in the upper row. The corresponding sequence of eye movements is shown in the insert on the upper left, where the red circles correspond roughly to the proportion of the image sampled. These saccades are driven by prior beliefs about the direction of gaze based upon the saliency maps in the second row. Note that these maps change with successive saccades as posterior beliefs about the hidden states, including the stimulus, become progressively more confident. Note also that salience is depleted in locations that were foveated in the previous saccade. These posterior beliefs provide both visual and proprioceptive predictions that suppress visual prediction errors and drive eye movements, respectively. Oculomotor responses are shown in the third row in terms of the two hidden oculomotor states corresponding to vertical and horizontal displacements. The associated portions of the image sampled (at the end of each saccade) are shown in the fourth row. The final two rows show the posterior beliefs in terms of their sufficient statistics and the stimulus categories, respectively. The posterior beliefs are plotted here in terms of conditional expectations and the 90% confidence interval about the true stimulus. The key thing to note here is that the expectation about the true stimulus supervenes over its competing expectations and, as a result, conditional confidence about the stimulus category increases (the confidence intervals shrink to the expectation). This illustrates the nature of evidence accumulation when selecting a hypothesis or percept the best explains sensory data.

to note here is that the expectation about the true stimulus supervenes over its competing representations and, as a result, posterior confidence about the stimulus category increases (the posterior confidence intervals shrink to the expectation): see Churchland et al. (2011) for an empirical study of this sort phenomena. The images in the lower row depict the hypothesis selected; their intensity has been scaled to reflect conditional uncertainty, using the entropy (average uncertainty) of the softmax probabilities.

This simulation illustrates a number of key points. First, it illustrates the nature of evidence accumulation in selecting a hypothesis or percept the best explains sensory data. One can see that this proceeds over two timescales; both within and between saccades. Within-saccade accumulation is evident even during the initial fixation, with further stepwise decreases in uncertainty as salient information is sampled. The within-saccade accumulation is formally related to evidence accumulation as described in models of perceptual discrimination (Gold and Shadlen, 2003; Churchland et al., 2011). This is reflected in the progressive elevation of the correct perceptual state above its competitors and the consequent shrinking of the posterior confidence interval. The transient changes in the posterior beliefs, shortly after each saccade, reflect the fact that new data are being generated as the eye sweeps toward its new target location. It is important to note that the agent is not just predicting visual contrast, but also how contrast changes with eye movements—this induces an increase in conditional uncertainty (in generalized coordinates of motion) during the fast phase of the saccade. However, due to the veracity of the posterior beliefs, the conditional confidence shrinks again when the saccade reaches its target location. This shrinkage is usually to a smaller level than in the previous saccade.

This illustrates the second key point; namely, the circular causality that lies behind perception. Put simply, the only hypothesis that can endure over successive saccades is the one that correctly predicts the salient features that are sampled. This sampling depends upon action or an embodied inference that speaks directly to the notion of active vision or visual palpation (O'Regan and Noë, 2001; Wurtz et al., 2011). This means that the hypothesis prescribes its own verification and can only survive if it is a correct representation of the world. If its salient features are not discovered, it will be discarded in favor of a better hypothesis. This provides a nice perspective on perception as hypothesis testing, where the emphasis is on the selective processes that underlie sequential testing. This is particularly pertinent when hypotheses can make predictions that are more extensive than the data available at any one time.

Finally, although the majority of saccades target the eyes and nose, as one might expect, there is one saccade to the forehead. This is somewhat paradoxical, because the forehead contains no edges and cannot increase posterior confidence about a face. However, this region is highly informative under the remaining two hypotheses (corresponding to the location of the nose in the inverted face and the left eye in the rotated face). This subliminal salience is revealed through inhibition of return and reflects the fact that the two competing hypotheses have not been completely excluded. This illustrates the competitive nature of perceptual selection induced by inhibition of return and can regarded, heuristically, as occasional checking of alternative hypotheses. This is a bit like a scientist who tries to refute his hypothesis by acquiring data that furnish efficient tests of his competing or null hypotheses.

CONCLUSION

This ideas reviewed in this paper suggest that the reward or cost-functions that underlie value in conventional (normative) models of optimal control can be cast as prior beliefs about future states, which are disclosed through active inference. In this setting, value becomes the evidence for generative models of our world—and valuable behavior is nothing more or less than accumulating evidence for our embodied models, through Bayesian updating of posterior beliefs. Subsequently, we saw that prior beliefs about future states are simply those that minimize the uncertainty of posterior beliefs. In this general formulation, we can understand exploration of the sensorium in terms of optimality principles based on ergodic or homoeostatic principles. In other words, to maintain the constancy of our external milieu, it is sufficient to expose ourselves to predicted and predictable stimuli. Being able to predict current observations also enables us to predict fictive sensations that we could experience from another viewpoint; where the best viewpoint is the one that confirms our predictions with the greatest precision or certainty. In short, action fulfills our predictions, while we predict the consequences of our actions will minimize uncertainty about those predictions. This provides a principled way in which to sample the world; for example, with visual searches using saccadic eye movements. These theoretical considerations are remarkably consistent with a number of compelling heuristics; most notably the Infomax principle or the principle of minimum redundancy, signal detection theory and formulations of salience in terms of Bayesian surprise.

An interesting perspective on active inference and embodied perception emerges from these considerations, in which percepts are selected through a form of circular causality: in other words, only the correct perceptual hypothesis can survive the cycle of action and perception, when the percept is used to predict where to look next. If the true state of the world and the current hypothesis concur, then the percept can maintain itself by selectively sampling evidence for its own existence. This provides an embodied (enactivist) explanation for perception that fits comfortably with the notion of visual sniffing or palpation (O'Regan and Noë, 2001; Wurtz et al., 2011). Furthermore, it resonates with neurodynamic accounts of self-generated behavior in a robotics context (Namikawa et al., 2011).

The arguments in this paper have been inspired by developments in theoretical neurobiology and machine learning. However, it is interesting to consider parallel developments in neurorobotics. Two decades ago most neurorobotics employed simple architectures with sensory-motor mappings implemented by perceptron-type networks and supervised learning; for example, the supervised learning of driving skills in robot cars (Pomerleau, 1991). In principle, active inference provides a formalism to revisit these sorts of problems using self-supervised schemes based upon deep hierarchical models. The usefulness of hierarchical schemes has been demonstrated by Morimoto and Doya, who show how a robot can stand up using hierarchical reinforcement learning (Morimoto and Doya, 2001). Furthermore, the idea of forward (predictive) modeling is now established in neurorobotics: Schaal (1997) has shown how learning a predictive forward model is beneficial in imitation learning, while Tani and Nolfi (1999) show how prediction error can be used to recognize self-generated behavior using a hierarchically organized mixture of predictive expert networks. There are clear parallels here with active inference under hierarchical generative (forward) models that suggest a theoretical convergence of neurobiology and neurorobotics. One can imagine exploiting the fairly simple and principled optimization schemes provided by free energy minimization to elaborate robots with deep hierarchical models, were these models that generally entail a separation of temporal scales and context sensitive behavior. On a more general note, active inference may provide a formal framework that connects the compelling work in neurorobotics on imitation and action observation to some of the highest level questions that currently preoccupy psychologists and cognitive neuroscientists—particularly those people interested in psychopathology and its mechanistic underpinnings.

The treatment of optimality in this paper has focused on the nature of value and its relationship to evidence. There are many other important issues that we have glossed over; such as the acquisition or learning of models. For example, as noted by one of our reviewers: "Many traditional (alternate) methods would be capable of arriving at optimal policies despite limitations in the model, owing to the properties of the approximation procedures. In the authors' proposal, the underlying generative model would need to capture the necessary dynamics through the definition of the priors and model structure (which the authors note may be learnt separately at a higher level). Do we know that this internal model can be learnt, in a tractable form given what can be known about the task? Do we know if the solutions to the two cases will be similar?"

In one sense, traditional methods are not necessarily alternative methods, because optimal policies can be cast as prior beliefs. In other words, the current framework just allows one to convert optimal control problems into pure inference problems. The motivation for this is to understand where prior beliefs (optimal policies) come from in a hierarchical setting. The hierarchical aspect is important because this necessarily induces empirical priors, which means that cost functions can themselves be optimized in relation to model evidence. This is illustrated nicely in the context of learning and model selection: a fuller treatment would show that the parameters of any given model can be learned in a Bayes optimal fashion by minimizing variational free energy (Friston, 2008). Furthermore, the model itself can also be optimized with respect to variational free energy, in exactly the same way that Bayesian model selection operates in data analysis. This hierarchical optimization may provide a nice metaphor for understanding selection at a neurodevelopmental or evolutionary timescale (Friston et al., 2006). Crucially, because we are dealing with approximate Bayesian inference, the models selected will necessarily be approximations and provide the simplest (most parsimonious) explanations for sampled outcomes. In answer to the reviewer's questions, any extant phenotype is an existence proof that its particular (approximate) model can be learnt. The question about the uniqueness of models is a bit more subtle—in the sense that (in active inference) models create their own data. This means that each phenotype may be a uniquely optimal model for its own sensorium but not that of another phenotype. These are clearly very important issues, which motivate the work reviewed in this paper.

The ideas described in this paper try to go beyond the formal similarity between optimal control and Bayesian inference schemes to suggest that optimal control is a special case of Bayes-optimal inference and that inference is the hard problem. In this setting, optimality reduces to sampling states prescribed by the priors of a generative model that specifies state transitions. So what are the practical advantages of casting optimal control as inference? In Friston et al. (2012b) we summarized the advantages of active inference as providing:

- A tractable approximate solution to any stochastic, nonlinear optimal control problem to the extent that standard (variational) Bayesian procedures exist. Variational or approximate Bayesian inference is well-established in statistics and data assimilation because it finesses many of the computational problems associated with exact Bayesian inference.
- The opportunity to learn and infer environmental constraints in a Bayes-optimal fashion; particularly the parameters of equations of motion and amplitudes of observation and hidden state noise.
- The formalism to handle system or state noise: currently, optimal control schemes are restricted to stochastic control (i.e., random fluctuations on control as opposed to hidden states). One of the practical advantages of active inference is that fluctuations in hidden states are modeled explicitly, rendering control robust to exogenous perturbations.
- The specification of control costs in terms of priors on control, with an arbitrary form: currently, most approximate stochastic optimal control schemes are restricted to quadratic control costs. In classical schemes that appeal to path integral solutions there are additional constraints that require control costs to be a function of the precision of control noise; e.g., Theodorou et al. (2010) and Braun et al. (2011). These constraints are not necessary in active inference.

The disadvantage of active inference is that one cannot prescribe optimality in terms of cost functions, because (Bayes) optimal behavior rests on a generative model that is specified by its likelihood and prior functions. Having said this, for every Bayes-optimal policy there is an associated cost function (Friston and Ao, 2012). Perhaps the most important advantage of active inference—for practical applications—is its simplicity and robustness. It simplicity stems from the fact that one only has to specify desired movements or trajectories in terms of prior beliefs (equations of motion in the generative model) as opposed to desired endpoints of movement (which requires the solution of a generally intractable optimal control problem). The robustness follows from the context sensitivity of active inference schemes and their ability to handle unpredicted (random) fluctuations or indeed changes in the motor plant—see Friston et al. (2010b). Finally, treating control problems as inference problems allows one to exploit the advances made in approximate Bayesian inference and model selection. A nice example here would be the hierarchal optimization of control architectures using Bayesian model selection and free energy as an approximation to log model evidence. This strategy is now used routinely to select among thousands of models within a few seconds (Friston and Penny, 2011) but has only been applied in a data analysis setting. In principle, these Bayesian procedures could also be used in a control setting.

In summary, we have tried to formalize the intuitive notion that our interactions with the world are akin to sensory experiments, by which we confirm our hypotheses about its causal structure in an optimal and efficient fashion. This mandates prior beliefs that the deployment of sensory epithelia and our physical relationship to the world will disclose its secrets—beliefs that are fulfilled by action. The resulting active or embodied inference means that not only can we regard perception as hypothesis testing, but we could regard action as performing experiments that confirm or disconfirm those hypotheses.

ACKNOWLEDGMENTS

This work was funded by the Wellcome trust. We would like to thank our three reviewers for very helpful comments and guidance.

REFERENCES

Ashby, W. R. (1947). Principles of the self-organizing dynamic system. *J. Gen. Psychol.* 37, 125–128.

Barlow, H. (1961). "Possible principles underlying the transformations of sensory messages," in *Sensory Communication*, ed W. Rosenblith (Cambridge, MA: MIT Press), 217–234.

Barlow, H. B. (1974). Inductive inference, coding, perception, and language. *Perception* 3, 123–134.

Baxter, J., Bartlett, P. L., and Weaver, L. (2001). Experiments with infinite-horizon, policy-gradient estimation. *J. Artif. Intell. Res.* 15, 351–381.

Beal, M. J. (2003). *Variational Algorithms for Approximate Bayesian Inference.* Ph.D. thesis, University College London.

Bellman, R. (1952). On the theory of dynamic programming. *Proc. Natl. Acad. Sci. U.S.A.* 38, 716–719.

Bialek, W., Nemenman, I., and Tishby, N. (2001). Predictability, complexity, and learning. *Neural Comput.* 13, 2409–2463.

Birkhoff, G. D. (1931). Proof of the ergodic theorem. *Proc. Natl. Acad. Sci. U.S.A.* 17, 656–660.

Bisley, J. W., and Goldberg, M. E. (2010). Attention, intention, and priority in the parietal lobe. *Annu. Rev. Neurosci.* 33, 1–21.

Botvinick, M. M., and An, J. (2008). "Goal-directed decision making in prefrontal cortex: a computational framework," in *Advances in Neural Information Processing Systems (NIPS)*, eds D. Koller, Y. Y. Bengio, D. Schuurmans, L. Bouttou, and A. Culotta, 21.

Braun, D., Ortega, P., Theodorou, E., and Schaal, S. (2011). *Path Integral Control and Bounded Rationality.* Paris: ADPRL.

Brown, L. D. (1981). A complete class theorem for statistical problems with finite sample spaces. *Ann. Statist.* 9, 1289–1300.

Camerer, C. F. (2003). Behavioural studies of strategic thinking in games. *Trends Cogn. Sci.* 7, 225–231.

Canolty, R. T., Edwards, E., Dalal, S. S., Soltani, M., Nagarajan, S. S., Kirsch, H. E., et al. (2006). High gamma power is phase-locked to theta oscillations in human neocortex. *Science* 313, 1626–1628.

Churchland, A. K., Kiani, R., Chaudhuri, R., Wang, X. J., Pouget, A., and Shadlen, M. N. (2011). Variance as a signature of neural computations during decision making. *Neuron* 69, 818–831.

Cooper, G. (1988). "A method for using belief networks as influence diagrams," in *proceedings of the Conference on Uncertainty in Artificial Intelligence,* 55–63.

Daw, N. D., and Doya, K. (2006). The computational neurobiology of learning and reward. *Curr. Opin. Neurobiol.* 16, 199–204.

Dayan, P., and Daw, N. D. (2008). Decision theory, reinforcement learning, and the brain. *Cogn. Affect. Behav. Neurosci.* 8, 429–453.

Dayan, P., and Hinton, G. E. (1997). Using expectation maximization for reinforcement learning. *Neural Comput.* 9, 271–278.

Dayan, P., Hinton, G. E., and Neal, R. (1995). The Helmholtz machine. *Neural Comput.* 7, 889–904.

Feldman, A. G., and Levin, M. F. (1995). The origin and use of positional frames of reference in motor control. *Behav. Brain Sci.* 18, 723–806.

Feldman, H., and Friston, K. J. (2010). Attention, uncertainty, and free-energy. *Front. Hum. Neurosci.* 4:215. doi: 10.3389/fnhum.2010.00215

Ferreira, F., Apel, J., and Henderson, J. M. (2008). Taking a new look at looking at nothing. *Trends Cogn. Sci.* 12, 405–410.

Feynman, R. P. (1972). *Statistical Mechanics.* Reading, MA: Benjamin.

Filatov, N., and Unbehauen, H. (2004). *Adaptive Dual Control: Theory and Applications.* Lecture Notes in Control and Information Sciences. Berlin: Springer Verlag.

Fox, C., and Roberts, S. (2011). "A tutorial on variational Bayes," in *Artificial Intelligence Review.* Spinger. doi: 10.1007/s10462-011-9236-8

Friston, K. (2008). Hierarchical models in the brain. *PLoS Comput. Biol.* 4:e1000211. doi: 10.1371/journal.pcbi.1000211

Friston, K. (2009). The free-energy principle: a rough guide to the brain? *Trends Cogn. Sci.* 13, 293–301.

Friston, K. (2010). The free-energy principle: a unified brain theory? *Nat. Rev. Neurosci.* 11, 127–138.

Friston, K. (2011). What is optimal about motor control? *Neuron* 72, 488–498.

Friston, K., Adams, R. A., Perrinet, L., and Breakspear, M. (2012a). Perceptions as hypotheses: saccades as experiments. *Front. Psychology.* 3:151. doi: 10.3389/fpsyg.2012.00151

Friston, K., Samothrakis, S., and Montague, R. (2012b). Active inference and agency: optimal control without cost functions. *Biol. Cybern.* 106, 523–541.

Friston, K., and Ao, P. (2012). Free-energy, value and attractors. *Comput. Math. Methods Med.* 2012, 937860.

Friston, K., and Kiebel, S. (2009a). Cortical circuits for perceptual inference. *Neural Netw.* 22, 1093–1104.

Friston, K. J., and Kiebel, S. J. (2009b). Predictive coding under the free-energy principle. *Philos. Trans. R. Soc. Lond. B. Biol. Sci.* 364, 1211–1221.

Friston, K., Kilner, J., and Harrison, L. (2006). A free energy principle for the brain. *J. Physiol. Paris* 100, 70–87.

Friston, K., Mattout, J., and Kilner, J. (2011). Action understanding and active inference. *Biol. Cybern.* 104, 137–160.

Friston, K., and Penny, W. (2011). *Post hoc* Bayesian model selection. *Neuroimage* 56, 2089–2099.

Friston, K., Stephan, K., Li, B., and Daunizeau, J. (2010a). Generalised filtering. *Math. Probl. Eng.* 2010, 621670.

Friston, K. J., Daunizeau, J., Kilner, J., and Kiebel, S. J. (2010b). Action and behavior: a free-energy formulation. *Biol. Cybern.* 102, 227–260.

Friston, K. J., Daunizeau, J., and Kiebel, S. J. (2009). Active inference or reinforcement learning? *PLoS ONE* 4:e6421. doi: 10.1371/journal.pone.0006421

Friston, K. J., Tononi, G., Reeke, G. N., Sporns, O., and Edelman, G. M. (1994). Value-dependent selection in the brain: simulation in a synthetic neural model. *Neuroscience* 59, 229–243.

Gibson, J. J. (1979). *The Ecological Approach to Visual Perception.* Boston, MA: Houghton Mifflin.

Gigerenzer, G., and Gaissmaier, W. (2011). Heuristic decision making. *Annu. Rev. Psychol.* 62, 451–482.

Ginzburg, V. L., and Landau, L. D. (1950). On the theory of superconductivity. *Zh. Eksp. Teor. Fiz.* 20, 1064.

Gold, J. I., and Shadlen, M. N. (2003). The influence of behavioral context on the representation of a perceptual decision in developing oculomotor commands. *J. Neurosci.* 23, 632–651.

Gomez, F., and Miikkulainen, R. (2001). *Learning Robust Nonlinear Control with Neuroevolution.* Technical Report AI01-292, The University of Texas at Austin, Department of Computer Sciences.

Gomez, F., Schmidhuber, J., and Miikkulainen, R. (2009). Accelerated neural evolution through cooperatively coevolved synapses. *J. Mach. Learn. Res.* 9, 937–965.

Gregory, R. L. (1980). Perceptions as hypotheses. *Philos. Trans. R. Soc. Lond. B Biol. Sci.* 290, 181–197.

Grossberg, S., Roberts, K., Aguilar, M., and Bullock, D. (1997). A neural model of multimodal adaptive saccadic eye movement control by superior colliculus. *J. Neurosci.* 17, 9706–9725.

Haken, H. (1983). *Synergetics: An introduction. Non-equilibrium Phase Transition and Self-Organisation in Physics, Chemistry and Biology, 3rd Edn.* Berlin: Springer Verlag.

Helmholtz, H. (1866/1962). "Concerning the perceptions in general," in *Treatise on Physiological*

Optics. Vol. III, 3rd Edn, ed J. Southall, Trans. (New York, NY: Dover).

Hinton, G. E., and van Camp, D. (1993). "Keeping neural networks simple by minimizing the description length of weights." in *Proceedings of COLT-93*, 5–13.

Hoffman, M., de Freitas, N., Doucet, A., and Peters, J. (2009). "An expectation maximization algorithm for continuous markov decision processes with arbitrary rewards," in *Twelfth International Conference on Artificial Intelligence and Statistics (AISTATS 2009),* 232–239.

Howard, R. A. (1960). *Dynamic Programming and Markov Processes.* Cambridge, MA: MIT Press.

Humphreys, G. W., Allen, H. A., and Mavritsaki, E. (2009). Using biologically plausible neural models to specify the functional and neural mechanisms of visual search. *Prog. Brain Res.* 176, 135–148.

Itti, L., and Baldi, P. (2009). Bayesian surprise attracts human attention. *Vis. Res.* 49, 1295–1306.

Itti, L., and Koch, C. (2001). Computational modelling of visual attention. *Nat. Rev. Neurosci.* 2, 194–203.

Jaynes, E. T. (1957). Information theory and statistical mechanics. *Phys. Rev. Ser. II* 106, 620–630.

Jensen, F., Jensen, V., and Dittmer, S. L. (1994). "From influence diagrams to junction trees," in *Proceedings of the Tenth Conference on Uncertainty in Artificial Intelligence,* Morgan Kaufmann, 367–373.

Kaelbling, L. P., Littman, M. L., and Cassandra, A. R. (1998). Planning and acting in partially observable stochastic domains. *Artif. Intell.* 101, 99–134.

Kappen, H. (2005a). Path integrals and symmetry breaking for optimal control theory. *J. Stat. Mech. Theory Exp.* 11, P11011.

Kappen, H. J. (2005b). Linear theory for control of nonlinear stochastic systems. *Phys. Rev. Lett.* 95, 200201.

Kiebel, S. J., Daunizeau, J., and Friston, K. J. (2009). Perception and hierarchical dynamics. *Front. Neuroinform.* 3:20. doi: 10.3389/neuro.11.020.2009

Linsker, R. (1990). Perceptual neural organization: some approaches based on network models and information theory. *Annu. Rev. Neurosci.* 13, 257–281.

MacKay, D. J. (1995). Free-energy minimisation algorithm for decoding and cryptoanalysis. *Electron. Lett.* 31, 445–447.

McKinstry, J. L., Edelman, G. M., and Krichmar, J. L. (2006). A cerebellar model for predictive motor control tested in a brain-based device. *Proc. Natl. Acad. Sci. U.S.A.* 103, 3387–3392.

Morimoto, J., and Doya, K. (2001). Acquisition of stand-up behavior by a real robot using hierarchical reinforcement learning. *Rob. Auton. Syst.* 36, 37–51.

Mumford, D. (1992). On the computational architecture of the neocortex. II. *Biol. Cybern.* 66, 241–251.

Namikawa, J., Nishimoto, R., and Tani, J. (2011). A neurodynamic account of spontaneous behaviour. *PLoS Comput. Biol.* 7:e1002221. doi: 10.1371/journal.pcbi.1002221

Neal, R. M., and Hinton, G. E. (1998). "A view of the EM algorithm that justifies incremental sparse and other variants," in *Learning in Graphical Models,* ed M. Jordan (Dordrecht: Kluwer Academic), 355–68.

Nicolis, G., and Prigogine, I. (1977). *Self-organization in Non-equilibrium Systems.* New York, NY: John Wiley.

O'Regan, J. K., and Noë, A. (2001). A sensorimotor account of vision and visual consciousness. *Behav. Brain Sci.* 24, 939–973.

Oja, E. (1989). Neural networks, principal components, and subspaces. *Int. J. Neural Syst.* 1, 61–68.

Olshausen, B. A., and Field, D. J. (1996). Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature* 381, 607–609.

Optican, L., and Richmond, B. J. (1987). Temporal encoding of two-dimensional patterns by single units in primate inferior cortex. II Information theoretic analysis. *J. Neurophysiol.* 57, 132–146.

Ortega, P. A., and Braun, D. A. (2010). A minimum relative entropy principle for learning and acting. *J. Artif. Intell. Res.* 38, 475–511.

Pomerleau, D. A. (1991). Effcient training of articial neural networks for autonomous navigation. *Neural Comput.* 3, 88–97.

Rao, R. P. (2010). Decision making under uncertainty: a neural model based on partially observable markov decision processes. *Front. Comput. Neurosci.* 4:146. doi: 10.3389/fncom.2010.00146

Rao, R. P., and Ballard, D. H. (1999). Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects. *Nat. Neurosci.* 2, 79–87.

Rawlik, K., Toussaint, M., and Vijayakumar, S. (2010). Approximate inference and stochastic optimal control. arXiv:1009.3958

Rayner, K. (1978). Eye movements in reading and information processing. *Psychol. Bull.* 85, 618–660.

Rescorla, R. A., and Wagner, A. R. (1972). "A theory of Pavlovian conditioning: variations in the effectiveness of reinforcement and nonreinforcement," in *Classical Conditioning II: Current Research and Theory,* eds A. Black and W. Prokasy (New York NY: Appleton Century Crofts), 64–99.

Robert, C. (1992). *L'analyse statistique Bayesienne.* Paris: Economica.

Schaal, S. (1997). "Learning from demonstration," in *Advances in Neural Information Processing Systems.* Vol. 9, eds M. C. Mozer, M. Jordan, and T. Petsche (Boston, MA: MIT Press), 1040–1046.

Shachter, R. D. (1988). Probabilistic inference and influence diagrams. *Oper. Res.* 36, 589–605.

Shen, K., Valero, J., Day, G. S., and Paré, M. (2011). Investigating the role of the superior colliculus in active vision with the visual search paradigm. *Eur. J. Neurosci.* 33, 2003–2016.

Shires, J., Joshi, S., and Basso, M. A. (2010). Shedding new light on the role of the basal ganglia-superior colliculus pathway in eye movements. *Curr. Opin. Neurobiol.* 20, 717–725.

Srihasam, K., Bullock, D., and Grossberg, S. (2009). Target selection by the frontal cortex during coordinated saccadic and smooth pursuit eye movements. *J. Cogn. Neurosci.* 21, 1611–1627.

Sutton, R. S., and Barto, A. G. (1981). Toward a modern theory of adaptive networks: expectation and prediction. *Psychol. Rev.* 88, 135–170.

Tani, J. (2003). Learning to generate articulated behavior through the bottom-up and the top-down interaction processes. *Neural Netw.* 16, 11–23.

Tani, J., and Nolfi, S. (1999). Learning to perceive the world as articulated: an approach for hierarchical learning in sensory–motor systems. *Neural Netw.* 12, 1131–1141.

Tatler, B. W., Hayhoe, M. M., Land, M. F., and Ballard, D. H. (2011). Eye guidance in natural vision: reinterpreting salience. *J. Vis.* 11, 5.

Theodorou, E., Buchli, J., and Schaal, S. (2010). A generalized path integral control approach to reinforcement learning. *J. Mach. Learn. Res.* 11, 3137-3181.

Todorov, E. (2006). "Linearly-solvable Markov decision problems," in *Advances in Neural Information Processing Systems.* Vol. 19, (Boston, MA: MIT Press), 1369–1376.

Todorov, E. (2008). "General duality between optimal control and estimation," in *IEEE Conference on Decisionand Control.*

Toussaint, M., and Storkey, A. (2006). "Probabilistic inference for solving discrete and continuous state Markov Decision Processes," in *Proceedings of the 23rd International Conference on Machine Learning,* 945–952.

Toussaint, M., Charlin, L., and Poupart, P. (2008). "Hierarchical POMDP controller optimization by likelihood maximization." in *Uncertainty in Artificial Intelligence (UAI 2008)* (AUAI Press). 562–570.

Tschacher, W., and Haken, H. (2007). Intentionality in non-equilibrium systems? The functional aspects of self-organised pattern formation. *New Ideas Psychol.* 25, 1–15.

Ungerleider, L. G., and Mishkin, M. (1982). "Two cortical visual systems," in *Analysis of Visual Behavior,* eds D. Ingle, M. A. Goodale, and R. J. Mansfield (Cambridge, MA: MIT Press), 549–586.

van den Broek, B., Wiegerinck, W., and Kappen, B. (2008). Graphical model inference in optimal control of stochastic multi-agent systems. *J. Artif. Int. Res.* 32, 95–122.

Wang, Z., and Klein, R. M. (2010). Searching for inhibition of return in visual search: a review. *Vision Res.* 50, 220–228.

Wardak, C., Olivier, E., and Duhamel, J. R. (2011). The relationship between spatial attention and saccades in the frontoparietal network of the monkey. *Eur. J. Neurosci.* 33, 1973–1981.

Watkins, C. J., and Dayan, P. (1992). Q-learning. *Mach. Learn.* 8, 279–292.

Williams, R. J. (1992). Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Mach. Learn.* 8, 229–256.

Wurtz, R. H., McAlonan, K., Cavanaugh, J., and Berman, R. A. (2011). Thalamic pathways for active vision. *Trends Cogn. Sci.* 5, 177–184.

Yarbus, A. L. (1967). *Eye Movements and Vision.* New York, NY: Plenum.

Yuille, A., and Kersten, D. (2006). Vision as Bayesian inference: analysis by synthesis? *Trends Cogn. Sci.* 10, 301–308.

Zetzsche, C., and Röhrbein, F. (2001). Nonlinear and extra-classical receptive field properties and the statistics of natural scenes. *Network* 12, 331–350.

Zhang, N. L. (1998). Probabilistic inference in influence diagrams. *Comput. Intell.* 14, 475–497.

Conflict of Interest Statement: 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.

Received: 12 August 2012; accepted: 16 October 2012; published online: 02 November 2012.

Citation: Friston K, Adams R and Montague R (2012) What is value—accumulated reward or evidence? Front. Neurorobot. ***6****:11. doi: 10.3389/fnbot.2012.00011*

frontiers in
NEUROROBOTICS

ORIGINAL RESEARCH ARTICLE
published: 29 April 2013
doi: 10.3389/fnbot.2013.00008

Reward-based learning for virtual neurorobotics through emotional speech processing

Laurence C. Jayet Bray[1,2]*, **Gareth B. Ferneyhough**[1], **Emily R. Barker**[1], **Corey M. Thibeault**[3] and **Frederick C. Harris Jr.**[1]

[1] Department of Computer Science and Engineering, University of Nevada, Reno, NV, USA
[2] Department of Bioengineering, George Mason University, Fairfax, VA, USA
[3] HRL Laboratories, LLC, Malibu, CA, USA

Edited by:
Jeffrey L. Krichmar, University of California, Irvine, USA

Reviewed by:
Quan Zou, The George Washington University, USA
Hsin Chen, National Tsing-Hua University, Taiwan

***Correspondence:**
Laurence C. Jayet Bray, Department of Bioengineering, George Mason University, 4400 University Drive, Fairfax, VA 22030, USA.
e-mail: ljayet@gmail.com

Reward-based learning can easily be applied to real life with a prevalence in children teaching methods. It also allows machines and software agents to automatically determine the ideal behavior from a simple reward feedback (e.g., encouragement) to maximize their performance. Advancements in affective computing, especially emotional speech processing (ESP) have allowed for more natural interaction between humans and robots. Our research focuses on integrating a novel ESP system in a relevant virtual neurorobotic (VNR) application. We created an emotional speech classifier that successfully distinguished happy and utterances. The accuracy of the system was 95.3 and 98.7% during the offline mode (using an emotional speech database) and the live mode (using live recordings), respectively. It was then integrated in a neurorobotic scenario, where a virtual neurorobot had to learn a simple exercise through reward-based learning. If the correct decision was made the robot received a spoken reward, which in turn stimulated synapses (in our simulated model) undergoing spike-timing dependent plasticity (STDP) and reinforced the corresponding neural pathways. Both our ESP and neurorobotic systems allowed our neurorobot to successfully and consistently learn the exercise. The integration of ESP in real-time computational neuroscience architecture is a first step toward the combination of human emotions and virtual neurorobotics.

Keywords: emotional speech processing, reward-based learning, virtual neurorobotics, biological computational model

1. INTRODUCTION

How does speech portray emotions? Many of our social cues and communication skills rely on emotional speech, but it is a challenging process to study. Affective computing, especially emotional speech processing (ESP) has helped elucidate the importance of human emotions. It is basically described as applying human like emotional effects to artificially produced speech. Speech contains acoustic features that vary with the speaker's affective state, and the ability to interpret these communication signals (e.g., emotions) affects social interaction (Warren et al., 2006). Humans also perceive how emotional environmental cues such as fear or anger indicate danger (Kanske and Hasting, 2010) and keep them fit for survival.

At the physiological level, speech is processed in specialized brain regions in the upper portion of the superior temporal sulcus, which is one of the voice-selective areas of the auditory cortex (Grossmann et al., 2010). These areas in monkeys and humans have been thought to provide social information to sensory systems. Recent studies on macaque monkeys have revealed they have a region in the superior temporal plane selective to speech similar to humans (Belin et al., 2000, 2004). These studies suggest that recognition of speech within species is an evolutionarily conserved brain function in primates and is independent of language (Petkov et al., 2008, 2009). Therefore, language requires more than simply linguistic information. Other studies in behavioral biology, psychology, and speech and communication sciences have suggested that many emotional states are communicated by specific acoustic characteristics of the speaker. Evidence reveals that listeners attend to changes in voice quality, articulation, pitch, and loudness to understand the speaker's emotion (Banse and Scherer, 1996). Emotions that are the most distinct in humans are anger, disgust, fear, joy, sadness, and surprise (El Ayadi et al., 2011).

As part of emotional processing, emotional speech recognition is a relatively recent research field, which is defined as extracting the emotional state of a speaker from her or his speech (El Ayadi et al., 2011). Automatic recognition of emotions from modalities such as speech has acquired expanding interest within the area of human-machine interaction research (Fu et al., 2010). Such emotional speech recognition is essential for facilitating realistic communication between robots and humans. Service robots are being designed to help humans with difficult or time-consuming tasks or help those with disabilities (Severinson-Eklundh et al., 2003). Appropriate communication allows robots to share human knowledge, and can potentially use human recognition capabilities to complete complex tasks (Ghidary et al., 2002). Thus, it will be important for future robotics to be able to understand emotion in speech in order to complete such tasks.

Biological-inspired human-robot interactions have become increasingly important as robots fascinate many researchers and become more common in our daily activities. For the past couple of years, we have worked on machine learning systems, and we developed a Virtual Neurorobotic (VNR) loop, which focuses on the coupling of neural systems with some form of physical actuation. This is based around the interoperability of a neural model, a virtual robotic avatar and a human participant (Goodman et al., 2007, 2008). Under all but the most basic scenarios this interoperability is accomplished through an organized network communication system (Thibeault et al., 2010b, 2012).

This paper provides an introduction to affective computing and emotional speech processing combined with one application of real-time virtual neurorobotics. We use our VNR to describe how our emotional speech system can be successfully used to reinforce learning and allow a neurorobot to make ideal choices based on visual cues.

2. AFFECTIVE COMPUTING

The curious nature of human emotions has been the subject of much research and philosophical debate. Why do humans have emotions, and what role do they have in human cognition and behavior? During the cognitive revolution that began in the second half of the twentieth century, the lingering influence of behaviorism helped downplay the role of emotion to little more than a side effect from instinctual and learned behavior (Hudlicka, 2003).

Recently, advancements in neuroscience and psychology have helped elevate the importance of emotion; within the last decade or so, research has shown that emotion plays a crucial role in human intelligence, including planning and decision making of all levels (Hudlicka, 2003; Picard, 2003). This renewed interest in emotional research has led to the birth of a growing research field, affective computing. Rosalind Picard's paper, *Affective Computing: Challenges* gave the field its name (Picard, 2003). In her paper, Picard discusses the three main areas of affective computing: emotional sensing and recognition, affect modeling, and emotion expression.

Several researchers have attempted to create emotionally intelligent robots. Perhaps the most famous is Kismet, an infant-like robotic creature developed at MIT (Breazeal and Aryananda, 2002). Kismet responds to the emotional state (typically acted) of its "caregiver" by analyzing the caregiver's speech in real-time. The system extracts statistics on the caregiver's voice pitch and energy, and classifies the underlying emotion using a Gaussian mixture model classifier. The robot responds to the caregiver's emotional intent by changing its facial expression. Naive test subjects were chosen to interact with the robot and many felt a strong emotional response while interacting with it, especially when Kismet showed sadness after being prohibited by the human. Kismet successfully shows that robots can be designed to react to human emotions, and in turn, elicit an emotional response from the human as well.

Another empathetic android robot is BARTHOC, developed at Bielefeld University, Germany (Hegel et al., 2006). BARTHOC can be given several different appearances by changing the latex mask that composes its face and head. For many experiments, the robot is given the appearance of a small child via a latex mask, although its appearance is decidedly less "cute" than Kismet, due to the difficulty in creating a realistic looking android face. Like Kismet, BARTHOC mimics the emotion of the human interacting with it by changing its facial expression. The emotion of the human is determined using emotional speech processing. BARTHOC can distinguish and portray six emotional states: neutral, happy, fear, anger, disgust, surprise, and sad.

Both Kismet and BARTHOC can mimic human emotions by recognizing the emotional content in a human's speech. Our system aims to further these advancements by using human emotional content as a training mechanism for a virtual robot.

3. THEORY BEHIND EMOTIONAL SPEECH PROCESSING (ESP)

ESP systems (also called emotional speech recognition systems) attempt to determine the underlying emotion in human speech. Unlike normal speech recognition systems, most ESP systems do not extract lexical information, but instead classify the speaker's emotion without any regard to context. This is typically accomplished by extracting prosodic features for each word or phrase uttered by the speaker, generating statistics on these features, and classifying the feature vector using a supervised learning algorithm.

Although the accuracy of ESP systems is typically lower than other emotional classification methods involving facial imaging and physiological features, their recognition rates are similar to those of humans (Hudlicka, 2003). Furthermore, emotional speech recognition is less computationally expensive and less invasive than other methods, and remains a popular method for emotion detection, especially in live environments.

3.1. FEATURES

There is currently little consensus on the best features for emotional speech recognition, however statistics on prosodic features, especially the fundamental frequency (pitch), are among the most common (Scherer et al., 1991; Dellaert et al., 1996; Oudeyer, 2003; Ververidis et al., 2004; Fu et al., 2010; Thibeault et al., 2010b; Koolagudi et al., 2011; Tahon et al., 2011). Other prosodic features used for ESP include energy and duration (Batliner et al., 2006). In addition to prosody, other common features include spectral features such as Mel-frequency cepstral coefficients (MFCCs), and non-linear Teager energy based features. In order to form a "good“ feature vector, ESP systems extract several statistical quantities from each feature contour such as the "mean, median, standard deviation, maximum, minimum, range, linear regression coefficients, 4th order Legendre parameters, vibrations, mean of first difference, mean of the absolute of the first difference, jitter, and ratio of the sample number of the up-slope to that of the down-slope of the pitch contour" (El Ayadi et al., 2011). By varying the number of features, and the statistics on each feature, ESP systems can have feature vectors of lengths ranging from 12 (Breazeal and Aryananda, 2002) to 988 (Eyben et al., 2009). To improve classification time and accuracy, several studies begin with large feature sets and then select the best features using exhaustive, sequential, or random searches (Fu et al., 2010).

3.2. FUNDAMENTAL FREQUENCY DETECTION

The fundamental frequency (F_0) of a voiced speech is typically defined as the rate of vibration of the vocal folds (de Cheveigné and Kawahara, 2002). Generally, the pitch humans perceive when someone is talking or singing is equivalent to the fundamental frequency, and ranges from about 40 to 600 Hz (Huang et al., 2001). We will therefore refer to the fundamental frequency simply as "pitch", and methods to determine F_0 as "pitch detection algorithms." Frequency-domain pitch detection approaches usually utilize the Fast Fourier Transform (FFT) to convert the signal to the frequency spectrum. This allows for polyphonic detection. Time-domain approaches, such as autocorrelation are typically less computationally expensive, but may be prone to errors and octave jumps, especially due to noise. As a method, robust algorithm for pitch tracking (RAPT) (Talkin, 1995) is a pitch tracking algorithm that attempts to return a smooth pitch contour, without the undesirable octave jumps and false detection problems present in the basic auto-correlation method. RAPT operates on two versions of the input signal, one at the original sample rate, and one at a significantly reduced rate. The algorithm first computes the normalized cross-correlation (NCFF) of a low-sample signal and records the locations of the local maxima. Next, NCFF is performed on the higher sample-rate signal in the vicinity of the peaks found in the previous step. This generates a list of several F_0 candidates for the input frame. Finally, dynamic programming is used to select the best F_0 candidates over the entire window.

3.3. CLASSIFIERS

After a feature vector has been created, it must be classified in order to determine its emotional class. A number of classifiers have been used in ESP systems, including hidden Markov models (HMM), Gaussian mixture models (GMM), k-nearest neighbor (k-NN), support vector machines (SVM), artificial neural networks (ANN), and decision trees (El Ayadi et al., 2011). Different classifiers can perform better in different situations, which can have a significant effect of a system's classification accuracy (El Ayadi et al., 2011). Therefore, it is important for the researcher to chose a classifier carefully, taking into account accuracy as well as computational requirements.

3.4. DATABASES

It can be difficult to compare the classification accuracies reported by different researchers due to the variety in emotional speech databases used. The Berlin emotional speech database (Burkhardt et al., 2005) contains recordings performed by professional actors in a noise-free environment, while (Morrison et al., 2007) provides actual recordings from call centers. Naturally, both humans and computers attain higher recognition accuracy on databases containing low-noise, acted recordings.

4. VIRTUAL NEUROROBOTICS (VNR)

VNR aims to develop combinations of biologically realistic neural simulations with robotic agents and human participants in closed-loop configurations (Thibeault et al., 2010b). As described by our previous studies by Goodman et al. (2007, 2008), we define VNR as follows: a computer-facilitated behavioral loop wherein a human interacts with a projected robot that meets five criteria: the robot is sufficiently embodied for the human to tentatively accept the robot as a social partner; the loop operates in real time, with no pre-specified parcellation into receptive and responsive time windows; the cognitive control is a neuromorphic brain emulation using our NeoCortical simulator (NCS) and incorporating realistic neuronal dynamics whose time constants reflect synaptic activation, membrane and circuitry properties, and most importantly learning; the neuromorphic architecture is expandable to progressively larger scale and complexity to track brain development; and the neuromorphic architecture can potentially provide circuitry underlying intrinsic motivation and intentionality, which physiologically is best described as emotional rather than rule-based drive.

NCS (Drewes, 2005; Wilson et al., 2005; Brette et al., 2007; Drewes et al., 2009; Jayet Bray et al., 2010) is a neural simulator that can model integrate-and-fire neurons with conductance-based synapses. It uses two clusters: four SUN 4600 machines (16-processors each) connected via Infiniband with 192 GB RAM per machine, 24 Terabytes of disk storage; and 208 Opteron cores, 416 GB RAM, and more than a Terabyte of disk storage. Note: for more information on NCS equations and related publications, please go to: www.cse.unr.edu/brain/publications.

As a part of our neurorobotics, learning can be based on many different experiences including making correct decisions and consequently being rewarded. As illustrated in **Figure 1**: (1) a human participant presents the robot with one external cue at a time. The robot sees and then processes the information (2), then a decision followed by an action associated with the initial cue is made (3). Then, there are two possible scenarios (4): If the decision/action is incorrect, then the robot does not receive any reward. However, if the decision is correct it does receive a reward (e.g., hears positive speech) by the human. In our correct case, the reward stimulates synapses (in our simulated model) that underwent spike-timing dependent plasticity (STDP) described by several studies

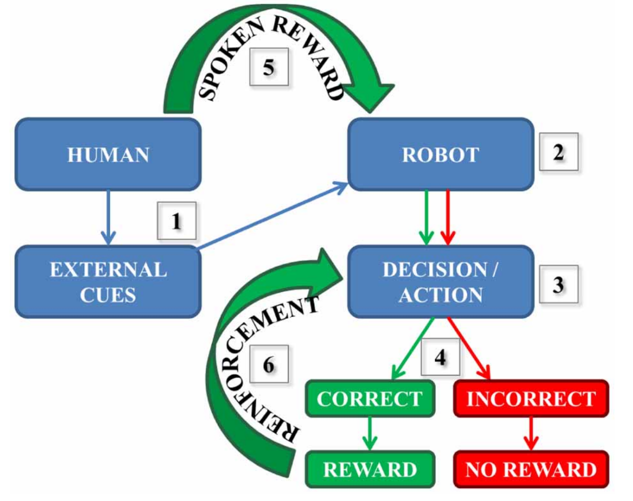


FIGURE 1 | Simplified reward-based learning scheme during human-robot interaction. (1) The external cue is presented by the human to the robot; (2) The information is seen and processed by the robot; (3) The decision and the related action are performed; (4) The robot chose the ball correctly or incorrectly; (5) The spoken reward (if correct action) is received by the robot (6) The reward reinforces Learning every time the decision is correct.

(Zhang et al., 1998; Song et al., 2000; Dan and Poo, 2004; Caporale and Dan, 2008; Markram et al., 2011) as:

$$W(\Delta t) = \begin{cases} A_{+} \exp\left(\frac{\Delta t}{\tau_{+}}\right) & \text{if } (\Delta t) < 0 \\ -A_{-} \exp\left(\frac{-\Delta t}{\tau_{-}}\right) & \text{if } (\Delta t) \geq 0 \end{cases} \qquad (1)$$

where A is the maximum amount of synaptic modification; Δt is the positive or the negative window; and τ is the positive or the negative decay constant.

Every time the robot receives a spoken reward (5), the neural pathway corresponding to the correct decision and the action is reinforced (6) until completely learned.

The integration of ESP in real-time computational neuroscience architecture is a first step toward the combination of human emotions and virtual neurorobotics. It was first described in our preliminary study by Thibeault et al. (2010b), and it is now being improved and further implemented in one of our neurorobotic applications. The improvements consisted on making the system a stand alone C++ application using a different classification and an ameliorated feature extraction method as described in Section 5.

5. METHODS

5.1. HUMAN EMOTIONAL SPEECH CLASSIFICATION

To provide a benchmark for our emotional speech classification system, we conducted a human trial in which seven individuals were asked to classify 40 random utterances (sentences) from the Berlin emotional speech database from four emotional classes: happy, sad, anger, and fear. An even amount of samples (10) was randomly played for each of the four emotions. Therefore, a total of 280 samples (70 for each emotion) were classified and displayed in a confusion matrix (**Table 1**). All the samples in the database were in German and the humans classifying the samples only spoke English. This allowed the listeners to only base their classifications on the prosody only, rather than the meaning of the words.

5.2. EMOTIONAL SPEECH RECOGNITION SYSTEM

Our emotional speech classification system operated in real-time by extracting several prosodic features for each utterance, and classifying them using the support vector machine library, libSVM (Chang and Lin, 2011) with the Radial Basis Function (RBF) kernel. This classifier was chosen because of its high accuracy for emotional speech classification tasks (El Ayadi et al., 2011).

To form the prosodic feature vector for each utterance, the pitch for each window was determined using RAPT (Talkin, 1995), as described in Section 3. The window size and overlap were 3361 and 2880 frames long, respectively. These values were suggested by the RAPT algorithm for our system's particular sample rate of 16 KHz. In addition to the pitch, RAPT also returned the signal energy for each window. If the energy was above a dynamic threshold, RAPT assumed that the speaker was talking. In this case, the energy and pitch for that window were saved. If the energy was too low, the speaker assumed to be silent and the window was discarded.

The system continued saving pitch and energy values for each window until a two second break in speech was was detected. This corresponded to the end of a utterance. After the end of an utterance, the feature vector was formed by calculating the mean, minimum, maximum, and range of the pitch values over the utterance. In addition to these four values, the feature vector also contained the mean speech energy during the voiced regions. In testing mode, the feature vector was then scaled and classified using libSVM. Before the system could classify emotions, it had to be trained (training mode). Features were extracted for 33% (offline) and 50% (live) of utterances, and they were given the appropriate emotion class labels. When the desired number of utterances was processed by the system, the feature vectors were scaled and used to create a libSVM model. The model file and scaling parameters were saved and used to classify the feature vectors in testing mode. To show the difference between pitch and emotion, the average pitch over 23 utterances was graphed comparing "happy" and "sad" emotion for both male and female speakers (**Figure 2**). This illustrates how the different pitch measurements change with respect to emotion and gender, as supported by Ververidis and Kotropoulos (2006).

There were two different experiments conducted to evaluate the classification accuracy of our system. The JACK Audio Connection Kit (Davis, 2013) was used to connect audio to the system, either from a separate audio player (offline mode) or the microphone (live mode). In the offline mode, the same prerecorded samples (23 "happy" and "sad" utterances for both male and female speakers) from the Berlin emotional speech database were used, which gave a total of 92 samples. In the live mode, four humans recorded samples at 16 KHz from a list of 10 neutral phrase samples. The following samples were recorded: "Look Jack, the ball is blue," "The ball is red," "You turned left toward the library," "Jack, you turned right toward the museum," "You pointed to the blue color," "You pointed to the red color, Jack," "Jack, you went over there," "Look what you've done," "Jack gave the rattle to his mom," "Jack kept the rattle for himself." Each sample was recorded twice with both "happy" and "sad" utterances giving a total of 160 phrase samples. For both experiments,

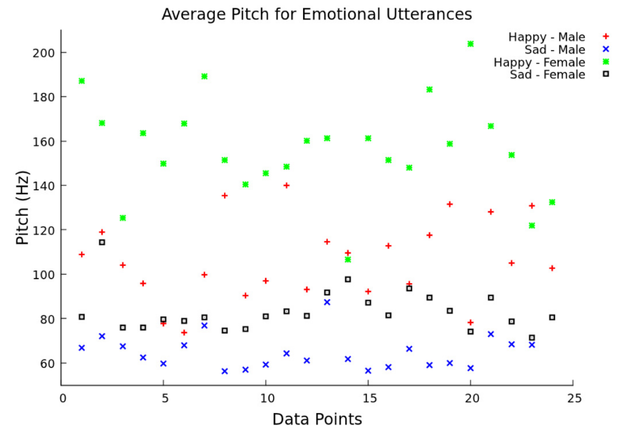


FIGURE 2 | Average pitch between two emotional utterances. The average pitch (Hz) is shown over 14 data points for both "happy" (female voice—green and male voice—red) and "sad" (female voice—black and male voice—blue) utterances.

the results were represented as confusion matrices distinguishing "happy" and "sad" utterances from both female and male speakers. These showed the accuracy of the system (in terms of % error) for both live and offline modes (**Tables 2, 3**).

5.3. A VIRTUAL NEUROROBOTIC APPLICATION

Our virtual neurorobotic loop used a virtual neurorobot as a remote agent, and the interaction between a camera and ESP. The design used in this project as well as the basic software engineering behind its implementation was further described in our previous research by Thibeault et al. (2012).

As a scenario example described in **Figure 3**, we designed an experiment using a spoken reward through ESP as reward-based learning. (1) A human presented a card with either a printed blue or red pattern to the neurorobot via the camera, which captured the image from the user and calculated the dominant color. (2) The information was processed by the virtual neurorobot, which was sent as the defined plain text statement ("saw red" or "saw blue") to NCSTools through the server interface (Thibeault et al., 2012). (3a) The configuration of NCSTools stimulated the appropriate regions of the remote NCS Model through the NCS network interface (Jayet Bray et al., 2012). Images were then processed and respective values were sent to simulated visual pathways (Thibeault et al., 2010a). (3b) The NCSTools server monitored the neurorobot and created the appropriate stimulus to send to proprioceptive feedback and premotor movements. The NCSTools software then received spiking information from the premotor region of the neural simulation. Such activity in the two premotor regions were monitored, and then compared as the stimulation progressed. The appropriate command was finally sent to the neurorobot once a configured threshold was reached (Anumandla et al., 2011; Jayet Bray et al., 2012). (4) This loop of events initiated the appropriate pointing motion/action toward a colored ball. (5) After the robot has pointed to the correct or incorrect colored ball, the human participant responded with a "happy" or "sad" spoken phrase. This was processed by the ESP which determined whether the participant encouraged or discouraged the action of the virtual neurorobot. (6) The output of the ESP was fed through NCS Tools to the neural model. This reward stimulus was injected into groups of neurons

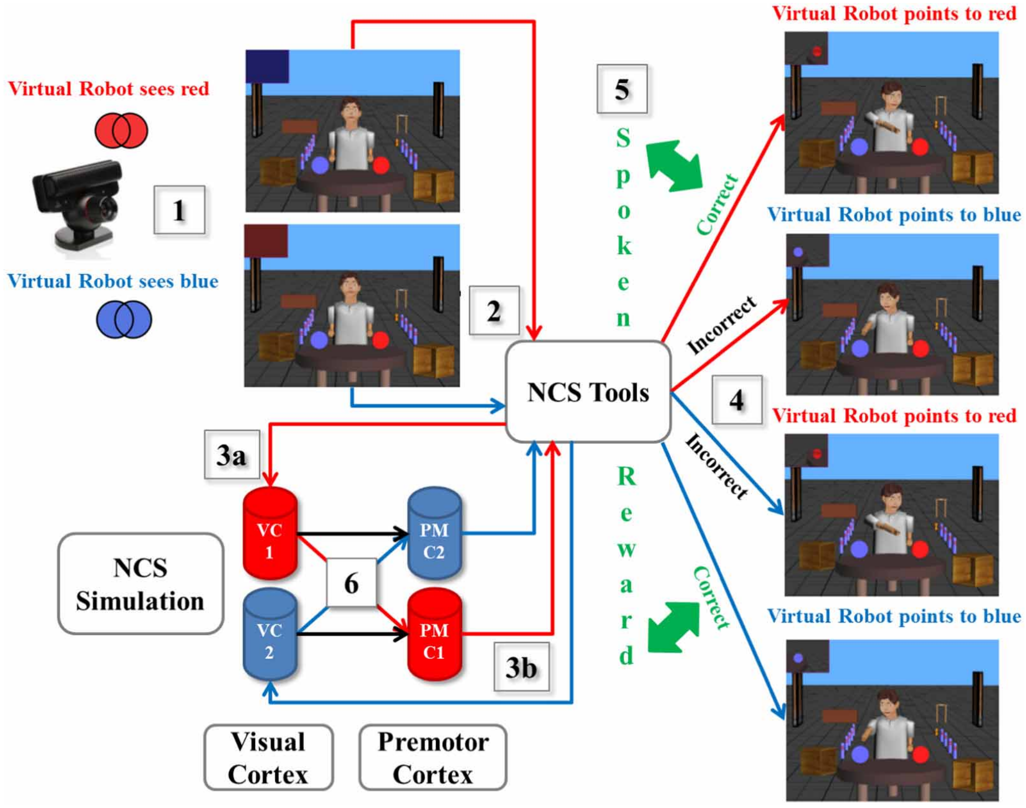


FIGURE 3 | A scenario of the virtual neurorobotic loop. (1) The external cue (red or blue card) is presented by the human to the neurorobot (via the camera); (2) The Information is processed by the neurorobot and is sent to NCSTools; (3a) NCSTools stimulates the appropriate regions of the modeled visual cortex; (3b) NCSTools server monitores the neurorobot and creates the appropriate stimulus to send to proprioceptive feedback and premotor movements; (4) This loop of events initiates the appropriate pointing motion/action toward a colored ball; (5) After the robot has pointed to the correct or incorrect colored ball, the human participant responded with a "happy" or "sad" spoken phrase. This was processed by the ESP which determined whether the participant encouraged or discouraged the action of the virtual neurorobot; (6) The output of the ESP was coupled with the neural model via NCSTools. This reward stimulus was injected into groups of neurons (VC1 and VC2) to stimulate the corresponding synapses (to PMC1 and PMC2) that underwent STDP, which reinforced or depressed learning through time.

(VC1 and VC2) to stimulate the corresponding synapses (to PMC1 and PMC2) that underwent STDP, which reinforced learning through time.

The experiment started as follows. The human showed a colored card randomly to the neurorobot (via the camera). On the first few attempts, the neurorobot had an equal chance of answering correctly or incorrectly since it was not familiar with the exercise. During this learning period, every time it chose the correct (incorrect) colored ball the "happy" ("sad") reward was given. It took about 4 to 5 trials for the neurorobot to fully learn the exercise. Once it was completely familiar with the drill, no more rewards were necessary, but it continued to correctly point to the right color for the rest of the experiment. Overall, as shown in **Figure 3** there were four possible scenarios: when the robot was shown the red (blue) pattern and correctly pointed to the red (blue) ball to its left (right), the human provided a happy spoken response. However, if the neurorobot incorrectly pointed right (left) to the blue (red) ball, a sad spoken response was given to the neurorobot.

For this simple example the reward was provided by correlated inputs between the previously activated visual column and the correctly chosen premotor column as well as reward activated STDP. In this case the plasticity of the synaptic connections was enabled during reward input. While the correlated firing encouraged the facilitation of the synapses resulting in an overall average increase in synaptic efficacy. It is important to emphasis that this reward mechanism is independent of the ESP system. The emotional classification can be used to activate any reward, punishment or input stimulus to the neural model. More sophisticated reward mechanisms such as those described in Florian (2007); Izhikevich (2007); Frmaux et al. (2010); Friedrich et al. (2011); O'Brien and Srinivasa (2013) will be explored in the future.

6. RESULTS

The results of our emotional speech classification system and its integration as a reward in a VNR scenario are presented below.

6.1. HUMAN EMOTIONAL SPEECH CLASSIFICATION PERFORMANCE

From the classification system, an English speaking human was able to classify German speakers' emotions (fear, anger, happy, and sad) with an accuracy of 88.6%, as shown in the confusion matrix in **Table 1**. The vertical category column represents the actual class (Berlin emotional speech database recordings) where the horizontal category row is the classification of the emotion by the human subjects. For instance, Out of the 70 German "happy" tones 56 were correctly classified and 14 were incorrectly interpreted as either "fear" or "anger." Additionally, the confusion table showed that most of the error occurred when the listener distinguished between "anger" and "happy," when listening to an angry emotion OR when the listener distinguished between "happy" and "fear," when listening to a happy utterance. This occurred because the utterances between these two emotions had similar features. This confusion can be expected between "anger" or "fear," and happy in similar systems. Therefore, the "happy" and "sad" emotions were chosen for our neurorobotic application below due to a classification accuracy of 98.6%.

Table 1 | Human classification confusion matrix.

Category	Anger	Fear	Happy	Sad	Error
Anger	62	3	5	0	11.4%
Fear	5	62	1	2	11.4%
Happy	5	8	56	1	20.0%
Sad	0	1	1	68	2.9%
Average error					11.4%

6.2. EMOTIONAL SPEECH RECOGNITION SYSTEM PERFORMANCE

In **Figure 2**, the average pitch is represented for the two chosen emotional classes (happy and sad) between the male and female groups from the Berlin emotional speech database. The "happy" utterance had a higher pitch frequency than the "sad" one, especially with female speakers. The "sad" male utterance had the lowest average pitch frequency overall.

During the offline mode, 92 samples from the Berlin emotional speech database (Burkhardt et al., 2005) were used to train (31 samples) and test (61 samples) the system. As shown in **Table 2**, 33 phrase samples of the 34 total happy samples (male and female combined) were correctly classified as happy while one was classified incorrectly as sad, giving an error of 5.6%. Out of the 27 total sad phrase samples (male and female combined), 25 were classified correctly while two were incorrectly classified as happy, giving an error of 13.3%. If we separate the male and female results, all 16 of the happy male phrase samples were correctly classified as happy, giving a 0% error. All of the 12 sad female samples were also correctly classified as sad, giving an error of 0%. The overall average error for all 61 phrase samples was 4.7%, which corresponds to a system accuracy of 95.3%. Note: Approximately 33% of the total 160 samples were used to train the system.

During the live mode, 160 samples from live recordings were used to train (83 samples) and test (77 samples) the system. As shown in **Table 3**, 41 phrase samples of the 42 total happy samples

Table 2 | Offline Mode Recognition confusion matrix.

Category	Happy-M	Sad-M	Happy-F	Sad-F	Error
Happy-M	16	0	0	0	0.0%
Sad-M	2	13	0	0	13.3%
Happy-F	0	0	17	1	5.6%
Sad-F	0	0	0	12	0.0%
Average error					4.7%

Table 3 | Live Mode Recognition confusion matrix.

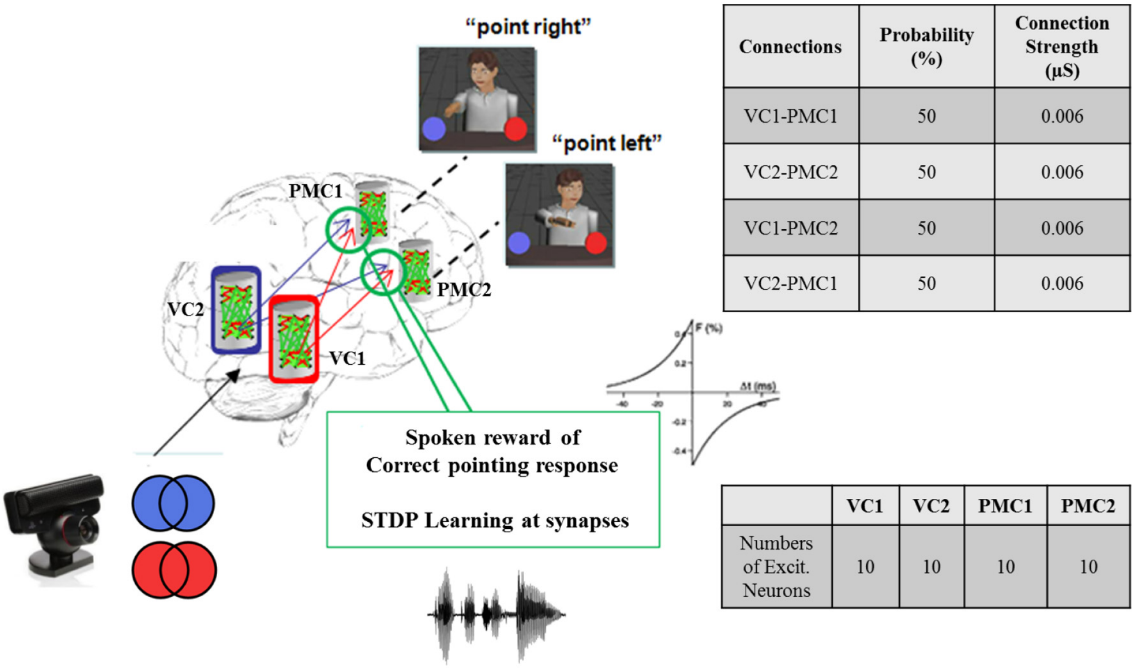
Category	Happy-M	Sad-M	Happy-F	Sad-F	Error
Happy-M	22	0	0	0	0.0%
Sad-M	0	16	0	0	0.0%
Happy-F	0	0	19	1	5.0%
Sad-F	0	0	0	19	0.0%
Average error					1.3%

were correctly classified as happy while 1 was classified incorrectly as sad, giving an error of 5%. Out of the 35 total sad phrase samples, none were classified incorrectly, giving an error of 0%. The overall average error for all 77 phrase samples was 1.3%, which corresponds to a system accuracy of 98.7%. Note: Approximately 50% of the total 160 samples were used to train the system.

6.3. VIRTUAL NEUROROBOTIC AND REWARD-BASED LEARNING

In our neurorobotic application, the simple spiking neuron model used was an important aspect of the system, and it is illustrated in **Figure 4**. Once the camera captured either red or blue color, the visual information was processed and sent to NCSTools, as described in Section 5. The information was then converted and sent to NCS running on a remote computing cluster. The brain architecture was composed of two areas: the visual cortex (VC) and the premotor cortex (PMC) divided into four areas of 10 neurons: VC1, VC2, PMC1, and PMC2. Each VC column was connected to both PMC columns with a probability of connections of 50% and a connection strength of 0.006 μS. Only the connections from VC1 → PMC1 and VC2 → PMC2 had reinforcement learning synapses (positive STDP) where the other connections got depressed though time (negative STDP). Therefore, as the red pattern was presented VC1 activity increased, and consequently increased PMC1 firing. On the other hand, when the blue pattern was presented VC2 activity increased, and consequently increased PMC2 firing. As the simulation proceeded, the competing neural areas of visual and motor processing were monitored by NCSTools. The resulting activity was correlated with a pointing action to one of two colored balls that matched the color presented. After the robot pointed, a spoken reward was given to the robot if it pointed to the correct colored ball. The reward, analogous to a dopaminergic increase, resulted in an STDP dependent increase in synaptic efficacy (Zou and Destexhe, 2007). STDP was defined in Section 4, and in the model the maximum positive and negative amounts of synaptic modification (A) were 20 and 10 respectively; the positive and negative windows (Δt) were 50 ms and 100 ms, respectively; and the positive and negative decay constants (τ) were both 5 ms.

The Graphical User Interface (GUI) is an option given to users for visualizing aspects of the neural model in real-time. The user can specify each tab with the information of either: main window, stimulation input (VCs), and motor areas (PMCs). As shown in **Figures 5A,B** the average synaptic weight over the simulation time can be monitored. As an example for a 9 s simulation, **Figure 5A** shows both average synaptic weights increase between VC1 (VC2) and PMC1 (PMC2), which shows evidence that the neurorobot's correct decisions were reinforced over time. However, the average synaptic weights between "non-learning" synapses (VC1 to PMC2 and VC2 to PMC1) show no increase over time (**Figure 5B**). To support these results, the firing activity of both PMC1 and PMC2 is represented in (**Figures 5C–F**).



Spoken reward of Correct pointing response

STDP Learning at synapses

Connections	Probability (%)	Connection Strength (μS)
VC1-PMC1	50	0.006
VC2-PMC2	50	0.006
VC1-PMC2	50	0.006
VC2-PMC1	50	0.006

	VC1	VC2	PMC1	PMC2
Numbers of Excit. Neurons	10	10	10	10

FIGURE 4 | Brain architecture in the virtual neurorobotic interface. Simple spiking neuronal model composed of two areas: the visual cortex (VC) and the premotor cortex (PMC). Each area is divided into two columns VC1, VC2, and PMC1, PMC2 (10 neurons each), respectively. Each VC area has feedforward connections to both PMC regions ($P = 50\%$ and Max. conductance = 0.006 μS). The synaptic connections from VC1 (VC2) to PMC1 (PMC2) use STDP as a learning mechanism. As the red (blue) is presented to VC1 (VC2) the activity of the corresponding column increases, which make PMC1 (PMC2) fire. When the neurorobot points correctly to "red" ("blue") a spoken reward is given, which stimulates the corresponding synapses VC1 → PMC1 (VC2 → PMC2).

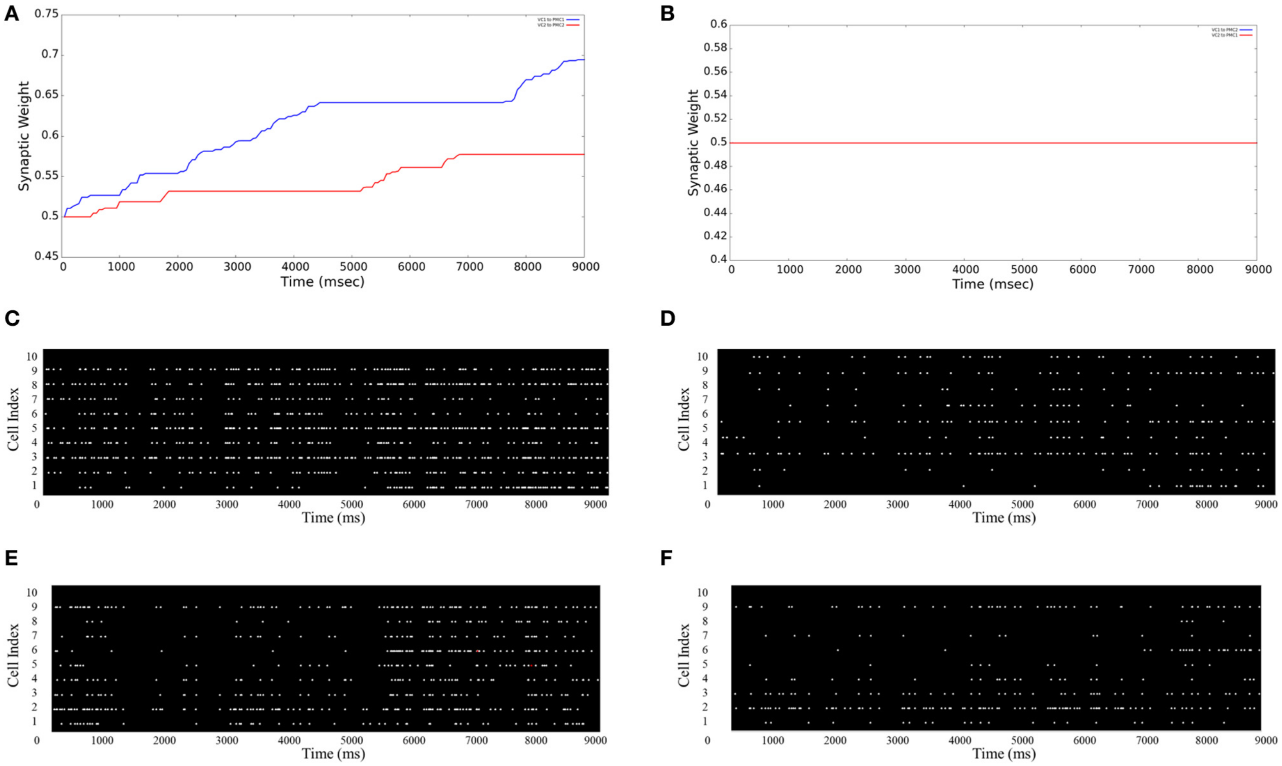


FIGURE 5 | Average synaptic weights and firing activities. The GUI monitors the synaptic strengths over time as an average of all synapses between different neuron groups. **(A)** VC1 → PMC1 and VC2 → PMC2 synpses show an increase in their strengths due to STDP reinforcement; **(B)** VC1 → PMC2 and VC2 → PMC1 synpses show no increase in their strengths (note: the blue and red lines are superimposed); (**C** and **E**) PMC1 and PMC2 firing rates increase as learning occurs—the neurorobot makes correct choices; (**D** and **F**) PMC1 and PMC2 firing rates do not increase when the neurorobot is not rewarded—makes incorrect choices.

They increase as reinforcement occurs (**Figures 5C,E**) when the neurorobot was rewarded, but they show no significant changes when the neurorobot is not rewarded (**Figures 5D,F**). The PMC1 and PMC2 average firing rates increased from 4.21 to 9.63 Hz and from 4.34 to 10.59 Hz, respectively (**Figures 5C,E**). However, the average rate changed from 3.89 to 3.95 Hz in **Figure 5D** and from 3.91 to 4.02 Hz in **Figure 5F**.

7. DISCUSSION AND FUTURE WORK

Robotic applications seem to be the future of our society due to a rapid evolution in advanced technologies. Many developers, researchers, and scientists have focused on physical robots (Breazeal and Aryananda, 2002; Hegel et al., 2006) that mimic human emotions by recognizing the emotional content in a human's speech. On the other hand, we have paid more attention to how the brain and its related biological processes, and cognition, are involved in human-robot interactions. The development of our VNR has emphasized the integration of ESP as a reward into a virtual neurorobotic system.

During our human emotional speech classification performance, seven English speaking humans were able to classify German speakers' emotions (fear, anger, happy, and sad) with an accuracy of 88.6%, which provided a benchmark for our emotional speech classification system. Since there was a 98.6% accuracy between the "happy" and "sad" utterances, these were chosen to be used as a spoken reward in our virtual neurorobotic application.

Using the Berlin emotional speech database, the average pitch (extracted from our system) between two emotional classes ("happy" and "sad") and groups of speakers (male and female) was significantly different. This confirmed that RAPT was a successful method for extracting the pitch out of every sample. Using the libSVM model, our offline mode system performance had an accuracy of 95.3% and our live recognition system performance attained similar accuracy by classifying the different emotions correctly 98.7% of the time.

Based on the system performances, we created a scenario where natural speech was used as a reward during a simple exercise. Our emotional speech processing system accurately distinguished between two classes of emotions, happy and sad, and provided a more natural and efficient way for training a child-like robot. ESP was translated to the presented VNR example to encourage or discourage the neurorobot's actions. The plasticity of the synaptic connections was shown as an increase in the synaptic strengths (between VC1 and PMC1, and VC2 and PMC2) and in the firing rates of PMC1 and PMC2 when a reward was given. On the other hand, the absence of reward showed no significant synaptic strengths nor firing rates increase in the concerned regions. These results give a preliminary evaluation when a spoken reward was used as an external stimulus into a neuromorphic brain architecture. In terms of applications, an emphasis was placed on robotic and automated agents. However, our system is by no means limited to that specific application.

Overall, we described how our spoken reward system was successfully used as reinforcement learning and allow our neurorobot to learn a simple exercise and make ideal choices based on visual cues. The ability to monitor and modify simulations in real-time was incredibly useful, especially when we further improve to spiking networks to a larger scale. More importantly, this could demonstrate another step towards multi-scale visualization of neural simulations in a virtual environment.

We are also currently working on the emotional classification system to accurately determine between additional classes in a live environment. Furthermore, the creation of additional virtual robotic scenarios could allow varying degrees of rewards, such as more emotions, and additional external cues, such as facial recognition. Ultimately, we plan create a biologically-realistic emotional classification system that extracts pitch features using a spiking cochlear model, and classifies emotions using a more biologically realistic neural network.

ACKNOWLEDGMENTS

The authors would like to thank Jacob Hansen for his help with the human emotional classification experiment. This work was supported by the U.S. Office of Naval Research (N000140110014).

REFERENCES

Anumandla, S. R., Jayet Bray, L. C., Thibeault, C. M., Hoang, R. V., Dascalu, S. M., Harris, F. C. Jr., et al. (2011). "Modeling oxytocin induced neurorobotic trust and intent recognition in human robot interaction," in *International Joint Conference on Neural Networks (IJCNN)* (San Jose, CA).

Banse, R., and Scherer, K. R. (1996). Acoustic profiles in vocal emotion expression. *J. Pers. Soc. Psychol.* 70, 614–636.

Batliner, A., Steidl, S., Schuller, B., Seppi, D., Laskowski, K., Vogt, T., et al. (2006). "Combining efforts for improving automatic classification of emotional user states," in *Proceedings IS-LTC 2006* (Ljubliana), 240–245.

Belin, P., Fecteau, S., and Bédard, C. (2004). Thinking the voice: neural correlates of voice perception. *Trends Cogn. Sci.* 8, 129–135.

Belin, P., Zatorre, R. J., Lafaille, P., Ahad, P., and Pike, B. (2000). Voice-selective areas in human auditory cortex. *Nature* 403, 309–312.

Breazeal, C., and Aryananda, L. (2002). Recognition of affective communicative intent in robot-directed speech. *Auton. Robots* 12, 83–104.

Brette, R., Rudolph, M., Carnevale, T., Hines, M., Beeman, D., Bower, J., et al. (2007). Simulation of networks of spiking neurons: a review of tools and strategies. *J. Comput. Neurosci.* 23, 349–398.

Burkhardt, F., Paeschke, A., Rolfes, M., Sendlmeier, W., and Weiss, B. (2005). "A database of german emotional speech," in *Proceedings of Interspeech* (Lissabon), 1517–1520.

Caporale, N., and Dan, Y. (2008). Spike timingdependent plasticity: a hebbian learning rule. *Annu. Rev. Neurosci.* 31, 25–36.

Chang, C.-C., and Lin, C.-J. (2011). LIBSVM: a library for support vector machines. *ACM Trans. Intell. Sys. Technol.* 2, 1–27.

Dan, Y., and Poo, M.-M. (2004). Spike timing-dependent plasticity of neural circuits. *Neuron* 44, 23–30.

Davis, P. (2013). Jack connecting a world of audio. Available online at: http://jackaudio.org/

de Cheveigné, A., and Kawahara, H. (2002). YIN, a fundamental frequency estimator for speech and music. *J. Acoust. Soc. Am.* 111, 1917–1930.

Dellaert, F., Polzin, T., and Waibel, A. (1996). "Recognizing emotion in speech," in *Proceeding of Fourth International Conference on Spoken Language Processing ICSLP 96* (Philadelphia, PA), 1970–1973.

Drewes, R. (2005). *Brainlab: a Toolkit to Aid in the Design, Simulation, and Analysis of Spiking Neural Networks with the NCS Environment*. Master's thesis, University of Nevada, Reno.

Drewes, R., Zou, Q., and Goodman, P. (2009). Brainlab: a python toolkit to aid in the design, simulation, and analysis of spiking neural networks with the neocortical simulator. *Front. Neuroinform.* 3:16. doi: 10.3389/neuro.11.016.2009

El Ayadi, M., Kamel, M. S., and Karray, F. (2011). Survey on speech emotion recognition: features, classification schemes, and databases. *Pattern Recognit.* 44, 572–587.

Eyben, F., Wollmer, M., and Schuller, B. (2009). "OpenEAR: introducing the munich open-source emotion and affect recognition toolkit," in *Affective Computing and Intelligent Interaction and Workshops, 2009. ACII 2009. 3rd International Conference* (Amsterdam), 1–6.

Florian, R. V. (2007). Reinforcement learning through modulation of spike-timing-dependent synaptic plasticity. *Neural Comput.* 19, 1468–1502.

Friedrich, J., Urbanczik, R., and Senn, W. (2011). Spatio-temporal credit assignment in neuronal population learning. *PLoS Comput. Biol.* 7:e1002092. doi: 10.1371/journal.pcbi.1002092

Frmaux, N., Sprekeler, H., and Gerstner, W. (2010). Functional requirements for reward-modulated spike-timing-dependent plasticity. *J. Neurosci.* 30, 13326–13337.

Fu, L., Wang, C., and Zhang, Y. (2010). "Classifier fusion for speech emotion recognition," in *Intelligent Computing and Intelligent Systems (ICIS), 2010 IEEE International Conference* (Xiamen), 407–410.

Ghidary, S. S., Nakata, Y., Saito, H., and Hattori, M. (2002). Multimodal interaction of human and home robot in the context of room map generation. *Auton. Robots* 13, 169–184.

Goodman, P. H., Buntha, S., Zou, Q., and Dascalu, S.-M. (2007). Virtual neurorobotics (VNR) to accelerate development of plausible neuromorphic brain architectures. *Front. Neurorobotics* 1:1. doi: 10.3389/neuro.12.001.2007

Goodman, P. H., Zou, Q., and Dascalu, S.-M. (2008). Framework and implications of virtual neurorobotics. *Front. Neurosci.* 2, 123–128. doi: 10.3389/neuro.01.007.2008

Grossmann, T., Oberecker, R., Koch, S. P., and Friederici, A. D. (2010). The developmental origins of voice processing in the human brain. *Neuron* 65, 852–858.

Hegel, F., Spexard, T., Vogt, T., Horstmann, G., and Wrede, B. (2006). "Playing a different imitation game: interaction with an empathic android robot," in *Proceedings 2006 IEEE-RAS International Conference on Humanoid Robots (Humanoids 06)* (Genova, Italy), 56–61.

Huang, X., Acero, A., and Hon, H.-W. (2001). *Spoken Language Processing: A Guide to Theory, Algorithm, and System Development*. 1st Edn., Upper Saddle River, NJ: Prentice Hall PTR.

Hudlicka, E. (2003). To feel or not to feel: the role of affect in human computer interaction. *Int. J. Hum. Comput. Stud.* 59, 1–32.

Izhikevich, E. M. (2007). Solving the distal reward problem through linkage of stdp and dopamine signaling. *Cereb. Cortex* 17, 2443–2452.

Jayet Bray, L. C., Anumandla, S. R., Thibeault, C. M., Hoang, R. V., Dascalu, S. M., Bryant, B. D., et al. (2012). Real-time human-robot interaction underlying neurorobotic trust and intent recognition. *J. Neural Netw.* 32, 130–137.

Jayet Bray, L. C., Quoy, M., Harris, F. C. Jr, and Goodman, P. H. (2010). A circuit-level model of hippocampal place field dynamics modulated by entorhinal grid and suppression-generating cells. *Front. Neural Circuits* 4:122. doi: 10.3389/fncir.2010.00122

Kanske, P., and Hasting, A. S. (2010). Decoding modality-independent emotion perception in medial prefrontal and superior temporal cortex. *J. Neurosci.* 30, 16417–16418.

Koolagudi, S., Kumar, N., and Rao, K. (2011). "Speech emotion recognition using segmental level prosodic analysis," in *Devices and Communications (ICDeCom), 2011 International Conference* (Mesra), 1–5.

Markram, H., Gerstner, W., and Sjstrm, P. (2011). A history of spike-timing-dependent plasticity. *Front. Syn. Neurosci.* 3:4. doi: 10.3389/fnsyn.2011.00004

Morrison, D., Wang, R., and De Silva, L. (2007). Ensemble methods for spoken emotion recognition in call-centres. *Speech Commun.* 49, 98–112.

O'Brien, M. J., and Srinivasa, N. (2013). A spiking neural model for stable reinforcement of synapses based on multiple distal rewards. *Neural Comput.* 5, 123–156.

Oudeyer, P.-Y. (2003). The production and recognition of emotions in speech: features and algorithms. *Int. J. Hum. Comput. Stud.* 59, 157–183.

Petkov, C. I., Kayser, C., Steudel, T., Whittingstall, K., Augath, M., and Logothetis, N. K. (2008). A voice region in the monkey brain. *Nat. Neurosci.* 11, 367–374.

Petkov, C. I., Logothetis, N. K., and Obleser, J. (2009). Where are the human speech and voice regions, and do other animals have anything like them? *Neuroscientist* 15, 419–429.

Picard, R. W. (2003). Affective computing: challenges. *Int. J. Hum. Comput. Stud.* 59, 55.

Scherer, K. R., Banse, R., Wallbott, H. G., and Goldbeck, T. (1991). Vocal cues in emotion encoding and decoding. *Motiv. Emotion* 15, 123–148.

Severinson-Eklundh, K., Green, A., and Hüttenrauch, H. (2003). Social and collaborative aspects of interaction with a service robot. *Rob. Auton. Syst.* 42, 223–224.

Song, S., Miller, K. D., and Abbott, L. F. (2000). Competitive hebbian learning through spike-timing-dependent synaptic plasticity. *Nat. Neurosci.* 3, 919–926.

Tahon, M., Delaborde, A., and Devillers, L. (2011). "Real-life emotion detection from speech in human-robot interaction: experiments across diverse corpora with child and adult voices," in *INTERSPEECH 2011, 12th Annual Conference of the International Speech Communication Association*, eds P. Cosi, R. De Mori, G. Di Fabbrizio, and R. Pieraccini (Florence), 3121–3124, August 27–31.

Talkin, D. (1995). A robust algorithm for pitch tracking (RAPT). *Speech Coding Synth.* 495, 495–518.

Thibeault, C. M., Harris, F. C. Jr., and Goodman, P. H. (2010a). Breaking the virtual barrier: real-time interactions with spiking neural models. *BMC Neurosci.* 11:73. doi: 10.1186/1471-2202-11-S1-P73

Thibeault, C. M., Sessions, O., Goodman, P., and Harris, F. C. Jr. (2010b). "Real-time emotional speech processing for neurorobotics applications," in *ISCA's 23rd International Conference on Computer Applications in Industry and Engineering (CAINE '10)* (Las Vegas, NV).

Thibeault, C. M., Hegie, J., Jayet Bray, L. C., and Harris, F. C. Jr. (2012). "Simplifying neurorobotic development with NCSTools," in *Conference on Computers and Their Applications (CATA)* (Las Vegas, NV).

Ververidis, D., and Kotropoulos, C. (2006). Emotional speech recognition: resources, features, and methods. *Speech Commun.* 48, 1162–1181.

Ververidis, D., Kotropoulos, C., and Pitas, I. (2004). "Automatic emotional speech classification," in *Acoustics, Speech, and Signal Processing, 2004. Proceedings. (ICASSP '04). IEEE International Conference* (Thessaloniki), I593–I596.

Warren, J. E., Sauter, D. A., Eisner, F., Wiland, J., Dresner, M. A., Wise, S. R., et al. (2006). Positive emotions preferentially engage an auditory-motor mirror system. *J. Neurosci.* 26, 13067–13075.

Wilson, C., Goodman, P., and Harris, F. Jr. (2005). *Implementation of a Biologically Realistic Parallel Neocortical-neural Network Simulator*. Master's thesis, University of Nevada, Reno.

Zhang, L. I., Tao, H. W., Holt, C. E., Harris, W. A., and Poo, M.-M. (1998). A critical window for cooperation and competition among developing retinotectal synapses. *Nature* 395, 37–44.

Zou, Q., and Destexhe, A. (2007). Kinetic models of spike-timing dependent plasticity and their functional consequences in detecting correlations. *Biol. Cybern.* 97, 81–97.

Conflict of Interest Statement: 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.

Received: 29 November 2012; accepted: 02 April 2013; published online: 29 April 2013.

Citation: Jayet Bray LC, Ferneyhough GB, Barker ER, Thibeault CM and Harris FC Jr. (2013) Reward-based learning for virtual neurorobotics through emotional speech processing. Front. Neurorobot. 7:8. doi: 10.3389/fnbot.2013.00008

frontiers in
NEUROROBOTICS

ORIGINAL RESEARCH ARTICLE
published: 05 April 2013
doi: 10.3389/fnbot.2013.00007

Evaluation of linearly solvable Markov decision process with dynamic model learning in a mobile robot navigation task

Ken Kinjo[1,2]*, **Eiji Uchibe**[2] **and Kenji Doya**[1,2]

[1] Neural Computation Laboratory, Graduate School of Information Science, Nara Institute of Science and Technology, Ikoma, Nara, Japan
[2] Neural Computation Unit, Okinawa Institute of Science and Technology, Onna-son, Okinawa, Japan

Edited by:
Florian Röhrbein, Technische Universität München, Germany

Reviewed by:
Marco Mirolli, Istituto di Scienze e Tecnologie della Cognizione, Italy
Evangelos Theodorou, University of Washington, USA

***Correspondence:**
Ken Kinjo, Neural Computation Unit, Okinawa Institute of Science and Technology, 1919-1 Tancha, Onna-son, Okinawa 904-0412, Japan.
e-mail: ken-k@oist.jp

Linearly solvable Markov Decision Process (LMDP) is a class of optimal control problem in which the Bellman's equation can be converted into a linear equation by an exponential transformation of the state value function (Todorov, 2009b). In an LMDP, the optimal value function and the corresponding control policy are obtained by solving an eigenvalue problem in a discrete state space or an eigenfunction problem in a continuous state using the knowledge of the system dynamics and the action, state, and terminal cost functions. In this study, we evaluate the effectiveness of the LMDP framework in real robot control, in which the dynamics of the body and the environment have to be learned from experience. We first perform a simulation study of a pole swing-up task to evaluate the effect of the accuracy of the learned dynamics model on the derived the action policy. The result shows that a crude linear approximation of the non-linear dynamics can still allow solution of the task, despite with a higher total cost. We then perform real robot experiments of a battery-catching task using our Spring Dog mobile robot platform. The state is given by the position and the size of a battery in its camera view and two neck joint angles. The action is the velocities of two wheels, while the neck joints were controlled by a visual servo controller. We test linear and bilinear dynamic models in tasks with quadratic and Guassian state cost functions. In the quadratic cost task, the LMDP controller derived from a learned linear dynamics model performed equivalently with the optimal linear quadratic regulator (LQR). In the non-quadratic task, the LMDP controller with a linear dynamics model showed the best performance. The results demonstrate the usefulness of the LMDP framework in real robot control even when simple linear models are used for dynamics learning.

Keywords: optimal control, linearly solvable Markov decision process, model-based reinforcement learning, model learning, robot navigation

1. INTRODUCTION

When we want to design an autonomous robot that can act optimally in its environment, the robot should solve non-linear optimization problems in continuous state and action spaces. If a precise model of the environment is available, then both optimal control (Todorov, 2006) and model-based reinforcement learning (Barto and Sutton, 1998) give a computational framework to find an optimal control policy which minimizes cumulative costs (or maximizes cumulative rewards). In recent years, reinforcement learning algorithms have been applied to a wide range of neuroscience data (Niv, 2009) and model-based approaches have been receiving attention among researchers who are interested in decision making (Daw et al., 2011; Doll et al., 2012).

However, a drawback is the difficulty to find an optimal policy for continuous states and actions. Specifically, the non-linear Hamilton-Jacobi-Bellman (HJB) equation must be solved in order to derive an optimal policy. Dynamic programming solves the Bellman equation, which is a discrete-time version of the HJB equation, for discrete states and actions problems. Linear Quadratic Regulator (LQR) is one of the well-known optimal control methods for a linear dynamical system with a quadratic cost function. It can handle continuous states and actions, but it is not applicable to non-linear systems.

Recently, a new framework of linearly solvable Markov decision process (LMDP) has been proposed, in which a non-linear Bellman's equation for discrete and continuous systems is converted into a linear equation under certain assumptions on the action cost and the effect action on the state dynamics (Doya, 2009; Todorov, 2009b). In fact, the basis idea of linearization of the HJB equation using logarithmic transformation has been shown in the book written by Flemming and Soner and its connection to risk sensitive control has been discussed in the field of control theory (Fleming and Soner, 2006). Their study has been receiving attention recently in the field of robotics and machine learning fields (Theodorou and Todorov, 2012) because there exist a number of interesting properties in the linearized Bellman equation (Todorov, 2009b). There are two major approaches in LMDP: the path integral approach (Kappen, 2005a,b) and the desirability function approach (Todorov, 2009b). They are closely related and new theoretical findings are reported (Theodorou and Todorov, 2012), but there are some differences in practice. In the path integral approach, the linearized Bellman is computed along paths starting from given initial states using sampling methods. The path integral approach has been successfully applied to learning of stochastic policies for robots with large degrees of freedom (Theodorou et al., 2010; Sugimoto and Morimoto, 2011; Stulp and Sigaud, 2012). The path integral approach is best suited for optimization around stereotyped motion trajectories. However, an additional learning is needed when a new initial state or a new goal state is given. In the value-based approach, an exponentially transformed state value function is defined as the *desirability function* and it is derived from the linearized Bellman's equation by solving an eigenvalue problem (Todorov, 2007) or an eigenfunction problem (Todorov, 2009c; Zhong and Todorov, 2011). One of the benefits of the desirability function approach is its compositionality. Linearity of the Bellman equation enables deriving an optimal policy for a composite task from previously learned optimal policies for basic tasks by linear weighting by the desirability functions (da Silva et al., 2009; Todorov, 2009a). However, the desirability function approach has so far been tested only in simulation. In this study, we test the applicability of the desirability function approach to real robot control.

In order to apply the LMDP framework to real robot applications, the environmental dynamics should be estimated through the interaction with the environment. This paper proposes a method which integrates model learning with the LMDP framework and investigates how the accuracy of the learned model affects that of the desirability function, the corresponding policy, and the task performance. Although Burdelis and Ikeda proposed a similar approach for the system with discrete states and actions (Burdelis and Ikeda, 2011), it is not applicable to a continuous domain. We test the proposed method in two tasks. The first task is a well-known benchmark, the pole swing-up problem. We use linear and non-linear models for approximation of the environmental dynamics and investigate how the accuracy of the dynamics model affects the estimated desirability function and the corresponding policy. The second task is a visually guided navigation problem using our Spring Dog robot which has six degrees of freedom. The landmark with the LED is located in the environment and the Spring Dog should approach the landmark. We compare linear and bilinear dynamics models with quadratic and Gaussian state cost functions. Experimental results showed that the LMDP framework with model learning is applicable to real robot learning even when simple dynamics models are used.

2. MATERIALS AND METHODS

2.1. LINEARLY SOLVABLE MARKOV DECISION PROCESS

At first, we show how a non-linear Bellman's equation can be made linear under the LMDP setting formulated by Todorov (2009b). Let $\mathcal{X} \subseteq \mathbb{R}^{N_{\boldsymbol{x}}}$ and $\mathcal{U} \subseteq \mathbb{R}^{N_{\boldsymbol{u}}}$ be the continuous state and continuous action spaces, where $N_{\boldsymbol{x}}$ and $N_{\boldsymbol{u}}$ are the dimensionality of the spaces, respectively. At time t, the robot observes the environmental current state $\boldsymbol{x}(t) \in \mathcal{X}$ and executes action $\boldsymbol{u}(t) \in \mathcal{U}$. Consequently, the robot receives an immediate cost $c(\boldsymbol{x}(t), \boldsymbol{u}(t))$ and the environment makes a state transition according to the following continuous-time stochastic differential equation,

$$d\boldsymbol{x} = \boldsymbol{a}(\boldsymbol{x})dt + \boldsymbol{B}(\boldsymbol{x})(\boldsymbol{u}dt + \sigma d\boldsymbol{\omega}), \tag{1}$$

where $\boldsymbol{\omega} \in \mathbb{R}^{N_{\boldsymbol{u}}}$ and σ denote Brownian noise and a scaling parameter for the noise, respectively. $\boldsymbol{a}(\boldsymbol{x})$ describes the passive dynamics of the system while $\boldsymbol{B}(\boldsymbol{x})$ represents the input-gain matrix. Note that Equation (1) is generally non-linear with respect to the state $\boldsymbol{x}$ but linear with respect to the action $\boldsymbol{u}$. It is convenient to represent Equation (1) in discrete time. By discretizing the time axis with step h, we obtain the following transition probability,

$$p^{\boldsymbol{u}_k}(\boldsymbol{x}_{k+1}|\boldsymbol{x}_k) = \mathcal{N}(\boldsymbol{x}_{k+1}|\boldsymbol{\mu}(\boldsymbol{x}_k, \boldsymbol{u}_k) + \boldsymbol{x}_k, h\boldsymbol{\Sigma}(\boldsymbol{x})), \tag{2}$$

where $\mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes a Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$, and

$$\boldsymbol{\mu}(\boldsymbol{x}, \boldsymbol{u}) = h(\boldsymbol{a}(\boldsymbol{x}) + \boldsymbol{B}(\boldsymbol{x})\boldsymbol{u}), \tag{3}$$

$$\boldsymbol{\Sigma}(\boldsymbol{x}) = \sigma \boldsymbol{B}(\boldsymbol{x})^{\mathrm{T}}\boldsymbol{B}(\boldsymbol{x}), \tag{4}$$

where $\boldsymbol{\mu}(\boldsymbol{x}, \boldsymbol{u})$ can be regarded as a deterministic state transition function. Note that $\boldsymbol{x}_k = \boldsymbol{x}(hk)$ and $\boldsymbol{u}_k = \boldsymbol{u}(hk)$. It should be noted that a state transition probability is defined as an uncontrolled probability when no control is applied ($\boldsymbol{u} = \boldsymbol{0}$), and otherwise, it is called a controlled probability.

A control policy or controller $\pi(\boldsymbol{u}|\boldsymbol{x})$ is defined as a probability of selecting the action $\boldsymbol{u}$ at state $\boldsymbol{x}$. When the goal of the robot is to find an optimal control policy π^* that can lead the robot to the desired state $\boldsymbol{x}_g \in \mathcal{X}_g \subseteq \mathcal{X}$, the objective function is formulated as minimization of the expected value of cumulative costs,

$$v^{\pi}(\boldsymbol{x}) = \mathbb{E}\left[\sum_{k=1}^{T_g - 1} c(\boldsymbol{x}_k, \pi(\boldsymbol{x}_k)) + g(\boldsymbol{x}_g)\right], \tag{5}$$

where and $c(\boldsymbol{x}, \boldsymbol{u})$ and $g(\boldsymbol{x})$, respectively denote the immediate and terminal cost. T_g represents an arrival time. $v^{\pi}(\boldsymbol{x})$ is known as a cost-to-go or value function. The optimal value function is the minimal expected cumulative cost defined by

$$v^*(\boldsymbol{x}) = \min_{\pi} v^{\pi}(\boldsymbol{x}). \tag{6}$$

It is known that the optimal value function satisfies the following Bellman's equation

$$v^*(\boldsymbol{x}) = \min_{\boldsymbol{u}} \left(c(\boldsymbol{x}, \boldsymbol{u}) + \mathbb{E}_{\boldsymbol{x}' \sim p^{\boldsymbol{u}}(\cdot|\boldsymbol{x})} v^*(\boldsymbol{x}')\right) \tag{7}$$

$$v^*(\boldsymbol{x}_g) = g(\boldsymbol{x}_g), \quad \boldsymbol{x}_g \in \mathcal{X}_g.$$

Since Equation (7) is non-linear, it is difficult to solve the optimal value function in general. However, the Bellman's equation is simplified if it is assumed that the immediate cost function is represented by

$$c(\boldsymbol{x}, \boldsymbol{u}) = hq(\boldsymbol{x}) + \mathrm{KL}(p^{\boldsymbol{u}}(\cdot|\boldsymbol{x}) \| p^{\boldsymbol{0}}(\cdot|\boldsymbol{x})), \tag{8}$$

where $q(\boldsymbol{x})$ is a non-negative state cost function and the second term on the right hand side of Equation (8) is a control cost given as the KL-divergence between controlled and uncontrolled probability distributions.[1] In this case, the non-linear Bellman's equation is converted to the following linear equation

$$\begin{aligned} z(\boldsymbol{x}) &= \exp(-hq(\boldsymbol{x}))\mathcal{G}[z](\boldsymbol{x}) \qquad (9) \\ z(\boldsymbol{x}_g) &= \exp(-g(\boldsymbol{x}_g)), \quad \boldsymbol{x}_g \in \mathcal{X}_g, \end{aligned}$$

where $z(\boldsymbol{x})$ is the desirability function defined by

$$z(\boldsymbol{x}) = \exp(-v^*(\boldsymbol{x})). \tag{10}$$

Hereafter, Equation (9) is called a linearized Bellman's equation. The operator $\mathcal{G}$ shown on the right hand side of the linearized Bellman's Equation (9) is the integral operator given by

$$\mathcal{G}[f](\boldsymbol{x}) = \int p^0(\boldsymbol{x}'|\boldsymbol{x}) f(\boldsymbol{x}') d\boldsymbol{x}'. \tag{11}$$

It should be noted that Equation (9) is always satisfied by the trivial solution ($z(\boldsymbol{x}) \equiv 0$ for all $\boldsymbol{x}$) if no boundary conditions are introduced.

2.2. LEARNING MODEL PARAMETERS

In the LMDP framework, the system dynamics (Equation 1) are assumed to be known in advance. When they are unknown, estimation of the dynamics is required from samples collected by the passive dynamics. Many methods exist which can estimate the system dynamics (Nguyen-Tuong and Peters, 2011; Sigaud et al., 2011), we adopt a simple least squares method to estimate $\boldsymbol{a}(\boldsymbol{x})$ and $\boldsymbol{B}(\boldsymbol{x})$ with basis functions. Specifically, we estimate a deterministic state transition (Equation 3). It should be noted that the scale parameter of noise σ is generally unknown, but it is determined by the experimenters here since it can be regarded as the parameter that controls exploration of the environment.

Let us suppose that the deterministic state transition $\boldsymbol{\mu}(\boldsymbol{x}, \boldsymbol{u})$ is approximated by the linear function with N_φ basis functions $\varphi_i(\boldsymbol{x}, \boldsymbol{u})$,

$$\boldsymbol{\mu}(\boldsymbol{x}, \boldsymbol{u}; \boldsymbol{W}) = \boldsymbol{W}^{\mathrm{T}} \boldsymbol{\varphi}(\boldsymbol{x}, \boldsymbol{u}). \tag{12}$$

where $\boldsymbol{W}$ is a weight matrix and $\boldsymbol{\varphi}(\boldsymbol{x}, \boldsymbol{u})$ is a vector consisting of basis functions. Suppose that the training samples $\{\boldsymbol{x}_1, \boldsymbol{u}_1, \ldots, \boldsymbol{x}_{N_s}, \boldsymbol{u}_{N_s}, \boldsymbol{x}_{N_s+1}\}$ are obtained by the passive dynamics. The objective function of model learning is given by the following sum-of-squares error function,

$$E = \frac{1}{2} \sum_{k=1} \left\{ \Delta \boldsymbol{x}_k - \boldsymbol{W}^{\mathrm{T}} \boldsymbol{\varphi}(\boldsymbol{x}_k, \boldsymbol{u}_k) \right\}^2, \tag{13}$$

where $\Delta \boldsymbol{x}_k = \boldsymbol{x}_{k+1} - \boldsymbol{x}_k$. Setting $\partial E / \partial \boldsymbol{W} = \boldsymbol{0}$ yields

$$\boldsymbol{W} = (\boldsymbol{\Phi}^{\mathrm{T}} \boldsymbol{\Phi})^{-1} \boldsymbol{\Phi}^{\mathrm{T}} \Delta \boldsymbol{X}, \tag{14}$$

where $\Delta \boldsymbol{X}$ is the matrix whose a row vector consisted of state transition in each sample $\Delta \boldsymbol{x}_k$ and $\boldsymbol{\Phi}$ is also the matrix whose a column vector consisted of the basis functions in each sample $\boldsymbol{\varphi}(\boldsymbol{x}_k, \boldsymbol{u}_k)$. The detail is as follow,

$$\Delta \boldsymbol{X} = \left[\Delta \boldsymbol{x}_1 \cdots \Delta \boldsymbol{x}_{N_s} \right]^{\mathrm{T}}, \quad \boldsymbol{\Phi} = \left[\boldsymbol{\varphi}(\boldsymbol{x}_1, \boldsymbol{u}_1) \cdots \boldsymbol{\varphi}(\boldsymbol{x}_{N_s}, \boldsymbol{u}_{N_s}) \right].$$

2.3. LEARNING A DESIRABILITY FUNCTION

The desirability function is approximated by

$$z(\boldsymbol{x}; \boldsymbol{w}, \boldsymbol{\theta}) = \sum_{i=1}^{N_z} w_i f(\boldsymbol{x}, \boldsymbol{\theta}_i) = \boldsymbol{w}^{\top} \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\theta}), \tag{15}$$

where w_i is a weight, $\boldsymbol{w}$ is the weight vector $[w_1, \ldots, w_{N_z}]^{\mathrm{T}}$, $f(\boldsymbol{x}, \boldsymbol{\theta}_i)$ is a basis function parameterized by $\boldsymbol{\theta}_i$, and $\boldsymbol{f}(\boldsymbol{x}; \boldsymbol{\theta})$ is the vector consisting of basis functions $[f(\boldsymbol{x}; \theta_1), \ldots, f(\boldsymbol{x}; \theta_{N_z})]^{\mathrm{T}}$. We adopt an unnormalized Gaussian function as Todorov suggested (Todorov, 2009c):

$$f(\boldsymbol{x}; \boldsymbol{\theta}_i) = \exp\left(-\frac{1}{2} (\boldsymbol{x} - \boldsymbol{m}_i)^{\mathrm{T}} \boldsymbol{S}_i (\boldsymbol{x} - \boldsymbol{m}_i) \right), \quad \boldsymbol{\theta}_i = \{\boldsymbol{m}_i, \boldsymbol{S}_i\} \tag{16}$$

where $\boldsymbol{m}_i$ and $\boldsymbol{S}_i$ denote a center position and a precision matrix of the i-th basis function, respectively. One advantage of using the Gaussian function that the integral operator (Equation 11) can be calculated analytically as follows:

$$\mathcal{G}[f_i](\boldsymbol{x}) = |\boldsymbol{V}_i|^{-\frac{1}{2}} \exp\left(-\frac{1}{2} \left(\boldsymbol{y} - \boldsymbol{m}_i\right)^{\mathrm{T}} \boldsymbol{H}_i \left(\boldsymbol{y} - \boldsymbol{m}_i\right) \right), \tag{17}$$

where $\boldsymbol{y}(\boldsymbol{x}) = \boldsymbol{x} + \boldsymbol{\mu}(\boldsymbol{x}, \boldsymbol{0}),\ f_i = f(\boldsymbol{x}, \boldsymbol{\theta}_i)$ for brevity and

$$\begin{aligned} \boldsymbol{H}_i &= \boldsymbol{S}_i - \boldsymbol{S}_i \boldsymbol{C} \boldsymbol{V}_i^{-1} \boldsymbol{C}^{\mathrm{T}} \boldsymbol{S}_i, \quad \boldsymbol{V}_i = \boldsymbol{I} + \boldsymbol{C}^{\mathrm{T}} \boldsymbol{S}_i \boldsymbol{C}, \\ \boldsymbol{C} &= \sigma h^{1/2} \boldsymbol{B}. \end{aligned}$$

It should be noted that $\boldsymbol{y}, \boldsymbol{H}_i, \boldsymbol{V}_i, \boldsymbol{C}$ are functions of $\boldsymbol{x}$.

The desirability function (Equation 15) should satisfy the linearized Bellman's equation (9). Therefore, in order to optimize $\boldsymbol{w}$ and $\boldsymbol{\theta}$ we can construct the following objective function for given collocation states $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{N_c}\}$:

$$e = \|\boldsymbol{r}(\boldsymbol{w}, \boldsymbol{\theta})\|^2, \quad \boldsymbol{r}(\boldsymbol{w}, \boldsymbol{\theta}) = \begin{bmatrix} \boldsymbol{F}(\boldsymbol{\theta}) - \boldsymbol{G}(\boldsymbol{\theta}) \\ \boldsymbol{f}(\boldsymbol{x}_g; \boldsymbol{\theta})^{\mathrm{T}} \end{bmatrix} \boldsymbol{w} - \begin{bmatrix} \boldsymbol{0} \\ \exp(-g(\boldsymbol{x}_g)) \end{bmatrix}, \tag{18}$$

[1] The Kullback–Leibler (KL) divergence measures the difference between two distributions. If two distributions are the same, the KL-divergence becomes 0. In the LMDP, the control cost is defined by how certain control $\boldsymbol{u}$ affects on state transition probability.

where $\boldsymbol{F}(\boldsymbol{\theta})$ and $\boldsymbol{G}(\boldsymbol{\theta})$ are $N_c \times N_z$ matrices and their (n, i) components are defined by

$$[\boldsymbol{F}(\boldsymbol{\theta})]_{ni} = f_i(\boldsymbol{x}_n), \tag{19}$$

$$[\boldsymbol{G}(\boldsymbol{\theta})]_{ni} = \exp(-hq(\boldsymbol{x}_n))\mathcal{G}[f_i](\boldsymbol{x}_n). \tag{20}$$

The objective function (Equation 18) is a quadratic function with respect to $\boldsymbol{w}$ and a non-linear function with respect to $\boldsymbol{\theta}$. See Appendix A for optimization of $\boldsymbol{w}$ and $\boldsymbol{\theta}$.

2.4. OPTIMAL CONTROL POLICY

In the LMDP framework, the optimal control policy is given by

$$p^{\boldsymbol{u}^*}(\boldsymbol{x}'|\boldsymbol{x}) = \frac{p^0(\boldsymbol{x}'|\boldsymbol{x})z(\boldsymbol{x}')}{\mathcal{G}[z](\boldsymbol{x})}. \tag{21}$$

Specifically, when the dynamics are represented in the form of the stochastic differential equation (1) and the basis function of the approximated desirability function is Gaussian, then the optimal control policy is represented by

$$\boldsymbol{u}^*(\boldsymbol{x}) = \sigma \sum_{i=1}^{N_z} \frac{w_i \mathcal{G}\left[f_i(\boldsymbol{x})\right]}{\sum_{k=1}^{N_z} w_k \mathcal{G}\left[f_k(\boldsymbol{x})\right]} \boldsymbol{d}_i(\boldsymbol{x}), \tag{22}$$

$$\boldsymbol{d}_i(\boldsymbol{x}) = \boldsymbol{V}_i^{-1} \boldsymbol{C}^{\mathrm{T}} \boldsymbol{S}_i \left(\boldsymbol{m}_i - \boldsymbol{x} - h\boldsymbol{a}(\boldsymbol{x})\right).$$

See Todorov (2009c) in more detail.

2.5. EXPERIMENT

In this paper, we conduct two experiments to evaluate the LMDP framework with model learning. One is a swing-up pole task in simulation. The other is a visually-guided navigation task using a real robot.

2.5.1. Swing-up pole

To verify that an appropriate control policy can be derived based on estimated dynamics, we conducted a computer simulation of the swing-up pole task. In the simulation, the one side of pole was fixed and the pole could rotate in plane around the fixed point as shown in **Figure 1**. The goal was to swing the pole to an upward position and stop at this position. The state in this task consisted of the vertical angle ϑ and the angular velocity $\dot{\vartheta}$, the origin of the state space was set at the goal position. It should be noted that ϑ was normalized to be in the range $(-\pi,\ \pi]$ (rad) while $\dot{\vartheta}$ was bounded: $\dot{\vartheta} \in [-4\pi,\ 4\pi]$ (rad /s). The control input and noise affected the torque of the pole. Therefore, the pole is assumed to obey the following stochastic state equation,

$$d\,\vartheta = \dot{\vartheta} d\,t \tag{23}$$

$$d\,\dot{\vartheta} = \left(m\frac{g}{l}\sin(\vartheta) - k\dot{\vartheta}\right) d\,t + u d\,t + \sigma d\,\omega,$$

where l, m, g, and k denote the length of the pole, mass, gravitational acceleration and coefficient of friction, respectively.

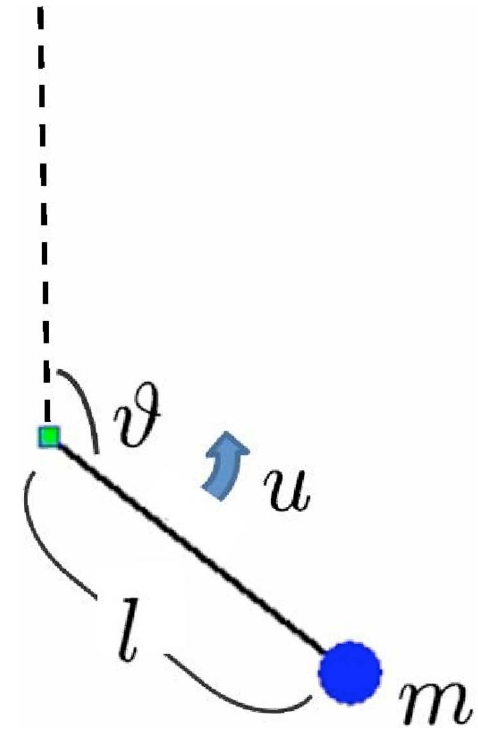


FIGURE 1 | Swing-up pole task.

The above state equation is represented in the form of Equation (1) as follows;

$$\boldsymbol{a}(\boldsymbol{x}) = \left[\dot{\vartheta},\ m\tfrac{g}{l}\sin(\vartheta) - k\hat{\vartheta}\right]^{\mathrm{T}}, \quad \boldsymbol{B} = \left[0,\ 1\right]^{\mathrm{T}}.$$

It should be note that the passive dynamics $\boldsymbol{a}(\boldsymbol{x})$ is a non-linear vector function of $\boldsymbol{x}$ while $\boldsymbol{B}$ is a constant vector. In this simulation, the physical parameters were $l = 1$ (m), $m = 1$ (kg), $g = 9.8$ (kg/s^2) and $k = 0.05$ (kg m^2/s). The state equation was discretized in time with a time step of $h = 10$ (ms) and the noise scale was set to $\sigma = 4$. The state cost was defined so that it was zero at the goal state, using the following unnormalized Gaussian function,

$$q(\boldsymbol{x}) = \left(1 - \exp\left(\boldsymbol{x}^{\mathrm{T}} \Sigma_{\mathrm{cost}}^{-1} \boldsymbol{x}\right)\right), \tag{24}$$

where diag $(\Sigma_{\mathrm{cost}}) = [0.1,\ 1.6]$.

As written in section 2.2, the weight matrix was estimated by Equation (14). In the sample acquisition phase we repeated simulations sufficiently, each simulation started from different initial states to avoid unevenly distributed samples. As a result, $N = 1000$ samples were extracted randomly as a training data set.

In this simulation, we prepared two types of basis functions $\boldsymbol{\varphi}(\boldsymbol{x}, \boldsymbol{u})$, as shown in **Table 1**, for approximation of the environmental dynamics. The first was a simple linear model with respect to $\boldsymbol{x}$ and $\boldsymbol{u}$ while the second model added the normalized radial

Table 1 | Basis functions used in the swing-up pole simulation.

	$\varphi(\boldsymbol{x}, \boldsymbol{u})$
Linear model	$[\boldsymbol{x}^{\mathrm{T}} \ \ \boldsymbol{u}^{\mathrm{T}}]^{\mathrm{T}}$
Linear-NRBF model	$[\boldsymbol{x}^{\mathrm{T}} \ \ \psi_1(\boldsymbol{x}) \ \ \psi_2(\boldsymbol{x}) \ \ \cdots \ \ \psi_M(\boldsymbol{x}) \ \ \boldsymbol{u}^{\mathrm{T}}]^{\mathrm{T}}$

basis functions (NRBF) $\psi_i(\boldsymbol{x}, \boldsymbol{u})$ to the linear model,

$$\psi_i(\boldsymbol{x}) = \frac{\exp\left(-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu}_i)^{\mathrm{T}} \boldsymbol{\Sigma}_{\psi_i}^{-1} (\boldsymbol{x} - \boldsymbol{\mu}_i)\right)}{\sum_k \exp\left(-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu}_k)^{\mathrm{T}} \boldsymbol{\Sigma}_{\psi_k}^{-1} (\boldsymbol{x} - \boldsymbol{\mu}_k)\right)}. \tag{25}$$

The centers, $\boldsymbol{\mu}_i$, of the basis functions, $\psi_i(\boldsymbol{x}, \boldsymbol{u})$, were determined by K-means clustering among the states of the training data. The covariance matrices $\boldsymbol{\Sigma}_{\psi_i}$ were determined experimentally and set to $\mathrm{diag}(\boldsymbol{\Sigma}_{\psi_i}) = [\pi/4, \ \pi]$. In the linear-NRBF model, $N_\psi = 25$ basis functions were used.

The set of collocation states $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{N_s}\}$, which were required to optimize the parameters of the desirability function, were uniformly distributed in the state space. The centers $\boldsymbol{m}_i$ of the basis functions $f_i(\boldsymbol{x})$ were initialized so as to distribute them uniformly in the state space. On the other hand, the covariance matrices $\boldsymbol{S}_i$ were determined empirically and set to $\mathrm{diag}([16, \ 1])$. The optimal control policy $\boldsymbol{u}^*(\boldsymbol{x})$ was derived from Equation (22).

2.5.2. Visually-guided navigation task

To evaluate the performance of the optimal control policy derived from the estimated dynamics and the desirability function, we conducted a visual navigation task using a wheel type robot called the Spring Dog. **Figure 2** shows the Spring Dog and the battery pack in the experimental field. The Spring Dog has six degrees of freedom: two fore legs, two rear wheels, and a pan-tilt camera head. There are several sensors such as a 3D accelerometer, a combined 3D gyroscope, and a USB camera mounted on the head, and so on. Three-color LED is attached to the top of the battery pack.

Figure 3 shows the control diagram, where three control policies were implemented in this experiment. The first one was a visual servoing controller, which controlled the camera head so as to keep tracking the battery pack continuously. The second one was a navigation controller using the two rear wheels, this was optimized by the LMDP framework. In other words, the navigation controller controlled the left and right wheels in order to move around in the environment. The desired velocities of left and right wheels correspond to control input $\boldsymbol{u}$ in Equation (1). The last one was a seeking behavior, in which the Spring Dog explored the environment to find the battery pack when the robot lost track of it. The navigation controller learned by the LMDP framework while the visual servoing and searching controllers were designed by the experimenters.

To realize a visually-guided navigation task, image binarization was applied to a captured image in order to separate the battery pack with the green LED from background. Some image features were calculated as shown in **Figure 4**. The state space consists of six variables described below: the center position of the battery pack (extracted pixels) in the image plane (x_{cx}, x_{cy}), average of absolute values around the center in horizontal and vertical axes of the extracted pixels (x_{ax}, x_{ay}), and the current joint angles of the neck controlled by the visual servoing controller. The state and action were summarized as follows:

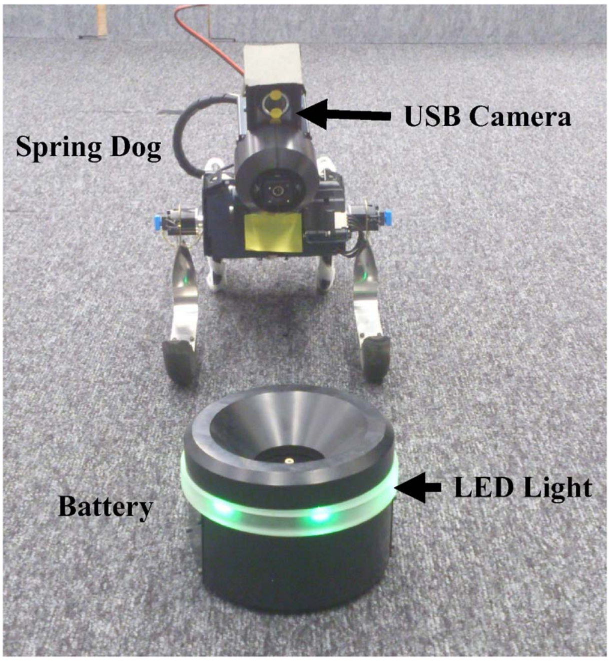


FIGURE 2 | Spring Dog, wheel typed robot and the battery pack.

$$\boldsymbol{x} = [x_{cx}, x_{cy}, x_{ax}, x_{ay}, x_{\mathrm{tilt}}, x_{\mathrm{pan}}]^{\mathrm{T}}, \quad \boldsymbol{u} = [u_{\mathrm{left}}, u_{\mathrm{right}}]^{\mathrm{T}}.$$

It should be noted that each value was scaled as follow,

$$\begin{aligned} -1 \leq x_{cx}, x_{cy}, x_{\mathrm{tilt}}, x_{\mathrm{pan}} \leq 1, \\ 0 \leq x_{ax}, x_{ay} \leq 1, \\ -1 \leq u_{\mathrm{left}}, u_{\mathrm{right}} \leq 1. \end{aligned}$$

The desired state, $\boldsymbol{x}_g$, was set to comprise of both a posture and location which allowed the Spring Dog to successfully capture of the battery. The view feed from the USB camera allowed recognition of the desired proximity and posture, as shown in **Figure 2**.

Two types of state dependent cost functions $q_1(\boldsymbol{x})$ and $q_2(\boldsymbol{x})$ were considered in the experiment. Each cost function was defined to be zero at the goal state as follows,

$$q_1(\boldsymbol{x}) = \alpha \left(\boldsymbol{x} - \boldsymbol{x}_g\right)^{\mathrm{T}} \boldsymbol{\Sigma}_{\mathrm{cost}}^{-1} \left(\boldsymbol{x} - \boldsymbol{x}_g\right) \tag{26}$$

$$q_2(\boldsymbol{x}) = \alpha \left(1 - \exp\left(-\left(\boldsymbol{x} - \boldsymbol{x}_g\right)^{\mathrm{T}} \boldsymbol{\Sigma}_{\mathrm{cost}}^{-1} \left(\boldsymbol{x} - \boldsymbol{x}_g\right)\right)\right), \tag{27}$$

where α was a scaling constant.

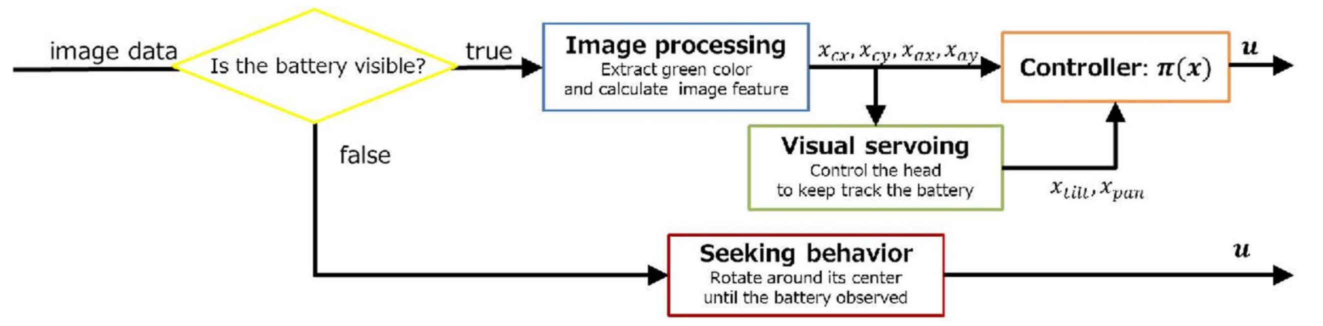


image data
Is the battery visible?
true
false
Image processing
Extract green color and calculate image feature
$x_{cx}, x_{cy}, x_{ax}, x_{ay}$
Controller: $\pi(x)$
u
Visual servoing
Control the head to keep track the battery
x_{tilt}, x_{pan}
Seeking behavior
Rotate around its center until the battery observed
u

FIGURE 3 | Control diagram in the Spring Dog.

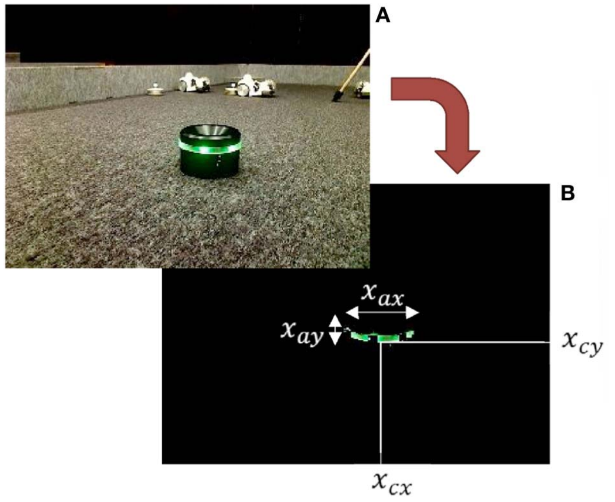


FIGURE 4 | Image binarization and image features. (A) Original captured image. **(B)** Binarized image.

Table 2 | Basis functions used in the robot experiment.

	$\varphi(\boldsymbol{x}, \boldsymbol{u})$
Linear model	$\left[(\boldsymbol{x}-\boldsymbol{x}_g)^{\mathrm{T}} \ \boldsymbol{u}^{\mathrm{T}}\right]^{\mathrm{T}}$
Bilinear model	$\left[(\boldsymbol{x}-\boldsymbol{x}_g)^{\mathrm{T}} \ u_{\text{left}}(\boldsymbol{x}-\boldsymbol{x}_g)^{\mathrm{T}} \ u_{\text{right}}(\boldsymbol{x}-\boldsymbol{x}_g)^{\mathrm{T}} \ \boldsymbol{u}^{\mathrm{T}}\right]^{\mathrm{T}}$

Procedure 1 | Setting initial position of the centers of the basis functions, $\mathbf{M}_{init}$.

```
Input: The date set of state, $\mathcal{D}_x$.
Output: The set of initial center positions, $\mathbf{M}_{init}$
  $\mathbf{M}_{init} \leftarrow \emptyset$
  while $\mathcal{D}_x \neq \emptyset$ do
    $x = \text{ChooseSample}(\mathcal{D}_x)$
    $\mathcal{D}_x \leftarrow \mathbf{X}_{\mathcal{D}} - \{x\}$
    if $\forall i \ f_i(x; m_i) < \tau$ or $\mathbf{M}_{init} = \emptyset$ then
      $\mathbf{M}_{init} \leftarrow \mathbf{M}_{init} \bigcup \{x\}$
    end if
  end while
  return $\mathbf{M}_{init}$
```

Next we explain the procedure for estimation of visual-motor dynamics. At first, the Spring Dog moved around using the fixed stochastic policy and obtained data. In the experiment, the control cycle was required to keep $h = 300 \pm 60$ (ms), but it was sometimes violated interference from other processes. To deal with this problem in sampling, we rejected the corresponding data. In addition, If the target became invisible, or the tilt or pan angle reached by setting, its limitation, the corresponding data was rejected from samples also. As a result, we obtained the data set, $\mathcal{D} = \left[\left[\boldsymbol{x}_1^{\mathrm{T}} \ \boldsymbol{u}_1^{\mathrm{T}}\right]^{\mathrm{T}}, \ldots, \left[\boldsymbol{x}_{N_{\mathcal{D}}}^{\mathrm{T}} \ \boldsymbol{u}_{N_{\mathcal{D}}}^{\mathrm{T}}\right]^{\mathrm{T}}\right]$. After normalizing this data set, the environmental dynamics were estimated as described in section 2.2.

In this experiment, we used two types of basis functions $\boldsymbol{\varphi}(\boldsymbol{x}, \boldsymbol{u})$, as shown in **Table 2**, to estimate visual-motor dynamics. If we apply the linear model for visual-motor dynamics and use a quadratic state cost function in Equation (26), the problem setting is identical to that of Linear Quadratic Regulator (LQR). Therefore, we can confirm that the LMDP finds the same optimal policy as LQR.

As well as the swing-up pole task, collocation states $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{N_s}\}$ were uniformly distributed in the state space, and the covariance matrices $\boldsymbol{S}_i$ were determined by hand. Moreover, only centers of basis functions of desirability were updated and covariance matrices were fixed in the experiment. The optimal control policy $\boldsymbol{u}^*(\boldsymbol{x})$ was derived from Equation (22). The initial position of the center $\boldsymbol{m}_i$ in each basis function $f_i(\boldsymbol{x})$ was taken from the data set of state, $\mathcal{D}_{\boldsymbol{x}} = \left[\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{N_{\mathcal{D}}}\right]$, which was extracted state data from the data set $\mathcal{D}$. However, it was not appropriate for the computational resources of the real robot to use all of the data. For this reason, the set of initial positions of the centers of the basis functions, $\mathbf{M}_{init} = [\boldsymbol{m}_1, \ldots, \boldsymbol{m}_{N_z}]$, were chosen from the data set of state $\mathcal{D}_{\boldsymbol{x}}$ following **Procedure 1**. As a result, at least one of the basis functions could return the value, which was over the threshold, τ, for every samples.

As already explained, to verify that LMDP can be apply to nonlinear state transition system and non-quadratic cost function and the obtained controller performs optimal. In the experiment we tested the following four conditions:

1. Linear model + quadratic state cost.
2. Bilinear model + quadratic state cost.
3. Linear model + Gaussian based state cost (non-quadratic).
4. Bilinear model + Gaussian based state cost (non-quadratic).

Note that LQR can be applicable in the first condition. Therefore, LQR was also implemented to compare the result of the LMDP framework to the ground truth obtained from LQR in the first condition.

3. RESULTS

3.1. COMPUTER SIMULATION

As described in the section 2.5.1, we used the linear and the linear-NRBF models to approximate the environmental dynamics of the swing-up pole. To evaluate the accuracy of estimation using these models, we measured the estimation errors. We extracted $N = 500$ samples randomly as a test data set and then calculated the estimates of the deterministic state transition $\boldsymbol{\mu}(\boldsymbol{x}, \boldsymbol{u}; \boldsymbol{W})$ when two models were applied, respectively. After that, we computed the mean squared error (MSE) of each component,

$$\text{MSE of the } k\text{-th component} = \frac{1}{N}\sum_{n=1}^{N}\left(\Delta x_{kn} - \boldsymbol{w}_k \boldsymbol{\varphi}(\boldsymbol{x}_n, \boldsymbol{u}_n)\right)^2, \tag{28}$$

where $\boldsymbol{w}_k$ denotes the elements of k-th row in the weight matrix $\boldsymbol{W}$.

Figure 5 shows the MSE of the angle and angular velocity component. According to the this result, the estimation of the angle component was quite accurate in both models because it was deterministic transition. On the other hand, the estimation of the angular velocity component was inaccurate as compared with the angle component since it was a stochastic state transition. According to Equations 2, 3 and the parameter setting of the time step, $h = 10$ (ms), the noise scale, $\sigma = 4$, and $\mathbf{B} = [0,\ 1]^{\mathrm{T}}$, the covariance matrix was derived $\operatorname{diag}(\Sigma) = [0,\ 0.04]$. The covariance matrix affects to the MSE by square, the MSE between real deterministic state transition and an observed temporal state transition should be at least 1.6×10^{-3}. The MSE of angular velocity component in the linear-NRBF model was also 1.6×10^{-3}, it was suggested that most of the error was caused by noise. Consequently, This result suggested that the environmental dynamics were accurately approximated by the linear-NRBF model. The estimated input gain matrices were given by

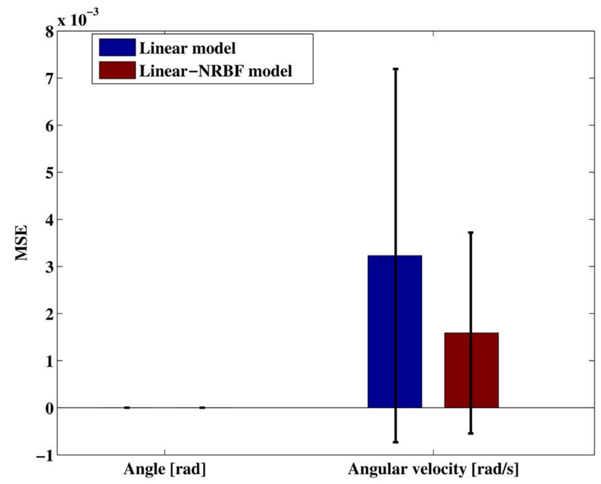


FIGURE 5 | Mean squared error of the joint angle and angular velocity. Each error bar represents the standard deviation.

$$\boldsymbol{B}_{\text{linear}} = \begin{bmatrix} 0.0000 \\ 0.9965 \end{bmatrix}, \quad \boldsymbol{B}_{\text{linear-NRBF}} = \begin{bmatrix} 0.0000 \\ 1.0113 \end{bmatrix}.$$

they were very close to the true matrix $\boldsymbol{B} = [0,\ 1]^{\mathrm{T}}$.

The desirability function was optimized using the estimated dynamics and the control policy derived from the obtained desirability function. **Figure 6** displays the results where the left panels show the desirability function $z(\boldsymbol{x})$ and the right panels show the learned policy $\boldsymbol{u}^*(\boldsymbol{x})$. The black line in the right panels shows a typical trajectory of learned behaviors starting from $\boldsymbol{x} = [\pi,\ 0]^{\mathrm{T}}$. The top panels of **Figure 6** display the results using the true dynamics. It should be noted that the desirability function is discontinuous around the central diagonal band since this system is under-actuated. Simulation results using the linear and linear-NRBF models are shown in the middle and bottom panels of **Figure 6**, respectively. As compared with the result based on the true dynamics, both of the linear and linear-NRBF models could approximate the desirability function. However, the policy obtained by the linear model was worse than that by the linear-NRBF model.

To evaluate the performance in more detail, we measured the cumulative costs corresponding to each of the obtained policies. In this test simulation, the initial state was set to $\boldsymbol{x} = [\pi,\ 0]^{\mathrm{T}}$ which corresponds to the bottom position. **Figure 7** shows mean cumulative costs of 50 episodes, each episode was terminated when the pole arrived at the goal state or the duration reached was over 20 (s) (2000 step). Note that the immediate cost in each step was calculated by $c(\boldsymbol{x}, \boldsymbol{u}) = h\left(q(\boldsymbol{x}) + \frac{1}{2\sigma^2}\|\boldsymbol{u}\|^2\right)$.

Figure 7 compares the cumulative costs among the three policies. Not surprisingly, the control policy derived from the true dynamics achieved the best performance. It should be noted that the control policy based on the dynamics estimated with the linear-NRBF model produced a comparable performance, and it was better than the performance of the linear model. As discussed in the previous section, the linear-NRBF model gave more correct estimation than the linear model. Consequently, these results suggest that we can obtain the better control policy by forming more accurate estimates.

3.2. REAL ROBOT EXPERIMENT

As described in section 2.5.2, we used the linear and bilinear models for environmental dynamics approximation. After the data acquisition phase, we obtained $N_{\mathcal{D}} = 9509$ samples and we extracted $N = 2500$ samples for a test data set, the rest of samples were used as a training data set. As well as the swing-up the pole

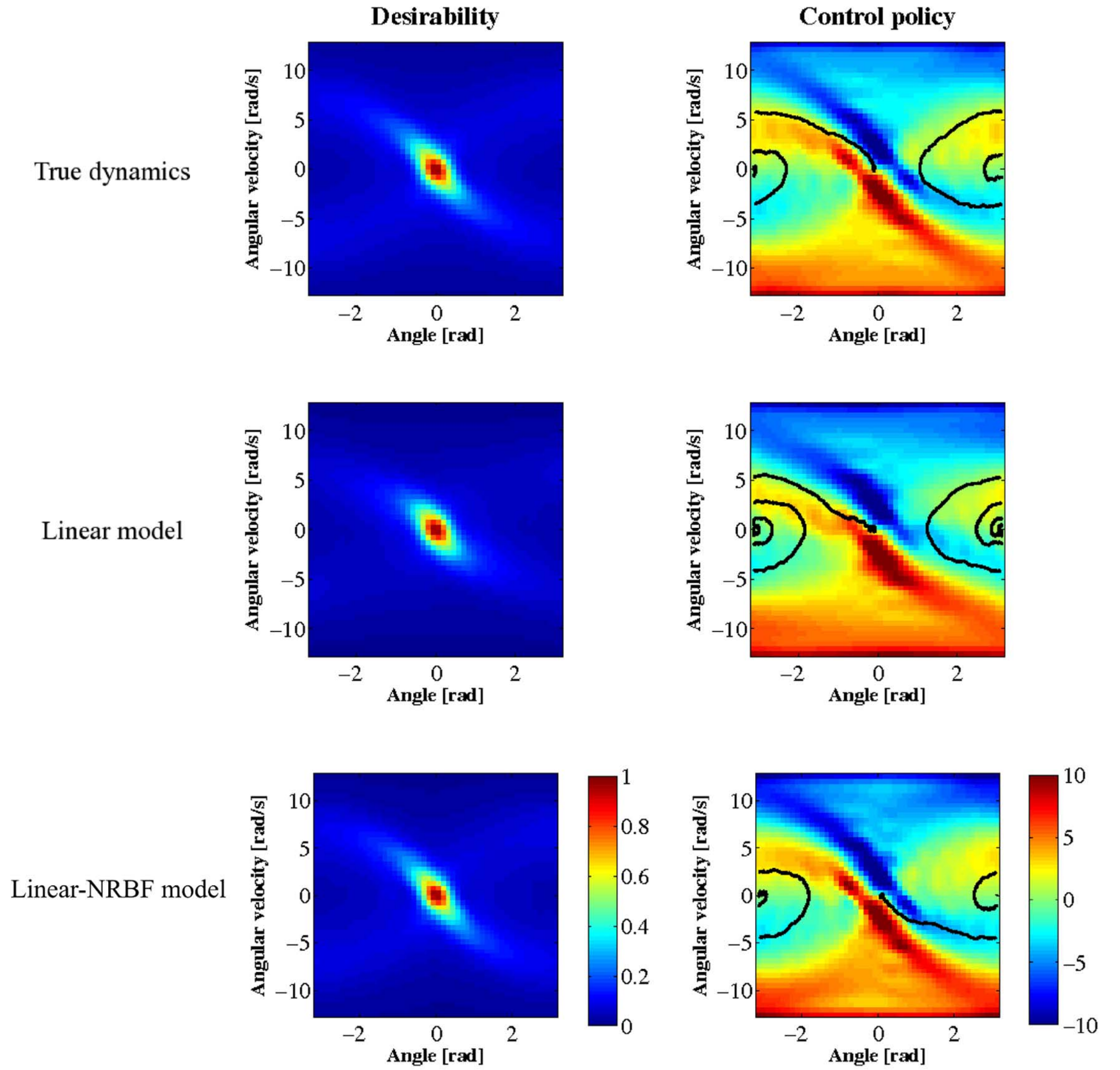


FIGURE 6 | Results of the swing-up pole task. $z(\boldsymbol{x})$ is on the left, $\boldsymbol{u}^*(\boldsymbol{x})$ on the right, true dynamics at the top, linear model in the middle, and linear and NRBF model at the bottom. The black line shows a typical trajectory.

task, we obtained weight matrix using Equation (14) and then calculated MSE in the test data set to evaluate the accuracy of estimation.

Figure 8 shows the result. There was no significant difference between linear and bilinilear models. It suggests these models have almost the same quality for approximating environmental dynamics. Comparing to other components, x_{cx} and x_{pan} derive larger MSE in both model. The reason is these components change more significantly than other components. During the sample acquisition phase, more movement in the rotatory direction occurred than in the translation direction. As a result, the variation of x_{cx}, which was caused by movement of rotatory direction, was large and the variation of x_{pan} also became large due to visual servoing to keep track of the battery in center of visual field.

Figure 9 shows one typical example of the obtained desirability function and the control policy when the cost function is quadratic and the visual-dynamics is estimated using the linear and bilinear models. The upper row corresponds to the LQR's case and the middle and bottom rows correspond to the LMDP trained with the proposed method using linear and bilinear models, respectively. In all figures, the horizontal and the vertical axes denote the pan and tile angle of the neck joint, respectively; the rest of the state components are set to the desired state. Blue dots plotted on middle and lower rows are $\boldsymbol{m}_i$, the center positions of the basis functions for approximating the desirability function. Although the peak of the desirability functions trained with the proposed method is broader than that of the desirability of LQR due to function approximation, obtained controllers show almost same tendency.

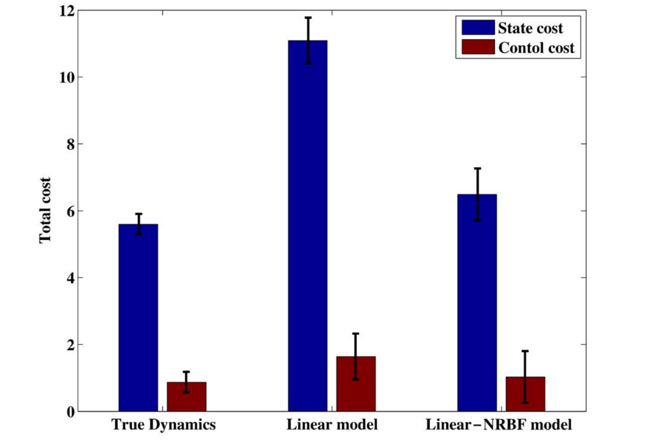


FIGURE 7 | Total costs collected by the obtained control policy. Each error bar represents the standard deviation.

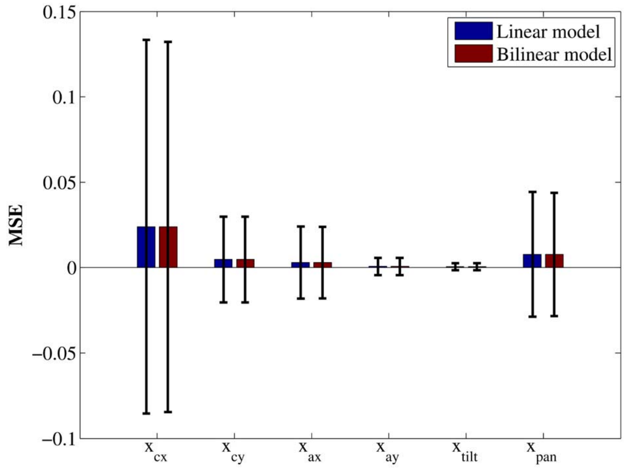


FIGURE 8 | Mean squared error of each state variable. Each error bar represents the standard deviation.

Next, to evaluate performance of obtained controllers, we tested the approaching behavior under the each controller. In the test, the initial position of the robot was set at a distance of 1.5 (m) left the target. The initial direction for each episode was selected randomly a set of three directions; target is placed directly in front of the robot, at a 15° offset to the right of the robot's line of motion or at a 15° offset to the left side, as shown in **Figure 10**. **Figure 11** shows the mean total costs of 30 episodes, the maximum period in one learning episode was 15 (s) (50 steps). For comparison, **Figure 11** shows only quadratic cost function case. Note that the immediate cost in each step was regarded as $c(\boldsymbol{x}, \boldsymbol{u}) = h\left(q_1(\boldsymbol{x}) + \frac{1}{2\sigma^2}\|\boldsymbol{u}\|^2\right)$, and was ignored when the target is not visible in the visual field.

Comparing the total cost among the three controllers using quadratic cost as shown in **Figure 11**, the controller using the linear model resulted in the almost same performance to the result using LQR controller. This result is reasonable because these controllers solve the same problem. The trajectories were very similar shown in **Figure 12**.

On the other hand, the controller using a bilinear model acquired marginally worse result as compared with the other controllers. One possible reason is that over fitting occurred in bilinear model.

In comparing performance among all obtained controllers, we cannot use the total cost because of the difference on state costs. For this reasons we calculated L-1 norm[2] between the current state and the goal state as quantity of controller performance which can be comparable in all controllers. **Figure 13** shows this. All of controllers brought the Spring Dog to almost the goal state in 10 s. Particularly, the controllers using the non-quadratic cost function brought the Spring Dog closer to the battery pack than other controllers. The reason can be considered that the non-quadratic cost function gave a lower cost in more narrow region than the quadratic cost.

4. DISCUSSION

Although it has been reported that the framework of LMDP can find an optimal policy faster than conventional reinforcement learning algorithms, the LMDP requires the knowledge of state transition probabilities in advance. In this paper, we demonstrated that the LMDP framework can be successfully used with the environmental dynamics estimated by model learning. In addition, our study is the first attempt to apply the LMDP framework to real robot tasks. Our method can be regarded as a of model-based reinforcement learning algorithms. Although many model-based methods includes model learning (Deisenroth et al., 2009; Hester et al., 2010) have been proposed in this field, they compute an optimal state value function which is a solution of a non-linear Bellman's equation. Experimental results show that our method is applicable to real robot behavior learning which is generally stochastic and including non-linear state transition. In our proposed method, a cost function is not estimated. However, it is possible to extend to estimate a cost function as well as system dynamics simultaneously, because it is usually formulated as a standard supervised learning problem. In addition, it is not so difficult to assume that a cost function is given in the real robot application, because the robot usually compute the reward by itself in many application.

In the swing-up pole task, the linear and linear-NRBF models were tested to approximate the pole dynamics. The policy derived from the linear model achieved the task of bringing the pole to the desired position even though it cannot represent the dynamics correctly. In the visually-guided navigation task, we compared the desirability function and control policy of LMDP with those of LQR if the environmental dynamics and the cost function were approximated by the linear model and the quadratic function, respectively. In this setting, the optimal state value function and the control policy were calculated analytically by LQR, and therefore, we obtained the optimal desirability function. The obtained desirability function and control policy were not exactly the same as those of LQR. However, we confirmed that the

[2] The L-1 norm of a vector $\boldsymbol{x} = (x_1, \ldots, x_n)^{\mathrm{T}}$ is the sum of the absolute value of the coordinate of $\boldsymbol{x}$, computed by $\|\boldsymbol{x}\|_1 = \sum_i |x_i|$.

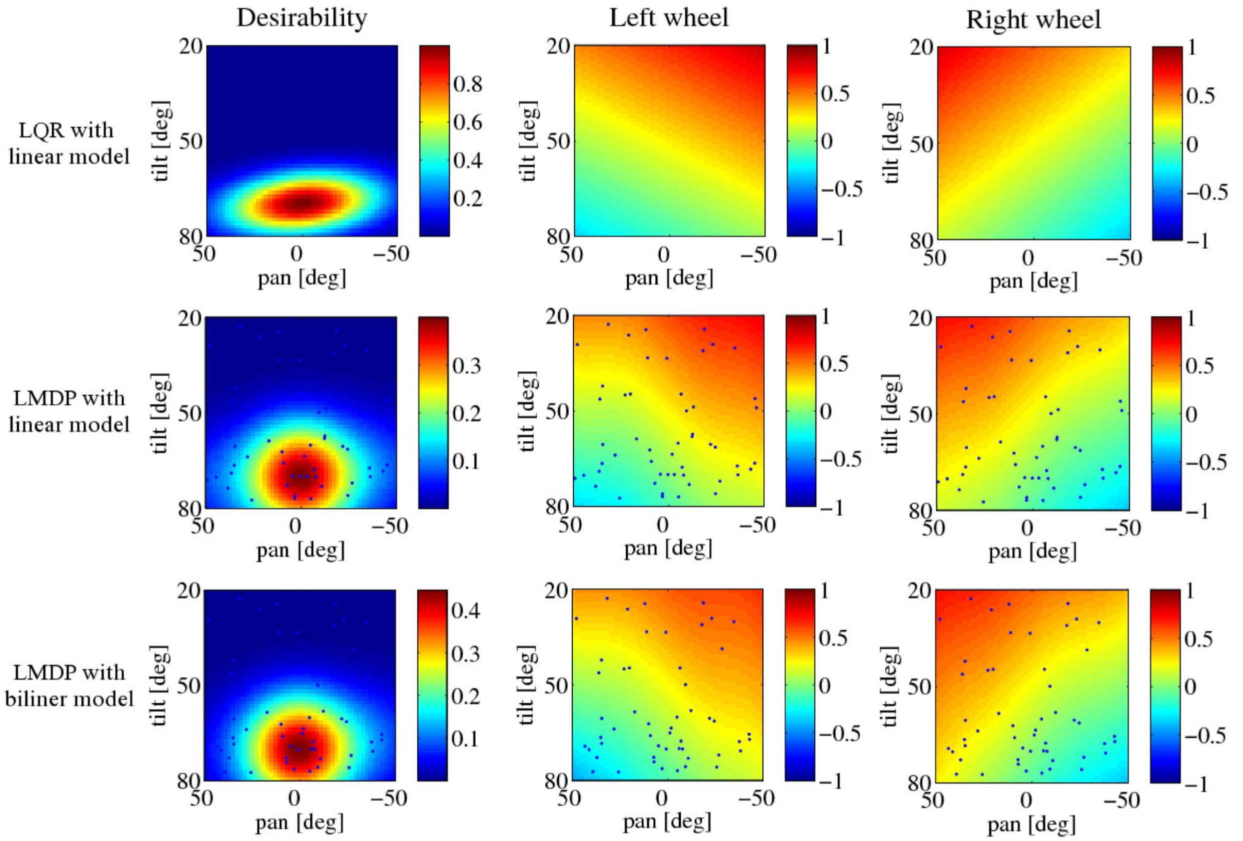


FIGURE 9 | Results of the robot navigation task. LQR with the linear model is at the top, LMDP with the linear model in the middle, LMDP with the bilinear model at the bottom, $z(\boldsymbol{x})$ on the left, $u^*_{\text{left}}(\boldsymbol{x})$ on the center, $u^*_{\text{right}}(\boldsymbol{x})$ on the right. Black dots represent the centers of the basis functions $\varphi(\boldsymbol{x}, \boldsymbol{u})$.

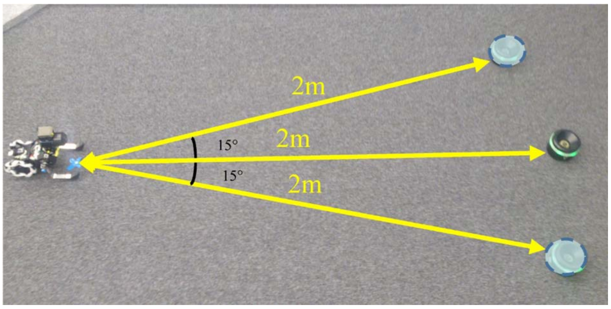


FIGURE 10 | Initial position of the Spring Dog and battery in the test phase. Three possible positions of the battery pack are considered.

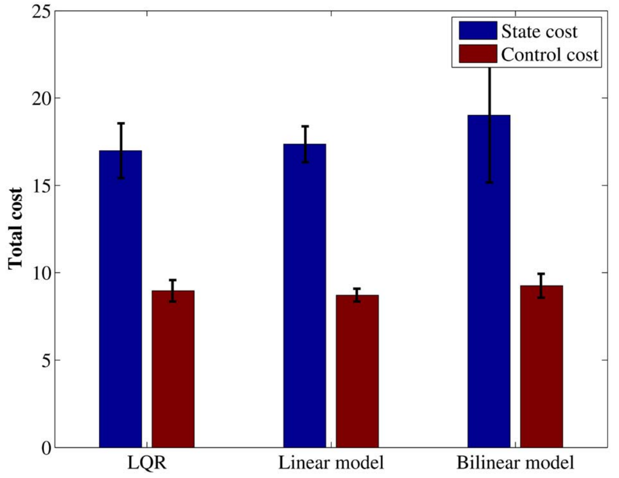


FIGURE 11 | Average of total cost using the quadratic state cost function. Each error bar represents the standard deviation.

performance using the obtained control policy was comparable to the performance using LQR. Both models prepared in this experiment failed to approximate a part of state transition such as x_{cx} and x_{pan}. This means that the Spring Dog could not predict the future position of the battery pack precisely when turned left or right. Nevertheless, the robot could approach the battery pack appropriately. This result suggests that LMDP with model learning is promising even though the estimated model was not so accurate. Fortunately, the control policy which brings the robot to the desired position can be obtained with simple linear model in both experiments. We plan to evaluate the proposed method to non-linear control tasks such as learning walking and running behaviors.

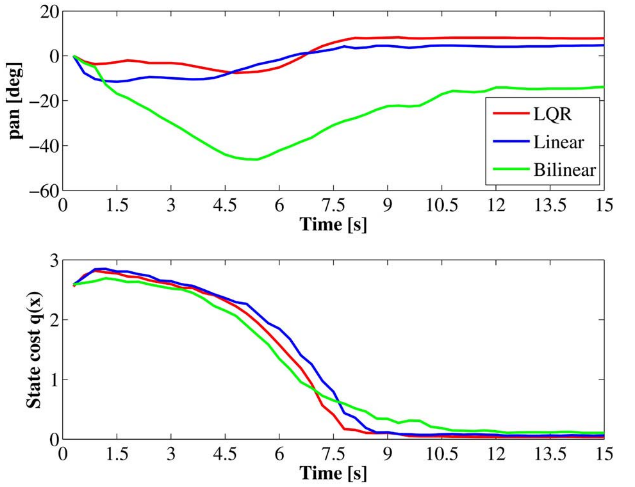


FIGURE 12 | Trajectories of the pan angle x_{tilt} and the immediate state cost under the quadratic state cost.

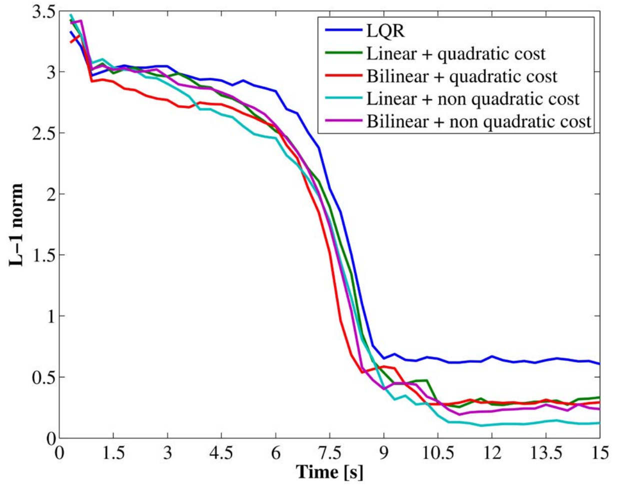


FIGURE 13 | Trajectories of the L-1 norm between the current and goal states.

As discussed in section 3, the quality of obtained control policy depends on the accuracy of the estimated environmental model. For instance, the bilinear model used in the robot experiment did not improve the approximation accuracy, as shown in **Figure 8**, even though its computational complexity is a rather than the linear model. In addition, a part of the conditional mean $\boldsymbol{\mu}(\boldsymbol{x}, \boldsymbol{u})$ was estimated by the least squares method in the current implementation but it would be more informative to estimate the state transition probability distribution $p^{\boldsymbol{u}_k}(\boldsymbol{x}_{k+1}|\boldsymbol{x}_k)$ itself. There exist several methods for estimating a probability distribution from samples. For example, Gaussian process is widely used to estimate environmental dynamics (Deisenroth et al., 2009; Deisenroth and Rasmussen, 2011). Sugiyama et al. (2010) proposed the method to estimate a conditional density distribution efficiently in the manner of density ratio estimation and applied it to state transition estimation in simulated environments. One advantage of their method is that it can estimate a multi-modal distribution by the least squares method. In this case, it is no longer tractable analytically to compute the integral operator even if Gaussian basis functions are used for approximation, and it should be replaced by the Monte Carlo estimates. Integration of sophisticated model learning methods with the LMDP framework is our future work.

The other extension is to develop a model free approach of learning desirability functions, in which the environmental dynamics is not estimated explicitly. Z learning is a typical model-free reinforcement learning method which can learn a desirability function for discrete states and actions, and it was shown that the learning speed of Z learning was faster than that of Q-learning in grid-world maze problems (Todorov, 2007, 2009b). Application of least squares-based reinforcement learning algorithms (Boyan, 2002; Lagoudakis and Parr, 2003) is promising direction. However, in the continuous state case, as mentioned in section 2.1, the optimality equation derive a trivial solution without boundary conditions. In addition, the desirability function should satisfy the inequality $0 \leq z(\boldsymbol{x}) \leq 1$ in order to recover a correct value function by $v(\boldsymbol{x}) = -\log(z(\boldsymbol{x}))$. Furthermore, values of the desirability "function tend to be too small" because of the exponential transformation. For these reasons boundary conditions must be carefully considered. Consequently, the constrained optimization methods should be solved to find the optimal desirability function while learning of the value function is considered as unconstrained optimization. For the extension of model-free learning, this issue have to be solved.

ACKNOWLEDGMENTS

This work was supported by Grant-in-Aid for Scientific Research on Innovative Areas: Prediction and Decision Making (24120527).

REFERENCES

Barto, A. G., and Sutton, R. S. (1998). *Reinforcement Learning*. Cambridge, MA: MIT Press/Bradford Books.

Boyan, J. A. (2002). Technical update: least-squares temporal difference learning. *Mach. Learn.* 49, 233–246.

Burdelis, M. A. P., and Ikeda, K. (2011). "Estimating passive dynamics distributions in linearly solvable markov decision processes from measured immediate costs in reinforcement learning problems," in *Proceedings of the 21st Annual Conference of the Japanese Neural Network Society* (Okinawa).

da Silva, M., Durand, F., and Popović, J. (2009). Linear Bellman combination for control of character animation. *ACM Trans. Grap.* 28. doi: 10.1145/1531326.1531388

Daw, N. D., Gershman, S. J., Seymour, B., Dayan, P., and Dolan, R. J. (2011). Model-based influences on humans' choices and striatal prediction errors. *Neuron* 69, 1204–1215.

Deisenroth, M. P., and Rasmussen, C. E. (2011). "PILCO: a model-based and data-efficient approach to policy search," in *Proceedings of the 28th International Conference on Machine Learning*, eds L. Getoor and T. Scheffer (Bellevue, WA, USA).

Deisenroth, M. P., Rasmussen, C. E., and Peters, J. (2009). Gaussian process dynamic programming. *Neurocomputing* 72, 1508–1524.

Doll, B. B., Simon, D. A., and Daw, N. D. (2012). The ubiquity of model-based reinforcement learning. *Curr. Opin. Neurobiol.* 22, 1075–1081.

Doya, K. (2009). How can we learn efficiently to act optimally and flexibly? *Proc. Natl. Acad. Sci. U.S.A.* 106, 11429–11430.

Fleming, W., and Soner, H. (eds.). (2006). "Logalithmic transformations and risk sensitivity," in *Controlled Markov Processes and Viscosity Solutions, Chapter 6* (New York, NY: Springer Science + Business Media, Inc.), 227–260.

Hester, T., Quinlan, M., and Stone, P. (2010). "Generalized model learning for Reinforcement Learning on a humanoid robot," in *Proceedings of IEEE International Conference on Robotics and Automation* (Anchorage, AK: IEEE), 2369–2374.

Kappen, H. (2005a). Linear theory for control of nonlinear stochastic systems. *Phys. Rev. Lett.* 95, 200201–200204.

Kappen, H. (2005b). Path integrals and symmetry breaking for optimal control theory. *J. Stat. Mech. Theor. Exp.* 11, P11011.

Lagoudakis, M. G., and Parr, R. (2003). Least-squares policy iteration. *J. Mach. Learn. Res.* 4, 1107–1149.

Nguyen-Tuong, D., and Peters, J. (2011). Model learning for robot control: a survey. *Cogn. Proc.* 12, 319–340.

Niv, Y. (2009). Reinforcement learning in the brain. *J. Math. Psychol.* 53, 139–154.

Sigaud, O., Salaün, C., and Padois, V. (2011). On-line regression algorithms for learning mechanical models of robots: a survey. *Robot. Auton. Syst.* 59, 1115–1129.

Stulp, F., and Sigaud, O. (2012). "Path integral policy improvement with covariance matrix adaptation," in *Proceedings of the 10th European Workshop on Reinforcement Learning (EWRL 2012)* (Edinburgh).

Sugimoto, N., and Morimoto, J. (2011). "Phase-dependent trajectory optimization for periodic movement using path integral reinforcement learning," in *Proceedings of the 21st Annual Conference of the Japanese Neural Network Society* (Okinawa).

Sugiyama, M., Takeuchi, I., Suzuki, T., Kanamori, T., and Hachiya, H. (2010). Least-squares conditional density estimation. *IEICE Trans. Inform. Syst.* E93-D, 583–594.

Theodorou, E., Buchli, J., and Schaal, S. (2010). A generalized path integral control approach to reinforcement learning. *J. Mach. Learn. Res.* 11, 3137–3181.

Theodorou, E., and Todorov, E. (2012). "Relative entropy and free energy dualities: connections to path integral and kl control," in *the 51th IEEE Conference on Decision and Control* (Maui), 1466–1473.

Todorov, E. (2006). "Optimal control theory," in *Bayesian Brain: Probabilistic Approaches to Neural Coding, Chapter 12*, eds D. Kenji, S. Ishii, A. Pouget, and R. P. Rao (Cambridge, MA: MIT Press), 269–298.

Todorov, E. (2007). Linearly-solvable Markov decision problems. *Adv. Neural Inform. Proc. Syst.* 19, 1369–1379.

Todorov, E. (2009a). Compositionality of optimal control laws. *Adv. Neural Inform. Proc. Syst.* 22, 1856–1864.

Todorov, E. (2009b). Efficient computation of optimal actions. *Proc. Natl. Acad. Sci. U.S.A.* 106, 11478–11483.

Todorov, E. (2009c). "Eigenfunction approximation methods for linearly-solvable optimal control problems," in *Proceedings of the 2nd IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning* (Nashville, TN), 161–168.

Zhong, M., and Todorov, E. (2011). "Aggregation methods for lineary-solvable Markov decision process," in *Proceedings of the World Congress of the International Federation of Automatic Control* (Milano).

Conflict of Interest Statement: 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.

Received: 01 December 2012; accepted: 15 March 2013; published online: 05 April 2013.

Citation: Kinjo K, Uchibe E and Doya K (2013) Evaluation of linearly solvable Markov decision process with dynamic model learning in a mobile robot navigation task. Front. Neurorobot. ***7****:7. doi: 10.3389/fnbot.2013.00007*

APPENDIX

AOPTIMIZATION OF FUNCTION APPROXIMATION PARAMETERS

When the cost function is non-negative, the value function $v(\boldsymbol{x})$ is also non-negative, and therefore, the inequality $0 \leq z(\boldsymbol{x}) \leq 1$ holds at any $\boldsymbol{x}$ by the definition of the desirability function (Equation 10). In order to satisfy this inequality, the constraint $w_i \geq 0$ for all i is required since we assume that the basis function is a non-normalized Gaussian function. This constrained optimization on $\boldsymbol{w}$ is efficiently solved by the following quadratic programming

$$\min_{\boldsymbol{w}} e, \quad \text{s.t.} \quad w_i \geq 0, \ \forall i. \tag{29}$$

To optimize $\boldsymbol{\theta}$, it is possible to apply the Levenberg–Marquardt algorithm to minimize the square error (Equation 18). However, it was reported that the desirability function become $z(\boldsymbol{x}_n; \boldsymbol{w}, \boldsymbol{\theta}) \approx 0$ during the minimization process because the center position of the basis functions $\boldsymbol{m}_i$ move away from collocation states $\boldsymbol{x}_n$ (Todorov, 2009b). To avoid the trivial solution $z(\boldsymbol{x}) = 0$, the following constraint is introduced,

$$\mathbf{1}^{\mathrm{T}} \boldsymbol{F}(\boldsymbol{\theta}) \boldsymbol{w} = \sum_{n=1} \hat{z}(\boldsymbol{x}_n; \boldsymbol{w}, \boldsymbol{\theta}) = \text{const.} \tag{30}$$

This constrained problem is optimized by the Levenberg–Marquardt algorithm. When we define $\boldsymbol{J} = \partial \boldsymbol{r} / \partial \boldsymbol{\theta}$ and $\boldsymbol{g} = \partial(\mathbf{1}^{\mathrm{T}} \boldsymbol{F}(\boldsymbol{\theta}) \boldsymbol{w}) / \partial \boldsymbol{\theta}$, then the objective function is given by

$$\min_{\boldsymbol{\delta}} \frac{1}{2} \boldsymbol{\delta}^{\mathrm{T}} (\boldsymbol{J}^{\mathrm{T}} \boldsymbol{J} + \gamma \boldsymbol{I}) \boldsymbol{\delta} + \boldsymbol{\delta}^{\mathrm{T}} \boldsymbol{J}^{\mathrm{T}} \boldsymbol{r} \qquad \text{s.t.} \quad \boldsymbol{g}^{\mathrm{T}} \boldsymbol{\delta} = 0, \tag{31}$$

where $\boldsymbol{\delta}$ and γ denote the gradient direction of the update rule and the parameter between 0 and 1, respectively. This is solved by the Lagrange multiplier methods.

frontiers in
NEUROROBOTICS

ORIGINAL RESEARCH ARTICLE
published: 05 February 2013
doi: 10.3389/fnbot.2013.00001

A neurorobotic platform to test the influence of neuromodulatory signaling on anxious and curious behavior

Jeffrey L. Krichmar [1,2]*

[1] *Department of Cognitive Sciences, University of California, Irvine, Irvine, CA, USA*
[2] *Department of Computer Science, University of California, Irvine, Irvine, CA, USA*

Edited by:
Florian Röhrbein, Technische Universität München, Germany

Reviewed by:
Eleni Vasilaki, University of Sheffield, UK
Benoît Girard, CNRS & UPMC, France

****Correspondence:***
Jeffrey L. Krichmar, Department of Cognitive Sciences, 2328 Social and Behavioral Sciences Gateway, University of California, Irvine, Irvine, CA 92697-5100, USA.
e-mail: jkrichma@uci.edu

The vertebrate neuromodulatory systems are critical for appropriate value-laden responses to environmental challenges. Whereas changes in the overall level of dopamine (DA) have an effect on the organism's reward or curiosity-seeking behavior, changes in the level of serotonin (5-HT) can affect its level of anxiety or harm aversion. Moreover, top-down signals from frontal cortex can exert cognitive control on these neuromodulatory systems. The cholinergic (ACh) and noradrenergic (NE) systems affect the ability to filter out noise and irrelevant events. We introduce a neural network for action selection that is based on these principles of neuromodulatory systems. The algorithm tested the hypothesis that high levels of serotonin lead to withdrawn behavior by suppressing DA action and that high levels of DA or low levels of 5-HT lead to curious, exploratory behavior. Furthermore, the algorithm tested the idea that top-down signals from the frontal cortex to neuromodulatory areas are critical for an organism to cope with both stressful and novel events. The neural network was implemented on an autonomous robot and tested in an open-field paradigm. The open-field test is often used to test for models anxiety or exploratory behavior in the rodent and allows for qualitative comparisons with the neurorobot's behavior. The present neurorobotic experiments can lead to a better understanding of how neuromodulatory signaling affects the balance between anxious and curious behavior. Therefore, this experimental paradigm may also be informative in exploring a wide range of neurological diseases such as anxiety, autism, attention deficit disorders, and obsessive-compulsive disorders.

Keywords: neuromodulation, anxiety, computer simulation, robotics, dopamine, serotonin, acetylcholine, norepinephrine

INTRODUCTION

The vertebrate neuromodulatory systems are critical for appropriate value-laden responses to environmental challenges (Krichmar, 2008). Whereas changes in the overall level of dopamine (DA) have an effect on the organism's reward or curiosity-seeking behavior (Schultz et al., 1997; Berridge, 2004), changes in the level of serotonin (5-HT) can affect its level of anxiety or harm aversion (Millan, 2003; Cools et al., 2008). The cholinergic (ACh) and noradrenergic (NE) systems affect the ability to filter out noise and irrelevant events (Vankov et al., 1995; Bucci et al., 1998; Aston-Jones and Cohen, 2005; Yu and Dayan, 2005). These neuromodulatory systems have broad and extensive projections to the central nervous system causing shifts in behavior and learning.

The frontal cortex, which projects to all the neuromodulatory systems (Briand et al., 2007), may be carrying a level of cognitive control through modulating the neuromodulators. For example, the medial prefrontal cortex (mPFC) can control the stress response by its interaction with the raphe nucleus, the main source of 5-HT in the central nervous system (Jasinska et al., 2012), and the orbitofrontal cortex (OFC) may exert control on the DA reward system (Frank and Claus, 2006). Empirical evidence and theoretical modeling have suggested that the mPFC, the anterior cingulate cortex, and the OFC control decision-making in the face of reward-cost tradeoffs (Rudebeck et al., 2006; Rushworth et al., 2007; Chelian et al., 2012). That is, the OFC's interaction with the DA system is monitoring the expected reward of an action, and the mPFC's interaction with the 5-HT system is monitoring the expected cost of an action (Zaldivar et al., 2010; Asher et al., 2012).

Previously, a general-purpose algorithm, based on principles of the brain's neuromodulatory systems, was presented for action selection in robots (Krichmar, 2012). Rather than presenting a neurobiologically detailed model of how the nervous system achieves this function through neuromodulation [see for example (Cox and Krichmar, 2009)], a general-purpose, but minimal model of neuromodulatory function was developed, which could be applied to robot control. Similar to classic robot control algorithms, such as subsumption architecture (Brooks, 1991) and behavior-based schemas (Arkin, 1998), the algorithm automatically arbitrated between actions based on current sensory input. The algorithm demonstrated the ability to adapt to changes in the environment by: (1) increasing sensitivity to sensory inputs, (2) responding to unexpected or rare events, and (3) habituating or ignoring uninteresting events. The algorithm showed several important features for autonomous robot control in general, such as, fluid switching of behavior, gating in important sensory events, and separating signal from noise.

The present paper extends this algorithm in several key ways to make it more neurobiologically realistic, and more adaptable. First, a frontal cortex layer, which loosely corresponds to the OFC and mPFC and projects to the DA and 5-HT systems, respectively, is added to the model. This provides a degree of top-down control on the neuromodulatory systems that handle sensory events. Second, an inhibitory projection from the 5-HT system to the DA system was added based on evidence that these systems are somewhat in opposition (Tops et al., 2009; Boureau and Dayan, 2011). From a behavioral standpoint, the 5-HT system causes the organism to be withdrawn and risk-averse, and the DA system causes the organism to be invigorated and risk-taking. From the algorithm's standpoint, this allowed sensory events to be shared with the appropriate action taken based on the current levels of DA and 5-HT. Lastly, a variable was added to model the tonic levels of DA and 5-HT. The previous model only considered phasic neuromodulatory responses, which resulted in decisive action. The tonic levels in the present model can set the agent's behavioral context or state and make the agent more likely to select a particular set of actions.

The present algorithm tested the hypothesis that high levels of 5-HT lead to withdrawn behavior by suppressing DA action and that high levels of DA or low levels of 5-HT lead to curious, exploratory behavior. It has been suggested that serotonin opposes activating or invigorating neuromodulators such as dopamine (Tops et al., 2009). Specifically, projections from raphe serotonin cells to DA areas may oppose the action of DA and mediate avoidance of threats (Deakin, 2003). Furthermore, the algorithm tested the idea that top-down signals from the frontal cortex to neuromodulatory areas are critical for an organism to cope with both stressful and novel events. A recent review suggested that the mPFC inhibited the serotonergic raphe nucleus after handling a stressful event (Jasinska et al., 2012). This feedback loop prevented the raphe from being overly active after the stressor had been handled. The present algorithm further suggests that projections from the OFC to the dopaminergic ventral tegmental area (VTA) have a similar function when responding to a positive valence event.

The algorithm was implemented in a neural network that controlled the behavior of an autonomous robot and tested in the open-field paradigm. The open-field test is often used for animal models anxiety or exploratory behavior and allows for qualitative comparisons with the neurorobot's behavior (Heisler et al., 1998; Lacroix et al., 2000; Lipkind et al., 2004; Fonio et al., 2009).

METHODS

ROBOT CONTROL

Experiments were run on an iRobot Create equipped with an URG-04-LX laser range finder (Hokuyo Automatic Co. LTD.) and a System 76 netbook running the Ubuntu Linux operating system for computation (see **Figure 1**). The Matlab Toolbox for iRobot Create (http://www.usna.edu/Users/weapsys/esposito/roomba.matlab/) was used to interface with the robot. The neural simulation and robot control algorithm for iRobot Create was written in Matlab (MathWorks) and can be downloaded at: http: // www.socsci.uci.edu/~jkrichma/krichmar_frontiers2012_carl_roomba.m

Robot control was achieved through processing events and states. States were pre-canned behaviors and events were driven by sensory signals. An event could cause a switching of behavior

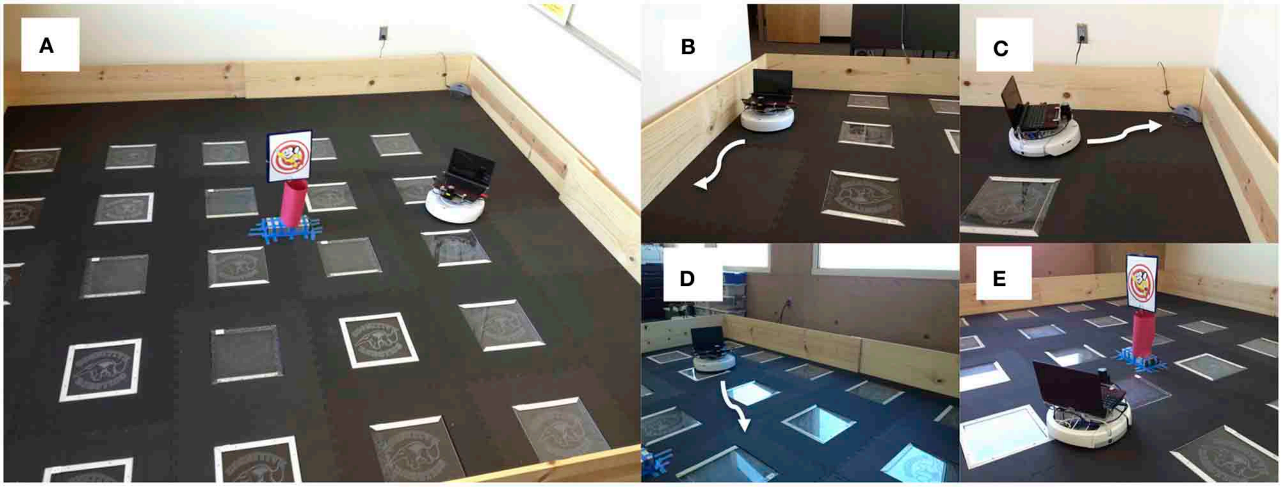


FIGURE 1 | Setup for neurorobotic experiments. Experiments were run on an iRobot Create equipped with an URG-04-LX laser range finder (Hokuyo Automatic Co. LTD.) and a System 76 netbook running the Ubuntu Linux operating system for computation. **(A)** Environment was a 3.7 m^2 arena enclosed with plywood. The picture in the middle was a novel object for the robot to explore. **(B)** Wall following behavior. Wall following was achieved using the Create's "Mouse" demo. **(C)** Find home behavior. Finding the docking station was achieved using the Create's "Cover and Dock" demo. **(D)** Open-field behavior. The robot moved toward open spaces in the environment based on laser range finder readings. **(E)** Explore object. The robot approached narrow objects based on laser range finder readings.

states. The neural simulation, which is described below, arbitrated between incoming events and decided when to switch states. A simulation cycle, *t*, occurred approximately once per second, which was roughly the time needed to read CarlRoomba's sensors, update the neural simulation, and send a motor command to CarlRoomba. The main limitation for cycle duration was Matlab handling of I/O. Future versions of the software will be written in C/C++ to speed up I/O and shorten simulation cycles.

In the present experiments, the robot, which is called CarlRoomba, handled three events: (1) *Object Detected*. This event was triggered if the laser detected an object between 12 and 30 degrees wide and closer than one meter. (2) *Light detected*. This event was triggered if the average pixel brightness in the grayscale image was greater than 50%. The netbook's built-in camera was used to detect light levels. (3) *Bump detected*. This event was triggered by iRobot Create's bump sensors or if the laser detected an object closer than 20 cm.

CarlRoomba switched between four behavior states: (1) *Wall Follow* (**Figure 1B**). Wall following was achieved by calling the iRobot Create's mouse demo routine. This caused CarlRoomba to follow the wall to its right. (2) *Find Home* (**Figure 1C**). Find home was achieved by calling the iRobot Create's cover and dock demo routine. This caused CarlRoomba to move in a random pattern until it detected the Roomba docking station via an IR beam that had a range of roughly 500 cm. (3) *Open-Field* (**Figure 1D**). CarlRoomba would drive toward the most open area of the environment, as judged by the laser range finder. If a collision with an object was detected, CarlRoomba would rotate clockwise. (4). *Explore Object* (**Figure 1E**). CarlRoomba would move toward the object found by the laser. If a collision with an object was detected, CarlRoomba would rotate clockwise.

NEURAL SIMULATION

Neuromodulatory systems receive sensory information and drive behavior by innervating downstream neural systems. The general framework of the present architecture is that sensory events can trigger neuromodulatory systems, which in turn drive behavior states (see **Figure 2**). Frontal areas (see OFC and mPFC in **Figure 2**) trigger action selection and exert cognitive control on the neuromodulatory areas (see DA and 5-HT in **Figure 2**) via inhibitory projections. The ACh and NE systems (see AChNE in **Figure 2**) act as an attentional filter allowing novel and unexpected events to gate through to the frontal cortex. Specifically, AChNE modulates connections from DA and 5-HT to cortical neurons and inhibitory connections between cortical neurons (see blue arrows and ellipses in **Figure 2**). It has been suggested that ACh and NE neuromodulation gates in sensory inputs and increases competition among frontal cortex neurons by upregulating GABAergic currents, but not glutamatergic connections (Hasselmo and McGaughy, 2004; Aston-Jones and Cohen, 2005). Although the architecture given in **Figure 2** is specific to the present problem space, the general framework could potentially be used to arbitrate any combination of sensory events and behavioral states.

In the present paper, the neural simulation consisted of three event neurons, each of which corresponded to one of the sensory events described above, four state neurons, each of which corresponded to one of the behavioral states described above, and neuromodulatory neurons. There was one DA neuron, one 5-HT, and three ACh/NE neurons, each of which corresponded to one of the sensory events described above. **Figure 2** shows the architecture and connectivity of the network.

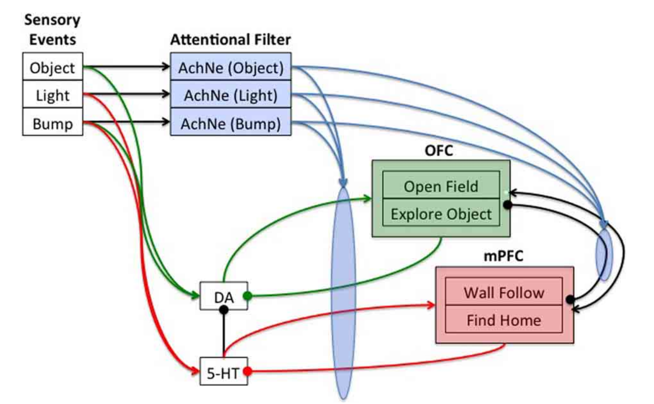


FIGURE 2 | Neural architecture to control robot behavior. Sensory events were handled by three binary neurons. These neurons projected to the attentional filter neurons (AchNE) and the dopaminergic and serotonergic neurons (DA and 5-HT). The DA and 5-HT neurons projected to the OFC and mPFC neurons. The most active OFC or mPFC neuron dictated the robot's behavioral state. The AChNE neurons had a modulatory effect on the projection from the DA and 5-HT to OFC and mPFC (see blue ellipse and arrows). OFC and mPFC projected to 5-HT and DA neurons with inhibitory connections. Excitatory and inhibitory connections within and between OFC and mPFC neurons were all-to-all. See text for details.

Initial simulations were carried out to set the weights and parameters given in the equations below. Weights were chosen such that the network demonstrated stable activity, and such that a phasic burst of neuromodulatory activity could efficiently drive action selection. Each OFC and mPFC neuron was connected to every other OFC and mPFC neuron with both excitatory (weight = +1.0) and inhibitory (weight = −1.0) connections. OFC neurons for OpenField and ExploreObject projected to the DA neuron with a weight equal to −1.0, and mPFC neurons for WallFollow and FindHome projected to the 5-HT neuron with a weight equal to −1.0. Neuromodulatory neurons selectively connected to OFC and mPFC neurons with weights set at 5, event neurons selectively connected to neuromodulatory neurons with weights set at 0.5, and event neurons connected to the corresponding ACh/NE neurons with weights set at 1.

In the present simulation, detecting an object with the laser signaled novelty or something potentially rewarding in the environment and worth taking a risk to investigate. Therefore, these events triggered dopaminergic neurons (Object→DA in **Figure 2**). A bright light signaled a potential danger, and thus triggered serotonergic neurons (Light→5-HT in **Figure 2**). A bump could signal either something interesting or noxious in the environment. Therefore, the bump event triggered both dopaminergic and serotonergic neurons (Bump→DA and Bump→5-HT in **Figure 2**). To model, serotonergic and dopaminergic opponency, 5-HT projected to DA with a weight set at −1.0.

Event neurons were binary and set to 1 when an event occurred and 0 otherwise. All other neurons were governed by the following activation function, which kept neural activity between 0 and 1:

$$n(t) = \frac{1}{1 + e^{-gI(t)}} \tag{1}$$

where g was the gain of the function and I was the input to the neuron. The initial weights, gains, and the baseline input, given in Equation 2, were set such that the range of synaptic input to the neuron would cover the full range of the sigmoid curve. Therefore, the gain was set to 2 for frontal cortex and neuromodulatory neurons, and 10 for ACh/NE neurons. Input to the neuron was based on pre-synaptic neural activity, $n_j(t)$, previous neural activity, $n_i(t-1)$, and neuromodulation:

$$I_i(t) = b + \sum_j n_j(t)w_{ji}(t) + pn_i(t-1) + \text{tonic}_{nm}(t) \tag{2}$$

where b was the baseline input set to -1.0 for DA and 5-HT, -0.5 for ACh/NE, and a random number that was drawn uniformly between negative one and zero for OFC and mPFC neurons. The baseline input was set such that the full range of the sigmoid curve (0 to 1 in Equation 1) was covered, and the random number value for b, which was drawn every time step for OFC and mPFC, added some stochasticity to cortical neural activity. p was the persistence set to 0.25 for frontal cortex, 0.5 for ACh/NE neurons, and zero for DA and 5-HT neurons. Synaptic input into neuromodulatory neurons had an additional term for tonic neuromodulation (tonic_{nm}). For all other neurons, tonic_{nm} was set to zero.

In our previous model, the ACh and NE system was introduced as an attentional filter (Krichmar, 2012). When the ACh/NE system was impaired in the algorithm, the robot lost its ability to filter out noise and responded to any incoming sensory event. This attentional filter, which is shown pictorially in **Figure 2** (see blue ellipse and arrows), was achieved by adding the following term to the synaptic input into OFC and mPFC neurons.

$$\begin{aligned} I_i(t) = I_i(t) &+ \sum_j AChNE(t-1)n\_fctx_j(t-1)w\_inh_{ji}(t-1) \\ &+ \sum_k AChNE(t-1)n\_nm_k(t-1)w\_nm_{ki}(t-1) \end{aligned} \tag{3}$$

where AChNE is the sum of all neural activity in the ACh and NE areas, $n\_fctx_j(t)$ is the activity from other frontal cortex neurons, $n\_nm_k(t)$ is the neuromodulatory input into a frontal neuron, $w\_inh_{ji}(t)$ is the weight of lateral inhibition from frontal cortex neuron j to frontal cortex neuron i, and $w\_nm_{ki}(t)$ is the weight of the connection from neuromodulatory neuron k to frontal cortex neuron i.

AChNE neurons acted as an attentional filter for events by adjusting weights from event neurons to AChNE neurons through the following update rule:

$$w_{ji}(t) = \begin{cases} p{*}wji(t-1) & \text{if } e_j = 1 \\ w_{ji}(t-1) + \frac{1-w_{ji}(t-1)}{\tau} & \text{otherwise} \end{cases} \tag{4}$$

where j is the index of the event neuron, i is the index of the ACh/NE neuron, p is the amount of change in response to an event, and τ, which was set to 25, was a time constant that governed the rate at which weights returned to their original value. Weights from event neurons to ACh/NE neurons were depressing, meaning that each event caused the weight to decrease ($p = 0.25$).

Tonic activity in the DA and serotonergic neurons was modeled by having a facilitating response to sensory events gated in by the AChNE neurons:

$$\text{tonic}_i(t) = \begin{cases} p{*}\text{tonic}_i(t-1) & \text{if } AChNE_j > 0.5 \\ \text{tonic}_i(t-1) + \frac{1-\text{tonic}_i(t-1)}{\tau} & \text{otherwise} \end{cases} \tag{5}$$

where i is the index of the neuromodulatory neuron, j is the index of the ACh/NE neuron, p is the amount of change in response to an event. The tonic levels rose every time there was a salient sensory event by setting $p = 1.25$. The time constant, τ, was related to neurotransmitter re-uptake, that is, how long a neuromodulator acted on its target neurons. For example, a larger value of τ meant that the re-uptake of a neuromodulator was slower and therefore the neuromodulator had a longer lasting effect. Initially, $\text{tonic}_{5\text{HT}}$ was set to 2.0 and tonic_{DA} was set to 1.0, which caused CarlRoomba to have higher levels of 5-HT at the start of an experimental trial.

These rates and parameters were set based on the expected occurrence of events during a four-minute session of running CarlRoomba. For example, in the control condition, the parameters p and τ were chosen such that salient events would trigger a long lasting increase in tonic neuromodulation. Multiple events should cause a change in the neurorobot's contextual state (e.g., become withdrawn) and a long interval between events would result in the neurorobot settling into a neutral state. In other conditions, parameter τ was set to demonstrate how low and high levels of tonic neuromodulation, relative to the control condition, might affect behavior.

Action selection occurred after the neural activities and weight updates were calculated. The maximally active state neuron was chosen as the new behavioral state if it had activity greater than 0.67. This threshold was set such that new actions would be selected roughly 4–5 times per minute. If no state neuron was above this threshold, the previous behavioral state continued.

EXPERIMENTAL PARADIGM

Experiments were run in an open-field arena, which was a 3.7 m^2 region blocked off by plywood (see **Figure 1A**). A cardboard column and picture that was detectable by the laser was placed in the center of the arena. The Roomba docking station was placed in one corner of the arena. Experiments were run in the dark for 240 s. At approximately 120 s into an experiment, which allowed CarlRoomba to acclimate to the environment, the lights were turned on for 10 s and then turned off again. CarlRoomba always started the experiment in the corner of the arena where the docking station was located, and always faced the center of the arena. Each parameter setting was run 5 times on CarlRoomba, each with different random number generator seeds.

The experimental setup was designed to mimic a rodent open-field experiment and CarlRoomba's ability to handle a stressful event. When placed in a new environment, rodents typically stay near their nest (i.e., the docking station) or follow closely along the walls of an environment (Fonio et al., 2009). As they become more comfortable in the environment, they will venture out into the open area of the arena or explore a novel object placed in the arena. This paradigm is often used to test animal models of anxiety (Simon et al., 1994; Heisler et al., 1998; Lipkind et al., 2004). The present experiments were designed to test how dopaminergic and serotonergic neuromodulation influence the ability to cope with a stressful event. In Fonio's experimental paradigm, the moving of a mouse to a novel environment is presumably a stressful event. However, this prior context would be difficult to mimic with the neurorobot CarlRoomba. Therefore, a light flash was used to mimic a stressful event in the open-field test, since rodents typically prefer the dark.

RESULTS

COGNITIVE CONTROL OF INTERESTING AND STRESSFUL EVENTS

CarlRoomba responded appropriately to sensory events in its environment. Novel objects resulted in it exploring the environment, stressful events, such as bright lighting caused it to seek safety. **Figure 3A** shows a representative trial from a CarlRoomba where there were balanced tonic levels of neuromodulation ($\tau_{DA} = \tau_{5HT} = 50$ in Equation 5). In **Figure 3A** and subsequent representative trial figures, the x-axis denotes time in seconds from the start of the trial until the end, which was approximately 240 s. The upper chart shows CarlRoomba's behavioral state over the course of the trial. The second through fifth charts show the neural activity of the State, Event, ACh/NE, and Neurmodulatory neurons, respectively, over the course of a trial where dark blue signifies no activity and bright red signifies maximal activity. The bottom chart denotes the level of tonic neuromodulation (see Equation 5). Note how initially when CarlRoomba was unfamiliar with the environment, serotonergic activity dominated, resulting in anxious behavior, such as WallFollow and FindHome actions. However, as CarlRoomba became more familiar and comfortable in its environment (approximately 60 s into the trial), DA levels were higher and there was more curious or exploratory behavior. Note that the AChNE neurons only gated through interesting and rare events. This was achieved through AChNE modulation of projections from neuromodulatory neurons to OFC and mPFC and through AChNE modulation of intrinsic inhibitory projections between frontal cortex neurons (see Equations 3 and 4 and **Figure 2**). For example, constant bump events were habituated (compare Bump event neuron activity with Bump AChNE activity in **Figure 3A**). At approximately 120 s into the trial, there was an unexpected Light event, which resulted in a phasic 5-HT response and a longer tonic increase in 5-HT (see Equations 2 and 5). This caused CarlRoomba to respond with withdrawn or anxious behavior until approximately 210 s into the trial when a pair of object events triggered exploration of the center of the environment (see **Figure 3A**). Specifically, tonic levels of 5-HT had decayed and the object events caused an increase in DA levels triggering a change in behavioral state.

Figure 3B shows the proportion of curious behavior (OpenField and ExploreObject) and anxious behavior (FindHome and WallFollow) for five experimental trials. In **Figure 3B** and subsequent figures summarizing five trials, histograms were calculated with 10 s bins over the course of the trial. Each bar was the average proportion of time spent in either curious (green bars) or anxious behavior (red bars) in a 10 s period of the trial. The error bar denoted the standard error. Note that on different trials, the timing of the light event varied (as early as 118 s and as late as 130 s). Thus, the increase in "Anxious" behavior at 110 s (see **Figure 3B**) is not due to a prediction of the stressful event, but rather trial variation. Because the initial state of CarlRoomba is not necessarily anxious or curious, and CarlRoomba pointed toward the center of the arena at the start of every trial, it is hard to quantify CarlRoomba's behavior over the first half of each trial. However, CarlRoomba's initial behavior appeared to be anxious, and then more curious as it became more familiar with the environment.

To resolve potential issues with comparing across conditions that result from trial and initial state variation, **Figure 3C** and subsequent population figures shows the behavior time-locked to the light event. The light event, which occurred at approximately the halfway point in the trial, was introduced to cause a stress response in CarlRoomba (see **Figure 3**). The ability of CarlRoomba to handle this stressful event was compared across all conditions. After the light event, the neurorobots' behavior rapidly switched to anxious behavior until roughly 200 s when it became curious again (see **Figure 3C**). Variation occurred due to different times of the light event, and random variations in other sensory events.

The neurorobots' behavior after a stressful event was qualitatively similar to a rodent's behavior when placed in a novel environment. For example, in Fonio et al.'s experiments (Fonio et al., 2009), mice progressed from staying near a nest (1–4 in their developmental sequence in Fonio et al., 2009, Figure 1), making circuits along the border of the environment (5–9 in Fonio et al., 2009, Figure 1), and then crossing the center of the environment (10–11 in Fonio et al., 2009, Figure 1). All their mice followed this behavioral pattern. In a similar way, CarlRoomba followed this pattern. In all five trials for the first 50 s following the light flash, CarlRoomba stayed near its docking station and the walls of the arena. By 100 s after the light flash, CarlRoomba spent over half its time either crossing the center of the environment or investigating a novel object in the center of the environment. These control experiments show that when CarlRoomba has an intact nervous system, it is able to respond appropriately to a stressor, and then resume exploratory behavior when the stressor has passed.

SEROTONIN AND THE ABILITY TO COPE WITH STRESSFUL EVENTS

It has been suggested that degradation of serotonin re-uptake can have detrimental effects on the ability to cope with stressors (Jasinska et al., 2012). To mechanistically test this notion, the time constant for tonic serotonin was increased ($\tau_{DA} = 50$, and $\tau_{5HT} = 150$ in Equation 5). This had the effect of serotonin staying in the system longer after a stressful event.

A stressful event, such as a bright light, still caused CarlRoomba to select anxious behaviors, but the increase in serotonin levels resulted in CarlRoomba never breaking out of this stressful behavior. **Figure 4A** shows a representative trial where τ_{5HT} was longer. Compared to **Figure 3A**, serotonin levels remain

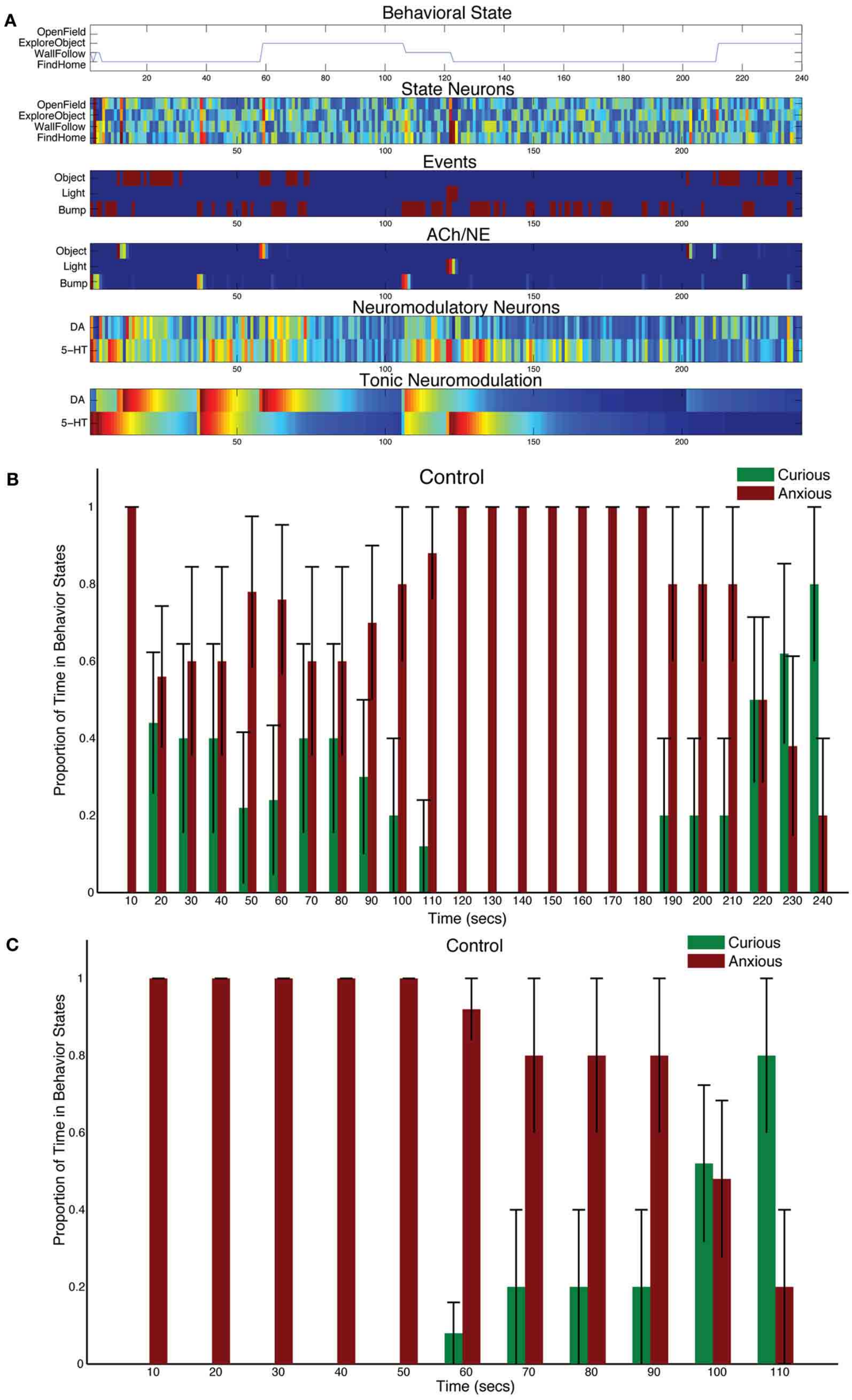


FIGURE 3 | Behavioral and neural responses in the intact model. The time constants τ_{DA} and τ_{5-HT} were both set at 50. **(A)** Behavioral and neural responses in a representative trial. The x-axis for all charts shows the time of the trial in seconds. The chart labeled "Behavioral State" denotes the state of the robot at a given time. The charts labeled "State Neurons," "Events," "ACh/NE," and "Neuromodulatory Neurons" show the neural activity over the trial, where dark blue equates to no activity and bright red equates to maximal activity. Note that Event neurons were binary. The chart labeled "Tonic Neuromodulation" denotes the level of tonic activation contributing to DA and 5-HT neurons. **(B)** The proportion of Curious (ExploreObject and OpenField) and Anxious (FindHome and WallFollow) behavior averaged over 5 trials. The error bars denote the standard error. The histogram binned the behavior in 10 s windows. **(C)** Similar to **(B)** except the behaviors were time-locked to the Light event.

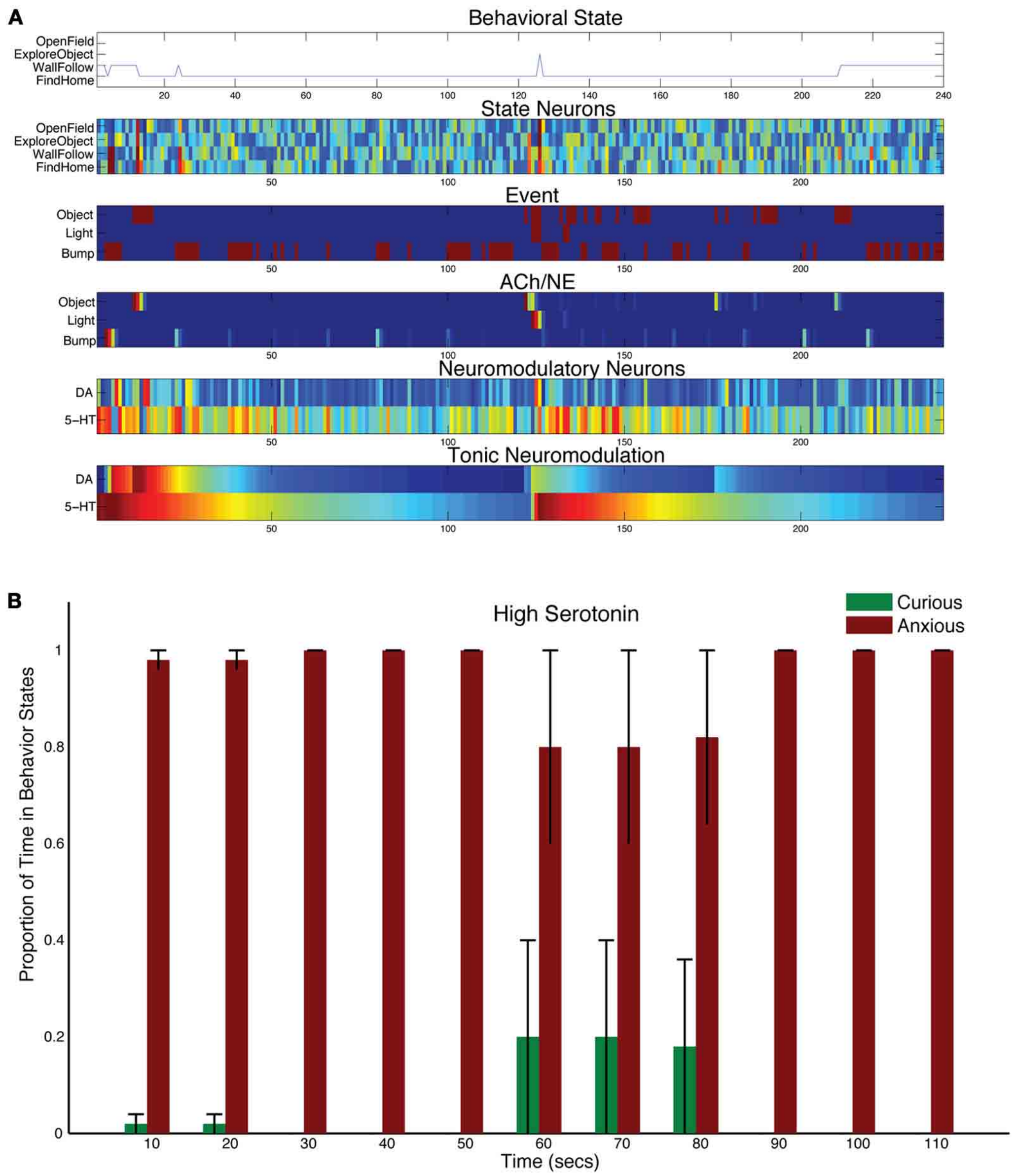


FIGURE 4 | Behavioral and neural responses with high serotonin levels. The time constant τ_{DA} was set to 50 and the time constant τ_{5-HT} was set to 150. **(A)** Behavioral and neural responses in a representative trial. Axes, labels, and color are the same as in **Figure 3A**. **(B)** The proportion of Curious (ExploreObject and OpenField) and Anxious (FindHome and WallFollow) behavior averaged over 5 trials. Axes, labels, and time locking is the same as in **Figure 3C**.

high and the resulting behavior is almost entirely wall following and finding home. **Figure 4B** shows the population behavior of five trials time locked to the light event. As in the control case, there is a strong response to the light. However, unlike the control behavior shown in **Figure 3C**, CarlRoomba with high serotonin levels never recovers from this stressful event, and demonstrates anxious behavior throughout the remainder of the trial. These results are qualitatively similar to that shown by Heisler and colleagues where genetically mice that were lacking in 5HT1A receptors spent less time in the center of the open-field arena (Heisler et al., 1998). 5-HT1A receptors located on serotonergic neurons act as autoreceptors and suppress serotonergic neuronal activity. Therefore, mice lacking in 5HT1A would have increased levels of serotonin in the nervous system. In the open-field test, these mice showed reduced time in the center of the arena, and were less likely to approach a novel object.

To test how lowering levels of serotonin affect behavior, the time constant for tonic serotonin was lowered with respect to control levels ($\tau_{DA} = 50$, and $\tau_{5HT} = 1$ in Equation 5). This drastically reduced the tonic levels of serotonin in the model, but the serotonergic system still responded phasically to sensory events (see **Figure 5A**). For example, there was a serotonergic response to the light event at 120 s into the trial. However, the object sensory event at 150 s and the bump event at 160 s resulted in

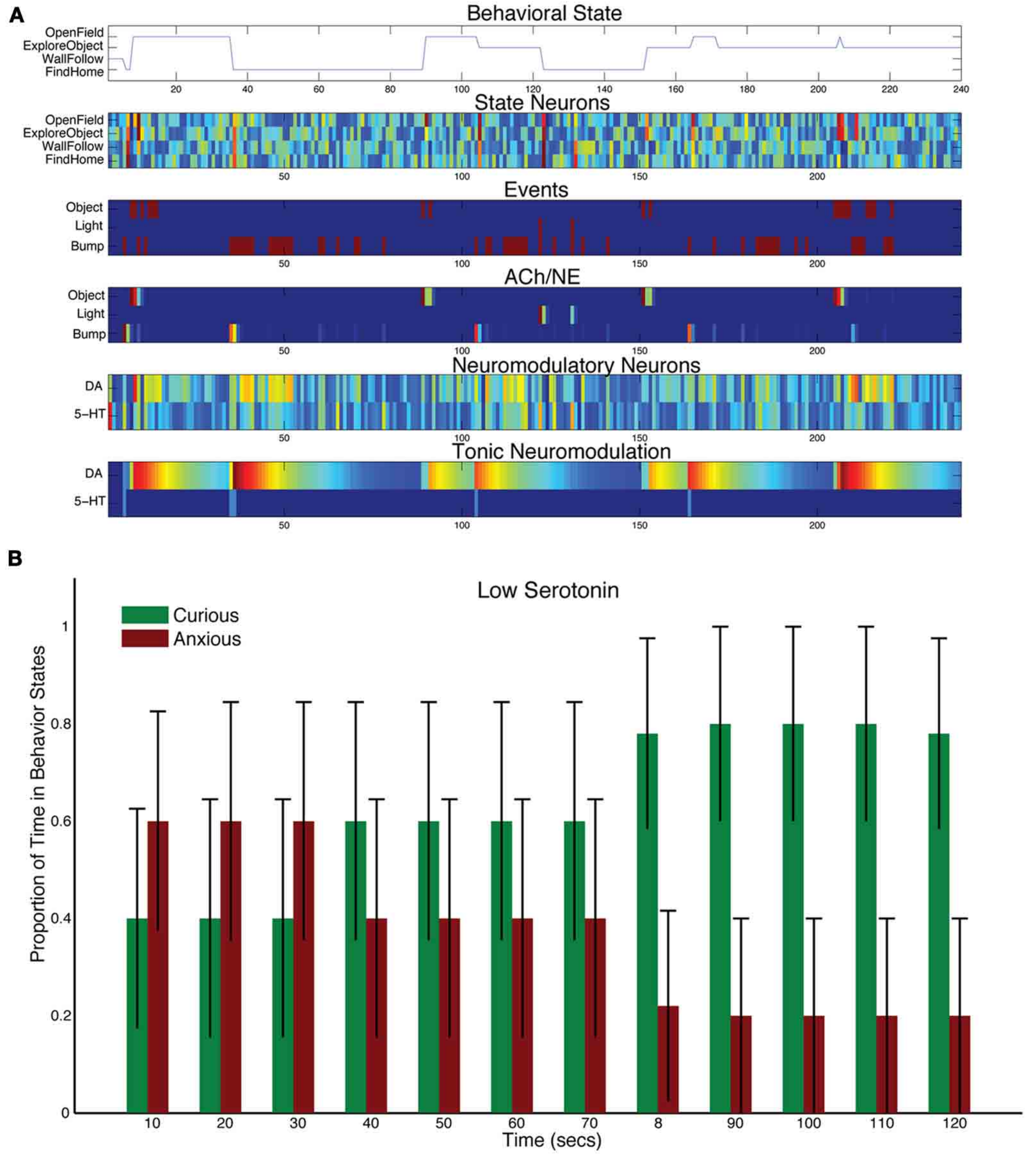


FIGURE 5 | Behavioral and neural responses with low serotonin levels. The time constant τ_{DA} was set to 50 and the time constant τ_{5-HT} was set to 1. **(A)** Behavioral and neural responses in a representative trial. Axes, labels, and color are the same as in **Figure 3A**. **(B)** The proportion of Curious (ExploreObject and OpenField) and Anxious (FindHome and WallFollow) behavior averaged over 5 trials. Axes, labels, and time locking is the same as in **Figure 3C**.

CarlRoomba taking exploratory behavior. **Figure 5B** shows the population behavior of five trials time locked to the light event. There is still some response to the light with anxious behavior, but CarlRoomba quickly switches to more curiosity seeking behavior, much more so than in the control experiments (compare **Figure 3C** with **Figure 5B**), by moving to the open part of the arena and exploring the object in the center.

Lowering serotonin levels through Acute Tryptophan Depletion (ATD) has been shown to reduce harm aversion and increase risk taking in humans (Crockett et al., 2008; Robinson et al., 2010). This is qualitatively similar to CarlRoomba's increased tendency to explore after a stressful event. Interestingly, ATD increased anxious behavior in the open-field test with rats (Blokland et al., 2002). In their discussion, they state that ATD only moderately lowers serotonin levels in rats (40%), but has a stronger effect in humans (80–90%). This may explain the difference between CarlRoomba's behavior and Blokland and colleagues' experiments. Future experiments with only a moderate change to τ_{5HT} may resolve this difference.

DOPAMINE AND RISK TAKING

Increasing the levels of DA by adjusting the tonic time constant ($\tau_{DA} = 150$, and $\tau_{5HT} = 50$ in Equation 5), resulted in more curiosity and risk taking, but did not abolish the stress response (see **Figure 6B**). For example, in the representative trial shown in **Figure 6A**, the light event did cause a strong increase in 5-HT

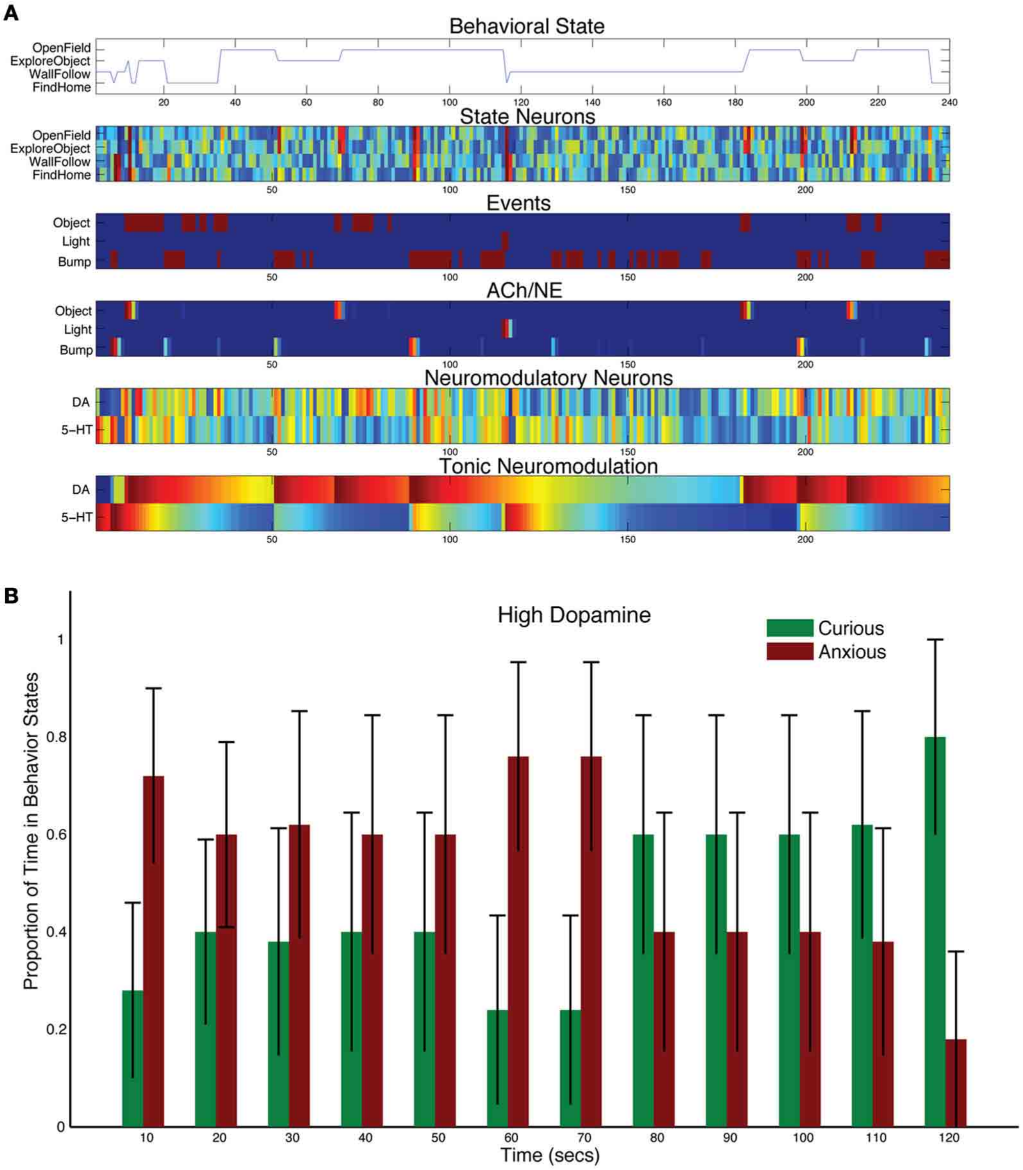


FIGURE 6 | Behavioral and neural responses with high dopamine levels. The time constant τ_{DA} was set to 150 and the time constant τ_{5-HT} was set to 50. **(A)** Behavioral and neural responses in a representative trial. Axes, labels, and color are the same as in **Figure 3A**. **(B)** The proportion of Curious (ExploreObject and OpenField) and Anxious (FindHome and WallFollow) behavior averaged over 5 trials. Axes, labels, and time locking is the same as in **Figure 3C**.

activity, which in turn inhibited DA activity. However, the next sensory events, which were gated through by the AChNE attentional filter at approximately 180, 200, and 220 s, resulted in strong DA activation and curiosity seeking behavior. The population data reflected this interplay between the DA and 5-HT system. CarlRoomba responded to the stressful event, but was much more curious than controls. In effect, CarlRoomba was taking more risks by venturing into the middle of the environment during or right after the stressful light event. Similarly, cocaine, which increases levels of DA in the nervous system, has been shown to increase activity in the open-field test with rats, as well as increase the exploration of novel objects (Carey et al., 2008).

Decreasing the levels of DA by adjusting the tonic time constant ($\tau_{DA} = 1$, and $\tau_{5HT} = 50$ in Equation 5) resulted in less curiosity, and more withdrawn behavior (see **Figure 7**). Object events did sometimes results in curious behavior (see 180 s into the trial shown in **Figure 7A**). But, in general, without much DA in the system, the 5-HT system dominated action selection leading to anxious behavior, such as following walls and searching for its home (i.e., docking station). For example, the bump event at 200 s into the trial in **Figure 7A**, triggered an anxious FindHome response by CarlRoomba. Overall, CarlRoomba's behavior was considerably more anxious when comparing the low DA condition (**Figure 7B**) to the control condition (**Figure 3C**).

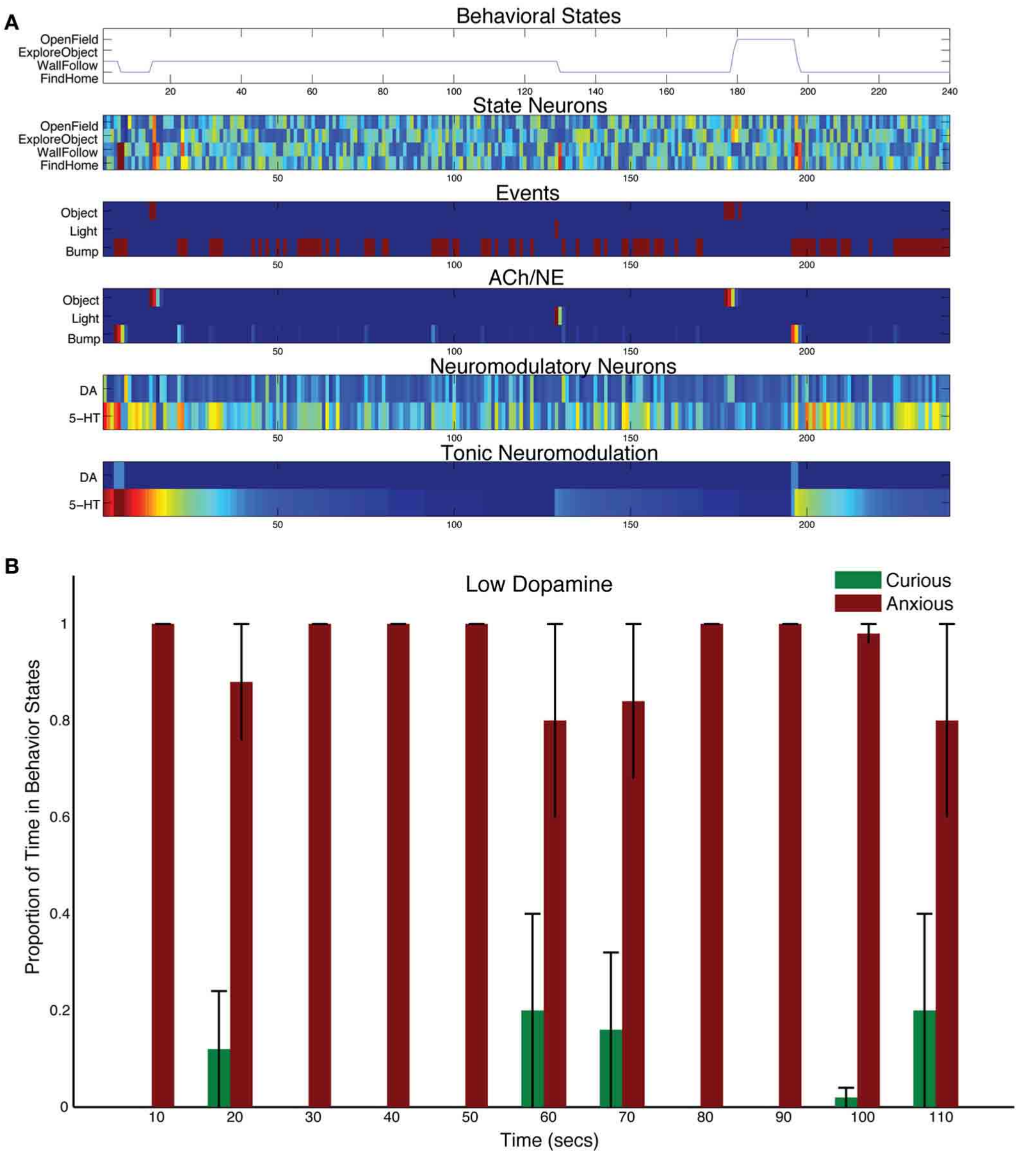


FIGURE 7 | Behavioral and neural responses with low dopamine levels. The time constant τ_{DA} was set to 1 and the time constant τ_{5-HT} was set to 50. **(A)** Behavioral and neural responses in a representative trial. Axes, labels, and color are the same as in **Figure 3A**. **(B)** The proportion of Curious (ExploreObject and OpenField) and Anxious (FindHome and WallFollow) behavior averaged over 5 trials. Axes, labels, and time locking is the same as in **Figure 3C**.

FRONTAL CORTEX AND COGNITIVE CONTROL

The OFC and mPFC areas of the model exert cognitive control on CarlRoomba's behavior by inhibiting the DA and 5-HT systems, respectively (see **Figure 2**). Activity in these areas initiated behavior selection, but also inhibited the neuromodulatory systems. This inhibition kept the appropriate neuromodulatory system in check and exerted cognitive control by signaling to the neuromodulatory system that the sensory event had been handled.

When the projections from mPFC to 5-HT were lesioned in the model, the serotonergic system was overactive and CarlRoomba acted anxious almost entirely (see **Figure 8A**). In all mPFC lesion cases, the light response triggered anxious behavior that persisted throughout the remainder of the trial (see **Figure 8B**).

When the projections from OFC to DA were lesioned in the model, DA levels dominated and more exploratory behavior was observed (see **Figure 9**). Although CarlRoomba showed more curious behavior, anxious behavior was not abolished (compare **Figure 8B** with **Figure 9B**). The asymmetry between these lesion experiments may be due to the opponency between the serotonergic and DA systems. The serotonergic system, through its inhibition of the DA system, can still trigger anxious behavior in response to a stressful event and may keep DA levels in check.

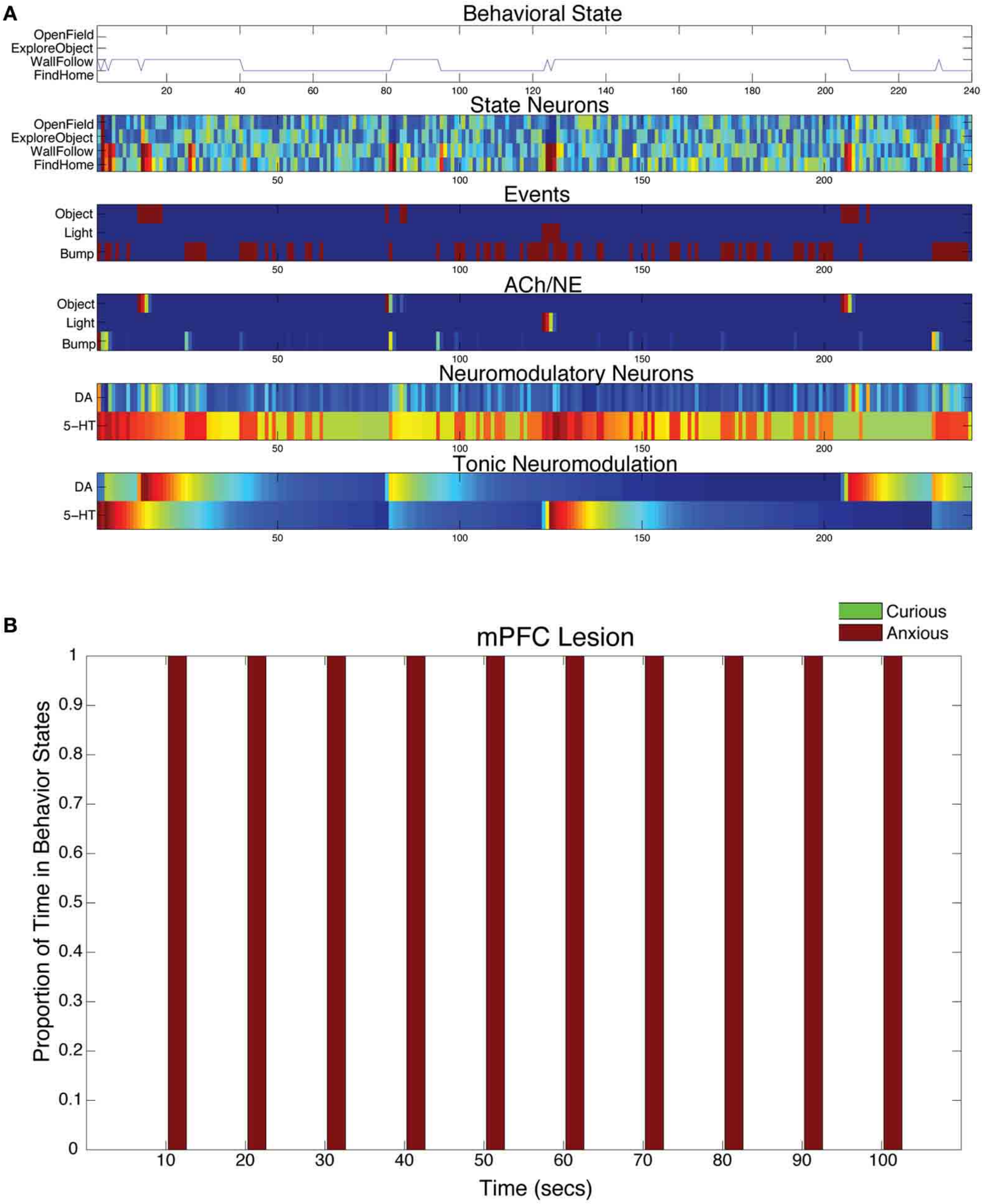


FIGURE 8 | Behavioral and neural responses with lesion to projection from mPFC to 5-HT. The time constants τ_{DA} and τ_{5-HT} were both set at 50. **(A)** Behavioral and neural responses in a representative trial. Axes, labels, and color are the same as in **Figure 3A**. **(B)** The proportion of Curious (ExploreObject and OpenField) and Anxious (FindHome and WallFollow) behavior averaged over 5 trials. Axes, labels, and time locking is the same as in **Figure 3C**.

DISCUSSION

The main purposes of the present neurorobotic study were to demonstrate that (1) high levels of serotonin lead to withdrawn behavior, and that (2) top-down signals from the frontal cortex to neuromodulatory areas are critical for coping with both stressful and novel events. Firstly, it has been suggested that serotonin opposes activating or invigorating neuromodulators such as dopamine (Tops et al., 2009). When the simulated nervous system was intact, the neurorobot appropriately responded to a stressful event with an increase in 5-HT activity. This led to withdrawn behavior by activating the mPFC and suppressing DA activity. Secondly, a recent review suggested that the mPFC inhibited the serotonergic raphe nucleus after handling a stressful event (Jasinska et al., 2012). In the present model, this feedback loop prevented the raphe from being overly active after the stressor had been handled. Over time, this allowed the DA system to become

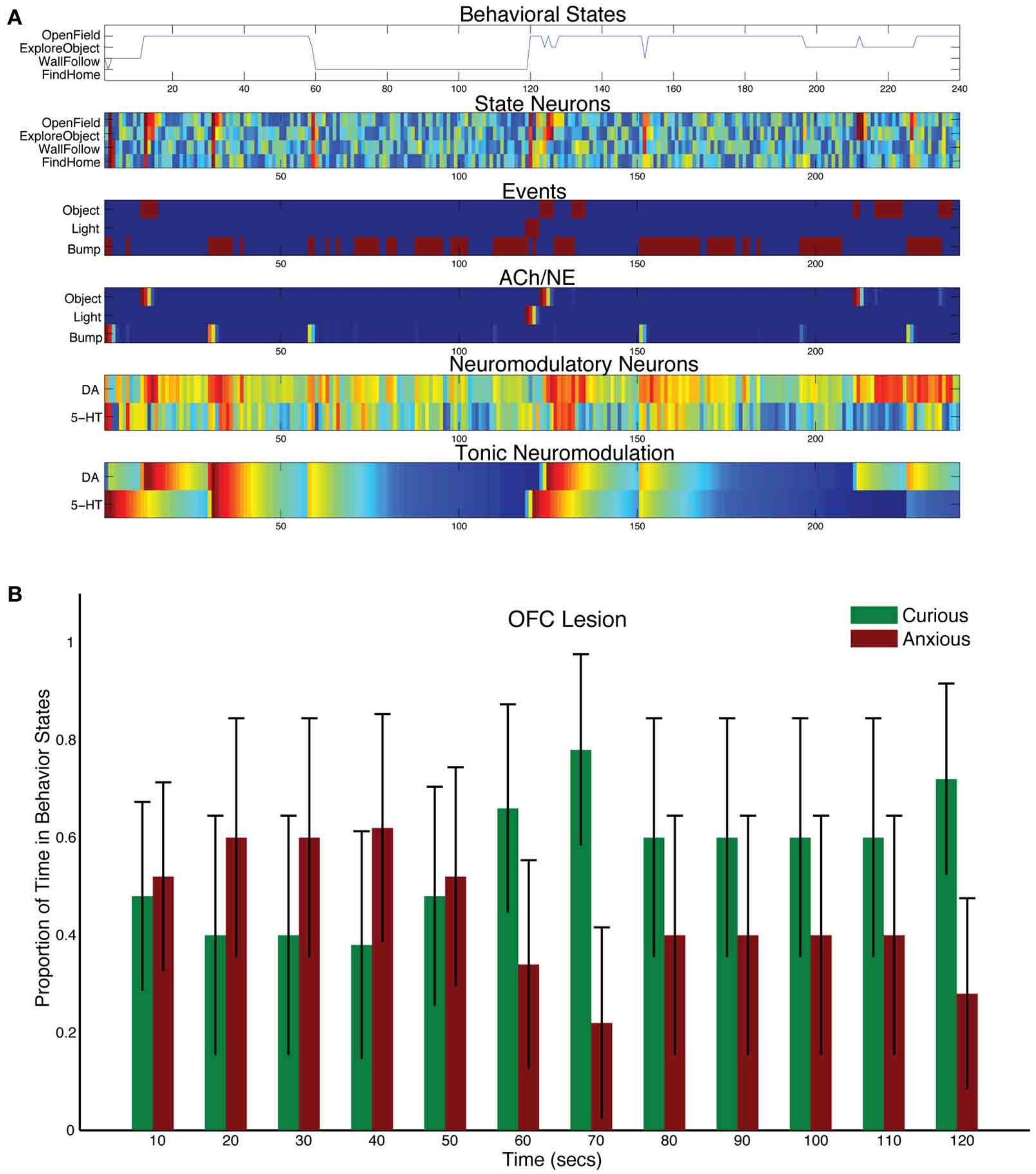


FIGURE 9 | Behavioral and neural responses with lesion to projection from OFC to DA. The time constants τ_{DA} and τ_{5-HT} were both set at 50. **(A)** Behavioral and neural responses in a representative trial. Axes, labels, and color are the same as in **Figure 3A**. **(B)** The proportion of Curious (ExploreObject and OpenField) and Anxious (FindHome and WallFollow) behavior averaged over 5 trials. Axes, labels, and time locking is the same as in **Figure 3C**.

active leading to exploratory behavior. The present algorithm further suggested that projections from the OFC to the DA function have a similar function when responding to positive novel events. Lastly, the introduction of the attentional filter in the ACh and NE systems allowed the neurorobot to respond to novel events and habituate to irrelevant events. As was shown in Krichmar (2012), when the ACh/NE system was compromised, the neurorobot was distracted by irrelevant events and switched behaviors constantly.

The behavior of the robot was similar to that observed in rodents under similar conditions. Specifically, the neurorobot, CarlRoomba, and the rodent are initially anxious or wary, resulting in staying near their nest or the walls of the arena (Fonio et al., 2009). After becoming familiar with the environment, both the rodent and CarlRoomba made forays into the middle of the arena. **Figure 3** summarizes this behavior in the neurorobot. Because CarlRoomba started each trial pointed directly at the object in the middle of the environment, there was some selection of OpenField and ExploreObject behaviors early on. In Fonio's experimental paradigm, the moving of a mouse to a novel environment is presumably a stressful event. However, this prior context would be difficult to mimic with the CarlRoomba. Therefore, a light flash was used to mimic a stressful event. In this case, CarlRoomba's behavior was qualitatively similar to the rodent. CarlRoomba tended to stay near its docking station or the walls of the arena. By 100 s after the light flash (see **Figure 3C**), CarlRoomba spent over half its time either crossing the center of the environment or investigating a novel object in the center of the environment.

Opponency between the serotonergic system and the DA system has been proposed behaviorally and in theoretical models (Daw et al., 2002; Tops et al., 2009). However, whether the anatomy supports uni-directional or bi-directional inhibition is an open issue (Boureau and Dayan, 2011). But there is evidence that projections from raphe serotonin cells to DA areas oppose the action of DA and mediate avoidance of threats (Deakin, 2003). Therefore, opponency in the present neurorobotic framework was modeled by inhibition from the raphe nucleus to the ventral tegmental area (shown as 5-HT→DA in **Figure 2**). There were also practical reasons for this projection. First, there was a need to arbitrate between sensory events that might trigger both DA and 5-HT, such as a bump event. Second, by having 5-HT inhibit DA, a bump event would cause anxious behavior early in a trial (Fonio et al., 2009) and after a stressor (Jasinska et al., 2012). This matches behavioral data and suggests that the serotonergic system may be actively opposing the dopaminergic system, and that dopaminergic system exerts its influence if serotonin levels are sufficiently low. Lastly, it may be advantageous, from a robot control perspective, to be initially conservative, but transition from conservative to riskier action over time if environmental conditions warrant such action.

SEROTONIN AND RISK-AVERSE BEHAVIOR

The serotonergic system is involved in the control of anxious states (Millan, 2003). For instance, a variation of an upstream promoter region of the serotonin transporter gene (5-HTTLPR) has been shown to influence both behavioral measures of social anxiety and amygdala response to social threats in humans (Hariri et al., 2002; Caspi et al., 2003, 2010). Lowering serotonin levels, through a dietary manipulation called ATD, has been shown to decrease cooperation and lower harm-aversion (Wood et al., 2006; Crockett et al., 2008). Moreover, manipulations of 5-HT receptor genes have an impact on stress and anxiety in mice (Heisler et al., 1998; Weisstaub et al., 2006; Holmes, 2008).

These serotonin-dependent traits and responses were shown in the present robot experiments. Increasing serotonin levels by lengthening the time constant for tonic 5-HT had a similar effect to the short allele variant of 5-HTTLPR. The robot showed stronger and long-lasting responses to a stressful event, that is, a bright light (see **Figure 4**). Indeed, these open-field responses are in agreement with mouse behavior, where manipulations to 5-HT1A and 5-HT2A receptors resulted in elevated anxiety in the open-field test as measured by center locomotion, overall distance traveled, rearing, and response to a novel object (Heisler et al., 1998; Weisstaub et al., 2006).

Similar to the decrease in harm aversion shown due to ATD (Wood et al., 2006; Crockett et al., 2008), decreasing serotonin levels in the model, through shortening the 5-HT time constant, had the effect of making the robot more risk taking (see **Figure 5**). The robot made more forays into the center of the environment, and more explorations of the object in the center of the environment.

DOPAMINE AND RISK-TAKING BEHAVIOR

The DA system has been implicated in the prediction of rewards and incentive salience or "wanting" (Schultz et al., 1997; Berridge, 2004), as well as novelty-seeking (Redgrave and Gurney, 2006; Bromberg-Martin et al., 2010). Variations in the DA system have been shown to affect risk-taking during gambling, the ability to filter out noise, and cognitive flexibility (Winterer and Weinberger, 2004; Roussos et al., 2008). A blockade of DA resulted in rats not making an extra effort of climbing over a barricade to get a high reward (Denk et al., 2005). This might be interpreted as low DA levels lead to less risk taking for potential rewards. Similarly, a human study has shown that individuals with a COMT polymorphism, which lowered levels of DA in the prefrontal cortex, tended to take fewer risks in a gambling task (Roussos et al., 2008). Moreover, individuals with this polymorphism persisted in accordance with prior instructions despite evidence that the rules had changed (Doll et al., 2011). Genetic variation in the DA system also has an effect on impulsivity. Polymorphisms in DA-related genes, including variable number tandem repeat (VNTR) polymorphisms in DRD4 and DAT1, have been associated with poor "action restraint" and "action cancellation" (Congdon et al., 2008; Munafo et al., 2008).

These DA-dependent behaviors and responses were observed in the robot's behavior and simulated nervous system. Similar to the Denk and Roussos findings, lowering tonic levels of DA led to a lack of risk-taking and more withdrawn behavior (Denk et al., 2005; Roussos et al., 2008). This was mainly due to the serotonergic system dominating and driving harm aversive behaviors, such as finding home or wall following (see **Figure 7**). It also led to behavior that could be regarded as impulsive since CarlRoomba perseverated with these behaviors. However, when the DA levels were elevated, the robot tended toward curious behavior (see **Figure 6**). It is interesting that in this condition, compared to others, the change in behavior is not as dramatic. It makes the prediction that the "anxious" behavior system (i.e., mPFC←→5-HT) may keep the "curiousity-seeking" behavior system (i.e., OFC←→DA) somewhat in check.

FRONTAL CORTEX AND COGNITIVE CONTROL

Recent experiments suggest that the reward and cost of actions are also partially represented in OFC and mPFC, respectively. In general, OFC appears to be involved in decision-making and planning with respect to rewards and preferences, and the mPFC appears to be involved in decision-making and planning having to do with effort, cost, and social valuation (Rushworth et al., 2007). Rudebeck et al., for example, trained rats to choose maze arms that yielded more food pellets either after a delay or after scaling a barrier (Rudebeck et al., 2006). When the OFC was lesioned, the rat was more likely to choose the lower (immediate) reward than the higher (deferred) reward. However, mPFC lesions, specifically the anterior cingulate cortex, caused rats to more often pick lower (less effortful) rewards than higher (more effortful) rewards. Moreover, unit recordings in the rat anterior cingulate cortex have shown that many of these neurons respond to effort during goal-directed actions (Cowen et al., 2012).

In the model, when CarlRoomba responded to a stressful event (e.g., bright light), there was first a phasic response in the 5-HT system, causing activity in the appropriate mPFC state neurons, resulting in the selection of a stress reducing behavior, and then the mPFC inhibited the 5-HT system, since it had dealt with the stressor. However, lesioning the connections from mPFC to the 5-HT system had a dramatic effect on behavior; anxious behavior completely dominated because cognitive control of the serotonergic systems was absent. CarlRoomba became withdrawn since the cognitive control of the serotonergic system was removed (see **Figure 8**).

Evidence suggests that mPFC mediates the cognitive control of stress by regulating the raphe nucleus (i.e., serotonergic system) (Maier and Watkins, 2010). In a study where rats were subjected to tailshocks, inactivation of the mPFC resulted in the elimination of the ability to control the stressor through regulation of raphe nucleus serotonin levels (Amat et al., 2005). Interestingly, Lacroix and colleagues found that lesions of the mPFC did not increase anxiety in rats during unconditioned fear paradigms, such as the open-field test, but increased anxiety during conditioning paradigms (Lacroix et al., 2000). The present model does not have the type of learning to support conditioning. Future models of CarlRoomba may need to investigate this dissociation with the addition of biologically plausible learning rules.

In a similar fashion to the model of mPFC's control of stress, CarlRoomba's OFC exerted control on incentive salience or reward-seeking. When CarlRoomba responded to a potentially interesting event, such as an object or a bump, there was first a phasic response in the DA system, causing activity in OFC state neurons, resulting in the choice of a reward-seeking behavior (e.g., OpenField or ExploreObject) and then the OFC inhibited the DA system, since it had responded to the event of interest. However, when the OFC was lesioned, the robot perseverated in its curious behavior (see **Figure 9**). In about 50% of the trials, CarlRoomba did not respond to the stressful light event and continued with its "Curious" behavior.

It has been suggested that the OFC is crucial for adaptation when reward values or contextual cues change (Rolls, 2004), and that the OFC is important for developing stimulus to reward associations, prediction, and expectancies (Schoenbaum et al., 2009). A recent rodent study showed that, depending on the conditions, the OFC is important for both of these roles (Riceberg and Shapiro, 2012). OFC lesions impaired reversal learning when the reversals occurred at low frequencies. However, when the contingencies changed at a high frequency, OFC lesions rats followed a Lose-Shift strategy. The authors suggest that OFC is computing reward expectancies based on reward history. Although CarlRoomba does not contain the learning machinery to calculate reward expectancies, it does show perseverative behavior when from the OFC to the DA system are lesioned. The OFC lesioned CarlRoomba also showed a lack of ability to assess the potential rewards for a given event (i.e., all events became highly rewarding). It will be of interest to add predictive reward learning (e.g., TD learning) to the model and test the system in a reversal learning task.

RELATED WORK

While there have been many models of action selection, the present work addresses how principles of neuromodulation and frontal cortex control could control autonomous robot behavior. It should be noted that other neural systems support action selection and behavioral switching. For example, the basal ganglia and its interaction with thalamocortical loops have been proposed as an action selection system (Prescott et al., 2006). This model, which was tested on a neurorobot, demonstrated behavioral switching in an open environment during a foraging task where the robot switched between wall-seeking, wall-following, approaching and placing objects. Similar to the present model, this basal ganglia model was able to choose between multiple, conflicting choices based on its context and motivation.

The present model was specifically designed to test how the opponency between the serotonergic and dopaminergic system, combined with top-down control from frontal cortex, could replicate rodent behavior. Moreover, it was able to show how altering the balance between these systems could influence anxious and exploratory behavior. These results can be compared to rodent studies under similar condition as described above (Heisler et al., 1998; Lacroix et al., 2000; Blokland et al., 2002; Lipkind et al., 2004; Bouwknecht et al., 2007). Future experiments may further delineate the role of these neuromodulators in balancing exploratory and anxious behavior. Moreover, the present neurorobotic experiments tests the feasibility of the architecture proposed by Jasinska and colleagues, where there is interaction between the mPFC and the raphe nucleus, for handling stressful events (Jasinska et al., 2012). CarlRoomba's neural architecture further suggests that there is a similar architecture between the OFC and DA system for handling positive valence stimuli.

Theoretical models have been proposed on neuromodulation, but they typically have not considered all of the neuromodulatory systems and their interactions with cortical and subcortical areas. The phasic response of the DA system has been proposed to signal temporal difference error (Schultz et al., 1997). Following this idea, the phasic response of DA has been modeled to shape behavior and action selection with reinforcement learning (Krichmar and Edelman, 2002; Sporns and Alexander, 2002; Arleo et al., 2004; Iida et al., 2004; Doya and Uchibe, 2005; Stone et al., 2005; Guenter et al., 2007; Nakamura et al., 2007).

Several neurorobot and computational neuroscience studies have investigated the interaction between multiple neuromodulatory systems. Our previous model took into consideration the phasic aspects of dopaminergic and serotonergic neuromodulation (Cox and Krichmar, 2009). This model postulated, similar to a model of noradrenergic neuromodulation (Aston-Jones and Cohen, 2005), that phasic neuromodulation causes an organism to be more decisive, whereas a lack of phasic response would result in more arbitrary action selection. A recent neurorobot study combined DA reinforcement learning with an exploration parameter related to the noradrenergic system (Khamassi et al., 2011). These simulated neuromodulatory systems interacted with an anterior cingulate cortex and prefrontal cortex. On two different robot platforms, they demonstrated that their model could deal with both expected and unexpected uncertainties in the real world. Our group has recently investigated the possible role of multiple neuromodulators in a resource allocation task (Chelian et al., 2012), and reversal learning on an autonomous robot (Oros and Krichmar, 2012).

However, few researchers have developed a model that includes the ACh, DA, NE, and 5-HT systems simultaneously. One exception was a theory proposed by Kenji Doya (Doya, 2002, 2008). In this theory, Doya subscribed a different functional role for each neuromodulatory system on different parameters of the temporal difference learning rule. Although this idea has not been implemented in a behaving robot, their group is actively exploring elements of this theory experimentally (Tanaka et al., 2007; Schweighofer et al., 2008). Our previous model showed how the combination of these neuromodulatory systems could produce effective action selection in robots (Krichmar, 2012).

The present model extends this prior work and takes into consideration the notion that the dopaminergic and serotonergic systems are in opposition. Specifically, the serotonergic system is inhibiting the dopaminergic system. One model that investigated these opponent interactions, suggested that tonic serotonin tracked the average reward rate and that tonic dopamine tracked the average punishment rate in a similar context, and speculated that a phasic serotonin signal might report an ongoing prediction error for future punishment (Daw et al., 2002). However, it has been difficult to find empirical evidence supporting these roles for tonic and phasic neuromodulation. Our prior modeling has shown that direct opponency between these systems is not necessary to achieve behavioral opponency (Asher et al., 2010, 2012; Zaldivar et al., 2010). In many cases there is an environmental tradeoff between the expected rewards and costs, and this can lead to opponency between active reward-seeking and withdrawn behavior. Indeed, by having different neuromodulatory systems handle different sensory events, this type of opponency emerged in the present model.

CONCLUSIONS

The neurorobotic experiments presented here demonstrate that the opposition of the serotoninergic system with the dopaminergic system can lead to the type of anxious and curious behavior observed in animals. Whereas high levels of 5-HT led to withdrawn, anxious behavior by suppressing DA action, high levels of DA or low levels of 5-HT led to curious, exploratory behavior. Moreover, it was shown that top-down signals from the frontal cortex to these neuromodulatory areas were critical for handling both stressful and positive valence events. The action of the neuromodulatory system and its interaction with areas important for action selection and planning are in a fine balance. It was shown that if any of these systems become out of balance, due to lesions or changes to the efficiency of neuromodulatory signaling, aberrant behavior occurs. This may have implications for understanding mood disorders, obsessive-compulsive disorders, and anxiety.

ACKNOWLEDGMENTS

Supported by the National Science Foundation (Award Nos: EMT/BSSE-0829752 & IIS/RI-0910710).

REFERENCES

Amat, J., Baratta, M. V., Paul, E., Bland, S. T., Watkins, L. R., and Maier, S. F. (2005). Medial prefrontal cortex determines how stressor controllability affects behavior and dorsal raphe nucleus. *Nat. Neurosci.* 8, 365–371.

Arkin, R. C. (1998). *Behavior-Based Robotics (Intelligent Robotics and Autonomous Agents)*. Cambridge, MA: The MIT Press.

Arleo, A., Smeraldi, F., and Gerstner, W. (2004). Cognitive navigation based on nonuniform Gabor space sampling, unsupervised growing networks, and reinforcement learning. *IEEE Trans. Neural Netw.* 15, 639–652.

Asher, D. E., Zaldivar, A., Barton, B., Brewer, A. A., and Krichmar, J. L. (2012). Reciprocity and retaliation in social games with adaptive agents. *IEEE Trans. Auton. Ment. Dev.* 4, 226–238.

Asher, D. E., Zaldivar, A., and Krichmar, J. L. (2010). "Effect of neuromodulation on performance in game playing: a modeling study," in *Paper Presented at: 2010 IEEE 9th International Conference on Development and Learning* (Ann Arbor, MI: IEEE Xplore).

Aston-Jones, G., and Cohen, J. D. (2005). An integrative theory of locus coeruleus-norepinephrine function: adaptive gain and optimal performance. *Annu. Rev. Neurosci.* 28, 403–450.

Berridge, K. C. (2004). Motivation concepts in behavioral neuroscience. *Physiol. Behav.* 81, 179–209.

Blokland, A., Lieben, C., and Deutz, N. E. (2002). Anxiogenic and depressive-like effects, but no cognitive deficits, after repeated moderate tryptophan depletion in the rat. *J. Psychopharmacol.* 16, 39–49.

Boureau, Y. L., and Dayan, P. (2011). Opponency revisited: competition and cooperation between dopamine and serotonin. *Neuropsychopharmacology* 36, 74–97.

Bouwknecht, J. A., Spiga, F., Staub, D. R., Hale, M. W., Shekhar, A., and Lowry, C. A. (2007). Differential effects of exposure to low-light or high-light open-field on anxiety-related behaviors: relationship to c-Fos expression in serotonergic and non-serotonergic neurons in the dorsal raphe nucleus. *Brain Res. Bull.* 72, 32–43.

Briand, L. A., Gritton, H., Howe, W. M., Young, D. A., and Sarter, M. (2007). Modulators in concert for cognition: modulator interactions in the prefrontal cortex. *Prog. Neurobiol.* 83, 69–91.

Bromberg-Martin, E. S., Matsumoto, M., and Hikosaka, O. (2010). Dopamine in motivational control: rewarding, aversive, and alerting. *Neuron* 68, 815–834.

Brooks, R. A. (1991). Intelligence without representation. *Artif. Intell.* 47, 139–159.

Bucci, D. J., Holland, P. C., and Gallagher, M. (1998). Removal of cholinergic input to rat posterior parietal cortex disrupts incremental processing of conditioned stimuli. *J. Neurosci.* 18, 8038–8046.

Carey, R. J., Damianopoulos, E. N., and Shanahan, A. B. (2008). Cocaine effects on behavioral responding to a novel object placed in a familiar environment. *Pharmacol. Biochem. Behav.* 88, 265–271.

Caspi, A., Hariri, A. R., Holmes, A., Uher, R., and Moffitt, T. E. (2010). Genetic sensitivity to the environment: the case of the serotonin transporter gene and its implications for studying complex diseases and traits. *Am. J. Psychiatry* 167, 509–527.

Caspi, A., Sugden, K., Moffitt, T. E., Taylor, A., Craig, I. W., Harrington, H., et al. (2003). Influence of life stress on depression: moderation by a polymorphism in the 5-HTT gene. *Science* 301, 386–389.

Chelian, S. E., Oros, N., Zaldivar, A., Krichmar, J., and Bhattacharyya, R. (2012). "Model of the interactions between neuromodulators and prefrontal cortex during a resource allocation task," in *Paper Presented at: IEEE International Conference on Development and Learning and Epigenetic Robotics (IEEE ICDL-EpiRob 2012)* (San Diego, CA).

Congdon, E., Lesch, K. P., and Canli, T. (2008). Analysis of DRD4 and DAT polymorphisms and behavioral inhibition in healthy adults: implications for impulsivity. *Am. J. Med. Genet. B Neuropsychiatr. Genet.* 147B, 27–32.

Cools, R., Roberts, A. C., and Robbins, T. W. (2008). Serotoninergic regulation of emotional and behavioural control processes. *Trends Cogn. Sci.* 12, 31–40.

Cowen, S. L., Davis, G. A., and Nitz, D. A. (2012). Anterior cingulate neurons in the rat map anticipated effort and reward to their associated action sequences. *J. Neurophysiol.* 107, 2393–2407.

Cox, B. R., and Krichmar, J. L. (2009). Neuromodulation as a robot controller: a brain inspired design strategy for controlling autonomous robots. *IEEE Robot. Autom. Mag.* 16, 72–80.

Crockett, M. J., Clark, L., Tabibnia, G., Lieberman, M. D., and Robbins, T. W. (2008). Serotonin modulates behavioral reactions to unfairness. *Science* 320, 1739.

Daw, N. D., Kakade, S., and Dayan, P. (2002). Opponent interactions between serotonin and dopamine. *Neural Netw.* 15, 603–616.

Deakin, J. F. (2003). Depression and antisocial personality disorder: two contrasting disorders of 5HT function. *J. Neural Transm. Suppl.* 64, 79–93.

Denk, F., Walton, M. E., Jennings, K. A., Sharp, T., Rushworth, M. F., and Bannerman, D. M. (2005). Differential involvement of serotonin and dopamine systems in cost-benefit decisions about delay or effort. *Psychopharmacology* 179, 587–596.

Doll, B. B., Hutchison, K. E., and Frank, M. J. (2011). Dopaminergic genes predict individual differences in susceptibility to confirmation bias. *J. Neurosci.* 31, 6188–6198.

Doya, K. (2002). Metalearning and neuromodulation. *Neural Netw.* 15, 495–506.

Doya, K. (2008). Modulators of decision making. *Nat. Neurosci.* 11, 410–416.

Doya, K., and Uchibe, E. (2005). The Cyber Rodent project: exploration of adaptive mechanisms for self-preservation and self-reproduction. *Adapt. Behav.* 13, 149–160.

Fonio, E., Benjamini, Y., and Golani, I. (2009). Freedom of movement and the stability of its unfolding in free exploration of mice. *Proc. Natl. Acad. Sci. U.S.A.* 106, 21335–21340.

Frank, M. J., and Claus, E. D. (2006). Anatomy of a decision: striato-orbitofrontal interactions in reinforcement learning, decision making, and reversal. *Psychol. Rev.* 113, 300–326.

Guenter, F., Hersch, M., Calinon, S., and Billard, A. (2007). Reinforcement learning for imitating constrained reaching movements. *Adv. Robot.* 21, 1521–1544.

Hariri, A. R., Mattay, V. S., Tessitore, A., Kolachana, B., Fera, F., Goldman, D., et al. (2002). Serotonin transporter genetic variation and the response of the human amygdala. *Science* 297, 400–403.

Hasselmo, M. E., and McGaughy, J. (2004). High acetylcholine levels set circuit dynamics for attention and encoding and low acetylcholine levels set dynamics for consolidation. *Prog. Brain Res.* 145, 207–231.

Heisler, L. K., Chu, H. M., Brennan, T. J., Danao, J. A., Bajwa, P., Parsons, L. H., et al. (1998). Elevated anxiety and antidepressant-like responses in serotonin 5-HT1A receptor mutant mice. *Proc. Natl. Acad. Sci. U.S.A.* 95, 15049–15054.

Holmes, A. (2008). Genetic variation in cortico-amygdala serotonin function and risk for stress-related disease. *Neurosci. Biobehav. Rev.* 32, 1293–1314.

Iida, S., Kuwayama, K., Kanoh, M., Kato, S., and Itoh, H. (2004). A dynamic allocation method of basis functions in reinforcement learning. *Adv. Artif. Intell.* 3339, 272–283.

Jasinska, A. J., Lowry, C. A., and Burmeister, M. (2012). Serotonin transporter gene, stress and raphe raphe interactions: a molecular mechanism of depression. *Trends Neurosci.* 35, 395–402.

Khamassi, M., Lallee, S., Enel, P., Procyk, E., and Dominey, P. F. (2011). Robot cognitive control with a neurophysiologically inspired reinforcement learning model. *Front. Neurorobot.* 5:1. doi: 10.3389/fnbot.2011.00001

Krichmar, J. L. (2008). The neuromodulatory system – a framework for survival and adaptive behavior in a challenging world. *Adapt. Behav.* 16, 385–399.

Krichmar, J. L. (2012). "A biologically inspired action selection algorithm based on principles of neuromodulation," in *Paper Presented at: IEEE World Congress on Computational Intelligence* (Brisbane, QLD).

Krichmar, J. L., and Edelman, G. M. (2002). Machine psychology: autonomous behavior, perceptual categorization, and conditioning in a brain-based device. *Cereb. Cortex* 12, 818–830.

Lacroix, L., Spinelli, S., Heidbreder, C. A., and Feldon, J. (2000). Differential role of the medial and lateral prefrontal cortices in fear and anxiety. *Behav. Neurosci.* 114, 1119–1130.

Lipkind, D., Sakov, A., Kafkafi, N., Elmer, G. I., Benjamini, Y., and Golani, I. (2004). New replicable anxiety-related measures of wall vs center behavior of mice in the open field. *J. Appl. Physiol.* 97, 347–359.

Maier, S. F., and Watkins, L. R. (2010). Role of the medial prefrontal cortex in coping and resilience. *Brain Res.* 1355, 52–60.

Millan, M. J. (2003). The neurobiology and control of anxious states. *Prog. Neurobiol.* 70, 83–244.

Munafo, M. R., Yalcin, B., Willis-Owen, S. A., and Flint, J. (2008). Association of the dopamine D4 receptor (DRD4) gene and approach-related personality traits: meta-analysis and new data. *Biol. Psychiatry* 63, 197–206.

Nakamura, Y., Mori, T., Sato, M. A., and Ishii, S. (2007). Reinforcement learning for a biped robot based on a CPG-actor-critic method. *Neural Netw.* 20, 723–735.

Oros, N., and Krichmar, J. L. (2012). "Neuromodulation, attention and localization using a novel Android™ Robotic Platform," in *ICDL-EpiRob 2012: IEEE Conference on Development and Learning and Epigenetic Robotics* (San Diego, CA: IEEE Explore).

Prescott, T. J., Montes Gonzalez, F. M., Gurney, K., Humphries, M. D., and Redgrave, P. (2006). A robot model of the basal ganglia: behavior and intrinsic processing. *Neural Netw.* 19, 31–61.

Redgrave, P., and Gurney, K. (2006). The short-latency dopamine signal: a role in discovering novel actions? *Nat. Rev.* 7, 967–975.

Riceberg, J. S., and Shapiro, M. L. (2012). Reward stability determines the contribution of orbitofrontal cortex to adaptive behavior. *J. Neurosci.* 32, 16402–16409.

Robinson, O., Cools, R., Crockett, M., and Sahakian, B. (2010). Mood state moderates the role of serotonin in cognitive biases. *J. Psychopharmacol.* 24, 573–583.

Rolls, E. T. (2004). The functions of the orbitofrontal cortex. *Brain Cogn.* 55, 11–29.

Roussos, P., Giakoumaki, S. G., Pavlakis, S., and Bitsios, P. (2008). Planning, decision-making and the COMT rs4818 polymorphism in healthy males. *Neuropsychologia* 46, 757–763.

Rudebeck, P. H., Walton, M. E., Smyth, A. N., Bannerman, D. M., and Rushworth, M. F. (2006). Separate neural pathways process different decision costs. *Nat. Neurosci.* 9, 1161–1168.

Rushworth, M. F., Behrens, T. E., Rudebeck, P. H., and Walton, M. E. (2007). Contrasting roles for cingulate and orbitofrontal cortex in decisions and social behaviour. *Trends Cogn. Sci.* 11, 168–176.

Schoenbaum, G., Roesch, M. R., Stalnaker, T. A., and Takahashi, Y. K. (2009). A new perspective on the role of the orbitofrontal cortex in adaptive behaviour. *Nat. Rev.* 10, 885–892.

Schultz, W., Dayan, P., and Montague, P. R. (1997). A neural substrate of prediction and reward. *Science* 275, 1593–1599.

Schweighofer, N., Bertin, M., Shishida, K., Okamoto, Y., Tanaka, S. C., Yamawaki, S., et al. (2008). Low-serotonin levels increase delayed reward discounting in humans. *J. Neurosci.* 28, 4528–4532.

Simon, P., Dupuis, R., and Costentin, J. (1994). Thigmotaxis as an index of anxiety in mice. Influence of dopaminergic transmissions. *Behav. Brain Res.* 61, 59–64.

Sporns, O., and Alexander, W. H. (2002). Neuromodulation and plasticity in an autonomous robot. *Neural Netw.* 15, 761–774.

Stone, P., Sutton, R. S., and Kuhlmann, G. (2005). Reinforcement learning for RoboCup soccer keepaway. *Adapt. Behav.* 13, 165–188.

Tanaka, S. C., Schweighofer, N., Asahi, S., Shishida, K., Okamoto, Y., Yamawaki, S., et al. (2007). Serotonin differentially regulates short- and long-term prediction of rewards in the ventral and dorsal striatum. *PLoS ONE* 2:e1333. 10.1371/journal.pone.0001333

Tops, M., Russo, S., Boksem, M. A., and Tucker, D. M. (2009). Serotonin: modulator of a drive to withdraw. *Brain Cogn.* 71, 427–436.

Vankov, A., Herve-Minvielle, A., and Sara, S. J. (1995). Response to novelty and its rapid habituation in locus coeruleus neurons of the freely exploring rat. *Eur. J. Neurosci.* 7, 1180–1187.

Weisstaub, N. V., Zhou, M., Lira, A., Lambe, E., Gonzalez-Maeso, J., Hornung, J. P., et al. (2006). Cortical 5-HT2A receptor signaling modulates anxiety-like behaviors in mice. *Science* 313, 536–540.

Winterer, G., and Weinberger, D. R. (2004). Genes, dopamine and cortical signal-to-noise ratio in schizophrenia. *Trends Neurosci.* 27, 683–690.

Wood, R. M., Rilling, J. K., Sanfey, A. G., Bhagwagar, Z., and Rogers, R. D. (2006). Effects of tryptophan depletion on the performance of an iterated Prisoner's Dilemma game in healthy adults. *Neuropsychopharmacology* 31, 1075–1084.

Yu, A. J., and Dayan, P. (2005). Uncertainty, neuromodulation, and attention. *Neuron* 46, 681–692.

Zaldivar, A., Asher, D. E., and Krichmar, J. L. (2010). "Simulation of how neuromodulation influences cooperative behavior," in *Simulation of Adaptive Behavior: From Animals to Animats*, eds S. Doncieux, J.-A. Meyer, A. Guillot, and J. Hallam [Berlin; Heidelberg: Springer-Verlag Lecture Notes on Artificial Intelligence (LNAI 6226)], 649–660.

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

Received: 25 November 2012; accepted: 11 January 2013; published online: 05 February 2013.

Citation: Krichmar JL (2013) A neurorobotic platform to test the influence of neuromodulatory signaling on anxious and curious behavior. Front. Neurorobot. 7:1. doi: 10.3389/fnbot.2013.00001

frontiers in
NEUROROBOTICS

ORIGINAL RESEARCH ARTICLE
published: 08 April 2013
doi: 10.3389/fnbot.2013.00005

Humanoids learning to walk: a natural CPG-actor-critic architecture

Cai Li*, Robert Lowe and Tom Ziemke

Interaction Lab, University of Skövde, Skövde, Sweden

Edited by:
Jeffrey L. Krichmar, University of California Irvine, USA

Reviewed by:
Mehdi Khamassi, CNRS, France
Poramate Manoonpong, Georg-August-Universität Göttingen, Germany
Calogero M. Oddo, Scuola Superiore Sant'Anna, Italy

****Correspondence:***
Cai Li, Interaction Lab, University of Skövde, 54128 Skövde, Sweden.
e-mail: gauss.lee@his.se

The identification of learning mechanisms for locomotion has been the subject of much research for some time but many challenges remain. Dynamic systems theory (DST) offers a novel approach to humanoid learning through environmental interaction. Reinforcement learning (RL) has offered a promising method to adaptively link the dynamic system to the environment it interacts with via a reward-based value system. In this paper, we propose a model that integrates the above perspectives and applies it to the case of a humanoid (NAO) robot learning to walk the ability of which emerges from its value-based interaction with the environment. In the model, a simplified central pattern generator (CPG) architecture inspired by neuroscientific research and DST is integrated with an actor-critic approach to RL (cpg-actor-critic). In the cpg-actor-critic architecture, least-square-temporal-difference based learning converges to the optimal solution quickly by using natural gradient learning and balancing exploration and exploitation. Futhermore, rather than using a traditional (designer-specified) reward it uses a dynamic value function as a stability indicator that adapts to the environment. The results obtained are analyzed using a novel DST-based embodied cognition approach. Learning to walk, from this perspective, is a process of integrating levels of sensorimotor activity and value.

Keywords: reinforcement learning, humanoid walking, central pattern generators, actor-critic, dynamical systems theory, embodied cognition, value system

1. INTRODUCTION

In recent years, with increasingly reforming ideas about how locomotion should be understood in a way that it is a result of the interaction of dynamical systems, bio-inspired approaches are attracting a lot of attention. Scientists claim that locomotion including its development or adaptivity emerges when the neural structure or the body with proper morphology interacts with the environment under the laws of physics (Pfeifer and Bongard, 2006; Ijspeert, 2008). Hence, the focus of investigating locomotive capabilities of artificial or biological agents should be shifted from how each body part moves in a kinematic chain to a generic view pertaining to how controllers (or neural systems), body, and environment interact as a *complete dynamic system.*

Recently, cutting-edge work in robotics shows the importance of the abovementioned ideas. According to Ijspeert, Central Pattern Generators (CPGs), the bio-inspired neural structures discovered in the middle of the last century (Hooper, 2001), work as a link connecting the sensori-motor level to the Mesencephalic Locomotor Region (MLR) in the brainstem which controls vertebrate locomotion. Thus, many robots under control of CPGs show their own adaptive behaviors when interacting with the environment (Fumiya et al., 2002; Pfeifer and Bongard, 2006; Degallier et al., 2011). A CPG network is a neural controller which can show adaptive network behaviors given sensory feedback. On the other hand, body flexibility, namely the so-called soft robotics, has been highlighted recently as a critical element for adaptive motor capabilities (Pfeifer and Bongard, 2006). However, there is no systematic way of evaluating flexibilities of different morphologies for locomotion.

On this basis, learning locomotion becomes more open and challenging in terms of integrating interactive information amongst the three parts: controllers, body, and context. Based on the *dynamic systems approach* proposed by Thelen in the 1990s from the perspective of development of cognition and action, locomotion is a consequence of self-organization and there is no "essence" for locomotive systems. Learning to walk is a formation process of a gait attractor dependent on the exploration of the state space in a dynamical system that consists of sensori-motor coupling of agent and environment. The attractor is a behavioral mode and *state space* is an abstract construct of space whose coordinates define the degrees of freedom of the system's behavior (Thelen and Smith, 1996). However, the learning mechanism which causes the formation of an attractor out of the state space in artificial systems still remains unclear in spite of Thelen's embodied theoretical stance. Adolph et al. (2012) posits that infants learn to walk through thousands of time-distributed, variable attempts including missteps and falls. She emphasizes the importance of the temporal-difference in the learning process. From the cognitive perspective, Schore (2012) indicates affective modulation is important for infants learning to walk. Particularly, the main caregiver plays a role as an "emotion system" outside assisting infants to evaluate their behaviors and scaffolding their affective systems. Pfeifer and Bongard (2006) explains locomotion learning from a robotics angle suggesting there is a "value" system in our body to evaluate the comfort of locomotion behaviors. Therefore, we assume there is an agent-centered mechanism related to learning how to walk and it has to comprise these properties: (1). It is an interactive-affective system. (2) It is capable of finding an optimized solution by exploring the state space through interaction with the environment in a time-sensitive manner. (3) The learning process is under control of the supervisor's "scaffolding." *We suggest, closely pertinent to the above three points, that reinforcement learning is an appropriate choice for the implementation of learning to walk.*

Reinforcement learning (RL) has, in recent years, evolved considerably especially in dealing with problems of continuous and high-dimensional state space (Doya, 2000b; Wiering and van Otterlo, 2012). Biologically, it sketches an interactive process of dopamine systems and the basal ganglia which is emotion-related (Schultz, 1998; Doya, 2000a; Graybiel Ann, 2005; Khamassi et al., 2005; Frank and Claus, 2006; Joel et al., 2012). Grillner et al. (2005) elucidate the functions of dopamine systems (striatum) and the basal ganglia (pallidum) with biological grounds on motor adaptation and selection. Moreover, RL proffers a computational formulation of learning, via the interaction of body, neural systems, and environment, to execute behaviors that deliver satisfying consequences. Grillner et al. (2007) also propose a layered architecture including basal ganglia, CPG network, and sensory feedback which may imply the interactive bond between CPGs and RL. In this article, by using RL, a meaning of "scaffolding" is given by manipulating the value function and update rules. Meanwhile, for the purpose of endowing a humanoid with a capability of learning to walk efficiently, the RL algorithm has to guarantee fast convergence.

Based on the above ideas and theories we propose a new architecture combining Natural Actor-Critic (NAC) and a CPG network to achieve a "learning to walk" task on a humanoid. This is the so-called Natural CPG-Actor-Critic. The natural actor-critic has been proposed by Kakade (2002) and further improved and used by Peters in the field of supervised motor learning (Peters and Schaal, 2006, 2008). This particular RL algorithm uses natural policy gradient methods which may achieve very efficient exploration and fast convergence of learning. Based on their ideas, Nakamura et al. (2007) proposed a natural CPG-Actor-Critic approach and implemented it with a $2D^1$-simulated stick walker in MATLAB. At the present time, the natural CPG-Actor-Critic has not been implemented on a humanoid platform. The reasons are clear: firstly, there exists no functional 3D CPG walking model that does not depend on inverse kinematics even though the motion of roll direction is of importance to walking (Collins et al., 2001). Nakamura's work fully adopted Taga's model (Taga, 1998) which similarly works on a 2D-simulated stick walker. Secondly, Taga's model is very complicated involving a very high-dimensional and difficult-to-reduce state space. This is why state value estimates take a long time to converge. Finally, the stick walker contacts the ground in an entirely different way to humanoids with foot interaction so that the body dynamics also differ. This is a morphology-related reason. Thus, in this article, we try to use another sensor-driven CPG architecture to avoid the problems faced by Nakamura and colleagues (For the comparison to Nakamura's model, please refer to Discussion A.1).

The main contribution of this article is to present a complete natural CPG-Actor-Critic architecture and implement it on a 3D-simulated humanoid by utilizing a state-of-the-art natural policy gradient in a relatively high-dimensional state space. In this work, it is shown not only how episodic NAC (eNAC) converges to optimal solutions by exploration-exploitation batch learning but also how eNAC helps a humanoid under control of CPGs learn to walk by searching appropriate posture and integrating sensory feedback. Meanwhile, by adopting a dynamic system perspective with respect to cognitive development, RL can be understood in a new light of state value estimates. Experiments introduced in this article consist of two parts. The first part will focus on the emergence of proper walking posture and integration of sensory feedback. The second part shows how the robot learns to walk on a slope and the relation between slope and posture change. The aim of this work is to glean how CPGs in a natural actor-critic architecture adapt to the environmental change in walking by balancing realization of body morphology and acquisition of sensory feedback.

2. MATERIALS AND METHODS

In order to fully comprehend how CPG networks work with the NAC architecture, a description of relevant theories applicable to the proposed architecture is offered in this section. With the cpg-actor-critic model, it is able to clearly show how the humanoid's body, the physical world, and neural controllers interactively cause the emergence of an appropriate walking gait. In order to learn walking, a proper upright standing posture is necessary. Scientific research shows that human infants learn to walk after they have learned to be able to maintain an upright posture (Kail and Cavanaugh, 1996; Adolph et al., 2012). After learning a standing posture, they can start to explore the world in an allocentric way. Through exploration, infants improve their walking behaviors (Clearfield, 2011). However, the exploration in a physical world consists of infinite possibilities increasing the difficulties in modeling this process. Thus, a limited but continuous state space has to be constructed for the purpose of learning to walk by exploring only in the state space of neural structure which is related to posture control and sensory feedback. Then walking can be considered as a Partially Observable Markov Decision Process (POMDP). In this article, we use a NAC architecture which appears as one good solution to bridge continuous state space and action space in a fast-learning way. We show that it can not only show the emergence of proper walking posture but also adaptation to environmental changes.

2.1. CENTRAL PATTERN GENERATORS

Modeling walking on a humanoid robot is a complicated task related to designing an autonomous control mechanism for a high degree-of-freedom (DOF) body. So the main challenge for developing modern control strategies concerns avoiding the problem of the "curse of dimensionality" which closely pertains to a large number of DOFs. Using CPGs, it is possible to transfer and restrict extremely high-DOF walking in Cartesian space to a low-dimensional sensory space of neural structure with neurophysiological theories and assumptions (Geng et al., 2006; Takamitsu et al., 2007; Endo et al., 2008).

[1] The 2D or 3D means a coordination system fixed on the torso of a robot. It has three axes: X (Pitch: pointing to front), Y(Roll: pointing to right), Z(Vertical: pointing upwards).

CPGs, as a group of presumed neurons existing in vertebrates' spinal cord (Latash, 2008), are the neural circuits generating rhythmic movement. With sensory feedback, the body or the robot under control of CPGs interacts with the environment in an adaptive way in which case the body dynamics are interactively entrained into a limit cycle. This limit cycle implies the following: firstly, structural-stability is imperative to a CPG architecture. This means CPG architectures should be able to shift to another limit cycle by adapting to contextual change and then recovering the original limit cycle without external disturbance (Righetti, 2008; Li et al., 2011). Secondly, the adaptive change of the limit cycle that CPGs converge to is generally done by updating the output or connection weights of CPGs. A lot of work has been done to emphasize the importance of these two points (Inada and Ishii, 2004; Ijspeert, 2008; Li et al., 2011, 2012).

Compared to a lot of work done with engineering models based on Zero Momentum Point (ZMP) (Lim et al., 2002; Strom et al., 2009) to model walking, CPGs also have many advantages (Nakamura et al., 2007). In terms of adaptive capabilities, as engineering models (including an accurate model of the controlled system and the environment) need to calculate the trajectories of motion with respect to very specific models, these models need to be recalculated or even remodeled when the context or the body changes. But, as for CPGs, it is just a matter of updating parameters to new adaptation capabilities. On the other hand, CPGs are proven to be more energy-efficient (Li et al., 2011) than those methods which need huge computer power to calculate complicated accurate models in each computation period.

From the perspective of the dynamic systems approach, just because of the excellent adaptivity of a CPG or its network, CPGs can be considered as an interface between the environment and high-level cognitive functionalities. As abovementioned, the shift and change of limit cycles could be viewed as results of CPGs interfacing to the high-level control system, like the RL system in this work.

2.1.1. Layered CPG structure

CPG structures have been explored by researchers for some time (Orlovskii et al., 1999; Amrollah and Henaff, 2010) but the integration of sensory feedback remains an unresolved open question to the research of CPGs without a conclusive structure. Recently, a proper layered CPG architecture has been proposed in Rybak et al. (2006) based on biological evidence (Amrollah and Henaff, 2010; **Figure 1**).

The layered CPG concept illustrates clearly not only the functions for each layer but also principles for the influence of afferent feedback in each layer. For instance, the rhythm generator (RG) layer is in charge of rhythm or frequency resetting depending on feedback. The PF layer functions like a network to keep synchronization of motorneuron activities as well as phase transition without altering the RG layer according to afferent feedback. The motorneuron level is an integrator where downward outputs and sensory feedback are fused together (details in **Figure 1**).

Based on this CPG structure, we propose a layered CPG architecture in our work which fulfills functions of each layer (**Figure 2**). In the structure, the four-cell recurrent network based on symmetric group theory (Golubitsky and Stewart, 2004) has the capability to be structurally stable (Righetti, 2008). It is of importance that this network can model the dynamics of different locomotion gaits (including walking, trotting, running, and crawling) by altering its connection weights and properties of each cell (Righetti, 2008). Crawling and walking on different humanoids have been implemented (Righetti and Ijspeert, 2006; Lee et al., 2011; Li et al., 2011). With this network, it keeps the synchronization of each oscillator cell within a specific phase difference by using typical negative neural connection (ipsilateral) and positive connection (contralateral) to keep ipsilateral oscillation out of phase and contralateral oscillation in phase. Each cell of the four-cell network is modeled with a Hopf oscillator (Equation 1–3) which is different from the one used in Nakamura's model (details in Discussion A.1).

$$\dot{z}_i = a\left(m - z_i^2 + s_i^2\right) z_i - \omega_i s_i \tag{1}$$

$$\dot{s}_i = a\left(m - z_i^2 + s_i^2\right) s_i + \omega_i z_i + \sum_j a_{ij} s_j \tag{2}$$

$$w_i = 2 \times \pi \left(\frac{\omega_{\mathrm{up}}}{1 + e^{-100 s_i}} + \frac{\omega_{\mathrm{down}}}{1 + e^{100 s_i}}\right) \tag{3}$$

where the z_i is the output of the Hopf Oscillator and s_i is the internal state. m is the amplitude and a is the convergence rate. ω_i is the internal weight in this coupled oscillator. It is usually set to 1. s_j is the output of the other cells except cell i and α_{ij} is the external weight (from cell j) of the four-cell network. Meanwhile, ω_i also represents the frequency of this oscillator. Interestingly, by changing values of ω_{up} and ω_{down}, you can change the duration of increase and decrease rate of the oscillator. For example, in our work $\omega_{up} = 5\omega_{down}$, the oscillation increases 5 times faster than decreases. This relation is derived from the experimental data by Hallemans et al. (2006) about joint kinematic trajectories of walking children. m and a are set to be 1 and 5 in our experiment.

If we assume the motorneurons work to integrate the internal oscillation and external sensory feedback, the whole physical system including the neural controller can be expressed like this:

$$\dot{x} = F(x, \tau) \tag{4}$$

where **x** denotes the state of the physical system, whose components are, for example, sensory angles of joints, and the dot $(\dot{})$ denotes the time derivative. τ denotes the control signal (torque or trajectory) from the controller, and $F(\mathbf{x},\tau)$ represents the vector field of the system dynamics. Then the motorneuron can be modeled by the firing neural structure (Buono and Palacios, 2004; Endo et al., 2008; Li et al., 2012), the dynamics of which can be given by:

$$\begin{aligned} \varsigma \dot{y}_{Ei} &= -y_{Ei} + \mathbf{I}_{Ei} \\ \tau_{Ei} &= G_E\left(y_{Ei}\right) \end{aligned} \tag{5}$$

$$\begin{aligned} \varsigma \dot{y}_{Fi} &= -y_{Fi} + \mathbf{I}_{Fi} \\ \tau_{Fi} &= G_F\left(y_{Fi}\right) \end{aligned} \tag{6}$$

where y_{Ei} and y_{Fi}, $\mathbf{I_{Ei}}$ and $\mathbf{I_{Fi}}$, ζ, τ_{Ei} and τ_{Fi} represent the state, input, damping constants (equal to 10 in our work), and the output of ith extensor and flexor motorneuron, respectively (if no exception,

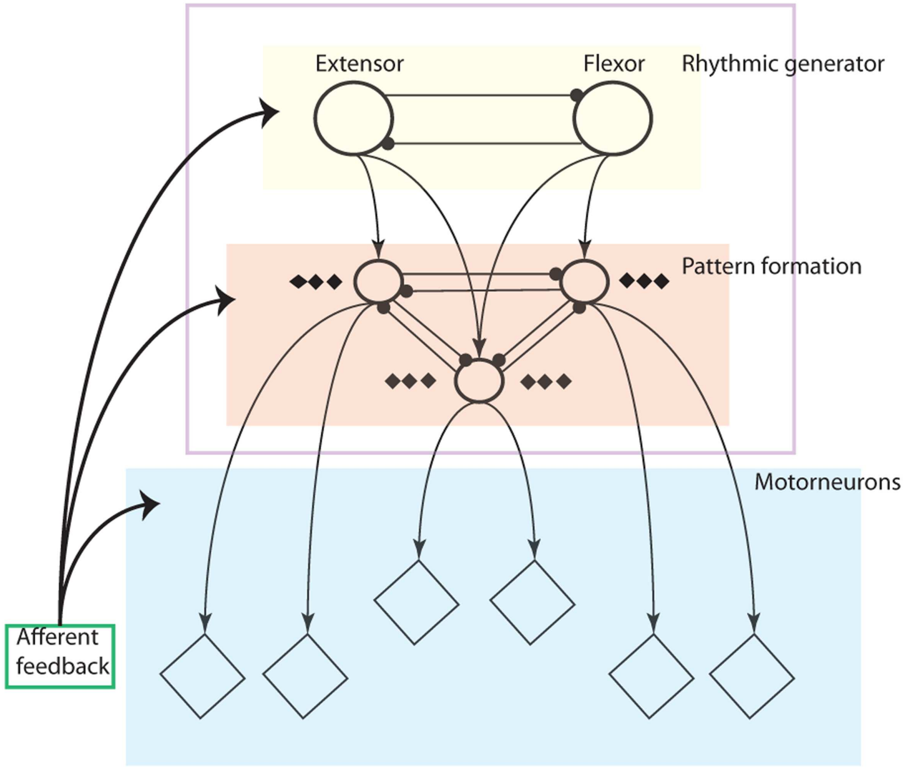


FIGURE 1 | Schematic illustration of the three-level central pattern generator (CPG) concept: The locomotor CPG consists of a half-center rhythm generator (RG), a pattern formation (PF) network and a motorneuron layer. Rhythmic generator layer (yellow area): this layer contains oscillators which generate rhythmic signals as the input to the PF layer. PF layer (red area: only three neurons are drawn with others neglected): The PF network contains interneuron populations, each of which provides excitation to multiple synergistic motorneuron pools (diamonds) and is connected with other PF populations via a network of inhibitory connections. It mediates rhythmic input from the RG to motorneurons and distributes it among the motorneuron pools. The network also synchronizes the oscillatory output of each interneuron. The motorneuron layer: It integrates the muscle sensory feedback and activation of PF network outputs. The extensor and flexor motorneurons together determine the output to the muscles (Rybak et al., 2006).

all the E and F in the lowerscripts represent extensor and flexor in this article). G_E and G_F are both activation functions, for example the sigmoid function. The input $\mathbf{I}_{Ei}$ and $\mathbf{I}_{Fi}$ are given by:

$$\mathbf{I}_{Ei} = \sum_j \mathbf{V}_{Eij}\mathbf{z}_j + \sum_k \mathbf{W}_{Eik}\mathbf{X}_{Ek} \tag{7}$$

$$\mathbf{I}_{Fi} = \sum_j \mathbf{V}_{Fij}\mathbf{z}_j + \sum_k \mathbf{W}_{Fik}\mathbf{X}_{Fk} \tag{8}$$

where $\mathbf{z}_j$ is the jth output of PF layer (the four-cell network). $\mathbf{V}_{Eij}$ and $\mathbf{V}_{Fij}$ are the connection weights from PF layer to motorneuron layer. $\mathbf{X}_{Ek}$ and $\mathbf{X}_{Fk}$ are the kth sensory feedback from sensory neurons in vector $\mathbf{X}_E$ and $\mathbf{X}_F$ weighted by the connection weight $\mathbf{W}_{Eik}$ and $\mathbf{W}_{Fik}$. Then the final output of the controller is given by:

$$\tau_i = T_{Ei}\tau_{Ei} + T_{Fi}\tau_{Fi} + \mathbf{W}_{pi}\mathbf{X}_{pi} \tag{9}$$

where τ_i is the ith output of CPGs and T_{Ei}, T_{Fi} are the connection weight. $\mathbf{X}_{pi}$ is the ith term in posture control vector $\mathbf{X}_p$ weighted by connection weight $\mathbf{W}_{pi}$.

2.1.2. *Sensor neurons*

The sensor neuron mechanism representing local reflex of muscles is very important for motorneurons (Latash, 2008). It has been proved to be biologically existent (Endo et al., 2008) and useful for robotic walking applications (Endo et al., 2008; Nassour et al., 2011). The general sensor neuron model is given by a sigmoid function:

$$\rho_{sn} = \frac{1}{1 = e^{a(\theta_{threshold} - \theta_{input})}} \tag{10}$$

where ρ_{sn} is the output of a sensor neuron. a is the sensitivity of a sensor neuron. $\theta_{threshold}$ and θ_{input} are the threshold and the input of a sensor neuron. The input can be raw or postprocessed sensor data and the threshold can be zero or a certain value depending on types of sensor neurons. The idea of using sensor neurons is to normalize the input of all the sensors and use them with different purposes (details see Appendix A).

According to existing robotic applications of CPGs, each CPG is used to control one joint of a robot. Each sensory connection weight (like $\mathbf{W}_{Eik}$ and $\mathbf{W}_{Fik}$) of each CPG is determined by the corresponding joint it controls and its specific sensory input. In the layered structure implemented on the physical robot NAO (Li et al., 2012), the 4-cell network is applied to a layered CPG architecture with manually tuned weights and it represents cognitive-related prior knowledge about the fundamental properties of walking. For example, as one property this network owns, the anti-phase contralateral leg movement is useful for walking. There is evidence suggesting that this typical movement is formed over many months of early infancy before infants learn to walk (Kail and Cavanaugh, 1996; Thelen and Smith, 1996). The main focus for learning to walk is shifted from learning very basic walking prerequisites to learning how each joint is coordinated with the whole-body and adaptively reacts to environmental change. Then RL proffers a very nice blueprint.

2.2. NAC MODEL

Actor-critic is a very typical but popular RL method broadly used in recent years (Kimura and Kobayashi, 1998; Sato and Ishii, 1998; Orlovskii et al., 1999; Sutton et al., 2000). In a typical implementation, an actor is a controller which emits actions or action-related control signals to a physical system. According to a certain policy, it observes the states of a physical system and determines the control signals on the basis of the states. A critic is a functional part which evaluates the states of a physical system and updates the controller and control policies. As a typical RL learning mechanism, it can be adapted by using some other updating rules. For example, the convergence of an actor-critic model based normal policy gradient approach is achieved in (Konda and Tsitsiklis, 2003) and a mathematical convergence of actor-critic is proved in (Dotan et al., 2008). The convergence of the actor-critic model with the natural policy gradient has been proved by Peters and Schaal (2008). Moreover, it has been proved to be faster than the normal "vanilla" policy gradient (Peters, 2007).

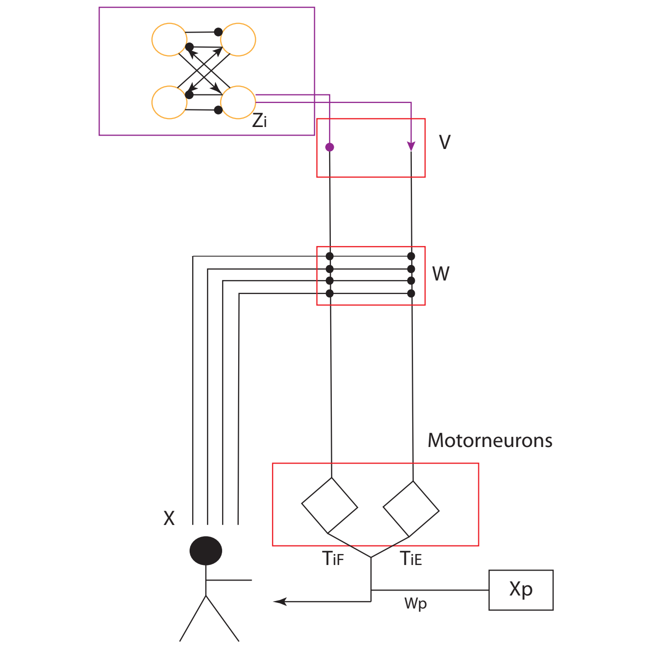


FIGURE 2 | CPG controller (Top: the four-cell network) and its layered structure. Yellow circles represent a coupled RG group corresponding to yellow area in **Figure 1**. The round-headed and sharp-headed arrows represent negative (−1) and positive (+1) connection weights (for details, please refer to text.) The four-cell network (purple-framed area) fulfills the function of the PF layer. The two diamonds represent the motorneuron layer which integrates sensory feedback and upper-layer outputs. V, W, W_p are weight vectors which integrate PF-layer outputs, sensory feedback and posture control terms respectively. T_{Ei} and T_{Fi} are the strength weights of extensor and flexor.

2.2.1. *Natural CPG-actor-critic model*

Natural CPG-Actor-Critic is an autonomous RL learning framework used for CPG network based on Actor-Critic learning with the natural policy gradient. It was proposed by Nakamura in 2007 and successfully implemented on Taga's stick walker in Matlab simulation (Taga, 1998; Nakamura et al., 2007). We adopted his approach but with an entirely different CPG architecture, learning schema, and basic RL algorithm (for details, refer to discussion). Since the output of our CPG model is based on the input of PF layer and the states of sensory feedback and posture control terms, a CPG is an adaptive controller whose output is dependent on all these inputs. As a matter of fact, the layered architecture proposed in our work can be viewed as a feed-forward neural network (**Figure 3**) where the posture control works as a bias. As a normal gradient approach used for the feed-forward neural network, the backpropagation approach is not suitable for our work. Firstly, the backpropagation normal gradient is too slow and cannot avoid the "plateau" problem (Peters and Schaal, 2008). Secondly, it needs a lot of computation and large storage for precedent states. Therefore, the natural gradient approach is adopted as it has been proved to be more efficient than the backpropagation for feed-forward neural networks by Amari (1998) who proposed natural gradient.

Compared to Nakamura's model, our model is naturally separated into two parts: the basic CPG and the actor part (details in **Figure 3** and Discussion A.1). This is similar to Nakamura's separation of his CPG model. The basic CPG part composed of an oscillatory network is to keep the phase relation and oscillation of the whole CPG as a core. The actor outputs the control signals based on its input. It covers two important functions of

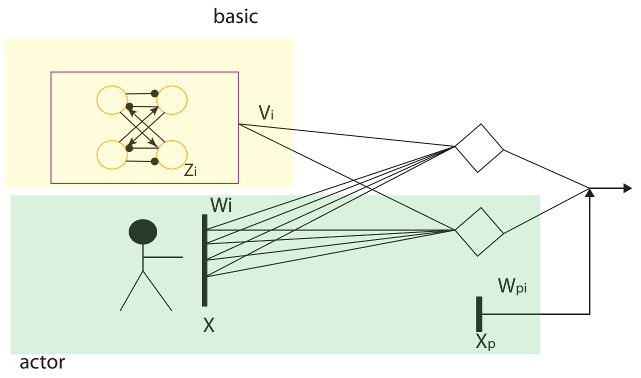


FIGURE 3 | The feed-forward two-layer neural network as the core of the CPG network. The yellow area is the basic CPG part with fixed connection weights and the green area functions for the output integration of sensor neurons and posture control.

a CPG: sensory feedback fusion and posture control (Orlovskii et al., 1999). The RL updating rule can be applied to this part to change the weights, leading to involvement of the adaptive change of the CPG controller based on interaction when a robot walks. RL state space is given as $\mathbf{X}$, a vector including all the sensory feedback and posture control terms. The action space is given by $\mathbf{U}$ which comprises all the control signals. The input and output of the CPG can be adapted to:

$$\mathbf{X} \sim \left\{\mathbf{X}_E, \mathbf{X}_F, \mathbf{X}_p\right\}, \mathbf{U} \sim \left\{\mathbf{U}_E, \mathbf{U}_F, \mathbf{U}_p\right\}$$

$$\mathbf{I}_{Ei} = \mathbf{I}_{Ei}^{basic} + \mathbf{I}_{Ei}^{actor} \tag{11}$$

$$\mathbf{I}_{Fi} = \mathbf{I}_{Fi}^{basic} + \mathbf{I}_{Fi}^{actor} \tag{12}$$

$$\mathbf{U}_{Ei} = \mathbf{I}_{Ei}^{\text{actor}} = \sum_k \mathbf{W}_{Eik}\mathbf{X}_{Ek} \tag{13}$$

$$\mathbf{U}_{Ei} = \mathbf{I}_{Fi}^{actor} = \sum_k \mathbf{W}_{Fi}\mathbf{X}_{Fi} \tag{14}$$

$$U_{pi} = \mathrm{W}_{pi}\mathrm{X}_{pi} \tag{15}$$

$$\mathbf{W} \sim \{\mathbf{W}_E,\ \mathbf{W}_F, \mathbf{W}_P\}$$

where $\mathbf{I}_{Ei}^{basic}$ and $\mathbf{I}_{Fi}^{basic}$ are the ith pair of the output of fixed basic CPG. $\mathbf{U}_E$ and $\mathbf{U}_F$ are vectors containing control signals emitted by the actor to the controller. $\mathbf{U}_{pi}$ is the ith element of a vector $\mathbf{U}_p$ including posture control terms. $\mathbf{U}_{Ei}$ and $\mathbf{U}_{Fi}$ are the ith terms in $\mathbf{U}_E$ and $\mathbf{U}_F$. $\mathbf{W}$ is a vector for all the connection weights. $\mathbf{W}_E$, $\mathbf{W}_F$, and $\mathbf{W}_p$ are vectors of connection weights for sensory feedback and posture control terms. Then the RL problem could be expressed as:

$$\mathbf{U} \sim \pi\left(\mathbf{U},\ \mathbf{X}\right) \tag{16}$$

where π is the stationary policy of the RL algorithm. Clearly, all the states $\mathbf{X}$ include two parts. $\mathbf{X}_E$ and $\mathbf{X}_F$ are called observable states. $\mathbf{X}_p$ is called unobservable states. They are assistive states which are provided to help the robot learn a proper posture. As our idea is to learn through interaction and to sense the body through peripheral systems, there is no full observability for the whole-body states. This condition is different from Nakamura et al. (2007) application. Hence, the whole control system is regarded as a POMDP. It is indicated that the actor determines the control signals sent to CPGs according to a static policy and CPGs act with the physical system. Then the critic evaluates the locomotion under control of CPGs changed by the actor and update the policy in the actor. This is the so-called CPG-Actor-Critic. Used with the natural policy gradient, it is called natural CPG-Actor-Critic. As a proper architecture for RL learning, we need to avoid a problem of RL "the curse of dimensionality." In order to reduce the dimensionality of the CPG controller, internal weights of the 4-cell network and $\mathbf{V}_{Eij}, \mathbf{V}_{Fij}\,(1,-1)$ are all fixed as primitive inputs of CPGs. This is different from Nakamura et al. (2007) idea of using an internal connection from the basic CPG (). The reason for not having internal connection weights is our flexible 4-cell network has already been endowed with prior knowledge or capabilities to keep synchronization and to reshape the output of oscillators. However, this prior knowledge must be learned in Nakamura's work. Meanwhile, using a sensory-driven CPG means there cannot be so much sensory feedback as the number of sensors on a given humanoid is always limited. Nakamura has full observability in state space of the accurate Taga walker but he only uses a subset of the available sensors. Since the aim of our work is to implement this architecture on a real humanoid to understand mechanisms of posture control and sensory feedback integration, a trial-and-error learning mechanism based on batch RL is needed (details in Discussion A.1).

2.2.2. Learning algorithm

The policy gradient (PG) approach is very useful for parameterized motor modeling. Peters summarizes and compares different PG approaches, including finite difference, likelihood ratio methods, and REINFORCE (Peters, 2007). It is concluded that the aim of the gradient approach is to find the correct updating direction of policy parameters in order to maximize expected reward. Assuming the stationary policy is $\pi^\theta(\mathbf{x}, \mathbf{u})$ which can determine action space $\mathbf{u}$ based on state space $\mathbf{x}$ with a static distribution $d^\pi(x)$, the immediate reward is $r(x, u)$, and then the expected reward $J(\theta)$ can be written as:

$$J\left(\theta\right) = \int_{\mathbf{x}} d^\pi\left(\mathbf{x}\right) \int_{\mathbf{u}} \pi^\theta\left(\mathbf{u}|\mathrm{x}\right) r\left(\mathbf{x}, \mathbf{u}\right) d\mathbf{x}d\mathbf{u} \tag{17}$$

where the policy $\pi^\theta(\mathbf{x}, \mathbf{u})$ is derivable at the policy parameters θ, namely $\triangledown_\theta \pi^\theta$ exists. For maximizing expected reward $J(\theta)$ with respect to θ, policy gradient will find the steepest increase direction $\triangledown_\theta J = J(\theta + \triangledown\theta) - J(\theta)$ to update the search policy $\pi^\theta(\mathbf{x}, \mathbf{u})$ until it converges. For this purpose, the update rule of the policy gradient can be expressed as:

$$\theta_{n+1} = \theta_n + \alpha\nabla_\theta J|_{\theta=\theta_n} \tag{18}$$

where n represents the nth step of update and α is the learning rate (equal to 0.01). If we directly take the 1st derivative of $J(\theta)$ with respect to θ, the gradient is given by:

$$\nabla_\theta J\left(\theta\right) = \int_{\mathbf{x}} d^\pi\left(\mathbf{x}\right) \int_{\mathbf{u}} \nabla_\theta \pi^\theta\left(\mathbf{u}|\mathbf{x}\right) r\left(\mathbf{x}, \mathbf{u}\right) d\mathbf{x}d\mathbf{u} \tag{19}$$

$$= \int_{\mathbf{x}} d^\pi\left(\mathbf{x}\right) \int_{\mathbf{u}} \pi^\theta\left(\mathbf{u}|\mathbf{x}\right) \nabla_\theta \log\left(\pi^\theta\left(\mathbf{u}|\mathbf{x}\right)\right) r\left(\mathbf{x}, \mathbf{u}\right) d\mathbf{x}d\mathbf{u} \tag{20}$$

where $\triangledown_\theta$ is the 1st derivative. This is the so-called normal gradient. If we use this gradient to update the policy, it is very slow to find the best policy for the maximization of expected reward. Therefore, the steepest gradient (natural policy gradient) is applied to our model. The adaptation of Equation 20 is at the core of the natural PG method. According to Peters' (2007) proof, the natural gradient is given by:

$$\theta_{\mathrm{n}+1} = \theta_{\mathrm{n}} + \alpha F_\theta^{-1}\nabla_\theta J|_{\theta=\theta_{\mathrm{n}}} \tag{21}$$

$$F_\theta = \int_T \pi^\theta \nabla_\theta \log \pi^\theta \nabla_\theta \log \pi^\theta d\theta \tag{22}$$

where F is the Fisher Matrix (FM). Multiplied by FM, the normal policy gradient is changed to the steepest one (here, all the

x,**u** are neglected for simplification reason). On the basis of policy gradient theorem (Peters, 2007), the PG could also be modified to:

$$\nabla_\theta J(\theta) = \int_{\mathbf{x}} d^\pi(\mathbf{x}) \int_{\mathbf{u}} \nabla_\theta \pi^\theta(\mathbf{u}|\mathbf{x}) \left(Q^\pi(\mathbf{x},\mathbf{u}) - b(\mathbf{x})\right) d\mathbf{x}d\mathbf{u} \tag{23}$$

where $Q(x,u)$ is the action-state function and $b(x)$ is a baseline which is a regularized term used to avoid large variance of gradient. With the theory of compatible function approximation, it is possible to apply basis functions $\nabla_\theta log^{\mathrm{T}}(\pi^\theta(\mathbf{u}|\mathbf{x}))$ to linearly approximate $Q^\pi(\mathbf{x}, \mathbf{u}) - b(\mathbf{x})$, then the above Equation 23 is adapted to:

$$\nabla_\theta J(\theta) = \int_{\mathbf{x}} d^\pi(\mathbf{x}) \int_{\mathbf{x}} \pi^\theta(\mathbf{u}|\mathbf{x}) \nabla_\theta \log\left(\pi^\theta(\mathbf{u}|\mathbf{x})\right) \times \nabla_\theta \log^T\left(\pi^\theta(\mathbf{u}|\mathbf{x})\right) wd\mathbf{x}d\mathbf{u} = F_\theta w \tag{24}$$

where **w** is a weight vector of the linear approximation. Then clearly, by replacing $\nabla_{\theta\mathrm{J}}(\theta)$ in (21) with (24), the natural PG becomes:

$$\theta_{\mathrm{n}+1} = \theta_{\mathrm{n}} + \alpha\mathbf{w} \tag{25}$$

The RL problem is transitioned from searching the steepest policy gradient to a normal regression problem about finding the best approximation of $Q^\pi(\mathbf{x}, \mathbf{u}) - b(\mathbf{x})$ with basis functions. Because $Q^\pi(\mathbf{x},\mathbf{u}) = b(\mathbf{x}) + \log\left(\pi^\theta(\mathbf{u}|\mathbf{x})\right)\mathbf{w}$ and $Q^\pi(\mathbf{x},\mathbf{u}) = r(\mathbf{x},\mathbf{u}) + \lambda \int_{\mathbf{x}'} p\left(x'|x,u\right) V\left(x'\right) dx'$ (where λ is the discounting factor, $\mathbf{x}'$ is the next state, $p(\mathbf{x}'|\mathbf{x},\mathbf{u})$ is the probability of state transition.), assume the value function is $V(\mathbf{x}) = b(\mathbf{x})$ and can be approximated by $\psi^T(\mathbf{x})\mathbf{v}$ (where **v** is the weight vector and ψ is the vector of basis function related to the value function; Baird, 1994). Therefore, the approximation can be re-written:

$$\log^T\left(\pi^\theta(\mathbf{u}_t|\mathbf{x}_t)\right)\mathbf{w} + \psi^T(\mathbf{x}_t)\mathbf{v} = \mathrm{r}(\mathbf{x}_t,\mathbf{u}_t) + \lambda\psi^T(\mathbf{x}_{t+1})\mathbf{v} + \in(\mathbf{x}_t,\mathbf{x}_{t+1},\mathbf{u}_t) \tag{26}$$

This is the equation for *LSTD*-$Q(\lambda)$ at time t. Then for the episodic learning, by summing up equation (26) with $t = 1,2\ldots H$, it is given by:

$$\frac{1}{H}\sum_{t=1}^{H} \log^T\left(\pi^\theta(\mathbf{u}_t|\mathbf{x}_t)\right)\mathbf{w} + \mathrm{J} = \frac{1}{H}\sum_{t=1}^{H} r(\mathbf{x}_t,\mathbf{u}_t) \tag{27}$$

where J is the value-function related term considered as a constant baseline. By means of the least square learning rule, the natural PG **w** can be obtained for each episode:

$$\begin{pmatrix} \mathrm{w} \\ J \end{pmatrix} = \left(\phi\phi^T\right)^{-1}\phi R.$$

$$\phi_t = \left[\frac{1}{H}\sum_{t=1}^{H} \log^T\left(\pi^\theta(\mathbf{u}_t|\mathbf{x}_t)\right)\mathbf{w},1\right] \tag{28}$$

$$R = \frac{1}{H}\sum_{t=1}^{H} r(\mathbf{x}_t,\mathbf{u}_t) \tag{29}$$

In our work, we use a monte-carlo like approach called episodic NAC (eNAC) (Peters, 2007) to make the robot repeat the walking episodes until it achieves final optimal performance. The eNAC is shown in Schema 1 with pseudocode.

Schema 1

```
*Repeat*: n = 1,2 …M trials
*input*: policy parameterization $\theta^n$
$\pi(\mathbf{U}|\mathbf{X})$ determines $\mathbf{U}_p$ before starting each trial
*Start the trial*: obtain $\mathbf{X}_{0:H}$, left $\mathbf{U}_{0:H}, r_{0:H}$ for each trial from $\pi(\mathbf{U}|\mathbf{X})$
Obtain the sufficient statistics
policy derivatives: $\phi_k = \nabla_\theta \log \pi_\theta(\mathbf{U}_t|\mathbf{X}_t)$
Fisher matrix $F_\theta = \left\langle \left(\sum_{k=0}^{H} \phi_k\right) \left(\sum_{l=0}^{H} \phi_l\right)^T \right\rangle$
Vanilla gradient $g = \left\langle \left(\sum_{k=0}^{H} \phi_k\right) \left(\sum_{l=0}^{H} \alpha_l r_l\right) \right\rangle$
Eligibility $\psi = \left\langle \left(\sum_{k=0}^{H} \phi_k\right) \right\rangle$
General reward $\bar{r} = \left\langle \left(\sum_{l=0}^{H} \alpha_l r_l\right) \right\rangle$, where $\alpha^l$ is the discount factor
Obtain natural gradient by computing
baseline $b = Q\left(\bar{r} - \psi^T F_\theta^{-1} g\right)$
with$Q = M^{-1}\left(1 + \psi^T\left(MF_\theta - \psi\psi^T\right)^{-1}\psi\right)$ *When updating rule is satisfied:*
$\theta_{n+1} = \theta_n + \alpha g$
until the convergence of algorithm
where $\langle\cdot\rangle$ means sum-up of all the previous values and current values.
```

2.3. EXPERIMENTAL SETTINGS

There are 2 main experiments presented in this article. The first one is to indicate that the proposed learning architecture can assist the robot learning to walk from the initial standing posture. The aim of this experiment is to prove the robot can adjust its posture and integrate sensory feedback simultaneously in the process of learning. The second experiment is to change the plane on which the robot stands to different angles to see how the learning architecture adaptively seeks out proper postures and walking gaits. By changing angles from −5° to +5°, this experiment also shows the relation between slope angles and posture change under the influence of gravity alternation.

2.3.1. Robotic platform and the neural controller

Figure 4 shows the robot and the neural network used to implement learning. We use the popular commercialized robot NAO. The advantages of using the NAO robot are summarized as: (1) There are locomotion-relevant sensors mounted on the NAO robot, such as gyro sensors which can detect acceleration of the body center in 3D space, joint sensors which can measure angle values, and foot pressure sensors which can sense ground contact of feet. All these sensors are useful for learning a proper walking gait. (2) Nao has a good firmware called Naoqi which is convenient for users to program and organize modules working together.

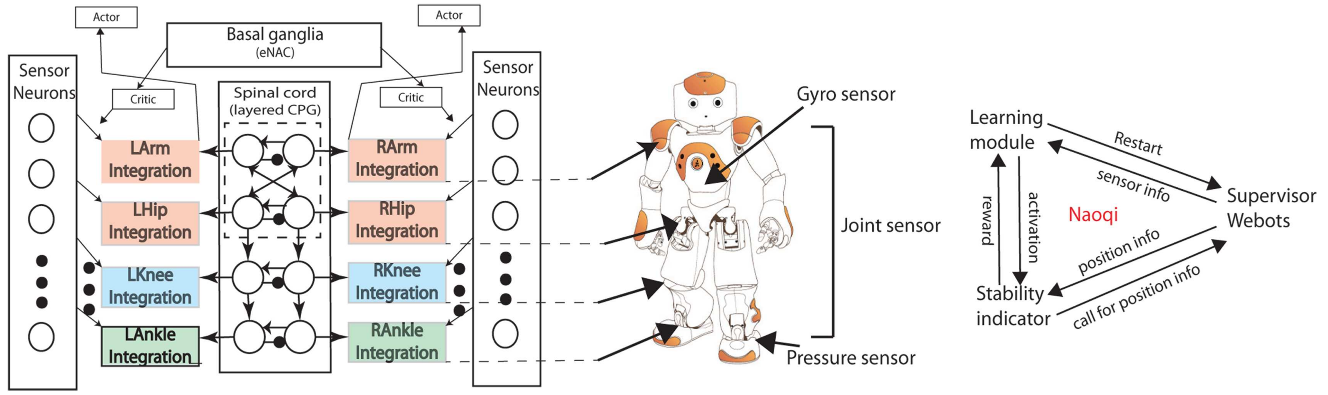


FIGURE 4 | Left: The complete architecture of cpg-actor-critic network. The integration with specific joint names represents the functions of motorneurons for each joint. The architecture abstractly represents the neurophysiological structure of the brainstem where the basal ganglia is in charge of RL, and the spinal cord is where the CPGs are located and motorneurons. Each layer of the three-layered architecture corresponds to different parts. The actor-critic learning mechanism works with basal ganglia as a RL functionality. Sensor neurons are different types of neurons which get information from different sensors of the robot (middle). Middle: The NAO robot. Arrows indicate the connection between the controller and the robot. Right: the software architecture. Naoqi is working as a middleware to handle the communication of three modules. The communicative information between every two modules are listed above the arrow.

The layered CPG network (**Figure 4** left) is used to control the NAO robot. Each output sends out position trajectories to each corresponding joint of NAO. Simultaneously, all the CPG neurons receive inputs from different kinds of sensor neurons based on the concept of sensor-driven CPG. There are three main sensor neurons with similar sigmoid form (refer to Appendix A): Proprioceptive (PP) sensor neurons for hips (joint sensors), anterior extremity (AE) sensor neurons for knees (joint sensors), and exteroceptive (ET) ankle sensor neurons (mixture of gyro sensors and pressure sensors). The motion of pitch direction is controlled by the CPG neural network while the roll motion (hips and ankles) is sensor-driven by the pitch motion (hips and ankles), respectively (Li et al., 2012; Appendix A).

2.3.2. *Software*

In this work, we use a simulated environment in the Webots simulator. Webots is an ODE (Open Dynamics Engine) based simulator in which users can not only simulate physics close to the real world but also move robots or objects and even change the environment. This is why there is a typical feature of Webots for simulating batch learning processes (Michel, 2004).

There are three main modules working together in the Naoqi of Webots. The supervisor module is in charge of restarting the simulation every episode by putting the robot in the initial position, changing the angle of the ground, measuring the distance the robot walks for each episode. The learning module is the main process where the CPG architecture and the learning algorithm are implemented. The stability indicator is a module working only for obtaining necessary sensory information from the supervisor module and the robot as well as calculating the immediate reward. It is an implementation of a basal ganglia like function. It sends a reward to the main process when activated by the learning module (**Figure 4**).

3. RESULTS

3.1. EXPERIMENT 1: WALKING ON THE FLAT GROUND

3.1.1. *Prerequisites*

In this experiment, the robot starts to walk from the same initial default standing posture and repeats the episode which lasts about 30 s until the algorithm converges. At the beginning of each episode, the policy gives two posture control signals for the knee and ankle parts as the posture change is very sensitive and should be explored as a basis for motion. Within each episode, the policy gives the other control signals related to sensory feedback every 1.5 ms. The policy used for balancing exploration and exploitation is given:

$$\begin{aligned}\pi_\theta(\mathbf{U},\mathbf{X}) &= N\left(\mathbf{U},\bar{\mathbf{U}},\sigma\right)\\ &= \frac{2\pi}{\sigma}\exp\left(\frac{\left(\mathbf{U}-\bar{U}\right)\left(\mathbf{U}-\bar{U}\right)^T}{\sigma^2}\right)\end{aligned}$$

where **U** is the output vector of the policy and $\bar{\mathbf{U}}$ is the input vector based on state space **X**. σ is the exploration rate which determines the variance of **U** from $\bar{\mathbf{U}}$. The value of σ cannot be so big (>0.1) that the system involves a lot of noise and it cannot be too small (<0.01) as the system will become very insensitive and diverges. In this experiment, for the posture control part $\mathbf{U}_p$, $\sigma = 0.05$. Otherwise $\sigma = 0.02$. As 0.02 is too small for the posture terms, a slightly bigger exploration rate is adopted. After having the continuous control signals sent to each joint, the robot needs to have the capability of evaluating different appearing walking gaits. The immediate fitness of a walking gait is acquired every 1.5 ms via the reward function which indicates the gait robustness, also called stability indicator. The stability of a walking gait should be considered in two directions: vertically, the SI is able to detect falling; horizontally, SI also considers the distance the robot moves.

In this way, SI reflects a trade-off between vertical and horizontal stability. Thus, the SI is given:

$$r = r_{height} + r_{acc} + r_{distance} \tag{30}$$

where $r_{height} = e^{25(H-H_{inil})}$, H is the height of gravity center and the NAO robot can detect the height based on the gyro sensor. H_{init} is the height of gravity center of the initial standing posture. Thus, this equation detects a dynamic change of height of the body when the robot is walking. When the robot falls, it is close to 0. $r_{acc} = 2\cos\left(\frac{accX}{10}\right) + 2\cos\left(\frac{accY}{22}\right)$, if $|accX| < 25$ and $|accY| < 50$. Otherwise, the robot is stopped and the episode is restarted. *accX* and *accY* are the acceleration of the robot's X axis (Pitch) and Y axis (Roll) of gravity center detected from the gyro sensor. For both directions, the gyro sensor is able to detect the acceleration from −70 to 70 which corresponds to −9.8 to +9.8 m/s^2. This part is implemented based on the inspiration of a vestibular system in the inner-ear mechanism for keeping body balance. It senses "falling" of the body by detecting the accelerations in 3D space (Thomas et al., 2009). Here, as we aim to study walking on the ground, the perpendicular acceleration is ignored. Twenty-five and 50 are the boundary values for the robot to fall. The even *cos* function is used to indicate this oscillatory motion of the walking in negative and positive directions of each axis. $r_{distance} = 2S$ and S is the walking distance detected by the supervisor module in Webots.

After each episode, two kinds of average reward are acquired. One is the average reward (AR) for each episode equal to $\sum_{l=0}^{H} a_l r_l$ and the other is the general average reward (GAR) equal to $\frac{\left\langle \sum_{l=0}^{H} a_l r_l \right\rangle}{M}$. If $AR > GAR$, the updating rule is satisfied. Otherwise, the episode is regarded as a failure. The algorithm converges when the learning process cannot find any episode which can satisfy the update rule.

3.1.2. Experiment 1 results

For each experiment, the algorithm starts with initialized $\theta = 0$ except that $\theta_5 = \theta_6 = 3$ as 3 is the weight value making ankle sensor neurons sensitive to external disturbance. 10 independent runs (different random seeds) were evaluated and 5 "good" results with top-five average reward are chosen for visualization in **Figure 5** (left column). We chose the one with highest average reward (run 5) to show how cpg-actor-critic finds the optimal learning gradient. Actually, the key feature of cpg-actor-critic is that it can find the best update directions of parameters quickly via balancing the exploration and exploitation. It is clearly observed that in the very first 10 episodes, the update directions of all the parameters are not stable, even opposite of right directions. However, after 10 update episodes, cpg-actor-critic can quickly find good and smooth update paths. Interestingly, **Figures 5B–E** shows the convergence of posture related parameters. In **Figure 5B**, $\theta_p 1$ and $\theta_p 2$ shows the posture change of the knee and the ankle. The knee posture is extending ($\theta_p 1$ turns negative) a lot to move the center of gravity toward the middle while the ankle position is only slightly changed to keep the balance with the knee posture. Meanwhile, θ_2 is increasing to 1 in order to limit the extension of the hip part and strengthen the flexion of the hip motion. The posture change of a chained-up three joints (ankle, knee, and hip) drives the robot to walk more robustly and for a longer distance. The final convergence of proper posture for walking is a consequence of the interaction of the morphology of NAO, the neural controller and the sensory feedback. For example, it is logical that NAO's ankle cannot be changed a lot as it is disproportionately big. The cpg-actor-critic realizes this obviously by the slight adjustment of the ankle posture with interaction.

As for the connection weights of AE and ankle sensor neurons, they only show the curves without flat convergence. The reason is that, in eNAC, the Q function is actually theoretically approximated by a linear combination of basis functions. However, practically it is only possible to averagely approximate without exact accurate convergence. This is also the reason we need to set up a specific convergence rule.

Finally, a specific walking gait is converged to by the interactive learning process and parameters are converged to $\theta = [0.4290, 1.0131, -11.7874, 21.6984, 3.2394, 3.8179, -0.6147, 0.1758, -12.8070]$.

3.2. EXPERIMENT 2: WALKING ON THE SLOPE

3.2.1. Prerequisites

The aim of experiment 2 is to test if the learning architecture can still function when there is different non-linear influence of the gravity for walking up and down the slope. Meanwhile, it is interesting to observe how the robot adaptively reacts to environmental change by achieving a trade-off between adaptation and learning. Finally, a conclusive relation between adaptive adjustment of CPG parameters and slope is explained.

In this experiment, we fully adopt the architecture in **Figure 4**. Since results in experiment 1 do not show any qualitative difference of walking gaits, each run in experiment 2 uses the parameter set developed in an arbitrarily selected good solution from experiment 1. The NAO, in each evaluation, is thus able to walk on a flat slope before attempting an upward or downward slope, depending on the condition. The good solution obtained for flat-ground walking consists of the following parameter set: $\theta = [1.3391, 0.4717, 3.1593, -0.6291, 3.4483, 3.1432, -0.6640, 0.2293, 0.4365]$ used as the set of values at the start of each experiment 2 run. In each experiment 2 run, the architecture is tested to learn to walk on the slopes from −0.08–0.08 rad (about −5–5°) by changing 0.01 rad each test. For each slope, there are 5 runs carried out for each condition where the aforementioned angles (8 in total) are gradually varied (get steeper) over the course of each simulation. Therefore, there is a total of $8 * 5$ upslope and $8 * 5$ downslope angles from which data points are derived (see **Figure 7**).

3.2.2. Experiment 2 results

Walking up and down the slope are two different cases with distinct gravitational effects. **Figure 6** shows how the walking posture and sensory feedback are autonomously changed by learning in those two situations (average data). From negative slope to positive slope, the change of gravity exerted on the robot is a non-linear alternation. So the posture change is required to cancel the influence of gravity in the moving direction (upslope and downslope: extra negative and positive force respectively). If we assume the

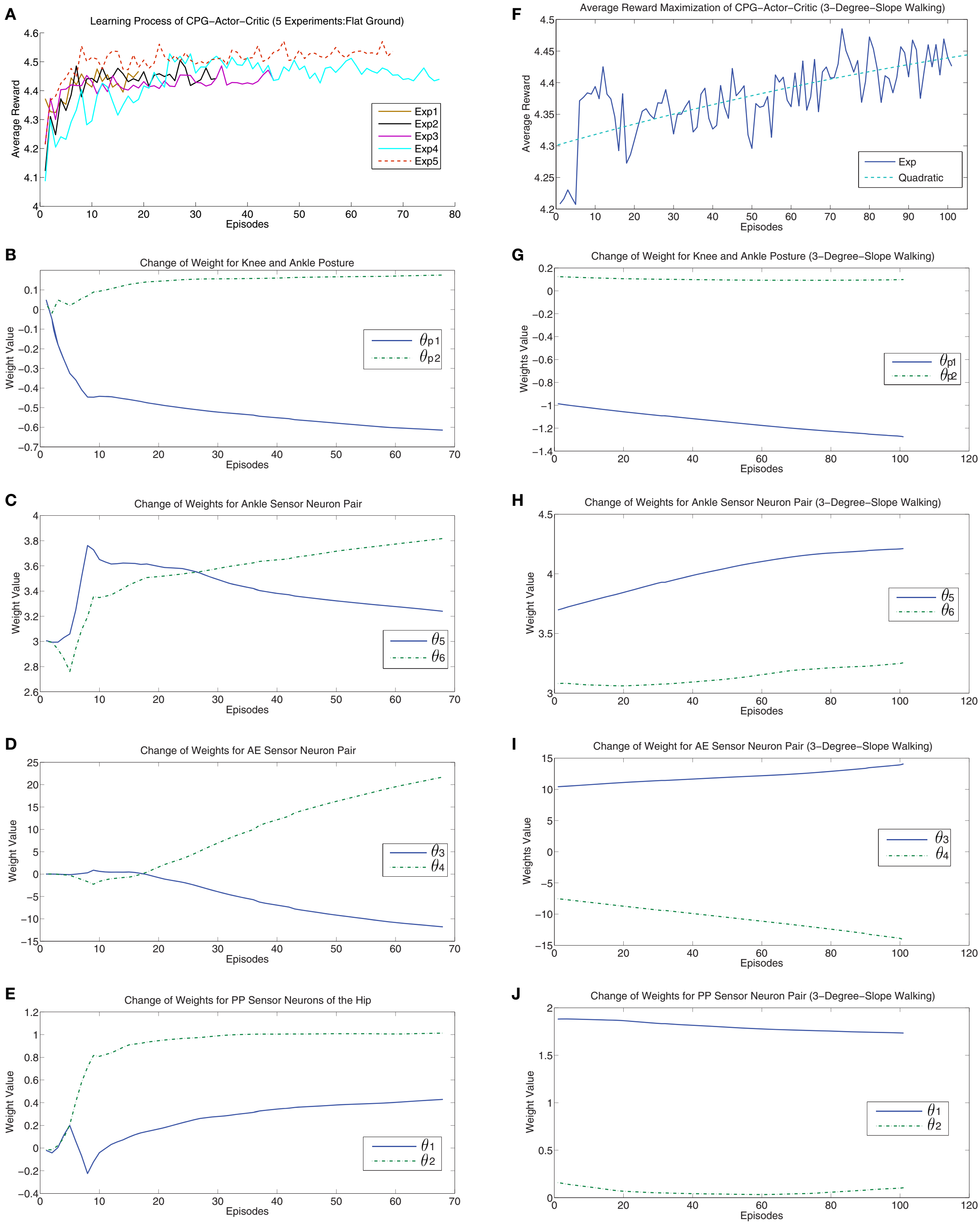


FIGURE 5 | Left column: The results of the runs with top-five reward on flat ground. (A) shows the maximization of average reward for the five runs. **(B–E)** show the results of the run with highest average reward (Exp 5) regarding how connection weights are updated in each CPG by learning process with respect to the contributions of each term respectively. Right column: The results of a run on 3° critical slope. **(F)** shows the "struggling" maximization of expected reward. The green dash line shows a quadratic fitting of the increasing learning curve. **(G–J)** show how connection weights of CPGs are adaptively updated on the critical slope. For details of explanation, please refer to main text. All the "Episodes" mean updating episodes which exclude the episodes unable to satisfy updating rule.

slope is β, then the gravity exerted in the walking direction is given by *f* = *mgsin*β, where m is the mass of the robot and g is the gravity constant. Therefore, **Figure 6A** shows a non-linear change of knee posture. When the robot walks up the slope, the gravity is a resistance force. When β is very small, *mgsin*β ≈ *mg*β shows a linear-like relation in which there is only small error. When the errors are accumulated until the resistance force *f* starts to prevent the robot moving forward, then the non-linear change has to be canceled. This is why there is an abrupt change when the robot walks up on the 3° slope (0.05 rad) which is called "critical" slope.

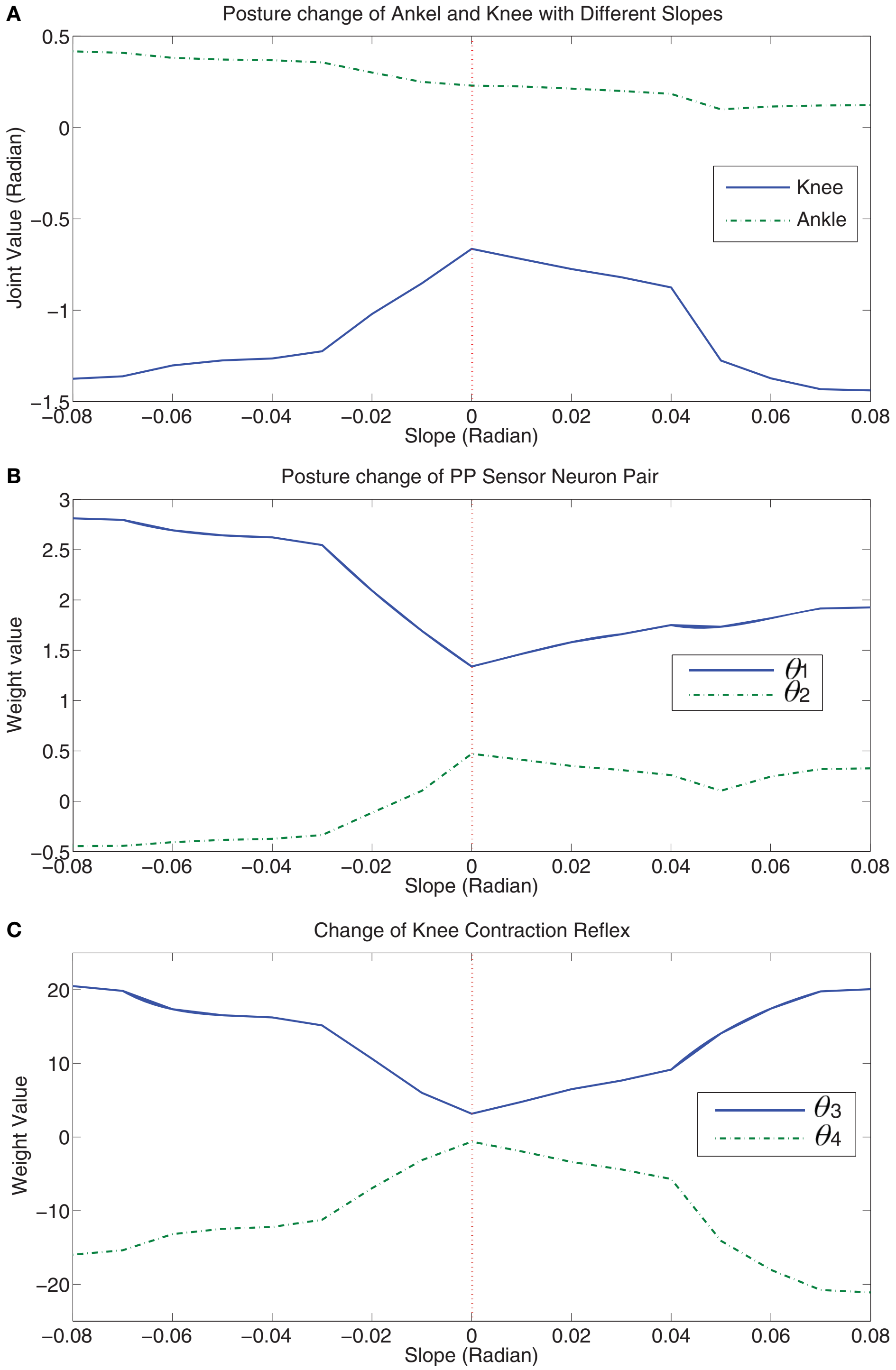


FIGURE 6 | Continued

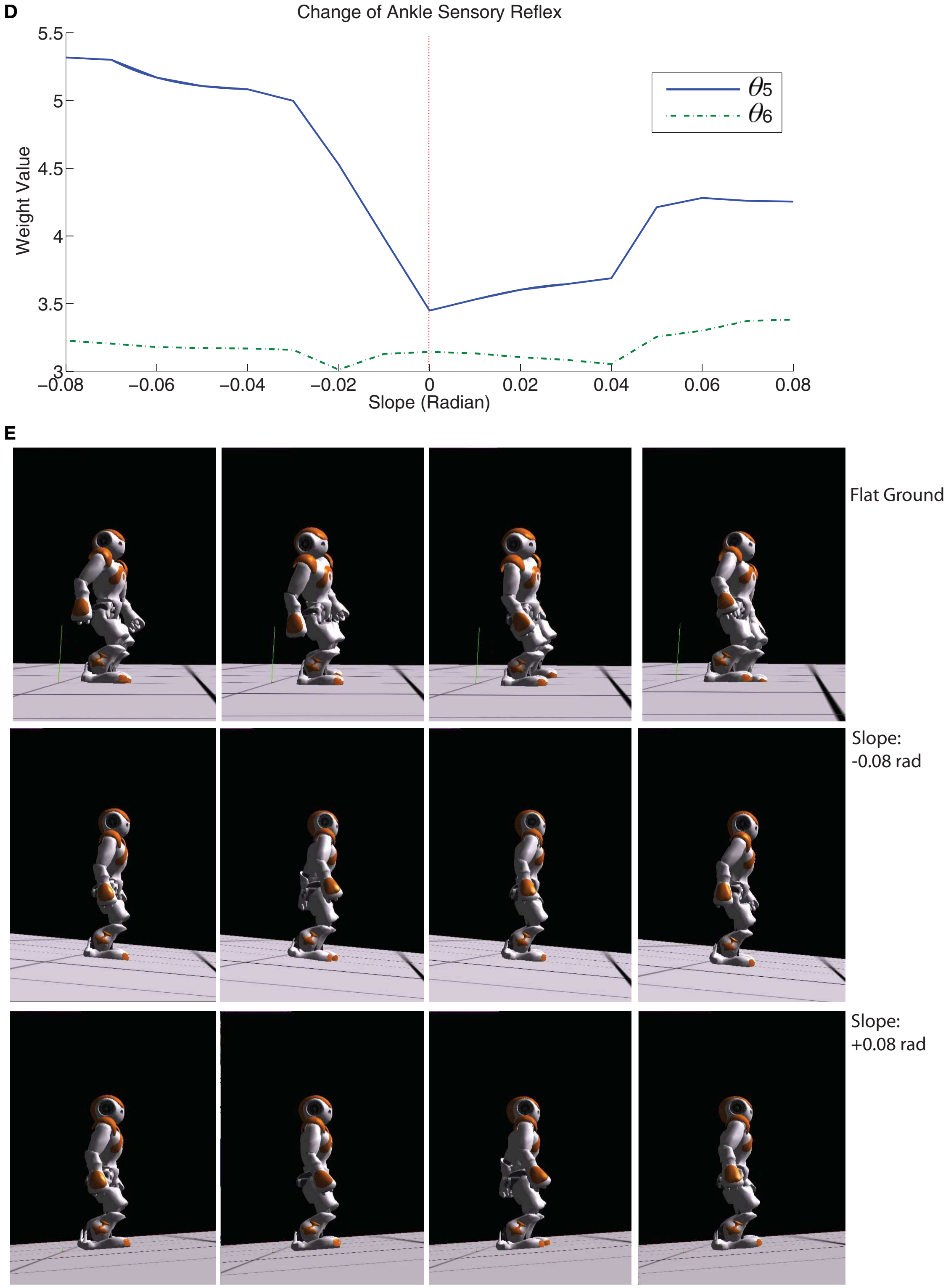


FIGURE 6 | (A) Posture change of ankle and knee joint with respect to slope (−0.08∼0.08). **(B)** shows how the hip joint is adjusted to adapt to slope changes. **(C,D)** show how the knee and ankle reflex change with respect to slope based on the strength of sensory feedback. **(E)** shows the different walking gaits on flat ground and slope (−0.08 and +0.08 rad). Please refer to video (Cai, 2013).

Then when the slope is slightly steeper than 0.05 rad, **Figure 6A** shows a new linear change of the knee posture. The same phenomenon happens to be the case that the robot walks down the slope (slope −0.04 is a turning point). **Figures 5E–J** show the updating of parameters for the "critical" slope. It is clearly visualized that a smooth parameter adjustment of the 3°-slope walking is achieved after the optimal update direction has been found by the learning process of previous slope walking. Interestingly, the posture alternation of the ankle part shows a nearly perfect linear change with respect to alternative slopes. The possible reason may be led by the sensory feedback (refer to the terms $\mathbf{X}_E 3$ and $\mathbf{X}_F 3$ in Appendix A) adaptively changing the ankle posture according to the inclination angle (detected by the gyro sensor) of the robot. This sensory feedback shows the natural adaptation of the CPG architecture which compensates accumulated errors (a non-linear weight change of ankle sensor neurons compensates the gravity in **Figure 6D**). As the key to maintaining stable walking is how to hold up the walking posture as upright as possible, the change of one joint in a kinematic chain of the leg leads to a posture alternation in other joints. Therefore, when the slope is turned from −0.08 to 0.08 rad, with nearly symmetric knee posture change and decreasing ankle change, the hip motion naturally flexes more on the upslope (pushing the body upward) and extends more (flexes less) on the downslope (using the gravity of the body). In **Figure 6B**, the alternation of θ_1 of downslope walking is larger than that of upslope walking indicates that the robot needs more hip flexion for walking on the upslope than the downslope. **Figures 6A,B** insinuates a maintenance of upright walking posture on different slopes.

As for the sensory feedback integration, the knee reflex has a symmetric tendency of upslope and downslope walking (**Figure 6C**). The ankle reflex changes non-linearly to compensate the effect of non-linear gravity change on the ankle joint (**Figure 6D**). Therefore, with an appropriate posture control and decent sensory information, the robot converges to different walking gaits on flat ground, upslope, and downslope (**Figure 6E**). The main difference between the gaits on flat ground and slope except posture is that the amplitude of roll motion is automatically reduced in slope walking in which case that slope walking needs more prudent gaits.

3.2.3. *Data analysis*

The distribution of experimental data is shown in **Table 1**. Based on the reward, the data is categorized into three groups in accordance with **Figure 7A** and the number of results are grouped into these three categories. It is shown both in **Figure 7A** and **Table 1** that most of learning results converge to the reward above 4.3 and 81.3% converged walking gaits are obtained with the reward above 4.4 which are dubbed as good results. In **Figure 7A**, the data shows two linearly increasing relations between the stability and walking distance, proving that the RL learning tries to optimize both of two key factors important for a good walking gait (According to Equation 30, the reward function is equal to the sum of stability and walking distance). **Figure 7B** indicates an interesting boost for the stability at the "critical" slope (0.04 rad) observed in the last section. Two stability clusters are observed in **Figure 7B** (upper). The learning algorithm maintains the stability on two levels separated by the "critical" slope and tries to imporve the walking distance as much as possible (**Figure 7B** (down)). Similarly, the same boost occurs for downslope walking with the separation of $|slope| = 0.04$. However, the stability of downslope walking is more than upslope walking as an acceleration in the forwarding direction is demanded in order to walk upward (In our work, stability is negatively proportional to the acceleration of the robot's pitch and roll directions). Therefore, with less force exerted on the body (less acceleration) and the same walking distance, downslope walking is easier compared to upslope walking in our experiments.

Table 1 | The Distribution of Experimental Data.

Reward	Upslope walking	Downslope walking
<4.3	1	0
4.3–4.4	9	5
>4.4	30	35

3.3. CONCLUSION

With the two experiments, the natural cpg-actor-critic architecture successfully learns different gaits through interaction according to environmental change. It also learns the correlation of posture changes amongst ankles, knees, and hips based on the NAO robot's morphology and the adaptability of neural controller. Meanwhile, it also achieves the implementation of CPG adjusting posture and integrating sensory feedback at the same time.

4. DISCUSSION

4.1. COMPARISON OF OUR WORK WITH RELATED WORK

4.1.1. *Comparison to Nakamura's model*

In order to explain the features of the proposed natural cpg-actor-critic in this article, the comparison of our model to Nakamura's is helpful to generally comprehend this complicated architecture.

4.1.1.1. Similarity. Based on the NAC, Nakamura's model and ours are both natural cpg-actor-critic architecture for learning walking gaits in different environments. The two architectures both layer into basic connections and training connections. The advantage of layering is to reduce the dimensionality of parameter space to avoid the typical problem for reinforcement learning (RL), curse of dimensionality.

4.1.1.2. Differences.

1. The use of a robot platform is different. Apparently, Nakamura's model only works on Taga's stick walker in Matlab. The work shown in this article covers an implementation on a real robot in a simulated physical world. The interaction of morphology, environment, and sensory feedback is closer to the physical world. This is the first implementation of natural cpg-actor-critic on a real robotic platform according to the authors' knowledge. The NAO robot is a robot which moves in 3D space and is more complicated than the 2D stick walker.
2. The type and use of CPG are both different. Nakamura's model is based on Matsuoka oscillators while Hopf oscillators are used

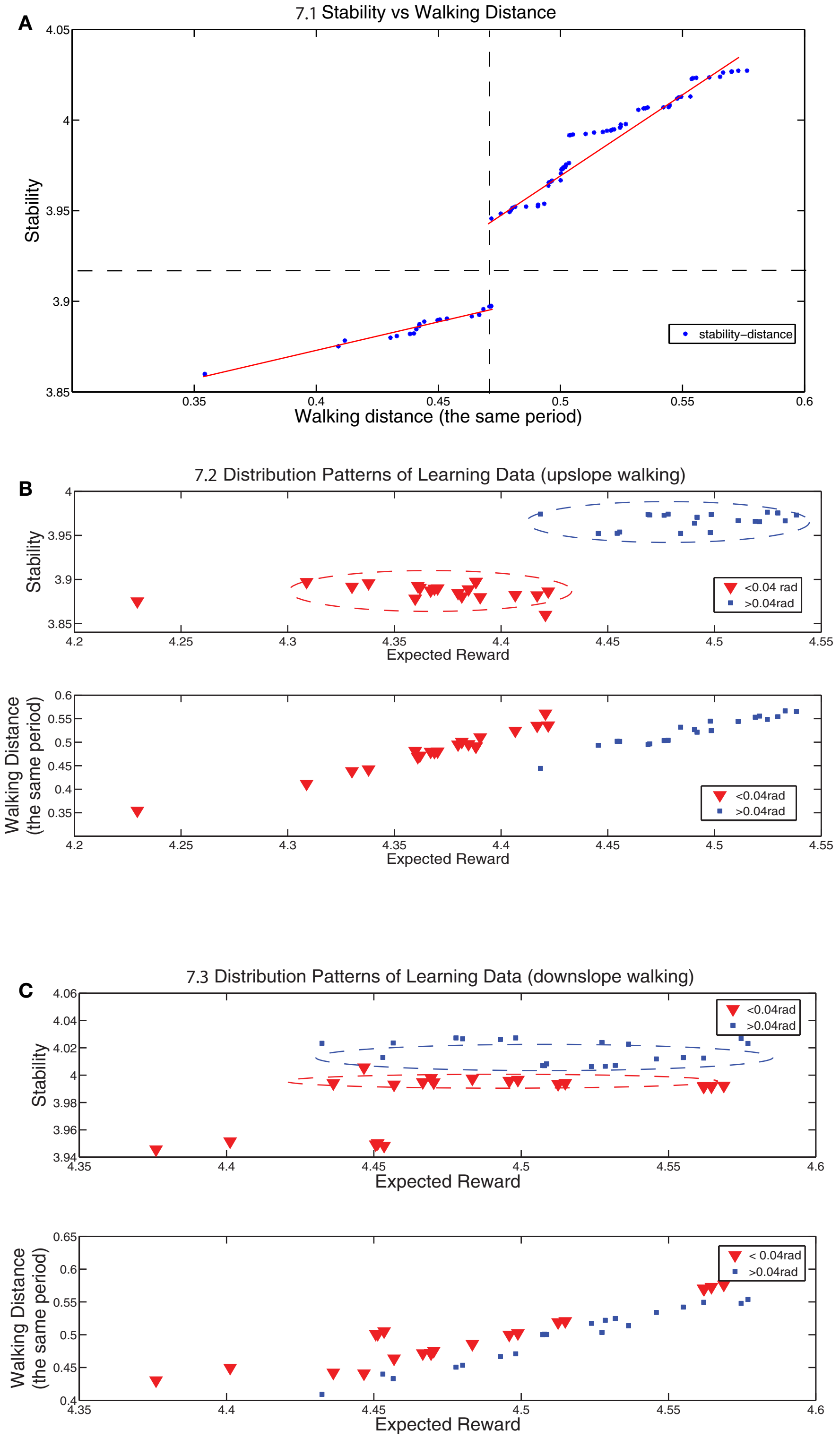


FIGURE 7 | (A) shows distribution points of stability vs walking distance for both upslope and downslope walking (80 data points). The dashed lines split the region into two regions: the left-upper cluster represents the results whose reward are above 4.4 and the right-down cluster represents the results whose reward are between 4.3 and 4.4 except one dot whose reward is below 4.3. Both of these two clusters are distributed around two hand-drawn lines. **(B,C)** show the distribution points of stability vs reward and walking distance vs reward for upslope and downslope walking respectively. The red-triangle dots represent the results for the cases in which $|slope| < 0.04\ rad$ and the blue-plus dots represent the results for $|slope| > 0.04\ rad$. Note that the walking distance is measured always for the same period and it also reflects the speed of walking.

in our work. The main difference of these two oscillators is that a Hopf oscillator can change its pattern simply by adjusting ω_i to preserve the basic characteristics (longer descending phase than ascending phase and anti-phase of the two legs) of walking behaviors but a Matsuoka oscillator cannot (Righetti, 2008). In this article, our CPG architecture is inspired not only by the layered biological structure but also by a sensor-driven mechanism. Sensor neurons are very useful to endow CPGs with preliminary adaptation.

3. The learning mechanism is distinct. As abovementioned, our model reduces more computation load and dimensions by grounding basic properties of walking in the PF layer. On the other hand, by using baseline b in eNAC is helpful in stabilizing the RL algorithm. This is why our model learns much faster and is more stable (not easily get diverged) than Nakamura's.

Generally speaking, the two natural cpg-actor-critic models are distinctly implemented in different bodies in heterogeneous physical worlds with dissimilar use of CPGs.

4.1.2. Features of our work

Except for the characteristics compared to Nakamura's model, our work also generally presents several novel features/perspectives compared to related work (Matsubara et al., 2006; Manoonpong et al., 2007; Endo et al., 2008; Nassour et al., 2013):

1. Morphology logic: the traditional inverse kinematics (IK) model is not used in our model. IK provides a mapping from cartesian space to joint space as long as a trajectory of the end-effector is known. However, walking does not necessarily need IK (McGeer, 1990; Manoonpong et al., 2007; Nassour et al., 2013). Even though IK is coined as a morphological logic for a rigid-body robot (Pfeifer and Bongard, 2006), our work may imply that IK is not the only logic and the interactive memory (Eligibility ψ for natural gradient) can also form a logic to help robot adjust the body posture adapting to environmental change. In Endo et al's. (2008) work, a walking CPG model (only on flat ground) based on IK is presented and the trajectory the foot follows is presumed to be a predefined ellipsoidal path. In our work, the posture is adjusted according to the gradient update interactively focusing on body stability and walking distance instead of recalculating the foot trajectory on different terrains (slope or flat ground). In Nassour et al's. (2013) work, the posture control is only implemented on the ankle part and it is manually tuned. However, our CPG model not only learns the weights of posture control term for the ankle part but also form an adaptive morphological logic by adapting posture alternation to different slopes. As for the work in Manoonpong et al. (2007); Matsubara et al. (2006), a simplified leggy walker without ankle joints is utilized, which seems to make it easier for the robot to walk.

   In a nutshell, in most of the work, an initial posture is manually chosen to be a basis/center which CPGs oscillate around but the evaluation of the posture remains unknown. In our work, we involve a posture control mechanism so that the posture is also adaptively changable to alternative terrains on the basis of past experience.
2. Learning mechanism: our work is the first implementation of natural cpg-actor-critic on a complete humanoid. "Natural" means the gradient approach applied in our model is the steepest and exploration-efficient in light of using natural gradient (Peters and Schaal, 2008). The RL learning presented in the work (Endo et al., 2008; Matsubara et al., 2006) is based on non-natural gradient which may not effectively avoid the "plateau" problem that the small gradient update causes learning to be stuck in a local optima without final convergence. On the other hand, in terms of dimensions of parameter space, our model has the ability to learn by adapting 9 parameters together. In Nassour et al's. (2013) work, there are only two parameters tuned and all the other connection weights are manually defined, including the posture change parameters for ankle parts. In Endo et al's. (2008) work, it is based on a speed-up normal gradient with three parameters to optimize. Therefore, our model seems to be able to work in a relatively high-dimensional parameter space.

However, there are still unsolved problems remaining in our work and they are summarized as follows:

1. Lack of memory: In our work, we demonstrate a CPG architecture leading the humanoid to learn to walk on different slopes. However, we acquire different adapted values of parameters with the same configuration of the parameter set. In order to adapt to the environmental change, this architecture needs spatio-temporal memory to memorize the relation between learned parameters and environmental variables. For example, in our work, contextual variables (the angle of the body) can be detected by gyro sensor. With the spatio-temporal memory, the robot can perform adaptive walking without learning when encountering the contextual changes it has experienced and learned before. The contextual transition may be solved by context-related transition based on bifurcations (Asa et al., 2009) or a context-switching mechanism with topological map (Caluwaerts et al., 2012).
2. Transferability: Even though most of related work demonstrates the results in a simulated robot (Matsubara et al., 2006; Manoonpong et al., 2007; Endo et al., 2008), whether our work is transferable to the physical robot still remains uncertain. In future work, we have to test different results on the physical robot.

4.1.3. Insights into RL approach selection

For the POMDP we concern in this article, function approximation is a very useful solution for solving problems in continuous action space (Orlovskii et al., 1999). Discretizing the state space with feature input of an agent is commonly used approach in actor-critic to representing the states of an agent under the condition that the state space is infinitely large (Orlovskii et al., 1999). Therefore, the value function can be approximated in a lot of ways. For example, it could be approximated based on state predictors (Doya et al., 2002; Gianluca, 2002; Khamassi et al., 2006), artificial neural network (ANN) (van Hasselt, 2011; Farkaš et al., 2012), and basis functions (Doya, 2000b; Peters and Schaal, 2006;

Nakamura et al., 2007; van Hasselt and Wiering, 2007). Regarding to the approximation based on state predictors, they mainly work for multi-model model dependent applications so it is not easy to compare the performance among them. It seems Cacla proposed by Hasselt can be adapted with ANN very easily for both actor and critic for the value-function approximation and action selection (van Hasselt, 2011). In our work, we mainly use episodic NAC to achieve steepest policy update. However, Hasselt et al compare NAC and Calca on cart-pole tasks, finding that Calca outperforms NAC (Orlovskii et al., 1999). The main difference between NAC and Calca is that the former optimizes the policy which maps state space to action space and the latter can search optimal solutions in action space directly. This is why Calca can update the action and approximate the value function separately with two sets of parameters and the action parameters are only updated with positive temporal difference (TD) (van Hasselt and Wiering, 2007). Normal NAC has to update also with negative-TD causing the action space to jump into an unknow space which may distablize and fail NAC. Inspired from Calca, in our work, we use the positive-TD update rule ($AR > GAR$) to avoid the suffering of negative-TD update for NAC. With initial trials for using Calca on cpg-actor-critic, it seems Calca cannot converge even after 300 episodes as it updates slowly.

4.2. DYNAMIC SYSTEMS APPROACH

Walking, in dynamic systems theory (DST), is regarded as a flexible limit-cycle behavior. Learning to walk entails finding out a proper limit cycle of the body motion in a certain environment through interaction. The cpg-actor-critic, as the architecture based on this theory, also covers a lot of aspects of the dynamic systems approach. According to Thelen, a dynamic system could be viewed as an equation $q = N(q, parameters, noise)$ where q is a vector representing all the subcomponents or states of the system and parameters are key factors to which the collective converged behavior is sensitive and that shift the system through different states. N is a non-linear function which determines q which reflects an attractor (Thelen and Smith, 1996). Similarly, the cpg-actor-critic could be written as $cpg = N(cpgstates, \theta, noise)$ where *cpg* is the vector of all the output of CPGs, cpg states are $\mathbf{X}$ and θ is a vector containing policy parameters. N represents the RL functionality which can find an attractor of CPGs. The noise is compressed with proper exploration rate of policies. The whole system is wrapped for a non-linear process of searching for attractors. In a dynamic system, *q* and *parameters* could be very high-dimensional. This is also the drawback of RL where a lot of work is done to reduce the dimensions of state space and parameters. Interestingly, the instability is observed at the beginning of learning (**Figure 5**) then stability emerges from instability. Clearfield argues that new motor capabilities of infants emerge from instabilities (Clearfield, 2004, 2011; Clearfield et al., 2008). In Thelen's theory, instability, including non-linearities, or phase shift or phase transition, is considered as the very source of new forms. In our implementation, the instabilities caused by exploration of an RL algorithm exactly leads to the final generation of a stable gradient. From the perspective of RL, instabilities in DST or infant learning may be the effects of preliminary exploration in order to seek the right direction of developmental tendency. Since the human body is an extremely sophisticated dynamic system which includes different levels (from microscopic to macroscopic) of high-dimensional parameter and state space, it takes more time and gets through more instabilities to finally converge to new behaviors. From the point of view of robotics, it also should be necessary to think about how a robot is able to learn in high-dimensional space with more intelligence. In this sense, cpg-actor-critic proffers a way to explore this open question of RL in a continuous space.

Interaction is of importance in locomotion learning. Inspired by infants learning to walk, the authors tested the use of assistive states ($\mathbf{X}_p$) in cpg-actor-critic architecture. Since "Parental scaffolding" is a necessary factor helping infant to stand up and learn to walk through a repeated process (Adolph et al., 2012), the proposed architecture also shows possibilities of external assistance in learning to walk. Firstly, the assistive states which are directly related to the posture of ankles and knees could be interpreted as external force or bias. Hence, these states could be representations of outer assistance, e.g., from parents' help. Secondly, infants start to learn to walk without mature value or emotion systems to evaluate their behaviors, parents play roles as infants' emotion systems telling them which is good or not thereby causing the maturation of their affective systems (Schore, 2012). In RL, different rules of learning (like update rules and avoidance of falling) are adopted to place a "scaffolding" function primarily in a learning process. However, it lacks a general and evolvable value system for different types of locomotion learning. In this article, the value function is fixed and task-oriented working as a stability indicator for walking. In modern RL approaches, except dealing with more complex high-dimensional learning tasks, a generic reward system which can be adaptive to dissimilar situations is also a challenge. This is why a mature emotion system is demanded in a lot of robotic learning applications (Breazeal and Scassellati, 1999).

4.3. CONCLUSION AND FUTURE WORK

In a nutshell, the work presented in this article simply shows the typical features of dynamic systems pertaining to instabilities, non-linearities, and adaptability to the environment. However, there is still a big difference in performance between an artificial, and a biological (human) adaptive dynamic system which solves more general problems in development and learning. Dynamic systems theory focuses on the development of systems in which new behaviors or attractors can emerge, disappear, and be memorized. In terms of this, RL, as a solver of general learning and developmental problems, needs further research.

In future work, we would like to test the results or the learning process on the physical NAO robot. Moreover, in order to testify the generality of our work and extend the adaptation of our model, experiments on different morphologies, and walking path planning (emphasized by Laumond; Arechavaleta, 2008; Mombaur et al., 2010) are also necessary.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at http://www.frontiersin.org/Neurorobotics/10.3389/fnbot.2013.00005/abstract

REFERENCES

Adolph, K. E., Cole, W. G., Komati, M., Garciaguirre, J. S., Badaly, D., Lingeman, J. M., et al. (2012). How do you learn to walk? Thousands of steps and dozens of falls per day. *Psychol. Sci.* 23, 1387–1394.

Amari, S. (1998). Natural gradient works efficiently in learning. *Neural. Comput.* 10, 251–276.

Amrollah, E., and Henaff, P. (2010). On the role of sensory feedbacks in rowat – selverston cpg to improve robot legged locomotion. *Front. Neurosci.* 4:113. doi:10.3389/fnbot.2010.00113

Arechavaleta, G. (2008). An optimality principle governing human locomotion. *IEEE Trans. Robot.* 24, 5–14.

Asa, K., Ishimura, K., and Wada, M. (2009). Behavior transition between biped and quadruped walking by using bifurcation. *Rob. Auton. Syst.* 57, 155–160.

Baird, L. C. III. (1994). "Reinforcement learning in continuous time: advantage updating," in *Proceedings of the Neural Networks, IEEE World Congress on Computational Intelligence* 4, 2448–2453.

Breazeal, C., and Scassellati, B. A. (1999). "Context-dependent attention system for a social robot," in *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence IJCAI '99* (San Francisco, CA: Morgan Kaufmann Publishers Inc) 1146–1153.

Buono, P. L., and Palacios, A. (2004). A mathematical model of motorneuron dynamics in the heartbeat of the leech. *Physica D* 188, 292–313.

Cai, L. (2013). *Video: Walking on Different Terrains.* Available at; http://www.youtube.com/watch?v=nlloyalhcqa [accessed January, 2013].

Caluwaerts, K., Staffa, M., N'Guyen, S., Grand, C., Dollé, L., Favre-Félix, A., et al. (2012). A biologically inspired meta-control navigation system for the psikharpax rat robot. *Bioinspir. Biomim.* 7:025009. doi:10.1088/1748-3182/7/2/025009

Clearfield, M. W. (2004). The role of crawling and walking experience in infant spatial memory. *J. Exp. Child. Psychol.* 89, 214–241.

Clearfield, M. W. (2011). Learning to walk changes infants' social interactions. *Infant Behav. Dev.* 34, 15–25.

Clearfield, M. W., Osborne, C. N., and Mullen, M. (2008). Learning by looking: infants social looking behavior across the transition from crawling to walking. *J. Exp. Child. Psychol.* 100, 297–307.

Collins, S. H., Wisse, M., and Ruina, A. (2001). A three-dimensional passive-dynamic walking robot with two legs and knees. *Int. J. Rob. Res.* 20, 607–615.

Degallier, R., Righetti, L., Gay, S., and Ijspeert, A. J. (2011). Toward simple control for complex, autonomous robotic applications: combining discrete and rhythmic motor primitives. *Auton. Robots* 31, 155–181.

Dotan, J., Volkinshtein, D. M., and Meir, R. (2008). "Temporal difference based actor critic learning – convergence and neural implementation," in *Proceedings of the NIPS*, eds D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou (Vancouver: Curran Associates Inc), 385–392.

Doya, K. (2000a). Complementary roles of basal ganglia and cerebellum in learning and motor control. *Curr. Opin. Neurobiol.* 10, 732–739.

Doya, K. (2000b). Reinforcement learning in continuous time and space. *Neural. Comput.* 12, 219–245.

Doya, K., Samejima, K., Katagiri, K., and Kawato, M. (2002). Multiple model-based reinforcement learning. *Neural. Comput.* 14, 1347–1369.

Endo, G., Morimoto, J., Matsubara, T., Nakanish, J., and Cheng, G. (2008). Learning cpg-based biped locomotion with a policy gradient method: application to a humanoid robot. *Int. J. Rob. Res.* 27, 213–228.

Farkaš, I., Malík, T., and Rebrová, K. (2012). Grounding the meanings in sensorimotor behavior using reinforcement learning. *Front. Neurorobot.* 6:1. doi:10.3389/fnbot.2012.00001

Frank, M. J., and Claus, E. D. (2006). Anatomy of a decision: Striato-orbitofrontal interactions in reinforcement learning, decision making, and reversal. *Psychol. Rev.* 113, 300–326.

Fumiya, L., Dravid, R., and Paul, C. (2002). "Design and control of a pendulum driven hopping robot," in *Proceeding of the IEEE/RSJ International Conference on Intelligent Robots and Systems* (Lausanne: IROS), 3, 2141–2146.

Geng, T., Porr, B., and Wörgötter, F. (2006). Fast biped walking with a sensor-driven neuronal controller and real-time online learning. *Int. J. Rob. Res.* 25, 243–259.

Gianluca, B. (2002). A modular neural-network model of the basal ganglias role in learning and selecting motor behaviours. *Cogn. Syst. Res.* 3, 5–13.

Golubitsky, M., and Stewart, I. (2004). *The Symmetry Perspective: From Equilibrium to Chaos in Phase Space and Physical Space* (*Progress in Mathematics*). Basel: Birkhäuser Basel.

Graybiel Ann, M. (2005). The basal ganglia: learning new tricks and loving it. *Curr. Opin. Neurobiol.* 15, 638–644.

Grillner, S., Hellgren, J., Ménard, A., Saitoh, K., and Wikström, M. A. (2005). Mechanisms for selection of basic motor programs roles for the striatum and pallidum. *Trends Neurosci.* 28, 364–370.

Grillner, S., Wallén, P., Saitoh, K., Kozlov, A., and Robertson, B. (2007). Review: neural bases of goal-directed locomotion in vertebrates-an overview. *Brain Res. Rev.* 57, 2–12.

Hallemans, A., De, C. lercqD., and Aerts, P. (2006). Changes in 3D joint dynamics during the first 5 months after the onset of independent walking: a longitudinal follow-up study. *Gait Posture* 24, 270–279.

Hooper, L. (2001). *Central Pattern Generators.* Athens: John Wiley and Sons Ltd.

Ijspeert, A. J. (2008). Central pattern generators for locomotion control in animals and robots: a review. *Neural. Netw.* 21, 642–653.

Inada, H., and Ishii, K. (2004). Bipedal walk using a central pattern generator. *Int. Congr. Ser.* 1269, 185–188.

Joel, D., Niv, Y., and Ruppin, E. (2012). Actor-critic models of the basal ganglia: new anatomical and computational perspectives. *Neural. Netw.* 15, 535–547.

Kail, R. V., and Cavanaugh, J. C. (1996). *Human Development.* Belmont: Brooks/Cole Publishing.

Kakade, S. (2002). A natural policy gradient. *Adv. Neural Inf. Process Syst.* 14, 1531–1538.

Khamassi, M., Lachèze, L., Girard, B., Berthoz, A., and Guillot, A. (2005). Actor-critic models of reinforcement learning in the basal ganglia: from natural to artificial rats. *Adapt. Behav.* 13, 131–148.

Khamassi, M., Martinet, L.-E., and Guillot, A. (2006). "Combining self-organizing maps with mixtures of experts: Application to an actor-critic model of reinforcement learning in the basal ganglia," in *From Animals to Animats 9, Proceedings*, Vol. 4095, eds S. Nolfi, G. Baldassarre, R. Calabretta, J. C. T. Hallam, D. Marocco, J.-A. Meyer, O. Miglino, and D. Parisi (Rome: Springer Berlin Heidelberg), 394–405.

Kimura, H., and Kobayashi, S. (1998). "An analysis of actor/critic algorithms using eligibility traces: reinforcement learning with imperfect value function," in *Proceedings of the Fifteenth International Conference on Machine Learning (ICML 1998)*, ed. J. W. Shavlik (Madison, WI*:* Morgan Kaufmann), 24, 278–286.

Konda, V. R., and Tsitsiklis, J. N. (2003). On actor-critic algorithms. *SIAM J. Control Optim.* 42, 1143–1166.

Latash, M. L. (2008). *Neurophysiological Basis of Movement*, 2nd Edn. Champaign: Human Kinetics.

Lee, G., Lowe, R., and Ziemke, T. (2011). Modelling early infant walking: testing a generic cpg architecture on the nao humanoid. *IEEE Int. Conf. Dev. Learn.* 2, 1–6.

Li, C., Lowe, R., Duran, B., and Ziemke, T. (2011). Humanoids that crawl: comparing gait performance of iCub and NAO using a CPG architecture. *IEEE Int. Conf. Comput. Sci. Automat. Eng.* 4, 577–582.

Li, C., Lowe, R., and Ziemke, T. (2012). "Modelling walking behaviors based on cpgs: a simplified bio-inspired architecture," in *From Animals to Animats 12, Volume 7426 of Lecture Notes in Computer Science*, eds T. Ziemke, C. Balkenius, and J. Hallam (Berlin: Springer), 156–166.

Lim, H.-O., Yamamoto, Y., and Takanishi, A. (2002). Stabilization control for biped follow walking. *Adv. Robot.* 16, 361–380.

Manoonpong, P., Geng, T., Kulvicius, T., Porr, B., and Wörgötter, F. (2007). Adaptive, fast walking in a biped robot under neuronal control and learning. *PLoS Comput. Biol.* 3:e134. doi:10.1371/journal.pcbi.0030134

Matsubara, T., Morimoto, J., Nakanish, J., Sato, M.-A., and Doya, K. (2006). Learning feedback pathway of cpg-based controller using policy gradient. *Rob. Auton. Syst.* 54, 911–920.

McGeer, T. (1990). Passive dynamic walking. *Int. J. Rob. Res.* 9, 62–82.

Michel, O. (2004). Webots: professional mobile robot simulation. *J. Adv. Rob. Syst.* 1, 39–42.

Mombaur, K., Laumond, J. P., and Yoshida, E. (2010). An optimal control based formulation to determine natural locomotor paths for humanoid robots. *Adv. Robot.* 24, 515–535.

Nakamura, Y., Mori, T., Sato, M. A., and Ishii, S. (2007). Reinforcement learning for a biped robot based on a CPG-actor-critic method. *Neural. Netw.* 20, 723–735.

Nassour, J., Henaff, P., Ouezdou, F. B., Cheng, G. (2011). "Bipedal Locomotion Control with Rhythmic Neural Circuits," in *Proceedings of International Workshop on Bio-Inspired Robots*, Nantes, France.

Nassour, J., Hugel, V., Benouezdou, F., and Cheng, G. (2013). Qualitative adaptive reward learning with success failure maps: applied to humanoid robot walking. *IEEE Trans. Neural Netw. Learn. Syst.* 24, 81–93.

Orlovskii, G. N., Deliagina, T. G., and Grillner, S. (1999). *Neuronal Control of Locomotion: From Mollusc to Man.* Oxford: Oxford University Press.

Peters, J. (2007). *Machine learning of motor skills for robotics.* Ph.D. thesis, Department of Computer Science, University of Southern California.

Peters, J., and Schaal, S. (2006). Policy gradient methods for robotics. *Rep. U S* 2006, 2219–2225.

Peters, J., and Schaal, S. (2008). Natural actor-critic. *Neurocomputing* 71, 1180–1190.

Pfeifer, R., and Bongard, J. C. (2006). *How the Body Shapes the Way We Think: A New View of Intelligence.* Cambridge, MA: The MIT Press.

Righetti, L. (2008). *Control of legged locomotion using dynamical systems.* Ph.D. thesis, EPFL, Lausanne.

Righetti, L., and Ijspeert, A. (2006). "Design methodologies for central pattern generators: an application to crawling humanoids," in *Proceedings of Robotics: Science and Systems*, Philadelphia.

Rybak, I. A., Shevtsova, N. A., Lafreniere-Roula, M., and McCrea, D. A. (2006). Modelling spinal circuitry involved in locomotor pattern generation: insights from deletions during fictive locomotion. *J. Physiol.* 577, 617–639.

Sato, M., and Ishii, S. (1998). "Reinforcement learning based on online em algorithm," in *Neural Information Processing Systems*, Vol. 11, eds M. J. Kearns, S. A. Solla and D. A. Cohn (Denver: MIT Press), 1052–1058.

Schore, A. N. (2012). *Affect Regulation and the Origin of the Self: The Neurobiology of Emotional Development.* Hillsdale: Taylor and Francis.

Schultz, W. (1998). Predictive reward signal of dopamine neurons. *J. Neurophysiol.* 80, 1–27.

Strom, J., Slavov, G., and Chown, E. (2009). "Omnidirectional walking using zmp and preview control for the nao humanoid robot," in *RoboCup, Volume 5949 of Lecture Notes in Computer Science*, eds J. Baltes, M. G. Lagoudakis, T. Naruse, and S. S. Ghidary (Berlin: Springer), 378–389.

Sutton, R. S., McAllester, D., Singh, S., and Mansour,Y. (2000). Policy gradient methods for reinforcement learning with function approximation. *Adv. Neural Inf. Process. Syst.* 12. 1057–1063.

Taga, G. (1998). A model of the neuro-musculo-skeletal system for anticipatory adjustment of human locomotion during obstacle avoidance. *Biol. Cybern.* 78, 9–17.

Takamitsu, M., Morimoto, J., Nakanishi, J., Sato, M.-A., and Doya, K. (2007). Learning a dynamic policy by using policy gradient: application to biped walking. *Syst. Comput. Jpn.* 38, 25–38.

Thelen, E., and Smith, L. B. (1996). *A Dynamic Systems Approach to the Development of Cognition and Action.* Cambridge: The MIT Press.

Thomas, M., Schweigart, G., and Fennell, L. (2009). Vestibular humanoid postural control. *J. Physiol. Paris* 103, 178–194.

van Hasselt, H., and Wiering, M. A. (2007). "Reinforcement Learning in Continuous Action Spaces," in *Proceedings of the IEEE International Symposium on Approximate Dynamic Programming and Reinforcement Learning 2007 ADPRL 2007*, 272–279.

van Hasselt, H. P. (2011). *Insights in Reinforcement Learning: Formal Analysis and Empirical Evaluation of Temporal-Difference Learning Algorithms.* Ph.D. thesis, SIKS Dissertation series, 2011.

Wiering, M., and van Otterlo, M. (2012). *Reinforcement Learning: State-of-the-Art (Adaptation, Learning, and Optimization).* Berlin: Springer.

Conflict of Interest Statement: 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.

Received: 30 November 2012; accepted: 06 March 2013; published online: 08 April 2013.

Citation: Li C, Lowe R and Ziemke T (2013) Humanoids learning to walk: a natural CPG-actor-critic architecture. Front. Neurorobot. 7:5. doi: 10.3389/fnbot.2013.00005

APPENDIX

A. THE DETAILS OF CPG-ACTOR-CRITIC IN THE IMPLEMENTATION

A.1. DIRECTIONS OF FLEXOR AND EXTENSOR FOR EACH JOINT

In the pitch motion, there are two kinds of moving directions for each joint of NAO: forward (F) and backward (B). The directions of extensor and flexor are given: (1) Hip: Flexor (B+) and Extensor (F−). (2) Knee: Flexor (F+) and Extensor (B−). (3) Ankle: Flexor (B+) and Extensor (F−). The "−" and "+" represent the decrease and increase of joint values.

A.2. DETAILS IN RL AND CPGs

In the RL, the policy parameters $\theta \sim \left[\theta_{1:6}, \theta_{p12}, \theta_9\right]$ are the weights $\mathbf{W}$ in CPGs (θ_9 is not shown in the main text as it is not related to CPGs). The state space is $\mathbf{X} \sim \left\{\mathbf{X}_E, \mathbf{X}_F, \mathbf{X}_p\right\}$, where $\mathbf{X}_E = \{\mathbf{X}_{E1}, \mathbf{X}_{E2}, \mathbf{X}_{E3}\}$, $\mathbf{X}_F = \{\mathbf{X}_{F1}, \mathbf{X}_{F2}, \mathbf{X}_{F3}\}$, $\mathbf{X}_p = \{1,1\}$. All the $\mathbf{X}_E$ and $\mathbf{X}_F$ are sensory feedback on sensor neurons with the functions given by: $\rho_{sn} = sigmoid\left(\theta_{threshold}, \theta_{input}, a\right) = \frac{1}{1+e^{a\left(\theta_{threshold}-\theta_{input}\right)}}$.
Then the $\bar{U}$ of RL policy could be written in details:

$$\bar{U}_{E1} = \theta_1 X_{E1},\ \bar{U}_{F1} = \theta_2 X_{F1} \tag{A1}$$

$$\bar{U}_{E2} = \theta_3 X_{E2},\ \bar{U}_{F2} = \theta_4 X_{F2} \tag{A2}$$

$$\bar{U}_{E3} = \theta_5 X_{E3},\ \bar{U}_{F3} = \theta_6 X_{F3} \tag{A3}$$

$$\bar{U}_{p1} = \theta_{p1} X_{p1},\ \bar{U}_{p2} = \theta_{p2} X_{p2} \tag{A4}$$

where for hip pitch motion $\mathbf{X}_{F1} = sigmoid$ (P_{sh}, Ph, 0.5) and $\mathbf{X}_{E1} = sigmoid$ (Psh, P, −0.5) are the proprioceptive (PP) sensor neurons, the Psh and Ph are the initial value of hip joint of standing posture and the value of the joint sensor. These two not only adjust the posture of hip but also can increase or limit the motion of the flexor or extensor. For the knee part, $\mathbf{X}_{F2} = sigmoid$ (P_{sk}, P_k, 16) and $\mathbf{X}_{E2} = sigmoid$ (P_{sk}, P_k, 16) are the same anterior extremity sensors. The P_{sk} and P_k are the basic posture of knee and the joint value of knee, respectively. 16 indicate a quick reflex when the knee joint reaches the extremity. As for the ankle part, $\mathbf{X}_{F3} = \Xi sigmoid$ (0, P_g, 8) and XE3 $= \Xi sigmoid$ (0, P_g, −8) are ankle sensor neurons. Ξ is a function which is equal to 1 when the foot contacts the ground and 0 when there is no contact. P_g is the angle of upright body based on the gyro sensor. These neurons are used to adjust the motion of ankle joint adaptively to the inclination angle of the body and work like a simple vestibular system. Therefore, the final output of CPGs is: (1) Hip: $\tau_1 = \tau_{\mathrm{E1}} - \tau_{\mathrm{F1}}$. (2) Knee: $\tau_2 = \tau_{E2} + \tau_{F2} + \mathrm{W}_{p1} X_{p1}$, where W_{p1} is equal to converged θ_{p1}. (3)Ankle: $\tau_1 = \tau_{\mathrm{E3}} - \tau_{\mathrm{F3}} + \mathrm{W}_p 2 X_p 2$, where W_{p2} is equal to converged θ_{p2}. The control signals $U = \bar{U} + \delta$, where δ is a vector containing exploration values generated by RL policy. All the abovementioned equations are implemented on one leg and the same is used on the other leg because of the symmetry.

The roll motion adopts sensor-driven CPGs. For the hip roll: $\tau_{hl} = sigmoid(P_{shl}, P_{hl}, 28) - sigmoid(P_{shr}, P_{hr}, 28)$ and $\tau_{hr} = sigmoid(P_{shr}, P_{hr}, 28)$-$sigmoid(P_{shl}, P_{hl}, 28)$ are the output of roll CPGs to left and right hip roll joints, where P_{shl}, P_{shr} are the standing posture of left and right hip pitch joints and P_{hl}, Phr are the values of joint sensors for left and right hip pitch joints. The same mechanism is for ankle roll: $\tau_{al} = sigmoid(P_{sal}, P_{al}, 28) - sigmoid(P_{sar}, P_{ar}, 28)$ and $\tau_{ar} = sigmoid(P_{sar}, P_{ar}, 28) - sigmoid(P_{sal}, P_{al}, 28)$ are the output of roll CPGs to left and right ankle roll joints, where P_{sal}, P_{sar} are the standing posture of left and right ankle pitch joints and P_{al}, P_{ar} are the values of joint sensors for left and right ankle pitch joints.

In order to better and stably approximate Q function in RL, we use another value-function related basis function $\psi = 0.1F$ to increase the stability of RL, where F is the joint value of hip. Since the Equation 27 $J = V^{\pi}(\mathbf{x}_H + 1) - V^{\pi}(\mathbf{x}_0)$, where $V^{\pi}(\mathbf{x}_H + 1)$ is the prediction of future value function dependent on state x_H. So by using $\theta_9\psi$ to approximate $V^{\pi}(\mathbf{x}_H + 1)$ can increase the stability of RL. $V^{\pi}(\mathbf{x}_0)$ is the value function of the initial state which is a constant approximated by baseline.

frontiers in
NEUROROBOTICS

ORIGINAL RESEARCH ARTICLE
published: 30 May 2013
doi: 10.3389/fnbot.2013.00009

An intrinsic value system for developing multiple invariant representations with incremental slowness learning

Matthew Luciw *†, Varun Kompella *†, Sohrob Kazerounian and Juergen Schmidhuber

IDSIA/SUPSI/USI, Lugano-Manno, Switzerland

Edited by:
Florian Röhrbein, Technische Universität München, Germany

Reviewed by:
Subramanian Ramamoorthy, The University of Edinburgh, UK
Minoru Asada, Osaka University, Japan

***Correspondence:**
Matthew Luciw and Varun Kompella, IDSIA, Galleria 2, Lugano-Manno 6928, Switzerland
e-mail: luciwmat@gmail.com; varun@idsia.ch

†Joint First Authors.

Curiosity Driven Modular Incremental Slow Feature Analysis (CD-MISFA; Kompella et al., 2012a) is a recently introduced model of intrinsically-motivated invariance learning. Artificial curiosity enables the orderly formation of multiple stable sensory representations to simplify the agent's complex sensory input. We discuss computational properties of the CD-MISFA model itself as well as neurophysiological analogs fulfilling similar functional roles. CD-MISFA combines 1. unsupervised representation learning through the slowness principle, 2. generation of an intrinsic reward signal through learning progress of the developing features, and 3. balancing of exploration and exploitation to maximize learning progress and quickly learn multiple feature sets for perceptual simplification. Experimental results on synthetic observations and on the iCub robot show that the intrinsic value system is essential for representation learning. Representations are typically explored and learned in order from least to most costly, as predicted by the theory of curiosity.

Keywords: slow feature analysis, intrinsic motivation systems, norepinephrine, neuromodulation, exploration-exploitation

1. INTRODUCTION

We describe a model called CURIOUSity-DRiven, Modular, Incremental Slow Feature Analysis (Curious Dr. MISFA), which autonomously explores various action contexts, learning low-dimensional encodings from the high-dimensional sensory inputs (i.e., video) that result from each such context. Autonomous behavior in this regard requires the coordinated interaction between a number of subsystems which enable an agent to balance exploration-exploitation, to engage in useful contexts while disengaging from others, and to organize representations such that newly learned representations do not overwrite previously learned ones. Ultimately, an agent making use of Curiosity Driven Modular Incremental Slow Feature Analysis (CD-MISFA) learns to seek out and engage contexts wherein it expects to make the quickest progress, learns an appropriate compact, context-dependent representation, and upon fully learning such a representation, disengages from that context to enable further exploration of the environment, and learning of subsequent representations. The goal of such an agent is to maximize intrinsic reward accumulation, and as a byproduct learn all such representations that are learnable given the contexts available to it. We not only show why the interacting subsystems of CD-MISFA are necessary for the kind of unsupervised learning it undertakes, but moreover, we show how the subsystems that enable the model to autonomously explore and acquire new sensory representations, mirror the functional roles of some of the underlying cortical and neuromodulatory systems responsible for unsupervised learning, intrinsic motivation, task engagement, and task switching.

Although difficult, attempts to integrate such disparate functional subsystems are not only helpful in understanding the brain, but are increasingly necessary for building autonomous artificial and robotic systems. It does not suffice, for example, to know the cortical mechanisms responsible for unsupervised learning of sensory representations, if these mechanisms aren't linked to the systems responsible for exploring one's environment. In the absence of external rewards, how should an agent decide which actions and contexts to explore, in order to determine which representations are relevant and learnable? If a sensory representation is deemed overly complex or even unlearnable, what are the mechanisms by which the agent can disengage from exploring its current context, in order to allow it to explore others? Although CD-MISFA is an algorithmic approach to developmental robotics, and does not explicitly model the neural mechanisms by which these functions are realized in the brain, it is notable that the functional roles of the various subsystems in CD-MISFA find counterparts in neurophysiology.

In the following, we first discuss background on CD-MISFA, Artificial Curiosity, and developmental learning, then provide a detailed computational description of how the various subsystems in CD-MISFA operate and interact, followed by a description of the neurophysiological correlates whose functional roles mirror those of CD-MISFA; namely, the interactions between the neuromodulatory systems involved in intrinsic motivation, task engagement, task switching, and value approximation. CD-MISFA is implemented in two situations: an environment composed of synthetic high-dimensional visual contexts, and a real-world environment, with an actively exploring humanoid iCub robot. A method for measuring the learning cost in the different contexts is introduced, and it is shown that the model is most likely to engage within the context where it can learn an as yet unlearned representation, where the cost is least among all possible contexts; this type of behavior is predicted by the theory of curiosity, and may be a general principle of development. The second result shows that IM-based exploration enables the embodied agent to learn interesting sensory representations, again in the predicted order, all while operating on high-dimensional video streams as sensory input.

2. CURIOUS Dr. MISFA

2.1. BACKGROUND

2.1.1. Artificial curiosity

Consider a setting in which an agent operates without a teacher or any other type of external motivation, such as external reward. In this case, an agent needs to be self-motivated, or *curious*. The *Formal Theory of Fun and Creativity* (Schmidhuber, 2006, 1991, 2010) mathematically formalizes driving forces behind curious and creative behaviors. This theory requires that a curious agent have two learning components: an adaptive predictor/compressor of the agent's growing history of perceptions and actions, and a reinforcement learner (Sutton and Barto, 1998). The learning progress or expected improvement of the compressor becomes an intrinsic reward for the reinforcement learner. To maximize intrinsic reward accumulation, the reinforcement learner is motivated to create new experiences such that the compressor makes quick progress.

2.1.2. Curiosity and development

Such a creative agent produces a sequence of self-generated tasks and their solutions, each task still unsolvable before learning, yet becoming solvable after learning. Further, there is an expected order in task-learning. Since the value function of the intrinsic reward contains the cost of learning, in the sense of an estimation of what type of progress it can expect, a task with the lowest cost of learning is preferentially learned next, among all possible tasks.

An orderly acquisition of competence can be seen as a developmental process. An important aspect to development is the gradual emergence of more and more types of skills, knowledge, etc (Schmidhuber, 1997, 2002; Prince et al., 2005). Such emergence, referred to as developmental stages, can observed through behavioral competence (Lee et al., 2007). More specifically, by developmental stages we mean that certain competencies are always seen to precede later ones, although the earlier competencies are not necessarily prerequisites for those learned later (which would be the case in continual learning Ring, 1994).

It has been shown in an *n*-armed bandit scenario that a system based on Artificial Curiosity undergoes developmental stages (Ngo et al., 2011). Further, when the goal is to maximize expected improvement of the predictor or other world model, it was shown that it is *optimal* to concentrate on the current easiest to learn task that has not yet been learned Lopes and Oudeyer (2012) (also in a bandit scenario).

However, the bandit setting involves initial knowledge of the number of possible tasks, in which case the learner can initially reserve learning resources for each task. This is unrealistic for open-ended autonomous development, in which the number of different tasks is initially unknown. What is learned in one part of the environment could apply to another part of the environment. To enable open-ended learning, CD-MISFA learns one module at a time, and if it finds a context that is represented well by one of its already stored modules, it will not need to assign learning resources or time to that context.

2.1.3. Developmental robotics

Developmental Robotics aims to discover and implement mechanisms that can lead to emergence of mind in a embodied agent (Lungarella et al., 2003). The underlying *developmental program* has several general requirements:

- *Not Task Specific.* The task(s) that the robot will handle, i.e., the skills that it can learn, are not explicitly coded in the program. In CD-MISFA, we have such a situation, as the perceptual representations that emerge are dependent on the statistics of the image sequences that are generated from autonomous exploration of the different contexts.
- *Environmental Openness.* Can the system handle a wide variety of possibly uncontrolled environments that the designers might not have explicitly thought of? Currently, CD-MISFA specifically requires a designer to define the environment contexts that the robot can explore over, and so this is a drawback of the system.
- *Raw Information Processing.* Learning is on raw (low-level) information, such as pixels and motor activation values. CD-MISFA slow features are updated directly from pixels, not symbolic inputs or hand-designed feature outputs. On the motor end, the active joints are extremely constrained, but this aspect is low-level as well.
- *Online Learning.* Batch data collection is avoided completely through the *incremental slow feature analysis (IncSFA)* technique, IncSFA (Kompella et al., 2012b), with which a perceptual representation is updated after each image.
- *Continual Learning Ring (1994).* For scaling up the machine's intelligent capabilities, it is necessary that learned skills lead to (or are combined to create) more complex skills. CD-MISFA has not yet demonstrated this, but skill development (albeit in a limited sense) has been shown (Kompella et al., 2012a) to be enabled by its representation learning (i.e., exploiting the learned representations for external reward). Potentially, a framework for continual learning can be built in here; this is to be explored in future work.

With respect to development, a main contribution of this paper is to show how non-task-specific and low-level visuomotor interactions can give rise to emergent behaviors, which are at a higher (but not yet conceptual) level. In a set-up environmental context, an agent's randomly moving effectors (motor babbling) lead to observable consequences involving interactions with the environment not directly controllable by the agent. Slowness learning leads to the emergent "higher-level" representations, since the learning is forced to pay attention to the events that occur on the slower time-scale instead of the regular, but more quickly changing parts of the sensorimotor data stream. For example, a robot that watches its arm can learn causality between its joint controls and the images quite quickly, since there is an abundance of data in babbling — in a sense, this is highly salient. But for the robot to learn about something external to it (i.e., an object), that it interacts with more infrequently, without human supervision, is more difficult, but nonetheless enabled by slowness learning.

2.1.4. Unsupervised visuomotor representation learning

There are many works on representation learning, but we are specifically interested in representation learning from high-dimensional image sequences where the sequence results from an agent's actions. Slow Feature Analysis (SFA; Wiskott and Sejnowski, 2002), is well-suited to this case. SFA applies to image sequences, and it provides *invariant* representations, unlike e.g., Principal Components Analysis (PCA; Jolliffe, 2005), which provides a compressed representation, but not invariance. SFA is also an appearance-based approach (Turk and Pentland, 1991; Murase and Nayar, 1995). Appearance-based approaches learn analog *world properties* (object identity, person identity, pose estimation, etc.) from a set of views. In the setting of a developmental embodied agent, SFA provides emergent invariant representations that resemble symbolic world knowledge; IncSFA provides this autonomously. When an agent is placed in the loop, such that its input sequence is caused by its selected actions, the emergent slow features have been shown to be useful decompositions of the environment (Mahadevan and Maggioni, 2007; Sprekeler, 2011), specifically for reinforcement learning (Sutton and Barto, 1998).

2.1.5. Related works

Related to CD-MISFA in terms of having similar motivations and being based in developmental principles (Weng et al., 2001) are the biologically-constrained intrinsic motivation model, and robotic implementation, of Baldassarre et al. (2012)[1], and the Qualitative Learner of Action and Perception (QLAP; Mugan and Kuipers, 2012). Powerplay (Schmidhuber, 2011; Srivastava et al., 2013) was also important in terms of motivating CD-MISFA.

The QLAP is a developmental robotics system designed to learn simplified predictable knowledge (potentially useful for skills) from autonomous and curiosity-driven exploration. It discretizes low-level sensorimotor experience through defining landmarks in the variables and observing contingencies between landmarks. It builds predictive models on the low-level experience, which it can use to generate plans of action later. It either selects its actions randomly or such that it expects to make fast progress in the performance of the predictive models (a form of artificial curiosity). A major difference between this system and ours is that we operate upon the raw pixels directly, rather than assuming the existence of a low-level sensory model. In QLAP, for example, the sensory channels are preprocessed so that the input variables track the positions of the objects in the scene. Through IncSFA, features emerge for raw visual processing, and this feature development is tightly coupled with the curiosity-driven learning.

The recently formulated PowerPlay can be viewed as a greedy variant of the Formal Theory of Creativity. In PowerPlay, an increasingly general problem solver is improved by searching for the easiest to solve, still not yet known, task, while ensuring all previously solved tasks remain solved. By its formulation, PowerPlay has no problems with *forgetting*, which can easily occur in an open-ended learning setup (Schaal and Atkeson, 1998; Pape et al., 2011). In CD-MISFA, when each new representation is learned well enough to be internally predictable (low error), it is frozen and added to a long-term memory storage, and therefore there will be no destruction of already learned representations. Further, CD-MISFA searches for the context corresponding to the easiest to encode new representation, thereby acting in a PowerPlay-esque manner.

2.2. CD-MISFA OVERVIEW

CD-MISFA (Kompella et al., 2012a) combines *representation learning* with *curiosity-driven exploration*.

The agent autonomously explores among m contexts, and builds a *representation library*, denoted as

$$\Phi^{\mathcal{L}} = \{\Phi_1^{\mathcal{L}}, \Phi_2^{\mathcal{L}}, \ldots, \Phi_n^{\mathcal{L}}\}. \tag{1}$$

There are, lets say $n(\leq m)$ different representations to learn in the environment (but the agent does not know n). So, one representation can suit more than one context. Learning resources are not assigned to each context individually. Instead, the agent learns one representation at a time.

Each representation $\Phi_i^{\mathcal{L}}$ is composed of two subsequent mappings. The first takes a sensory input vector (e.g., pixels) $\mathbf{x}(t)$ (where t indicates discrete time), and encodes it via slow features. A straightforward example is *linear* SFA, which projects $\mathbf{x}$ from I to J dimensions ($J << I$) via matrix $\mathbf{W} = (\mathbf{w}_1, \ldots, \mathbf{w}_J)$, composed of J column vectors which are the slow features. In this case, for the i-th representation,

$$\mathbf{y}^i(t) = \mathbf{x}^T(t)\mathbf{W}^i. \tag{2}$$

The second mapping produces internal state s^i from slow feature output $\mathbf{y}^i$. To this end, it has a set of cluster centers $\mathbf{C} = (\mathbf{c}_1, \ldots, \mathbf{c}_\xi)$ in the slow feature output space, and assigns the current state as the one with the smallest error from the current output:

$$s^i(t) = \arg\min_j \|\mathbf{y}^i(t) - \mathbf{c}_j^i\|. \tag{3}$$

These mappings provide a simplification of the raw sensory data expected to be perceived when the agent is within the context. The first provides invariance, suppressing irrelevant information. The second provides specificity in the remaining (relevant) information.

2.2.1. Contexts

The agent explores different *contexts*, by switching between them. Example contexts are rooms to explore, objects to interact with, or types of videos to perceive. As a specific example context, see **Figure 1**. We do not specifically define context, but note the following. (1) For convenience, a context can be thought of as having some state and action space, that is known to the agent. Thus, each context involves a set of states, a set of actions, and transition probabilities, from one state to another, given some action. (2) There is some *exploration policy*, internal to the context, by which the agent interacts with this environment. Exploration policies define how the randomized exploration (i.e., motor babbling) will occur on the given states and actions, e.g., Brownian

[1] A detailed comparison with the model of Baldassarre *et al* is presented in Section 4.3.

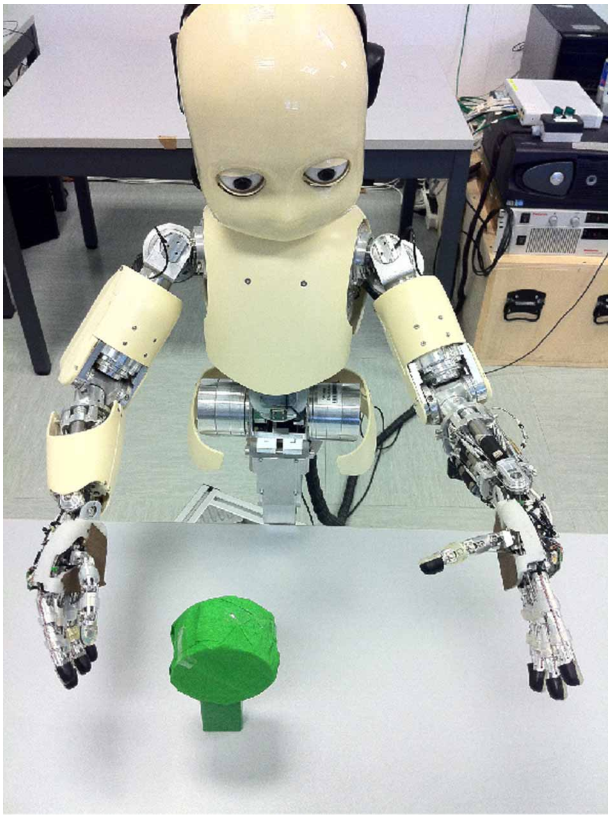


FIGURE 1 | Setup of an environmental context, in which the robot randomly moves its right arm (via single joint babbling). The robot is not equipped with an object detection module, so it does not initially "know" about the object in its field of view. In each episode, the arm reliably displaces the object, and through training on this image data, a slow feature representation emerges that provides information about the state of the object (perturbed or not), invariant to what the robot already "knows" about (its shoulder joint settings).

motion on a mobile robot's wheel velocities with an innate reflex to turn away from obstacles sensed through the distance sensors (Franzius et al., 2007). (3) There is a potentially unobservable world state that defines the high-dimensional observations that will be the input to IncSFA. In **Figure 1** is defined an example of a robot perched over an object. Here, a state space is a discretization of the right arm shoulder joint angles to 20 states, while the actions are (1) increase or (2) decrease the joint angle enough to move to an adjacent state. The world state includes the condition of the object, which is not known to the agent initially. But this becomes "known" through the slow feature encoding, after it is learned. Another example context is a simple grid world (Sutton and Barto, 1998) where the agent explores via random selection of one of four actions (up, down, left, and right). Its state space is given by the grid with observations of high-dimensional images showing the grid and the agent as viewed from above (Lange and Riedmiller, 2010; Luciw and Schmidhuber, 2012).

Each interaction with a context is an episode. There is a start condition to the episode, and an ending condition, which must occur at some point in the random exploration. After the ending condition, the agent must decide whether to continue exploration of this context, or to move to another[2].

The agent uses curiosity to explore among multiple contexts. Rewards and motivation are intrinsic to the agent, and this intrinsic reward is calculated from representation learning progress. The agent can choose to remain engaged in its current context (exploitation), or seek to engage in another context (exploration). These decisions are due to utility judgements, where the utility is an estimate of *expected* learning progress of remaining engaged in the current context versus the expected learning progress of another contexts. If the former is higher, remaining engaged within the current context is most valuable, and, if the latter is higher, disengagement and switching is the more valuable choice.

Figure 2 shows the architecture of CD-MISFA. The "adaptive module" encompasses the unsupervised learning part, which involves a combination of IncSFA and Robust Online Clustering (ROC). The representation library is shown by the "trained modules." Estimation errors are denoted by ϵ, while intrinsic reward is denoted by $\dot{\epsilon}$. The intrinsic reward signal feeds into the value function estimation module. The possible environmental contexts are shown at the bottom, the current context is the "state" (with respect to the higher-level value function), while the "action" is either to remain engaged in that context, or to disengage and go to another.

Next, in Section 2.3, we discuss learning of a single representation. Specifically, we use IncSFA and the ROC method, respectively. Some details of these algorithms are described below, but more thorough descriptions can be found elsewhere (Guedalia et al., 1999; Weng et al., 2003; Peng and Yi, 2006; Zhang et al., 2005; Kompella et al., 2012b).

2.3. UNSUPERVISED REPRESENTATION LEARNING: IncSFA

SFA is concerned with the following optimization problem:

Given an I-dimensional input signal $\mathbf{x}(t) = [x_1(t), \ldots, x_I(t)]^T$, find a set of J instantaneous real-valued functions $\mathbf{g}(x) = [g_1(\mathbf{x}), \ldots, g_J(\mathbf{x})]^T$, which together generate a J-dimensional output signal $\mathbf{y}(t) = [y_1(t), \ldots, y_J(t)]^T$ with $y_j(t) := g_j(\mathbf{x}(t))$, such that for each $j \in \{1, \ldots, J\}$

$$\Delta_j := \Delta(y_j) := \langle \dot{y}_j^2 \rangle \quad \text{is minimal—} \tag{4}$$

under the constraints

$$\langle y_j \rangle = 0 \quad \text{(zero mean)}, \tag{5}$$

$$\langle y_j^2 \rangle = 1 \quad \text{(unit variance)}, \tag{6}$$

$$\forall i < j : \langle y_i y_j \rangle = 0 \quad \text{(decorrelation and order)}, \tag{7}$$

with $\langle \cdot \rangle$ and $\dot{y}$ indicating temporal averaging and the derivative of y, respectively.

[2] We note that there are some similarities with the *options* framework Sutton et al. (1999) here. One could link the start condition idea to an initiation state, and the end condition would correspond to having a termination probability of one at some state, and zero elsewhere. Indeed, one can view the problem of representation learning as analogous to abstraction learning in options, which remains an important open problem. However, we do not need to formalize contexts as options in this paper.

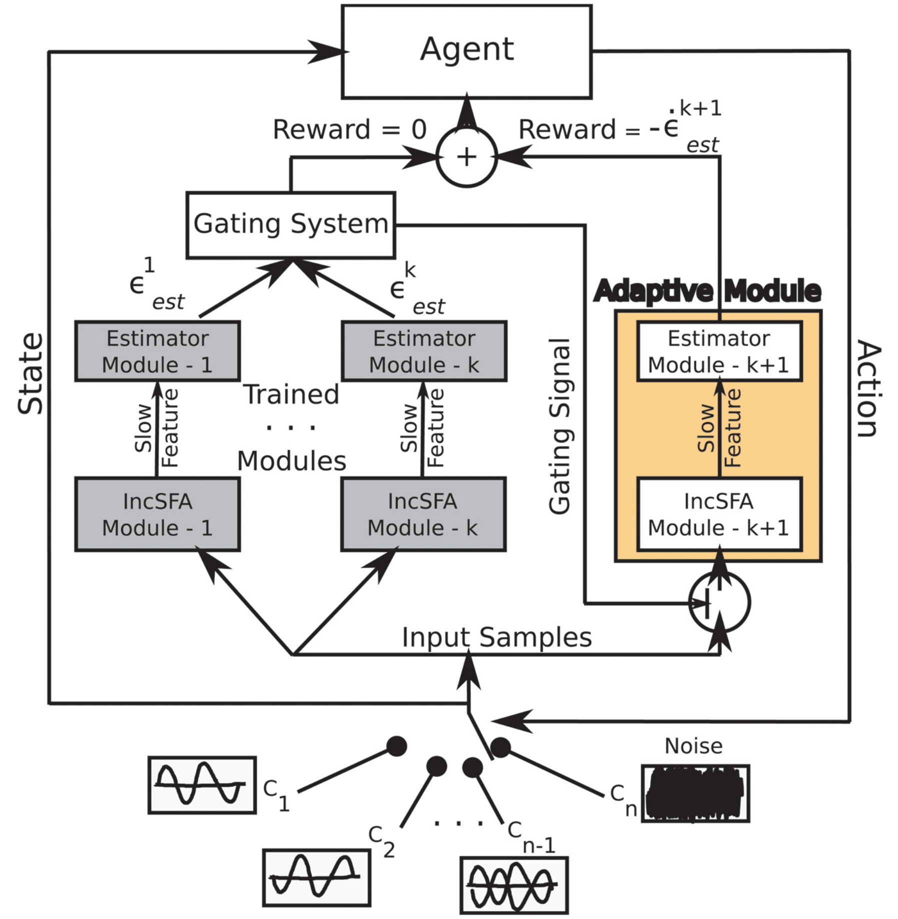


FIGURE 2 | Architecture of CD-MISFA.

The goal is to find instantaneous functions g_j generating different output signals that are as *slowly varying* as possible. The decorrelation constraint (7) ensures that different functions g_j do not code for the same features. The other constraints (5) and (6) avoid trivial constant output solutions.

In a linear sense, the optimization problem can be solved through an eigenvector approach, involving two uses of principal component analysis (PCA)—first, of the covariance matrix of the inputs (for whitening) and, second, of the covariance matrix of the whitened approximate *derivative* measurements (Wiskott and Sejnowski, 2002). IncSFA uses incremental algorithms for the two required PCAs. For the first, Candid Covariance-Free Incremental PCA (Zhang and Weng, 2001; Weng et al., 2003), is used, which can also reduce the dimensionality by only computing the K highest eigenvectors. For the second, Minor Components Analysis (MCA; Oja, 1992; Peng and Yi, 2006; Peng et al., 2007) updates the J slowest features.

The overall framework of IncSFA is shown in **Algorithm 1**. IncSFA needs to update the signal mean (learned incrementally by simple online average estimation), the K principal components, and the J slow features. In general $K < I$ and $J < K$ — and K and J are parameters of the algorithm. The learning methods use "amnesic" learning rate schedule, so they are potentially suited to non-stationary input sequences. IncSFA uses Hebbian (CCIPCA) and anti-Hebbian (CIMCA) update rules Dayan and Abbott (2001) to compute slow-features from a time-varying input signal.

CCIPCA updates estimates of eigenvalues and eigenvectors from each centered observation. CCIPCA combines a statistically efficient Hebbian update with the residual method (Kreyszig, 1988; Sanger, 1989) to generate observations in a complementary space in order to update components besides the first, dealing with the requirement that any component must also be orthogonal to all higher-order components. The CCIPCA algorithm is presented in **Algorithm 2**. The principal component estimates are used to construct a whitening matrix. After whitening, the signal is (approximately) normalized and decorrelated.

Minor Components Analysis preferentially learns the least significant principal components. The update for each slow-feature vector $\mathbf{w}_i$ from 1 to J, is

$$\mathbf{w}_i \leftarrow (1-\eta^{MCA})\mathbf{w}_i - \eta^{MCA}\left((\dot{\mathbf{z}}\cdot\mathbf{w}_i)\,\dot{\mathbf{z}} + \gamma\sum_j^{i-1}(\mathbf{w}_j\cdot\mathbf{w}_i)\mathbf{w}_j\right) \tag{8}$$

Algorithm 1: IncSFA (J, K, θ)

```
//Incremental update of $J$ slow features from
  samples $\mathbf{x} \in \mathcal{R}^I$
// $\mathbf{V}$ : $K$ columns: PCs of $\mathbf{x}$
// $\mathbf{W}$ : $J$ columns: SFs
// $\mathbf{v}^\gamma$ : First PC in $\dot{\mathbf{z}}$-space
// $\bar{\mathbf{x}}$ : Mean of $\mathbf{x}$
$\{\mathbf{V}, \mathbf{W}, \mathbf{v}^\gamma, \bar{\mathbf{x}}\} \leftarrow$ INITIALIZE ()
for $t \leftarrow 1$ to $\infty$ do
    $\check{\mathbf{x}} \leftarrow$ SENSE($worldstate$)
    $\{\eta_t^{PCA}, \eta_t^{MCA}\} \leftarrow$ LRNRATESCHEDULE ($\theta, t$)
    $\mathbf{x} \leftarrow (1 - \eta_t^{PCA})\, \bar{\mathbf{x}} + \eta_t^{PCA}\, \mathbf{x}$ //Update mean
    $\mathbf{u} \leftarrow (\mathbf{x} - \bar{\mathbf{x}})$ //Centering
    //Candid Covariance-Free Incremental PCA
    $\mathbf{V} \leftarrow$ CCIPCA-UPDATE ($\mathbf{V}, K, \mathbf{u}, \eta_t^{PCA}$)
    $\mathbf{S} \leftarrow$ CONSTRUCTWHITENINGMATRIX ($\mathbf{V}$)
    If $t > 1$ then ($\mathbf{z}_{prev} \leftarrow \mathbf{z}_{curr}$) //Store prev.
    //Whitening and dim. reduction
    $\mathbf{z}_{curr} \leftarrow \mathbf{S}^T \mathbf{u}$
    if $t > 1$ then
        $\dot{\mathbf{z}} \leftarrow \left(\mathbf{z}_{curr} - \mathbf{z}_{prev}\right)$ //Approx. derivative
        //For seq. addition ($\gamma$)
        $\mathbf{v}^\gamma \leftarrow$ CCIPCA-UPDATE ($\mathbf{v}^\gamma, 1, \dot{\mathbf{z}}, \eta_t^{PCA}$)
        $\gamma \leftarrow \mathbf{v}^\gamma / \|\mathbf{v}^\gamma\|$
        //Covariance-free Incremental MCA
        $\mathbf{W} \leftarrow$ CIMCA-UPDATE ($\mathbf{W}, J, \dot{\mathbf{z}}, \gamma, \eta_t^{MCA}$)
    end
    $\mathbf{y} \leftarrow \mathbf{z}_{curr}^T \mathbf{W}$ //Slow feature output
end
```

Algorithm 2: CCIPCA-Update ($\mathbf{V}$,K,$\mathbf{u}$,η)

```
//Candid Covariance-Free Incremental PCA
$\mathbf{u}_1 \leftarrow \mathbf{u}$
for $i \leftarrow 1$ to $K$ do
    //Principal component update
    $\mathbf{v}_i \leftarrow (1 - \eta)\, \mathbf{v}_i + \eta \left[\frac{\mathbf{u}_i \cdot \mathbf{v}_i}{\|\mathbf{v}_i\|}\, \mathbf{u}_i\right]$
    //Residual
    $\mathbf{u}_{i+1} = \mathbf{u}_i - \left(\mathbf{u}_i^T \frac{\mathbf{v}_i}{\|\mathbf{v}_i\|}\right) \frac{\mathbf{v}_i}{\|\mathbf{v}_i\|}$
end
return $\mathbf{V}$
```

where η_{MCA} is a learning rate. This update is based on anti-Hebbian learning with an additional Gram–Schmidt term inside the summation that enforces different features to be orthogonal. After updating, a feature is normalized for stability.

The feature output is an instantaneous function,

$$\mathbf{y}(\mathbf{t}) = \mathbf{z}(\mathbf{t})^T \mathbf{w}(t). \tag{9}$$

2.4. ADAPTING THE STATES WITH ROC

In a context's pre-defined state space, each state has its own instance of an online clustering algorithm. Clustering is done in an associative space that *combines* this pre-defined state space with the slow feature output space. These clusters, once learned, act as augmented internal states, potentially providing information about invariants captured with IncSFA.

As an example, consider again the robot viewing its arm move eventually toppling an object in the scene. The state space here is a quantization of the joint angles of the shoulder into 20 bins, thereby providing 20 states, leading to 20 instances of the clustering algorithm. A developed slow feature output here is a step function, e.g., when the object is not toppled, the feature output equals zero, and when the object is toppled the feature output equals one. Upon convergence of, first, IncSFA and, second, the clustering, each joint-angle state will be replaced by two internal states, which inform whether the object is or is not toppled.

Learning these clusters is not as straightforward as the above example makes it seem, since the signal is highly non-stationary during the early learning phases, due to its input being a function of adapting slow features. The slow feature outputs can change rapidly during the training phase. The estimator therefore has to be able to change its estimates to this non-stationary input, while converging to a good estimate when the input becomes stable. To this end, we use a clustering algorithm to specifically handle non-stationary data, called ROC (Guedalia et al., 1999; Zhang et al., 2005).

ROC is similar to an incremental K-means algorithm—a set of cluster centers is maintained, and with each new input, the most similar cluster center (the winner) is adapted to become more like the input. Unlike k-means, with each input, it follows the adaptation step by *merging* the two most similar cluster centers, and *creating a new cluster center* at the latest input. In this way, ROC can quickly adjust to non-stationary input distributions by directly adding a new cluster for the newest input sample, which may mark the beginning of a new input process.

But is this plasticity at the cost of stability? No. In order to enforce stability, clusters maintain a weight, which increases faster for more similar (to the cluster center) inputs. A large weight prevents a cluster center from changing that much. When two clusters are merged, their weights are also combined.

A sketch of the ROC per-sample update is in **Algorithm 3**. The ROC algorithm repeatedly iterates through the following steps. For every input sample, the algorithm finds the closest cluster *winner* and updates the center $\mathbf{c}_{winner}$ toward it, also increasing the weighting parameter a_{winner}. Next, the closest two clusters are merged into one cluster. Then, a new cluster is created around sample $\mathbf{y}$. Finally, all clusters weights decrease slightly. Parameters required are ξ, the maximum number of clusters, an amnesic parameter ϕ to prevent convergence, and the response function for similarity measurement.

Algorithm 3: ROC-Amnesic($\mathbf{y}, s, \xi, \phi$)

```
// Cluster SFA-encoded samples $\mathbf{y} \in \mathcal{R}^J$
// $\mathbf{y}$ : Slow feature encoded input
// $s$ : Context state
// $\xi > 1$ : Maximum number of clusters
// $0 \le \phi \le 1$ : Amnesic parameter
//Determine which set of clusters to use
// $\mathbf{C}$ : Set of cluster centers
// $\mathbf{a}$ : Set of cluster weights
$\{\mathbf{C}, \mathbf{a}\} \leftarrow$ GetClusteringInstance ($s$)
if $|\mathbf{C}| < \xi$ then
    // Cluster center is $\mathbf{y}$, weight is 0
    $\{\mathbf{C}, \mathbf{a}\} \leftarrow$ AddNewCluster ($\mathbf{y}, \mathbf{C}, \mathbf{a}$)
else
    $winner \leftarrow \arg\max_i \text{Response}(\mathbf{y}, \mathbf{c}_i)$
    $\mathbf{c}_{winner} \leftarrow \mathbf{c}_{winner} + \frac{\mathbf{y} - \mathbf{c}_{winner}}{a_{winner} + 1}$
    $a_{winner} \leftarrow a_{winner} + \text{Response}(\mathbf{y}, \mathbf{c}_{winner})$
    // Merge the two closest
    $\{\gamma, \delta\} \leftarrow \arg\max_{\gamma,\delta,\gamma\neq\delta} \text{Response}(\mathbf{c}_\gamma, \mathbf{c}_\delta)$
    $\mathbf{c}_\gamma \leftarrow \frac{\mathbf{c}_\gamma a_\gamma + \mathbf{c}_\delta a_\delta}{a_\gamma + a_\delta}$
    $a_\gamma \leftarrow a_\gamma + a_\delta$
    // Latest input becomes new cluster
    $\mathbf{c}_\delta \leftarrow \mathbf{y}$
    $a_\delta \leftarrow 0$
    // Forgetting (leak)
    for $i \leftarrow 1$ to $\xi$ do
        $a_i \leftarrow a_i(1 - \phi)$
    end
end
```

2.5. INTRINSIC REWARD

The intrinsic reward is expected learning progress. Learning progress is approximated as the decrease in context-specific cumulative estimation error. Each context state i has an associated error ϵ^i_{est}. These errors are updated whenever the agent visits that state—

$$\epsilon^i_{est}(t) = \min_j ||\mathbf{y}(t) - \mathbf{c}_j|| \tag{10}$$

where $\mathbf{y}(t)$ is the slow-feature output vector and $\mathbf{c}_j$ is the j-th cluster center associated with this state. The context's current estimation error is the sum of stored errors, over all M context states,

$$\epsilon_{est}(t) = \sum_{i=1}^{M} \epsilon^i_{est}(t), \tag{11}$$

and the intrinsic reward is the *derivative* of the total estimation error $\dot{\epsilon}_{est} = \epsilon_{est}(t) - \epsilon_{est}(t-1)$. **Figure 3** shows an example with a 20-state estimator.

2.6. MODULE STORAGE AND GATING

Once the slow feature outputs stabilize, the estimator clusters converge and the error will become very low. Next, estimator clusters with small weights a_i are eliminated, to avoid having spurious internal states. Finally, this overall representation *module* is frozen, considered *learned*, and placed in long-term memory.

The already trained set of modules are the abstraction library $\Phi^{\mathcal{L}}$ (Equation 1). If one of these module's estimation error within a context is below a threshold, that context is assigned that module's representation and the adaptive training module will be prevented from learning, by this gating signal. There will no intrinsic reward in this case. On the other hand, if the estimation error of all the trained modules for the incoming data is above the threshold, the gating signal enables the single adaptive module to be trained on the input data. Hence the training module will encode only data from input streams that were not encoded earlier.

2.7. ENGAGE/DISENGAGE MECHANISM

Every time the agent exits a context, the agent needs to make a decision. To this end, the agent can take two *internal-actions*, $\mathcal{A}^o = \{\text{engage, disengage}\}$. The internal-action *engage* allows the agent to stay in the same context (starting over), while *disengage* causes the agent to switch to another context. For the purposes of our model, we do not allow the agent to select the context it will switch to, instead having it randomly selected. Thus, the transition-probability model P of the internal environment (modeling transition probabilities between all pairs of contexts i and j, conditioned on the two internal-actions) is given by:

$$P^{engage}_{ij} = \begin{cases} 1, & \text{if } i = j \\ 0, & \text{if } i \neq j \end{cases} \tag{12}$$

$$P^{disengage}_{ij} = \begin{cases} 0, & \text{if } i = j \\ \frac{1}{N-1}, & \text{if } i \neq j \end{cases} \tag{13}$$

$\forall i, j \in [1, \ldots, N]$.

2.8. REWARD AND VALUE FUNCTION

The agent maintains an estimated *reward function*, which is the expected change in estimation error when transitioning from context o to context o' (and $o = o'$ is possible). The agent's reward function is updated at every engage-disengage decision, from the intrinsic rewards, as a sample average:

$$R^{o,o'}_a := (1 - \eta)\, R^{o,o'}_a + \eta \sum_t^{t+T} -\dot{\epsilon}_{est}(t) \tag{14}$$

where $0 < \eta < 1$ is a learning rate, T was the duration of the previous context until its termination, $(o, o') \in \{O_1, \ldots, O_n\}$ and $a \in \{\text{engage, disengage}\}$.

Using the current updated model of the reward function R and the internal-state transition-probability model P, the agent's policy ($\mathcal{O} \times \mathcal{A}^o \rightarrow [0, 1]$) is updated.

It is important that the policy adapts quickly enough to adapt to the quickly changing reward function. Intrinsic rewards can

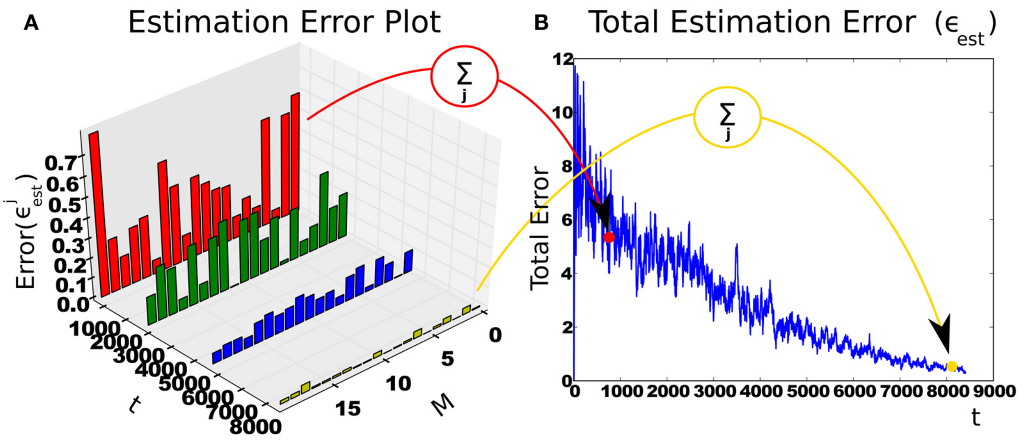


FIGURE 3 | Intrinsic reward is calculated from reduction of context-specific cumulative estimation error. (A) The change in estimation error over time in a context with 20 states (M). With more experience, the features stabilize and estimator errors decrease. **(B)** The sum of estimation errors. The subsequent difference is the intrinsic reward.

change quickly as learning progresses, and the RL must adapt quicker than the underlying representation learner. We used model-based Least Squares Policy Iteration (Lagoudakis and Parr, 2003), which is an efficient value-estimation technique, although in principle the more biologically-plausible temporal-difference (TD) methods could also work.

2.8.1. *Epsilon-greedy*

The agent cannot take the value-maximizing decision from the very beginning, since it needs time to build its value estimates to a more accurate level. Early on, it can make decisions more or less randomly so that it can gather experience in the different contexts, and to learn good estimates of value over all contexts. Given good value estimates, it can choose to engage within the context where it should learn quickly, in other words, make the fastest learning progress, and to lead to a quick learning of the next representation. To this end, the model augments its internal action selection with decaying ϵ-greedy exploration.

3. NEURAL CORRELATES TO CD-MISFA

3.1. SFA AND COMPETITIVE LEARNING—ENTORHINAL CORTEX AND HIPPOCAMPUS

Slow Feature Analysis variants have been used to simulate representation learning in a number of biological scenarios. Based on the general principle that underlying driving forces manifest through slow changes in sensory observations, the features that emerge from SFA often encode important *invariants*. Hierarchical SFA has been shown to develop *grid cells* from high-dimensional visual input streams (Franzius et al., 2007). Grid cells, found in entorhinal cortex (EC) (Hafting et al., 2005), have a pattern of firing that effectively represent hexagonal codes of any two-dimensional environment. As such, grid cells are effective *general* representations for spatial navigation in typical environments.

A competitive learning layer, over the top-layer of slow features, leads to features acting as *place cells* or *head-direction cells*, depending on what changes more slowly from the observation sequences. A place cell will fire when the animal is in a specific location in the environment, typically invariant to its heading direction. Head-direction cells fire when the animal faces a certain direction, no matter what coordinate position it is in. Place cells and head-direction cells are found in hippocampus (O'Keefe and Dostrovsky, 1971; Taube et al., 1990), which has input from EC. It's been hypothesized that hippocampus acts as a relatively fast encoder of specific, episodic information, on top of cortex, which learns general structure from lots of data over a long period (Cohen and O' Reilly, 1996)—"It has been proposed that this universal spatial representation might be recoded onto a context-specific code in hippocampal networks, and that this interplay might be crucial for successful storage of episodic memories (Fyhn et al., 2007)."

SFA's biological plausibility was furthered by IncSFA, which avoids batch processing and has Hebbian and anti-Hebbian updating equations. Hierarchical SFA (Franzius et al., 2007) and IncSFA (Luciw et al., 2012), with competitive learning on top, was shown to develop place and head-direction cell representations. For the representations learned in CD-MISFA, we use the basic structure suggested by these results: A slow feature learner (possibly hierarchical) for *global* features (IncSFA), inputs into a competitive learner for development of *local* features (ROC).

3.2. NEUROMODULATORY SUBSYSTEMS FOR INTRINSIC REWARD AND CONTEXT SWITCHING

3.2.1. *Intrinsic rewards: dopamine and learning progress*

Dopaminergic projections originate from the ventral tegmental area (VTA). Dopamine has been implicated in reward prediction (Schultz et al., 1997), leading to plausible relation to the theory of reinforcement learning (Sutton and Barto, 1998)—specifically, dopamine may be acting as a TD error signal. However, this account remains controversial (Redgrave et al., 1999; Kakade and Dayan, 2002). A major deviation from the dopamine as TD-error theory comes from data implicating dopamine in responding to novel salient stimuli (Schultz, 1998; Redgrave and Gurney, 2006), even for stimuli that are not predictive of reward. Dopaminergic responses to such stimuli fade over subsequent trials. It has been proposed that this characteristic serves the purpose of a "novelty bonus"—e.g., a reward addendum serving as a "optimistic initialization."

These data present intriguing correlations to the curiosity theory. Dopamine release in response to novel stimuli could potentially signal a predicted intrinsic reward—an expectation of *learning progress*. Could DA in some situations signal the intrinsic reward? Dopamine's potential role in intrinsic motivation has been discussed before (Redgrave and Gurney, 2006; Kaplan and Oudeyer, 2007), but not with respect to the formal theory of curiosity Schmidhuber (2010), which predicts that intrinsic reward should be proportional to compression progress. Computational models in neuroscience often treat intrinsic reward as resulting from the novelty of a stimulus. If intrinsic reward really does result from novelty, we would expect persistent high levels of dopamine in response to unpredictable noisy stimuli (as it remains novel from moment to moment). On the other hand, if intrinsic rewards encode compression progress, we would expect decreases in the level of dopamine as the predictive model becomes unable to learn anything more about the structure of the noise[3].

3.2.2. *Engagement and disengagement (and switching): norepinephrine*

Neurons of the locus coeruleus (LC), in the brainstem, are the sole source of norepinephrine (NE). NE is linked to arousal, uncertainty, vigilance, attention, motivation, and task-engagement. The LC-NE system is more traditionally thought to affect levels of arousal, but more recently has been implicated in optimization of behavioral performance (Usher et al., 1999; Aston-Jones and Cohen, 2005; Sara, 2009).

In that context, the activity of the LC-NE system can be understood as modulation of exploration-exploitation. The tonic differences in LC-NE response are associated with levels of arousal. Tonic NE response is correlated with task performance levels (Usher et al., 1999). Low tonic activity coincides with low attentiveness and alertness (Aston-Jones et al., 1991), while high tonic activity coincides with agitation and distractibility (Aston-Jones and Cohen, 2005). Good task performance coincides with an intermediate tonic level during which phasic bursts of activity are observed, while poor task performance due to distraction is associated with high tonic activity. In phasic mode during periods of intermediate tonic NE activity, NE is released in response to task-relevant events (Dayan and Yu, 2006). As suggested by Usher et al. and others (Usher et al., 1999; Aston-Jones and Cohen, 2005), the phasic modes might correspond to exploitation, whereas high tonic states of NE activity might correspond to exploration.

When it is beneficial for the agent to remain engaged in the current task, the tonic NE level stays moderate, and only relevant task stimuli will be salient. However, when it is not beneficial to remain engaged in the current task, the NE level raises and task-irrelevant stimuli become more salient. This drives the agent to distractibility, and task performance suffers. Attending to some distractor stimuli could have the effect of causing the agent to switch to another task in which this distractor becomes relevant, ostensibly with the purpose of exploring among available tasks (i.e., it "throws the ball in the air so another team can take it" Aston-Jones and Cohen, 2005).

In CD-MISFA, the agent's two internal-actions, (engage or disengage), and the reasons they are taken, links to the NE-driven task engagement/disengagement model. Boredom (low NE) indicates that a good representation already has been learned, leading to low estimation error, and thereby low potential intrinsic reward. Distractibility (high NE) indicates that the errors are too high, not decreasing quickly enough, or they cannot be reduced. In this case, it becomes valuable to disengage and find some other context, where learning may progress faster (or at all). When the agent has found a good context, the estimation errors decrease regularly, providing intrinsic reward that leads to a high value estimate (and a desire to remain engaged in that context).

3.3. FRONTAL CORTEX: VALUE FUNCTION AND REPRESENTATION SELECTION

The NE and DA neuromodulatory systems each have reciprocal connectivity with the prefrontal cortex—executive areas, which deal with cognitive aspects such as decision making, and top-down control of other functions, such as selective attention (Miller, 2000). If the LC-NE system is handling task-engagement and disengagement based on some value judgement, then this system needs to be controlled by another system that is estimating these values. The prefrontal cortex (PFC) plausibly plays a role in value estimation, and might use the utility information to provide top-down regulation of the activities of the LC neurons (Ishii et al., 2002).

PFC and nearby structures, specifically the anterior cingulate cortex (ACC) and the orbital frontal cortex (OFC), are implicated in value-based judgements. The ACC is involved in error detection (i.e., recognizing a prediction error) and estimating the costs of these errors (Bush et al., 2002). OFC is thought to be of import in motivational control of goal-directed behaviors (Rolls et al., 1996)—OFC damage leads to responses to objects which are no longer rewarding (Rolls et al., 1994; Meunier et al., 1997). The dorsolateral pre-frontal cortex (DLPF) is implicated in value-based working memory (Rao et al., 1997). Thus, these structures could possibly work together to estimate a value function, in the RL sense (Ishii et al., 2002).

Another important property of PFC is to maintain an appropriate task representation, i.e., imposing internal representations that guide subsequent performance, and switching these for another when it is no longer appropriate (Miller, 2000; Cohen et al., 2004). This property requires mechanisms to keep goal-relevant information (i.e., what should be considered salient and what should be considered a distractor) enabled in resonance with lower structures. Further, it requires a mechanism to maintain a context despite bottom-up disturbances, and a mechanism to switch the context. The PFC has connections from and to higher-order associative cortices, so it is in a good position to impose task-relevant representations from the top-down. Such "executive attention" enables memory representations to be "maintained in a highly accessible state in the presence of interference, and

[3] To our knowledge, this has not been tested yet.

these representation may reflect action plans, goal states or task-relevant stimuli in the environment (Kane and Engle, 2002)."

4. EXPERIMENTS AND RESULTS

4.1. SYNTHETIC SIGNALS

In other works, we have studied the types of representations uncovered by IncSFA, and their applicability (Kompella et al., 2012b; Luciw and Schmidhuber, 2012). The experiments here will focus moreso on the curiosity-driven behavior, especially in comparison to what the formal theory of curiosity predicts. We also explore the potential link of CD-MISFA to neuromodulatory task-switching—what quantities in our experimental results might be analogous to associated neuromodulators dopamine and norepinephrine?

CD-MISFA's typical behavior involves cycles of exploration, exploitation, and module storage. Exploration involves context switching, enabling accumulation of learning progress estimates about each context. The exploitation period has it settle into a single context where progress is easiest, until the representation is stored in long-term memory. Based on the formal theory of curiosity, we expect CD-MISFA to learn the representations in inverse order of their learning difficulty. Further, it will not waste time on anything unlearnable, corresponding to noise—which we note is novel and surprising in the traditional sense of Shannon et al. (1949), however, uninteresting since no learning progress can be made.

To this end, the first experiment involves a synthetic learning environment, with four types of *sources*—also known as *driving forces* Wiskott (2003). The simple driving forces are the fundamental "causes" of the complex observations. For example, an observation sequence given by an onboard camera of a mobile robot is "caused" by the robot's position, orientation, and camera angle. One cannot reconstruct the observations from the driving forces alone, of course, but tasks and rewarding conditions are often associated with the driving forces, and knowledge of the driving forces leads to useful (potentially rewarding) predictive power.

At any time the agent is experiencing one of five contexts. Two contexts are generated based on the same driving force, while the other three each have a different driving force. In **Figure 4A**, the 2 × 1000 (dimension by time steps) signal sources can be seen (S-A, S-B, S-C, S-D), ordered via learning difficulty, with the easiest signal at the top. The blue curve shows the first dimension, while the red dotted curve shows the second dimension. At the bottom, we have a highly non-stationary source, which changes irregularly, so as to be unlearnable to IncSFA. We want to hide each of these sources within a different high-dimensional process, albeit linearly, so that linear IncSFA will be able to extract them and it will take enough effort to do so. A high-dimensional observation is generated from a source by multiplication with one of four 400 × 2 matrices, which are randomly generated before each experiment. The 400 resulting values are rearranged into a 20 × 20 and value-normalized from zero to one to be pixel values for each image. Each input observation $\mathbf{x}(t)$ is an image of 20 × 20 pixels. In **Figure 4B**, one can see a few sample observations. The task for CD-MISFA is to extract all three learnable driving force signals from a single stream of high-dimensional observations.

Figure 4C shows the CD-MISFA agent's environment, which contains the five contexts (C1–C5; which can be considered states in the RL sense), and has two actions—stay (engage) or switch (disengage). Each time a context is entered, 100 steps of observations are fed to IncSFA. Each context has a local clock, so that the local time step will pick up where it left off if the agent returns from another. At the end of the 1000 time steps, the local time step resets[4].

4.1.1. Measuring learning difficulty

In order to test predictions of the Formal Theory of Curiosity, we need to analytically establish a definition of learning cost for slow features, by which we will measure the relative complexity of the signals within each context. We introduce here a measure denoted as Ω, to quantify the learning progress of IncSFA.

$$\Omega(\mathbf{x}) = \left[1 - \frac{\eta^{mca}(\lambda_{n-1} - \lambda_n)}{1 - \eta^{mca} - \eta^{mca}\lambda_n}\right] \qquad (15)$$

where λ_n, and λ_{n-1} are the eigenvalues corresponding to the lowest-order and second lowest-order (respectively) principal components in the whitened derivative space. We will discuss the origins of Ω further in Section 4.4, with full derivation.

In this experiment, the three learnable signals are quantified as $\Omega^A = 0.9933$ (for S-A), $\Omega^B = 0.9988$ (for S-B), and $\Omega^C = 0.9997$ (for S-C). They are quite close due to the similarities of the last and second to last eigenvalues in each distribution, however, there is a non-linear relationship between Ω and learning time. For S-A, about 2–3 epochs are needed. For S-B, about 15 epochs are needed. For S-C, about 40 epochs are needed.

4.1.2. Experiment setup

The experiment setup is as follows. Since there are 1000 different time-steps, we use 1000 states for the clustering. Thus, there will be 1000 different clusterers, each with maximum number of clusters set as 2. The estimator error is measured as an average of the estimation errors after each episode—an interaction with a single context of 100 time steps. The intrinsic reward estimates and policy are updated after each episode. Once the estimation error gets below the threshold 0.3, the module is frozen, and a new module created. The initial setting for ϵ-greedy is 0.6, which decreases via multiplication with 0.995 after every episode, and is reset when a new module is created. The learning rates for IncSFA: for CCIPCA, a $1/t$ learning rate is used, with amnesic parameter $l = 2$ (Weng et al., 2003), the MCA learning rate is a constant $\eta_{MCA} = 0.05$. We collected results over 25 different runs. Each run has a different initializations of all aspects, wherein CD-MISFA operates for time 3.5×10^5.

We note here some implementation details about the gating system. The gating system prevents corruption of the adapting IncSFA with samples from an already known/learned representation. This is implemented as a buffer that fills during each episode,

[4] We use "time step" in a local sense—it refers to one of 1000 steps that make up each context. We will use "time" to refer to global time (over all contexts).

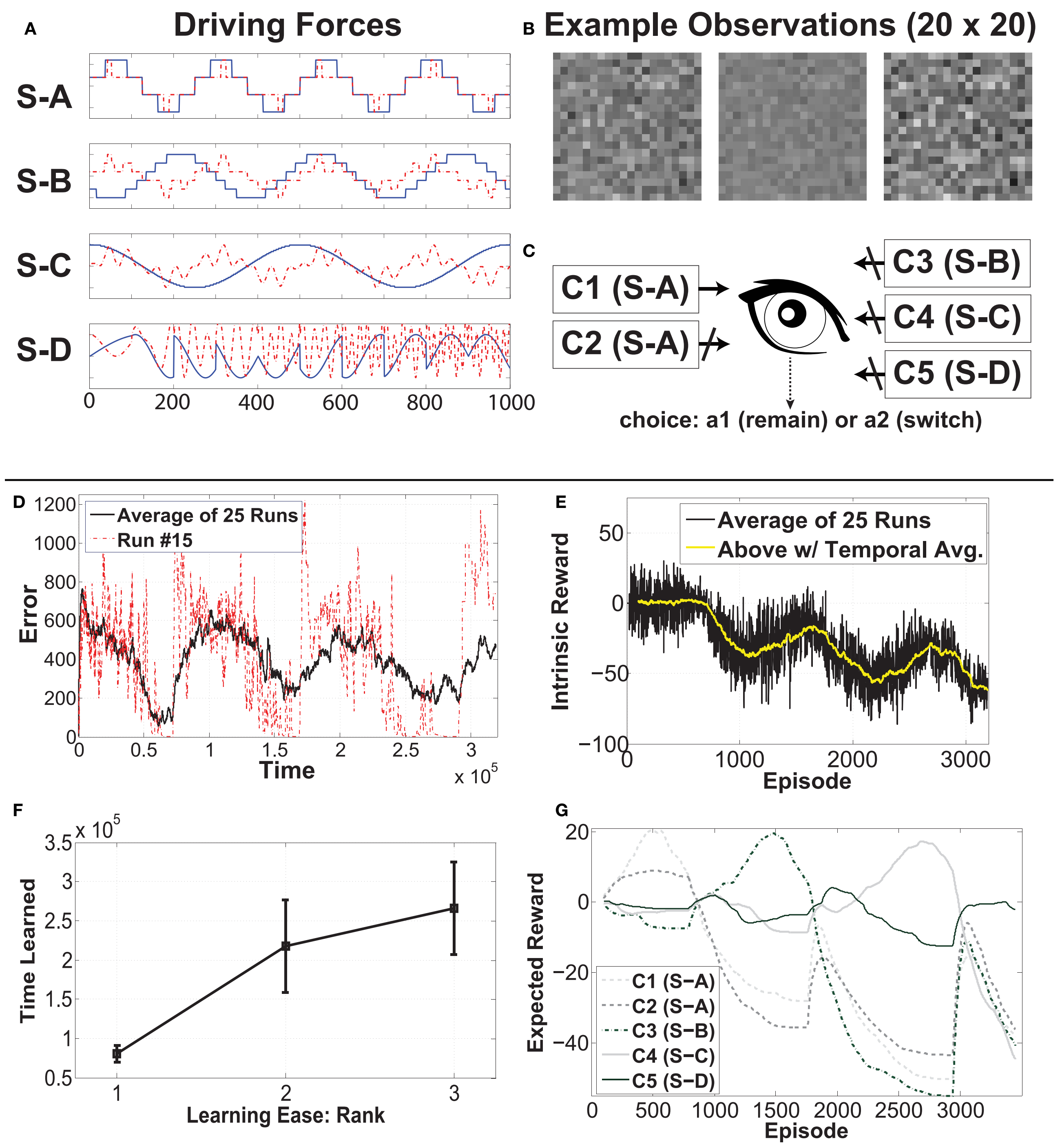


FIGURE 4 | Experimental setup and results of Curious Dr. MISFA with synthetic data. See text for details.

at the end of which the 100 observations are sent to all feature sets, from which the output is calculated. That output is then sent to the clusters in each SF output space, enabling error calculation for all modules. If the minimum module error is less than 0.3, the previous 100 samples are not used for learning, and a negative reward of −100 given. Otherwise, the samples are fed to IncSFA for learning. In this case, the intrinsic reward is calculated as the difference between the current estimation error of the adaptive learner and the same context's previously measured estimation error. The negative reward serves only to speed up the learning process. If it were removed, each run would simply take longer to complete.

4.1.3. Behavior

In all 25 runs behavior of CD-MISFA involved alternating phases of mostly exploration among all contexts, and exploitation once it settles on a context where it expects to make the most progress. We will call this exploitation-exploration process a cycle. Exploration is caused by the initial high amount of change in the adapting slow features, so that the estimator, which is on top of the slow features, cannot make progress. Once CD-MISFA remains within a context for enough time, the features become predictable enough so that an advantageous intrinsic reward can result. Due to ϵ-greedy, it continues to switch between contexts, allowing it to accumulate good estimates for all (the previous intrinsic reward accumulations are captured in the reward function). As ϵ decays, the policy converges to the simple but optimal strategy to disengage from all contexts except the easiest to learn new context.

4.1.4. Results

Results are shown in **Figures 4D–G**.

Part **(D)** shows the average cumulative estimator error (a single run is also plotted for perspective). In each cycle, the error starts high, then trends down as representations are learned; finally a module is created. Within each run, this is a rather noisy signal, as the agent jumps from context to context. The end of each run has only the unlearnable context remaining, so the error cannot reduce enough to store another module.

Part **(E)** shows the run-averaged and temporally-averaged (for smoothness) intrinsic reward. Each cycle (notably except the first) involves a rise and fall. Relatively low intrinsic reward that trends higher is associated with disengagement behavior. Relatively high intrinsic reward that trends lower is associated with engagement behavior. The high punishment for boring experiences within a learned context tends to drag the values down, moreso later in each run, when more representations have been learned. The first cycle seems to lack the typical rise, which we posit is due to the simplicity of the signal.

Part **(F)** shows average learning times and standard deviations for the three learnable signals. The ordering tends to be as predicted, but not always: A module for S-A emerges first all 25 times, S-B's module occurs second 18 times, and third 7 times, while S-A's module is mostly third (18/25). Due to the 7 runs when S-B and S-C were learned opposite as expected, the average learning time for S-B is higher than the average time when the second module is typically learned (as can be seen in **Figure 4D**), and the average learning time for S-C is lower than when the third module is learned.

Part **(G)** illustrates the reward function for run number 15, which is a fairly typical run. C1 and C2 are associated with initial rising reward. Once the shared source (S-A) is learned, both have their expectations of reward drop. We see C3 subsequently rise, followed by C4, then C5 (unlearnable).

4.1.5. On invariance

There are two independent dimensions to each source, which together generate the observations. The corresponding representation thereby also contains two parts. One part of the driving force is (trivially) invariant to the other part, and, after learning, the invariance property is observable at the representation outputs. For example, if (after learning) the first dimension of our source is held constant but the second allowed to change, then the observations will change, but the output of the first feature of the corresponding component will be constant, while the second changes. **Figure 5** illustrates this concept. As a real world example, consider place cells and head-direction cells. The output of the place cells are invariant to changes in orientation, and vice versa.

4.1.6. On neuromodulators

The estimation error profile observed in **Figure 4D** and associated behavior mirrors the findings regarding the LC-NE system and the "inverted U." High levels of estimation error correspond with predominantly disengagement and switching ("agitation"), while low levels of estimation error correspond with switching ("boredom"). There is a "sweet spot" of error, where the agent mostly engages in a single context. In this sweet spot, the intrinsic reward, representing learning progress, is at its relative peak. The intrinsic reward signal could link to dopamine, although, as we mentioned, there is no conclusive evidence about this.

4.2. EMERGENT REPRESENTATION FROM SENSORIMOTOR LOOPS—AN iCub EXPERIMENT

This experiment uses an embodied agent (iCub) with real high-dimensional images (grayscale 75 × 100), from the robot's eyes. There are two contexts here. In each, the iCub explores via random movement of its shoulder joint, causing the outstretched hand to eventually displace the single object in its field of view. It then observes the outcome while the hand continues to move. It is not given any prior knowledge about the objects, itself, or any concepts at all. It merely observes the pixel values, and uses CD-MISFA for learning and decision making. In one context, the object is a cup, which topples over upon contact with very predictable outcome. In the other, the object will roll in different

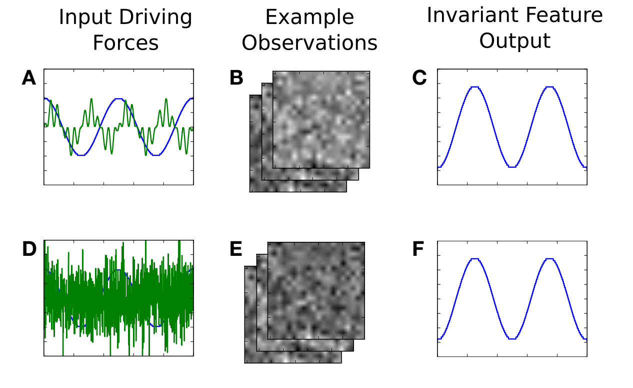


FIGURE 5 | An illustrative example of invariance, in the context of our synthetic signal experiment. A two dimensional driving force **(A)** generates high-dimensional observations **(B)**, from which IncSFA learns features that extract the original driving force. **(C)** The output of the first, slowest, feature. After learning, the second part of the driving force is replaced by noise **(D)**, causing different images **(E)**. However, the previously learned first feature output does not change **(F)**.

directions. About 70 episodes of image sequences were collected for each context. The eventual slow features, emergent from the holistic images, will code for the state of the objects.

Three example images from each of the two contexts are shown in **Figure 6A**. Each episode involves random exploration and an object-robot interaction event, and has between 50 and 250 images. We can say the "topple" context is easier to learn than the other, since the Ω value for the "topple" images is 0.9982, and the Ω value for the "push" images is 0.9988.

For the desired encoding to emerge requires careful setup, since IncSFA (and SFA, generally), applied to images with no pre-processing, is an appearance based vision technique (Turk and Pentland, 1991). To enable learning, we need to keep certain aspects of the images consistent. First, the robot's head is kept stable, so the image background doesn't noticeably shift. If the image shifts, it is possible the features would code for head position. Second, at the beginning of the episode, the object is always placed in the same position.

4.2.1. Setup

The joint angles were quantized into 20 distinct bins, yielding 20 states for each context, leading to 20 different clustering algorithms operating. Each clustering implementation had its maximum number of clusters set to 3. The estimation error threshold, below which the current module is saved and a new module is created, was set to 2.3. The initial ϵ-greedy value was 0.6, with a 0.93 decay multiplier. CCIPCA used learning rate $1/t$ with amnesic parameter 0.4, while the MCA learning rate was 0.01. CCIPCA did variable size dimension reduction by calculating how many eigenvalues would be needed to keep 98% of the input variance — typically this was between 10 and 15— so the 7500 pixels could be effectively reduced to only about 10 dimensions.

Unlike in the synthetic signals experiment, the slowest feature here encodes the context identity, which is to be expected when the input signals from widely different clusters; in a sense this is similar to a multiple rooms case (Mahadevan and Maggioni, 2007), where the features code for room ID. In order to prevent learning progress from continual switching, the following rule was implemented: when the agent decided to remain in its current context, it experienced two subsequent episodes, but when it decided to switch to the other, it only experienced one. In other words, the agent is given more time to learn by staying rather than by switching.

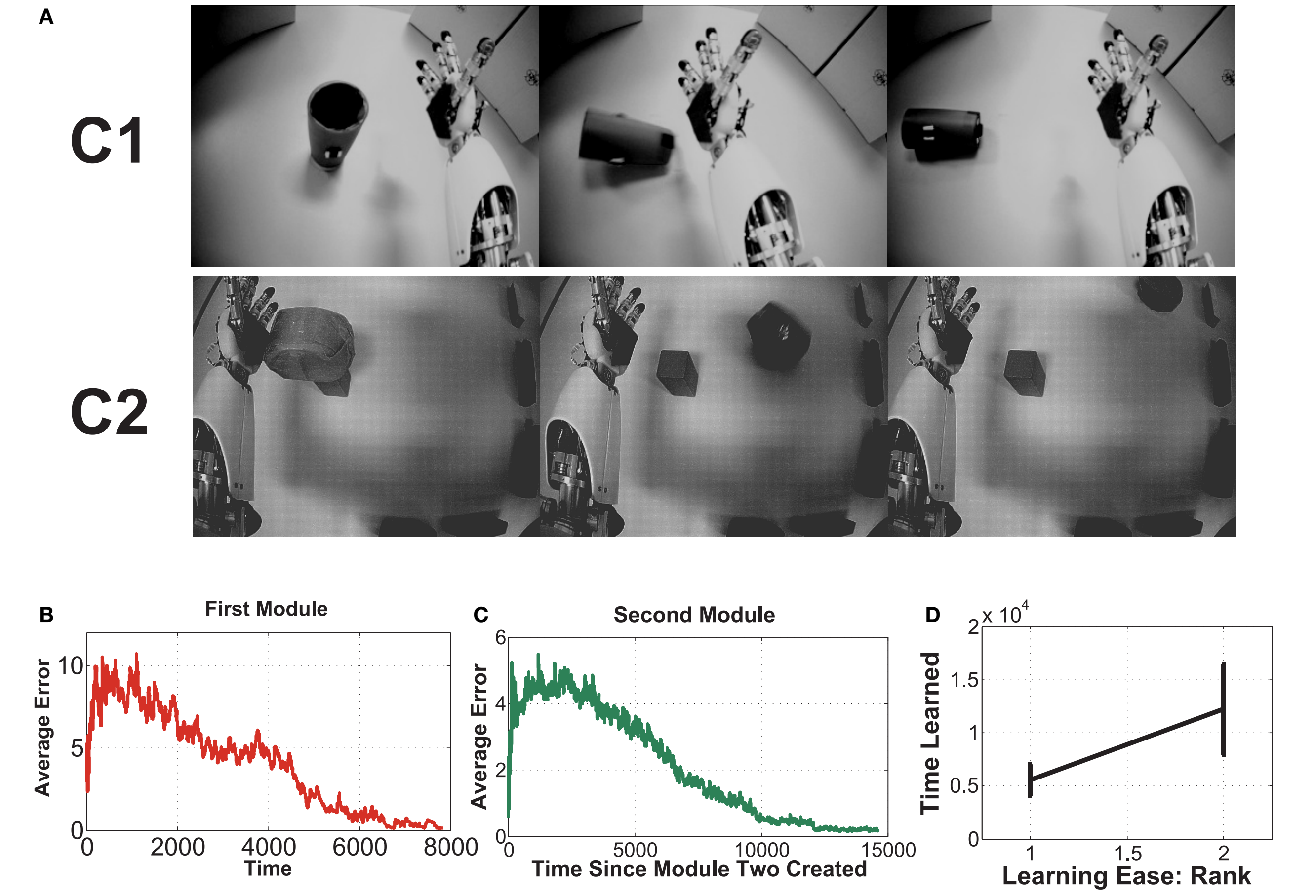


FIGURE 6 | This experiment uses image sequences from our iCub's cameras, while it moves its arm and interacts with objects. See text for details.

4.2.2. Results

Fifteen experimental runs were performed. **Figures 6B–D** show results. Part (**B**) shows the average estimation error during the first module's learning, while part (**C**) shows average estimation error for the second. Part (**B**) has a higher error, with more fluctuation than part (**C**), which mostly involves learning in a single context, since it will learn to quickly disengage away from the already learned context due to boredom punishment. In part (**D**), one can see the easier representation was indeed mostly learned first (in 14 of the 15 runs, this was the case).

Examples of the context-specific representations over time are shown in **Figure 7**. Both representations eventually encode whether the object is displaced or not. Most of the information in the image sequences can be broken down into three components: a baseline (the background), the object, and the arm. The object changes slower than the arm, so it is preferentially extracted by SFA. Moreover, the object-based features are invariant to the arm's position. Generalization is also possible, in a limited sense. If the arm were replaced by some other object (e.g., a stick), the feature output would not be perturbed. For more robust generalization, a better pre-processing is probably needed, as is typical with appearance-based vision techniques (Cui and Weng, 2000).

Once the features are learned, the feature output space creates a reduced-dimension state space for reinforcement learning techniques, if an external reward is in play. For examples, see our illustrative video of state-space simplification[5] Kompella et al. (2012a,b) for an example of using RL to maximize external reward upon the previously learned features.

4.2.3. On concepts

How do the learned representations relate to concepts? CD-MISFA could be the basis for something more substantial in the direction of concept learning, but, by itself, it is limited.

The representations learned by CD-MISFA correspond to compressed descriptions of image feeds, emerging from an eigen-decomposition of the covariance of temporally subsequent image differences. In some cases the resulting representations loosely resemble concepts, as when the slowest feature is shown to invariantly capture the state of some object in the images. But we are hesitant to explicitly refer to these representations as concepts, for a number of reasons. First, the notion of concept is itself up for debate. Arguments about what constitutes a concept will necessarily jump disciplinary boundaries, including philosophy, linguistics, and artificial intelligence. We do not wish to wade into this debate however, and we instead concern ourselves with the manner in which an agent or robot, starting with little prior knowledge, might direct its own behavior so as to increase what it knows about the world around it. Second, the types of representations learned by CD-MISFA are generally too low-level to be considered conceptual. For example, if CD-MISFA used intrinsic rewards to guide it to areas which enabled it to develop low-level feature detectors, such as edge detectors (which SFA can learn from a moving fovea Berkes and Wiskott, 2002), would we want refer to the edge detectors or the edges themselves as concepts? Likely not, despite the fact that it could develop from the same learning mechanisms that led to a representation for a toppling event.

[5] http://www.idsia.ch/~luciw/videos/IncSFAArm.mp4

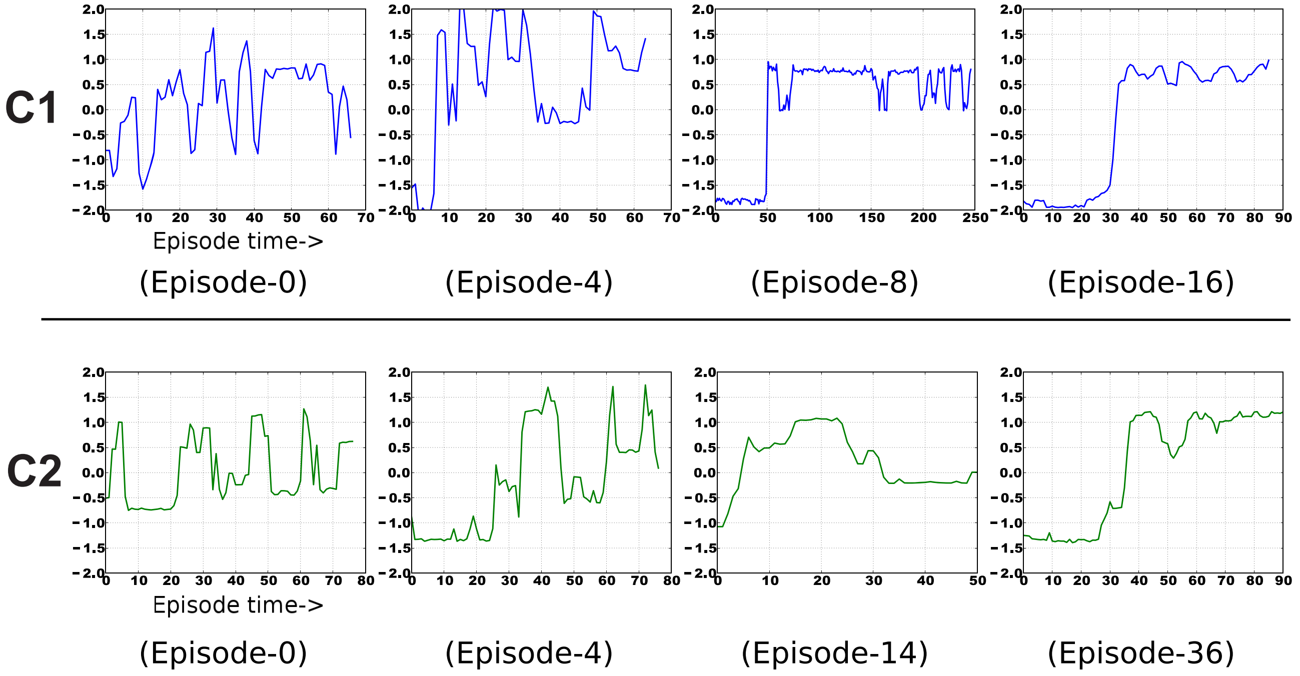


FIGURE 7 | Example emergence of object-centric slow features for both contexts. The final result encodes two states of each object—upright or displaced.

4.3. COMPARISON

Baldassarre et al. (2012) recently presented a biologically-constrained model of IM, which is also applicable to developmental robotics. Although the Baldassarre model (TBM) is more closely tied to neuroanatomical function than CD-MISFA, we argue that a number of functional and theoretical drawbacks of TBM make CD-MISFA a superior choice.

TBM was implemented on an iCub robot, and tested in an environment motivated by psychological studies, which includes a box with 3 buttons, 3 doors, and a series of lights. The robot can take 6 oculomotor actions (eyes fixating at either of the 3 buttons or 3 boxes) and 3 arm-motor actions ("reach and press," "reach and point," "reach and wave"). The IM reward function is modeled based on illumination change, considered as an automatically extracted salient event, and is a value that decays with recurrence of the salient event. In a learning phase, the model is allowed to explore by selecting any of its oculomotor/arm-motor actions, and observing the result (i.e., the opening of a door). In a test phase, external rewards are "hidden" inside one of the doors, and the goal of the agent becomes: press the correct button to retrieve the reward.

A primary drawback of TBM and its experimental validation is, although it makes use of IM, it is not clear to what extent (if any) IM is necessary for appropriate learning to occur. The model is not tested without an IM reward function, and in principle, the task undertaken would be learnable simply through *random exploration without any IM reward whatsoever*. Conversely, the role of IM in CD-MISFA and its associated experiments is essential, since if CD-MISFA is not presented with intrinsic reward, the model will not stay in any particular context long enough to learn the underlying representations. If CD-MISFA simply explores its environments in a random fashion, it is incapable of learning any meaningful representation.

A major advantage of CD-MISFA over TBM is the former's grounding in the Formal Theory of Fun and Creativity. Whereas the decay of the intrinsic reward value in TBM arbitrarily depends on the number of times the agent repeats a given action, CD-MISFA makes use of the more appropriate learning progress measure. In CD-MISFA, information ceases to be intrinsically rewarding as a function of how and when those visits lose informational value.

Lastly (and perhaps most importantly), TBM does not operate on realistic sensory/motor spaces. Whereas CD-MISFA explicitly shows how IM can operate in a model learning from high-dimensional input streams, and how action selection can operate on low-level motor outputs, TBM only shows how a model of IM can learn a small subset of predefined actions, operating on abstract representations of visual input.

4.4. QUANTIFYING THE LEARNING COST

We discuss here the measure denoted as Ω, which is used to quantify the learning cost of various types of signals for IncSFA. For simplicity, we consider here signals with similar input-variance but that have a different temporal structure. This assumption allows CCIPCA to approximately have a similar progress for the signals. Therefore, our focus remains here only on the progress of the CIMCA algorithm.

In an approach similar to the proof provided by Peng et al. (2007) for the convergence of MCA, we present here an analysis on quantifying the learning progress of the CIMCA algorithm. For the sake of simplicity, we just consider here only the first output component, but this can trivially be extended for higher output components.

The weight-update rule of CIMCA is given by:

$$\begin{aligned} w^{mca}(k) &= \left(1-\eta^{mca}\right)\, w^{mca}(k-1) \\ &\quad -\eta^{mca}\, (\mathbf{x}(k)\cdot w^{mca}(k-1))\, \mathbf{x}(k) \end{aligned} \tag{16}$$

$$w^{mca}(k) = w^{mca}(k)/\|w^{mca}(k)\| \tag{17}$$

To analyze the "average" dynamics of Equation 16, we reformulate it to a deterministic discrete time (DDT) system by taking the conditional expected value

$$E[w^{mca}(k+1)|w^{mca}(0), \mathbf{x}(i), i<k] \tag{18}$$

at each iteration:

$$\begin{aligned} w^{mca}(k) &= \left(1-\eta^{mca}\right)\, w^{mca}(k-1) \\ &\quad -\eta^{mca}\, E[\mathbf{x}(k)\mathbf{x}(k)^T]w^{mca}(k-1) \end{aligned} \tag{19}$$

Here, $E[\mathbf{x}\mathbf{x}^T]$ is the correlation matrix (R) of the input data ($\mathbf{x} \in \mathcal{R}^n$). The weight vector $w^{mca}(k)$ is shown to converge to minor component of input data Peng et al. (2007), if the following conditions are satisfied:

$$\begin{aligned} &\eta^{mca}\lambda_1 < 0.5, \quad ||w^{mca}(0)||^2 = 1, \\ &0 < \eta^{mca} \le 0.5, \quad w^{mca}(0)^T w^{mca*} \ne 0 \end{aligned} \tag{20}$$

where λ_1 is the largest eigenvalue of R, $w^{mca}(0)$ is the initial weight vector and w^{mca*} is the eigenvector with the smallest eigenvalue of R. Since the correlation matrix R is a symmetric non-negative definite matrix, it can be factorized into QDQ^{-1}, where Q is the eigenvector matrix (columns representing unit-eigenvectors v_i) and D is a diagonal matrix with corresponding eigenvalues (λ_i). In addition, the eigenvectors $\{v_i|i=1,2,\ldots,n\}$ form an orthonormal basis spanning $\mathcal{R}^n$. The weight vector w^{mca} can then be represented as

$$w^{mca}(k) = \sum_{i=1}^{n} a_i(k)v_i \tag{21}$$

where $a_i(k)$ are some constant coefficients.

Definition 1. *Given a stationary input distribution* $\mathbf{x} \in \mathcal{R}^n$ *and its eigendecomposition:* $\{v_i, \lambda_i\}$, $\forall i \in \{1,\ldots,n\}$, *where* v *denotes the set of eigenvectors and* λ *their corresponding eigenvalues (such that* $\lambda_1 > \cdots > \lambda_n \ge 0$*). Then, we define* $\Omega(\mathbf{x})$ *as a measure to indicate the learning progress of CIMCA for the input distribution* $\mathbf{x}$.

The following lemmas are useful to derive an analytical expression for Ω. Note that for all the following lemmas to hold true, the convergence conditions in (20) have to be satisfied.

Lemma 1. *Let V_i be denoted as*

$$V_i = \left[1 - \eta^{mca} - \eta^{mca}\lambda_i\right] \tag{22}$$

then,

$$a_i(k) = \frac{V_i^k a_i(0)}{\sqrt{\sum_j^n V_j^{2k} a_j^2(0)}}, \forall i \in \{1, 2, \ldots, n\} \tag{23}$$

Proof: We prove the result by mathematical induction.
$k = 1$: Substituting (21) in (19) for $k = 0$, we get

$$a_i(1) = V_i a_i(0), \quad \forall i \in \{1, 2, \ldots, n\}$$

At each update, the weight vector $w^{mca}(k)$ is normalized according to (17).

$$a_i(1) = \frac{V_i a_i(0)}{\sqrt{\sum_j^n V_j^2 a_j^2(0)}}, \quad \forall i \in \{1, 2, \ldots, n\} \tag{24}$$

Therefore, (23) is true for k=1.
$k = m$: Assuming the result to be true for some $k = m > 1$

$$a_i(m) = \frac{V_i^m a_i(0)}{\sqrt{\sum_j^n V_j^{2m} a_j^2(0)}}, \quad \forall i \in \{1, 2, \ldots, n\}$$

let P denote

$$P = \sqrt{\sum_j^n V_j^{2m} a_j^2(0)}$$

$k = m + 1$: Substituting (21) in (19) for $k = m$, we get

$$a_i(m+1) = V_i a_i(m) = \frac{V_i^{m+1} a_i(0)}{P} \tag{25}$$

Upon normalizing,

$$\begin{aligned} a_i(m+1) &= \frac{\frac{V_i^{m+1} a_i(0)}{P}}{\sqrt{\sum_j^n \frac{V_j^{2m+2} a_j^2(0)}{P^2}}} \\ &= \frac{V_i^{m+1} a_i(0)}{\sqrt{\sum_j^n V_j^{2m+2} a_j^2(0)}}, \quad \forall i \in \{1, 2, \ldots, n\} \end{aligned}$$

which is same as substituting $k = m + 1$ in (23). Therefore, by the principle of mathematical induction the result (23) is true for any $k > 1$. □

Lemma 2. *Let σ_i be denoted as*

$$\sigma_i = \left[1 - \frac{\eta^{mca}(\lambda_i - \lambda_n)}{1 - \eta^{mca} - \eta^{mca}\lambda_n}\right] \tag{26}$$

then,

$$0 < \sigma_1 < \cdots < \sigma_{n-1} < 1 \tag{27}$$

Proof: If we show that

$$0 < \frac{\eta^{mca}(\lambda_i - \lambda_n)}{1 - \eta^{mca} - \eta^{mca}\lambda_n} < 1 \tag{28}$$

then the condition (27) is straightforward.

We first prove the left inequality. Clearly, since $\lambda_1 > \cdots > \lambda_n \geq 0$ and $0 < \eta^{mca} \leq 0.5$, the numerator

$$\eta^{mca}(\lambda_i - \lambda_n) > 0, \ \forall i \in \{1, \ldots, n-1\} \tag{29}$$

and the denominator

$$\begin{aligned} 1 - \eta^{mca} - \eta^{mca}\lambda_n &> 1 - \eta^{mca} - \eta^{mca}\lambda_1 \\ &> 0.5 - \eta^{mca}\lambda_1, \quad \because \eta^{mca} < 0.5 \\ &> 0, \qquad \because \eta^{mca}\lambda_1 < 0.5 \end{aligned} \tag{30}$$

To prove the right inequality, it holds

$$\begin{aligned} \text{iff,} \quad \eta^{mca}(\lambda_i - \lambda_n) &< 1 - \eta^{mca} - \eta^{mca}\lambda_n \\ \text{iff,} \quad \eta^{mca}(\lambda_1 - \lambda_n) &< 1 - \eta^{mca} - \eta^{mca}\lambda_n \\ \text{iff,} \quad \eta^{mca}\lambda_1 &< 1 - \eta^{mca} \\ \text{iff,} \quad 0.5 &< 1 - \eta^{mca}, \quad \text{which is true} \end{aligned}$$

□

Lemma 3. *Let $C_i = \left[\frac{a_i(0)}{a_n(0)}\right]$ then,*

$$a_i(k) = C_i \sigma_i^k a_n(k), \quad \forall i \in \{1, \ldots, n-1\} \tag{31}$$

$$a_n(k) = \frac{1}{\sqrt{\sum_j^{n-1} \sigma_j^{2k} C_j^2 + 1}} \tag{32}$$

Proof: Using Equation (23) and the condition (30), we get

$$\begin{aligned} \frac{a_i(k+1)}{a_n(k+1)} &= \left[\frac{1 - \eta^{mca} - \eta^{mca}\lambda_i}{1 - \eta^{mca} - \eta^{mca}\lambda_n}\right] \cdot \left[\frac{a_i(k)}{a_n(k)}\right], \\ \forall i \in \{1, \ldots, n-1\} &= \left[1 - \frac{\eta^{mca}(\lambda_i - \lambda_n)}{1 - \eta^{mca} - \eta^{mca}\lambda_n}\right] \cdot \left[\frac{a_i(k)}{a_n(k)}\right] \\ &= \sigma_i \cdot \left[\frac{a_i(k)}{a_n(k)}\right] \\ &= \sigma_i^{k+1} \cdot \left[\frac{a_i(0)}{a_n(0)}\right] \end{aligned}$$

This implies,

$$a_i(k) = C_i \sigma_i^k a_n(k), \quad \forall i \in \{1, \ldots, n-1\}$$

Using the result from Lemma 1 and substituting for i = n, we get

$$\begin{aligned} a_n(k) &= \frac{V_n^k a_n(0)}{\sqrt{\sum_j^n V_j^{2k} a_j^2(0)}} \\ &= \frac{1}{\sqrt{\sum_j^{n-1} \left(\frac{V_j}{V_n}\right)^{2k} \left(\frac{a_j(0)}{a_n(0)}\right)^2 + 1}} \\ &= \frac{1}{\sqrt{\sum_j^{n-1} \sigma_j^{2k} C_j^2 + 1}} \end{aligned}$$

□

Lemma 3 gives an expression for each of the coefficients. Since $a_n(k)$ is bounded $(0 < a_n(k) < 1)$, $a_i(k)$'s $(\forall i \in \{1, \cdots, n-1\})$ belong to a family of exponential-decay functions: $C_i a_n(k) e^{-kln(1/\sigma_i)}$. Therefore,

$$\lim_{k \to \infty} a_i(k) = 0, \ \forall i \in 1, \ldots, n-1 \tag{33}$$

$$\lim_{k \to \infty} a_n(k) = 1 \tag{34}$$

Therefore, from (21) $w^{mca}(k)$ converges to the minor-component vector v_n.

Theorem 1. *Let* $\tau_i^{1/2}$ *denote the half-life period of* $a_i(k)$*, then the following inequality holds:*

$$\tau_1^{1/2} < \cdots < \tau_{n-1}^{1/2} \tag{35}$$

Proof: Since $a_n(k)$ is bounded $(0 < a_n(k) < 1)$, $a_i(k)$'s $(\forall i \in \{1, \ldots, n-1\})$ belong to a family of exponential-decay functions: $C_i a_n(k) e^{-kln(1/\sigma_i)}$. Half-life period $\tau_i^{1/2}$ is the time when the value $a_i(k)$ becomes equal to half its initial value. Therefore,

$$C_i a_n(k) \sigma_i^k = C_i a_n(0)/2$$

Using Lemma 3 and simplifying we get,

$$k = -\frac{ln(2)}{ln\sigma_i} + \frac{0.5}{ln\sigma_i} \times \frac{\sum_j^{n-1} \sigma_j^{2k} C_j^2 + 1}{\sum_j^{n-1} C_j^2 + 1} \tag{36}$$

Let us denote the term $\frac{\sum_j^{n-1} \sigma_j^{2k} C_j^2 + 1}{\sum_j^{n-1} C_j^2 + 1}$ as ξ. It is clearly evident by using Lemma 2 that for $k > 0$, $0 < \xi < 1$ and ξ is a monotonically decreasing function w.r.t k. However, for larger values of k and for consecutive σ_i's, ξ can be assumed to be a constant. Substituting the term ξ in Equation (36), we get

$$\begin{aligned} \tau_i^{1/2} &= -\frac{ln(2) - 0.5 * ln(\xi)}{ln\sigma_i} \\ &= \frac{ln(2) - 0.5 * ln(\xi)}{ln(1/\sigma_i)} \end{aligned} \tag{37}$$

Therefore, from Equation (37) and Lemma 2 we have,

$$\tau_{j-1}^{1/2} < \tau_j^{1/2}, \ \forall j \in \{2, \ldots, n-1\} \tag{38}$$

□

Theorem 1 gives the order in which the individual components $a_i(k)$ decay over time.

Theorem 2. *Given two input distributions* $\mathbf{x}^1, \mathbf{x}^2 \in \mathcal{R}^n$ *and the eigendecomposition of their corresponding expected correlation-matrix:* $\{v_i^1, \lambda_i^1\}$, $\{v_i^2, \lambda_i^2\}$, $\forall i \in \{1, \ldots, n\}$, *where* v *denotes the set of eigenvectors and* λ *their corresponding eigen-values* $(\lambda_1 > \cdots > \lambda_n \geq 0)$. *For an* η^{mca} *that satisfies,*

$$\eta^{mca} \lambda_1^1 < 0.5\,, \quad \eta^{mca} \lambda_1^2 < 0.5 \tag{39}$$

Then, the signal with a lower σ_{n-1} *will have quicker convergence and therefore quicker learning progress.*

Proof: From Theorem 1, it is clear that, the weight-vector $w^{mca}(k)$ converges to the minor component v_n when the penultimate coefficient $a_{n-1}(k)$ tends to 0. Therefore, a signal with lower σ_{n-1} will have a lower half-life period $\tau_{n-1}^{1/2}$ and hence the weight-vector $w^{mca}(k)$ converges quicker. □

Definition 2. *We therefore define* $\Omega(\mathbf{x})$ *as a measure to indicate the learning progress of CIMCA for an input-distribution* $\mathbf{x}$ *equal to* σ_{n-1}^{th} *value, that is,*

$$\Omega(\mathbf{x}) = \left[1 - \frac{\eta^{mca}(\lambda_{n-1} - \lambda_n)}{1 - \eta^{mca} - \eta^{mca}\lambda_n}\right] \tag{40}$$

5. CONCLUSIONS

A CD-MISFA agent autonomously explores multi-context environments. Compact context representations are learned from high-dimensional inputs through incremental slow feature analysis. Intrinsic rewards for measurable learning progress tell the agent which context is temporarily "interesting," and when to actively engage in/disengage from a context or task. Such mechanisms are necessary from a computational perspective, and biological systems have evolved methods of achieving similar functional roles. In particular, while cortical regions of the brain are involved in unsupervised learning from sensory data (among other things), neuromodulatory systems are responsible for providing intrinsic rewards through dopamine, and regulating levels of attention to allow for task engagement and disengagement through norepinephrine. As artificial and robotic agents become increasingly sophisticated, they will not only look to biological solutions for inspiration, but may begin to resemble those solutions simply through the pressure of computational constraints.

ACKNOWLEDGMENTS

We thank the IDSIAni for many stimulating discussions. This work was funded through the 7th framework program of the EU under grants #231722 (IM-CLeVeR project) and #270247 (NeuralDynamics project), and through SNF grant #138219 (Theory and Practice of Reinforcement Learning II).

REFERENCES

Aston-Jones, G., Chiang, C., and Alexinsky, T. (1991). Discharge of noradrenergic locus coeruleus neurons in behaving rats and monkeys suggests a role in vigilance. *Prog. Brain Res.* 88, 501–520.

Aston-Jones, G., and Cohen, J. (2005). An integrative theory of locus coeruleus-norepinephrine function: adaptive gain and optimal performance. *Annu. Rev. Neurosci.* 28, 403–450. doi: 10.1146/annurev.neuro.28.061604.135709

Baldassarre, G., Mannella, F., Fiore, V., Redgrave, P., Gurney, K., and Mirolli, M. (2012). Intrinsically motivated action-outcome learning and goal-based action recall: a system-level bio-constrained computational model. *Neural Netw.* 41, 168–187. doi: 10.1016/j.neunet.2012.09.015

Berkes, P., and Wiskott, L. (2002). "Applying slow feature analysis to image sequences yields a rich repertoire of complex cell properties," in *Procedings 12th International Conference on Artificial Neural Networks (ICANN)*, Madrid: Springer. doi: 10.1007/3-540-46084-5-14

Bush, G., Vogt, B., Holmes, J., Dale, A., Greve, D., Jenike, M., et al. (2002). Dorsal anterior cingulate cortex: a role in reward-based decision making. *Proc. Natl. Acad. Sci.* 99, 523–528. doi: 10.1073/pnas.012470999

Cohen, J., Aston-Jones, G., and Gilzenrat, M. (2004). "A systems-level perspective on attention and cognitive control: guided activation, adaptive gating, conflict monitoring, and exploitation vs. exploration, chapter 6," in *Cognitive*, ed M. I. Posner (New York, NY: Guilford Press), 71–90.

Cohen, J., and O' Reilly, R. (1996). "A preliminary theory of the interactions between prefrontal cortex and hippocampus that contribute to planning and prospective memory," *Prospective Memory: Theory and Applications*, eds M. Brandimonte, G. O. Einstein, and M. A. McDaniel (New Jersey, NJ: Erlbaum), 267–295.

Cui, Y., and Weng, J. (2000). Appearance-based hand sign recognition from intensity image sequences. *Comput. Vis. Image Underst.* 78, 157–176. doi: 10.1006/cviu.2000.0837

Dayan, P., and Abbott, L. (2001). *Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems.* MIT press.

Dayan, P., and Yu, A. (2006). Phasic norepinephrine: a neural interrupt signal for unexpected events. *Network* 17, 335–350. doi: 10.1080/09548980601004024

Franzius, M., Sprekeler, H., and Wiskott, L. (2007). Slowness and sparseness lead to place, head-direction, and spatial-view cells. *PLoS Comput. Biol.* 3:e166. doi: 10.1371/journal.pcbi.0030166

Fyhn, M., Hafting, T., Treves, A., Moser, M., and Moser, E. (2007). Hippocampal remapping and grid realignment in entorhinal cortex. *Nature* 446, 190–194. doi: 10.1038/nature05601

Guedalia, I., London, M., and Werman, M. (1999). An on-line agglomerative clustering method for nonstationary data. *Neural Comput.* 11, 521–540. doi: 10.1162/089976699300016755

Hafting, T., Fyhn, M., Molden, S., Moser, M., and Moser, E. (2005). Microstructure of a spatial map in the entorhinal cortex. *Nature* 801. doi: 10.1038/nature03721 /marginparQ: Volume

Ishii, S., Yoshida, W., and Yoshimoto, J. (2002). Control of exploitation–exploration meta-parameter in reinforcement learning. *Neural Netw.* 15, 665–687. doi: 10.1016/S0893-6080(02)00056-4

Jolliffe, I. (2005). *Principal Component Analysis*. New York, NY: Wiley Online Library.

Kakade, S., and Dayan, P. (2002). Dopamine: generalization and bonuses. *Neural Netw.* 15, 549–559. doi: 10.1016/S0893-6080(02)00048-5

Kane, M., and Engle, R. (2002). The role of prefrontal cortex in working-memory capacity, executive attention, and general fluid intelligence: an individual-differences perspective. *Psychon. Bull. Rev.* 9, 637–671. doi: 10.3758/BF03196323

Kaplan, F., and Oudeyer, P. (2007). In search of the neural circuits of intrinsic motivation. *Front. Neurosci.* 1:225. doi: 10.3389/neuro.01.1.1.017.2007

Kompella, V., Luciw, M., Stollenga, M., Pape, L., and Schmidhuber, J. (2012a). "Autonomous learning of abstractions using curiosity-driven modular incremental slow feature analysis," in *Proceedings. Joint International Conference Development and Learning and Epigenetic Robotics (ICDL-EPIROB-2012)*, San Diego, CA. doi: 10.1109/DevLrn.2012.6400829

Kompella, V. R., Luciw, M. D., and Schmidhuber, J. (2012b). Incremental slow feature analysis: adaptive low-complexity slow feature updating from high-dimensional input streams. *Neural Comput.* 24, 2994–3024. doi: 10.1162/NECO_a_00344

Kreyszig, E. (1988). *Advanced Engineering Mathematics*. New York, NY: Wiley.

Lagoudakis, M., and Parr, R. (2003). Least-squares policy iteration. *J. Mach. Learn. Res.* 4, 1107–1149.

Lange, S., and Riedmiller, M. (2010). "Deep learning of visual control policies," in *European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN)*, (Bruges, Belgium).

Lee, M., Meng, Q., and Chao, F. (2007). Staged competence learning in developmental robotics. *Adapt. Behav.* 15, 241–255. doi: 10.1177/1059712307082085

Lopes, M., and Oudeyer, P. (2012). "The strategic student approach for life-long exploration and learning," in *Proceedings of the 2012 IEEE Conference on Development and Learning and Epigenetic Robotics (ICDL-EPIROB-2012)*, San Diego.

Luciw, M., Kompella, V. R., and Schmidhuber, J. (2012). "Hierarchical incremental slow feature analysis," in *Workshop on Deep Hierarchies in Vision*, Vienna.

Luciw, M., and Schmidhuber, J. (2012). "Low complexity proto-value function learning from sensory observations with incremental slow feature analysis," in *Proceedings of the 22nd International Conference on Artificial Neural Networks (ICANN)*, Lausanne. doi: 10.1007/978-3-642-33266-1_35

Lungarella, M., Metta, G., Pfeifer, R., and Sandini, G. (2003). Developmental robotics: a survey. *Connect. Sci.* 15, 151–190. doi: 10.1080/09540090310001655110

Mahadevan, S., and Maggioni, M. (2007). Proto-value functions: a laplacian framework for learning representation and control in markov decision processes. *J. Mach. Learn. Res.* 8, 2169–2231.

Meunier, M., Bachevalier, J., and Mishkin, M. (1997). Effects of orbital frontal and anterior cingulate lesions on object and spatial memory in rhesus monkeys. *Neuropsychologia.* 35, 999–1015. doi: 10.1016/S0028-3932(97)00027-4

Miller, E. (2000). The prefrontal cortex and cognitive control. *Nat. Rev. Neurosci.* 1, 59–66. doi: 10.1038/35036228

Mugan, J., and Kuipers, B. (2012). Autonomous learning of high-level states and actions in continuous environments. *IEEE Trans. Auton. Mental Dev.* 4, 70–86.

Murase, H., and Nayar, S. (1995). Visual learning and recognition of 3-d objects from appearance. *Int. J. Comput. Vis.* 14, 5–24. doi: 10.1007/BF01421486

Ngo, H., Ring, M., and Schmidhuber, J. (2011). "Compression progress-based curiosity drive for developmental learning," in *Proceedings of the 2011 IEEE Conference on Development and Learning and Epigenetic Robotics (ICDL-EPIROB-2011)*, Frankfurt. doi: 10.3389/conf.fncom.2011.52.00003

Oja, E. (1992). Principal components, minor components, and linear neural networks. *Neural Netw.* 5, 927–935. doi: 10.1016/S0893-6080(05)80089-9

O'Keefe, J., and Dostrovsky, J. (1971). The hippocampus as a spatial map: preliminary evidence from unit activity in the freely-moving rat. *Brain Res.* 34, 171–175. doi: 10.3410/f.13284975.14644075

Pape, L., Gomez, F., Ring, M., and Schmidhuber, J. (2011). "Modular deep belief networks that do not forget," in *Proceedings of the 2011 International Joint Conference on Neural Networks (IJCNN)*, (San Jose, CA), 1191–1198. doi: 10.1109/IJCNN.2011.6033359

Peng, D., and Yi, Z. (2006). A new algorithm for sequential minor component analysis. *Int. J. Comput. Intell. Res.* 2, 207–215.

Peng, D., Yi, Z., and Luo, W. (2007). Convergence analysis of a simple minor component analysis algorithm. *Neural Netw.* 20, 842–850. doi: 10.1016/j.neunet.2007.07.001

Prince, C., Helder, N., and Hollich, G. (2005). "Ongoing emergence: a core concept in epigenetic robotics," in *Proceedings of the 5th International Workshop on Epigenetic Robotics: Modeling Cognitive Development in Robotic Systems*. Nara, Japan: Lund University Cognitive Studies.

Rao, S., Rainer, G., and Miller, E. (1997). Integration of what and where in the primate prefrontal cortex. *Science* 276, 821–824. doi: 10.1126/science.276.5313.821

Redgrave, P., and Gurney, K. (2006). The short-latency dopamine signal: a role in discovering novel actions? *Nat. Rev. Neurosci.* 7, 967–975. doi: 10.1038/nrn2022

Redgrave, P., Prescott, T., and Gurney, K. (1999). Is the short-latency dopamine response too short to signal reward error? *Trends Neurosci.* 22, 146–151.

Ring, M. (1994). *Continual Learning in Reinforcement Environments*. PhD thesis, University of Texas at Austin.

Rolls, E., Everitt, B., Roberts, A., Rolls, E., Everitt, B., and Roberts, A. (1996). The orbitofrontal cortex [and discussion]. *Philos. Trans. R. Soc. Lond. B Biol. Sci.* 351, 1433–1444. doi: 10.1098/rstb.1996.0128

Rolls, E., Hornak, J., Wade, D., and McGrath, J. (1994). Emotion-related learning in patients with social and emotional changes associated with frontal lobe damage. *J. Neurol. Neurosurg. Psychiatry*, 57, 1518–1524. doi: 10.1136/jnnp.57.12.1518

Sanger, T. (1989). Optimal unsupervised learning in a single-layer linear feedforward neural network. *Neural Netw.* 2, 459–473.

Sara, S. (2009). The locus coeruleus and noradrenergic modulation of cognition. *Nat. Rev. Neurosci.* 10, 211–223. doi: 10.1038/nrn2573

Schaal, S., and Atkeson, C. (1998). Constructive incremental learning from only local information. *Neural Comput.* 10, 2047–2084. doi: 10.1162/089976698300016963

Schmidhuber, J. (1991). "Curious model-building control systems," in *Proceedings of the International Joint Conference on Neural Networks, Singapore*, Vol. 2. (Seattle, WA: IEEE press), 1458–1463. doi: 10.1109/IJCNN.1991.170605

Schmidhuber, J. (1997). What's interesting? Technical Report IDSIA-35-97, IDSIA. Available online at: ftp://ftp.idsia.ch/pub/juergen/interest.ps.gz; extended abstract in Proc. Snowbird'98, Utah, 1998; see also Schmidhuber (2002).

Schmidhuber, J. (2002). "Exploring the predictable," in *Advances in Evolutionary Computing*, eds A. Ghosh and S. Tsuitsui (Springer), 579–612. doi: 10.1007/978-3-642-18965-4_23

Schmidhuber, J. (2006). Developmental robotics, optimal artificial curiosity, creativity, music, and the fine arts. *Connect. Sci.* 18, 173–187. doi: 10.1080/09540090600768658

Schmidhuber, J. (2010). Formal theory of creativity, fun, and intrinsic motivation (1990–2010). *IEEE Trans. Auton. Mental Dev.* 2, 230–247. doi: 10.1109/TAMD.2010.2056368

Schmidhuber, J. (2011). Powerplay: training an increasingly general problem solver by continually searching for the simplest still unsolvable problem. arXiv:1112.5309v1 [cs.AI].

Schultz, W. (1998). Predictive reward signal of dopamine neurons. *J. Neurophysiol.* 80, 1–27.

Schultz, W., Dayan, P., and Montague, P. (1997). A neural substrate of prediction and reward. *Science* 275, 1593–1599. doi: 10.1126/science.275.5306.1593

Shannon, C., Weaver, W., Blahut, R., and Hajek, B. (1949). *The Mathematical Theory of Communication*, Vol. 117. Urbana: University of Illinois press.

Sprekeler, H. (2011). On the relation of slow feature analysis and laplacian eigenmaps. *Neural Comput.* 23, 3287–3302. doi: 10.1162/NECO_a_00214

Srivastava, R. K., Steunebrink, B. R., and Schmidhuber, J. (2013). First Experiments with PowerPlay. *Neural Netw.* 41, 130–136. doi: 10.1016/j.neunet.2013.01.022

Sutton, R., and Barto, A. (1998). *Reinforcement Learning: An Introduction*. Cambridge, UK: MIT Press.

Sutton, R., Precup, D., and Singh, S. (1999). Between mdps and semi-mdps: a framework for temporal abstraction in reinforcement learning. *Artif. Intell.* 112, 181–211. doi: 10.1016/S0004-3702(99)00052-1

Taube, J., Muller, R., and Ranck, J. (1990). Head-direction cells recorded from the postsubiculum in freely moving rats. i. description and quantitative analysis. *J. Neurosci.* 10, 420.

Turk, M., and Pentland, A. (1991). "Face recognition using eigenfaces," in *Proceedings IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, (Cambridge, MA: MIT press), 586–591. doi: 10.1109/CVPR.1991.139758

Usher, M., Cohen, J., Servan-Schreiber, D., Rajkowski, J., and Aston-Jones, G. (1999). The role of locus coeruleus in the regulation of cognitive performance. *Science* 283, 549–554. doi: 10.1126/science.283.5401.549

Weng, J., McClelland, J., Pentland, A., Sporns, O., Stockman, I., Sur, M., et al. (2001). Autonomous mental development by robots and animals. *Science* 291, 599–600. doi: 10.1126/science.291.5504.599

Weng, J., Zhang, Y., and Hwang, W. (2003). Candid covariance-free incremental principal component analysis. *IEEE Trans. Pattern Anal. Mach. Intell.* 25, 1034–1040. doi: 10.1109/TPAMI.2003.1217609

Wiskott, L. (2003). Estimating driving forces of nonstationary time series with slow feature analysis. *arXiv.org* 1–8. Available online at: http://arxiv.org/abs/cond-mat/0312317

Wiskott, L., and Sejnowski, T. (2002). Slow feature analysis: unsupervised learning of invariances. *Neural Comput.* 14, 715–770. doi: 10.1162/089976602317318938

Zhang, D., Zhang, D., Chen, S., Tan, K., and Tan, K. (2005). Improving the robustness of online agglomerative clustering method based on kernel-induce distance measures. *Neural Process. Lett.* 21, 45–51. doi: 10.1007/s11063-004-2793-y

Zhang, Y., and Weng, J. (2001). *Convergence analysis of complementary candid incremental principal component analysis*. East Lansing, MI: Michigan State University.

Conflict of Interest Statement: 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.

Received: 02 December 2012; accepted: 08 May 2013; published online: 30 May 2013.

Citation: Luciw M, Kompella V, Kazerounian S and Schmidhuber J (2013) An intrinsic value system for developing multiple invariant representations with incremental slowness learning. Front. Neurorobot. **7***:9. doi: 10.3389/fnbot.2013.00009*

frontiers in
NEUROROBOTICS

ORIGINAL RESEARCH ARTICLE
published: 02 April 2013
doi: 10.3389/fnbot.2013.00006

Rare neural correlations implement robotic conditioning with delayed rewards and disturbances

Andrea Soltoggio*, Andre Lemme, Felix Reinhart and Jochen J. Steil

Faculty of Technology, Research Institute for Cognition and Robotics (CoR-Lab), Bielefeld University, Bielefeld, Germany

Edited by:
Jeffrey L. Krichmar, University of California Irvine, USA

Reviewed by:
Denis Sheynikhovich, Universite Pierre et Marie Curie, France
Eiji Uchibe, Okinawa Institute of Science and Technology, Japan

***Correspondence:**
Andrea Soltoggio, Faculty of Technology, Research Institute for Cognition and Robotics (CoR-Lab), Bielefeld University, Universitaet Strasse 25, 33615 Bielefeld, Germany.
e-mail: andrea@soltoggio.net

Neural conditioning associates cues and actions with following rewards. The environments in which robots operate, however, are pervaded by a variety of disturbing stimuli and uncertain timing. In particular, variable reward delays make it difficult to reconstruct which previous actions are responsible for following rewards. Such an uncertainty is handled by biological neural networks, but represents a challenge for computational models, suggesting the lack of a satisfactory theory for robotic neural conditioning. The present study demonstrates the use of rare neural correlations in making correct associations between rewards and previous cues or actions. Rare correlations are functional in selecting sparse synapses to be eligible for later weight updates if a reward occurs. The repetition of this process singles out the associating and reward-triggering pathways, and thereby copes with distal rewards. The neural network displays macro-level classical and operant conditioning, which is demonstrated in an interactive real-life human-robot interaction. The proposed mechanism models realistic conditioning in humans and animals and implements similar behaviors in neuro-robotic platforms.

Keywords: classical conditioning, instrumental conditioning, distal reward, robotics, neuromodulation

1. INTRODUCTION

In reward learning, the results of actions, manifested as rewards or punishments, occur often seconds after the actions that caused them. For this reason, it is not always easy to determine which previous stimuli and actions are causally associated with following rewards. This problem was named *distal reward problem* (Hull, 1943), or credit assignment problem (Sutton and Barto, 1998). This problem and the ability of animals to solve it emerged originally in behavioral psychology (Thorndike, 1911; Pavlov, 1927; Skinner, 1953). More generally, the distal reward problem can be seen as a particular instance of the broader ontological problem of discovering apparent cause-effect relationships in the external world. The ability of determining such relationships is distinctive of human and animal intelligence.

Such abilities were observed for example by Pavlov (1927), who induced a dog to believe that the ringing of a bell predicted the arrival of food. After conditioning, the ringing of the bell alone triggered salivation. Thorndike (1911) was also the first to describe how animals learn from experience which course of actions leads to best outcomes. Even organisms with relatively simple neural systems, like the marine mollusk Aplysia, are capable of associating neutral stimuli with following noxious stimuli in classical (Kandel and Tauc, 1965; Carew et al., 1981) and operant conditioning (Brembs et al., 2002). The capability of discovering relationships among stimuli, actions, and rewards in the world is therefore not a prerogative of human cognition, but it is also largely exploited in animal intelligence. Such a notion implies that relatively basic neural dynamics, as those of the Aplysia, can associate stimuli, actions, and reward across time and lead to what can be seen as a primordial version of temporal inductive inference (Osherson et al., 1990).

An important topic in neural computation is the understanding of how small neural networks discover relationships among events, even in the presence of interfering stimuli, or considerable time delays between cues, actions, and outcomes. One hypothesis that has gathered consensus in the last decade is that of *synaptic tagging* (Frey and Morris, 1997; Redondo and Morris, 2011) or *eligibility traces* (Wang et al., 2000; Sarkisov and Wang, 2008). The idea is that particular neural events, deriving for example from performing an action or perceiving a cue, leave slowly decaying traces in the network. The traces expire for unrelated and disturbing stimuli, but get promoted to long term synaptic changes when a reward follows. The utility of synaptic tags in the solution of the distal reward problem was shown in simulation in Päpper et al. (2011).

Conditioning occurs with the delivery of rewards or punishments in the form of pleasant or noxious stimuli. Reward signals were found to be mediated both in vertebrate and invertebrate organisms by neuromodulation (Carew et al., 1981; Hammer, 1993; Schultz et al., 1993; Menzel and Müller, 1996). The increasing evidence of the important role of neuromodulation in reward-driven learning led to the formulation of models of modulated plasticity with rate-based neurons (e.g., Montague et al., 1996; Alexander and Sporns, 2002; Sporns and Alexander, 2002; Ziemke and Thieme, 2002; Soltoggio et al., 2008; Soltoggio and Stanley, 2012), and with spiking neurons and modulated spike-timing-dependent-plasticity (STDP) (Soula et al., 2005; Farries and Fairhall, 2007; Florian, 2007; Legenstein et al., 2008; Potjans et al., 2009, 2011; Vasilaki et al., 2009). This evidence suggests that neuromodulation is both a biological (Schultz et al., 1993, 1997; Hasselmo, 1995) and a computational (Montague et al., 1996; Porr and Wörgötter, 2007) effective medium to convey reward information to a neural substrate. Neuromodulation, however, involves a variety of modulatory chemicals, which are observed to regulate a spectrum of neural functions, from arousal to attention, exploration, exploitation, memory consolidation, and other (Hasselmo, 1995; Marder and Thirumalai, 2002; Aston-Jones and Cohen, 2005). The implementation of such functions is investigated in a number of computational models (Fellous and Linster, 1998; Doya, 1999, 2002) and neural robotic controllers (Krichmar, 2008; Cox and Krichmar, 2009), in particular with focus on the role of neuromodulation in attention (Avery et al., 2012).

Relatively few studies focus on the particular neural mechanisms that bridge the temporal gap between sequences of cues, actions, and rewards (Izhikevich, 2007; Päpper et al., 2011; Soltoggio and Steil, 2013). In Izhikevich (2007), the precise spike-timing of neurons was indicated as the essential feature to perform classical and operant conditioning with modulated STDP. This position was challenged in a recent study (Soltoggio and Steil, 2013) in which the *rarity* of both correlating neural activity and eligibility traces was identified as the main feature that allowed for the solution of the distal reward problem also in rate-based models. The rarity of correlations was shown in simulation to be responsible for selecting rare neural events. Such events are then propagated further in time and enable weight updates if rewards occur.

The identification of the neural principles that solve the distal reward problem is fundamental in understanding how biological networks find relationships among stimuli and improve behavioral responses over time. Robots provide a realistic means for testing computational models that deal with similar timing and complexity of sensory information as those of living organisms. Cognitive developmental robotics (Asada et al., 2001), for example, is an area in which human feedback is used during learning. In such contexts, the asynchrony of flows of inputs and outputs implies that a learning neural network must cope with imprecise timing and unreliability of signals and actions. When people provide cues and feedback in a human-robot interaction, different operators, errors, and disturbances create a complex input-output pattern from which to extract correct relationships among stimuli and actions.

The principle of rare correlations, first introduced in Soltoggio and Steil (2013), is tested in the current study precisely in robotic scenarios in which learning is guided by human feedback. Classical and operant conditioning are tested in a setting in which a neural network serves as controller. Inputs from the robot cameras (the eyes) and tactile sensors (on the hands) are processed by a neural network, which in turn controls robotic actions like displaying a smiling expression, recognizing the tutor and learning to identify the correct color of objects. The learning is guided by the rewards given by the human participants, specifically the tutor, who interacts with the robot in a natural and spontaneous way, thereby affecting the robot perception with uncertain timing, delayed reward and disturbances. The successful achievement of conditioning and of behavior reversal proves the validity of the method to simulate realistic conditioning with the proposed neural model.

This paper is organized as follows. The principle of rare correlations and the plasticity mechanism are explained in Section 2. The robotic experimental settings, the conditioning problems and the details of the learning networks are illustrated in Section 3. The results, including both robotic runs and simulations, are presented in Section 4 and discussed in more detail in Section 5. The paper ends with concluding remarks in Section 6. An appendix provides further implementation details.

2. USING RARE CORRELATIONS TO SOLVE THE DISTAL REWARD PROBLEM

When a reward occurs, several previous cues and actions are, in general, equally likely to be the cause. One trial is therefore not enough to understand the correct relationship. When more trials are attempted with variable conditions, the responsible cues and actions will be invariant and always present, whereas the disturbing and unrelated cues and actions may change from trial to trial. How can a neural network discern, over multiple trials, which stimuli and actions lead to rewards, and which are instead unrelated? Secondly, how can the network make the association despite the temporal gap, or delay, between stimuli, actions, and rewards?

Eligibility traces (Wang et al., 2000; Sarkisov and Wang, 2008) or synaptic tags (Frey and Morris, 1997; Redondo and Morris, 2011) are synapse-specific values with relatively slow dynamics believed to express the *eligibility* of a specific synapse for later changes. The duration of traces must be at least as long as the delays between cues, actions, and rewards. A reward is generally conveyed by means of a modulatory signal (Montague et al., 1996; Farries and Fairhall, 2007; Florian, 2007; Porr and Wörgötter, 2007; Soltoggio et al., 2008; Pfeiffer et al., 2010). However, when rewards are delayed, the neural activity that caused such reward is not present anymore. When rewards are delayed, modulation cannot act on the current neural activity, because that may not be related to the present reward. In such cases, it makes sense that modulation multiplies the eligibility traces to give a weight update. Such a modulatory signal changes the synaptic weights of those synapses that are eligible, and leaves the other synapses unchanged (Izhikevich, 2007; Päpper et al., 2011; Soltoggio and Steil, 2013). One fundamental and open question in this approach is what rule promotes or downgrades synapses to be eligible or ineligible at any time. Izhikevich (2007) uses the precise spike-timing to create traces according to a traditional STDP rule. Alternatively, the principle of rare correlations (Soltoggio and Steil, 2013), also used in the present study, prescribes that spiking neurons are not necessary so long as traces express correlating events and are created parsimoniously. The fundamental aspects in the creation of traces is the maintenance of a low balance of traces with respect to the overall number of synapses. Those rare traces allow the network to isolate the reward-triggering synapses in a few trials. The decay time of traces is related to their production rate, in a way that longer-lasting traces can be maintained if the rate of production is further decreased. By means of this balance, rewards with longer delays can be correctly associated with previous cues and actions.

The principle is illustrated by the following example. Assume that in a relatively small network with 100,000 synapses, high activity across one single synapse σ triggers a reward. Such a reward, however, is delivered with a delay between 1 and 3 s. Assume that correlations between connected neurons across the whole network are 1%/s of the total number of synapses. Those correlations generate eligibility traces at the specific synapses. If the traces have a time constant of 1 s, they decay exponentially and are negligible after 3 s. Therefore, at any time, approximately 3,000 synapses are eligible (i.e., 3% of the total). When correlating activity across σ triggers a reward, which is conveyed as a modulatory signal to the whole network, the reward episode reinforces approximately 3,000 synapses (the eligible synapses). In other words, the synapse σ caused a reward, but because the network is not silent and because the reward is delayed, thousand of other synapses also carried correlating activity before the reward delivery. If σ carries correlating activity more times, and more rewards are delivered, each time approximately 3,000 random synapses are reinforced. Only σ, because is the reward-triggering synapse, is reinforced consistently. Other synapses that are reinforced consecutively by chance become fewer and fewer at each reward episode. The number of synapses that are reinforced twice consecutively is the 3% of 3%, i.e., 0.09%, or 90 synapses from a total of 100,000. After only four reward episodes, $0.03^4 = 0.0027\%$, i.e., three or fewer synapses have been reinforced consecutively. By the fifth reward episode, σ is likely to be the only synapse that was reinforced consistently. Thus, the use of rare correlations allows for a logarithmic-like search among noisy and spontaneous network activity where one single synapse among hundred of thousand triggers a reward. For more detail of this experiment, see (Soltoggio and Steil, 2013).

If correlations are not rare, e.g., 10%/s of the total or more, too many synapses are reinforced at each reward episode, causing some synapses to reach high values even when they are not triggering a reward. The rarer the correlations, the fewer are the unrelated synapses that are reinforced, and therefore the learning is more precisely targeted to the reward-triggering synapses. On the other hand, extremely rare correlations results in a network that selects synapses for reinforcement on a very sporadic basis, thereby resulting in a robust but slower learning.

The principle of rare correlations leads to the question of what rule can be used to extract them from the neural activity. The rarely correlating Hebbian plasticity (RCHP) was proposed in Soltoggio and Steil (2013) to address this question. This mechanism, described in detail in the next section, is employed for the first time in this study with a neuro-robotic experiment to learn associations of stimuli, actions, and rewards.

2.1. RARELY CORRELATING HEBBIAN PLASTICITY

The Rarely Correlating Hebbian Plasticity (RCHP) (Soltoggio and Steil, 2013) is a type of Hebbian plasticity that filters out the majority of correlations and produces non-zero values only for a small percentage of synapses. Rate-based neurons can use a Hebbian rule augmented with two thresholds to extract low percentages of correlations and decorrelations. The RCHP rule is expressed by

$$\mathrm{RCHP}_{ji}(t) = \begin{cases} +\alpha & \text{if } v_j\left(t - t_{pt}\right) \cdot v_i(t) > \theta_{hi} \\ +\beta & \text{if } v_j\left(t - t_{pt}\right) \cdot v_i(t) < \theta_{lo} \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where j and i are a presynaptic and a postsynaptic neuron, α and β two positive learning rates (in this study set to 0.1) for correlating and decorrelating synapses respectively, $v(t)$ is the neural output, t_{pt} is the propagation time of the signal from the presynaptic to the postsynaptic neuron, and θ_{hi} and θ_{lo} are the thresholds that detect highly correlating and highly decorrelating activities.

The rule expressed by equation (1) has two main features. The first is that the majority of neural activity does not correlate. Only a small percentage of synapses, determined by the thresholds θ_{hi} and θ_{lo}, has correlating values different from zero. This feature makes the RCHP different from a classical Hebbian rule in which all activity correlates along a continuous spectrum of values. A neural model that modulates classical Hebbian plasticity changes all synapses to a various extent because all synapses that carry non-zero activity are expected to correlate. Such an overall weight change can potentially wipe existing neural connections without reinforcing sufficiently those synapses that are responsible for a reward. On the contrary, the RCHP rule extracts a small percentage of synapses to be eligible for a weight update, leaving the majority of synapses unchanged and stable. A second feature of the RCHP rule is that detected correlations attempt to capture the cause-effect relationship of signal propagation across synapses. Similarly to STDP, when a high presynaptic activity value leads to a high postsynaptic activity value, the event is captured by the RCHP rule. In fact, the activity of the presynaptic neuron at time t is multiplied by the activity of the postsynaptic neuron at time $t + t_{pt}$, which is the time when the signal from the presynaptic neuron reaches the postsynaptic neuron. It is later explained that the propagation time and sampling time can be equivalent. In this way, the time window for detecting a correlation is effectively one time step.

The thresholds θ_{hi} and θ_{lo} are estimated online to target an average rate μ of approximately 0.5%/s of rare correlations. θ_{hi} and θ_{lo} are assigned initially arbitrary values of 0.1 and −0.1 respectively. A first-in first-out queue of correlations $cq(t)$ holds the number of correlations registered at each step during the recent past (in this implementation for the last 10 s). If the number of measured correlations during the last 10 s is higher than 5 times the target μ, i.e., higher than 2.5%, θ_{hi} is increased of a small step $\eta = 0.002$/s. If the correlations are too few, i.e., less than $1/5\,\mu$ (0.1%), the threshold is decreased of the same small step. The same procedure is applied to estimate θ_{lo}. It is important to note that such a procedure is an heuristic devised to implement a rudimentary homeostatic mechanism to extract rare correlations. The precise parameters used to implement the homeostasis are not particularly crucial as long as correlations are rare on average. In fact, the instantaneous rate of correlations and the long term dynamics vary considerably according to fluctuations of the neural activity, various input regimes, and weight changes. The self-tuning of the thresholds, as it is used in the present algorithm, is not meant to be a precise rule, but it is devised to ensure that, on average, only rare correlations are detected throughout the neural network. The large majority of synapses carry activity across neurons that do not correlate. A summary of the algorithm above is provided in the Appendix 6.

2.2. A NEURAL MODEL WITH ELIGIBILITY TRACES AND MODULATION

The RCHP rule acts on eligibility traces c_{ji} on each synapse between a presynaptic neuron j and a postsynaptic neuron i. A modulatory signal m, which is governed by a fast decay and by the exogenous input reward $r(t)$, converts eligibility traces to weight changes. The changes of the eligibility traces c_{ij}, weights w_{ij}, and modulation m are governed by

$$\dot{c}_{ji} = -c_{ji}/\tau_c + \mathrm{RCHP}_{ji}(t) \tag{2}$$

$$\dot{w}_{ji}(t) = m(t) \cdot c_{ji}(t) \tag{3}$$

$$\dot{m}(t) = -m(t)/\tau_m + \lambda \cdot r(t) + b. \tag{4}$$

where a reward episode at time t sets $r(t) = 1$, which increases the value of $m(t)$ proportionally to a constant λ. A baseline modulation b can be set to a small value and has the function of maintaining a small level of plasticity. The modulatory signal decays relatively quickly with a time constant $\tau_m = 1$ s, while traces have $\tau_c = 4$ s. The neural state u_i and output v_i of a neuron i are computed with a rate-based model expressed by

$$u_i(t) = \sum_j \left(w_{ji} \cdot v_j(t) \cdot \kappa_j\right) \tag{5}$$

$$v_i(t + \Delta t) = \begin{cases} \tanh(\gamma \cdot u_i(t)) + \xi_i(t) & \text{if } u_i \geq 0 \\ \xi_i(t) & \text{if } u_i < 0 \end{cases} \tag{6}$$

where w_{ji} is the connection weight from a presynaptic neuron j to a postsynaptic neuron i; κ_j is +1 and −5 for excitatory and inhibitory neurons respectively to reflect the stronger effect of less numerous inhibitory neurons; γ is a gain parameter; $\xi_i(t)$ is a uniform noise source drawn in the interval $[-0.1, 0.1]$. The sampling time is set to 200 ms, which is also assumed to be the propagation time t_{pt} [equation (1)] of signals among neurons. The values of all parameters are specified in Appendix 6. The architecture of the network with the inputs and outputs is outlined in the next section.

3. CONDITIONING IN A HUMAN-ROBOT INTERACTION

The principle of rare correlations is applied to a network model to perform classical and operant conditioning with the robotic platform iCub. The robot iCub and the hardware set-up are described in the following section. The classical and operant conditioning scenarios are illustrated in Sections 3.2 and 3.3. The learning networks with the inputs and outputs are described in Section 3.4.

3.1. THE ROBOTIC PLATFORM

The iCub is a child-sized humanoid robot of 90 cm of height, weighing 23 kg, and comprising 53° of freedom (Tsakarakis et al., 2007). **Figure 1** shows a rendered photo of the iCub interacting with people in the experimental environment. The robot facilitates human-robot interaction by means of haptic sensors in the hands, cameras, and its capability to display facial expressions. Expressions are produced by means of light-emitting diode arrays below the shell of the head. The position of the eye lids also add expressivity. In the current study, the facial expressions are limited to neutral, happy, and sad. Synthesized speech is produced via speakers mounted at the robot rack and it is used in the current scenario to provide additional feedback.

Cameras in the artificial eyes provide visual information of the surroundings. The visual input is used to detect people and objects in the room. In particular, markers are attached to people to make them easily identifiable (**Figure 1**). Additionally, object trackers signal the appearance of colored balls in the visual field of iCub. Additional details on the type and meaning of the inputs and outputs are explained in the following sections.

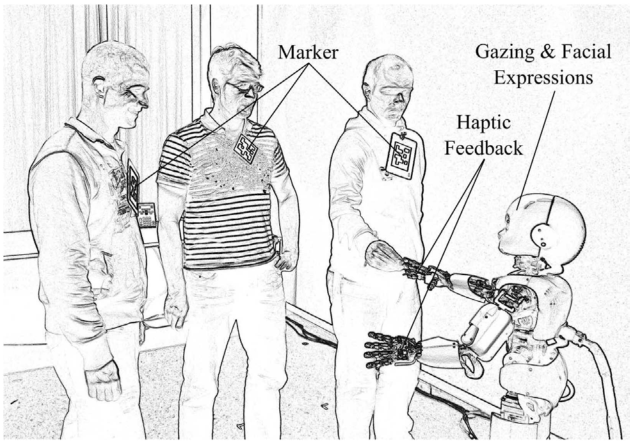


FIGURE 1 | The humanoid robot iCub in the experimental environment. The robot detects people in its field of view with the help of markers. Haptic sensing delivers rewarding or punishing signals to the learning networks. Gazing by means of head movements, speech output, and facial expressions provide feedback to the human participants.

3.2. LEARNING WHO IS THE TUTOR (CLASSICAL CONDITIONING)

This experimental scenario aims at testing the capability of the proposed network model to perform classical conditioning in a realistic human-robot interaction.

The robot monitors the environment moving his head and shifting its gaze over the room. This movement has the purpose of enlarging the field of view and endowing the iCub with a naturally looking behavior. The iCub is capable of recognizing different people identified by markers. Of all the people taking part in the experiment, one particular person is designated to be the tutor. The tutor is a person who takes care of the iCub, and signals that by conveying an haptic input with the touch of the iCub's hand. This signal represents an unconditioned stimulus that triggers an *innate*, i.e., pre-wired and fixed, positive reaction. Such a reaction corresponds also to a burst of modulatory activity as described in following sections. The haptic input can be interpreted as the delivery of food to Pavlov's dog. The iCub reacts to the unconditioned stimulus displaying a smiling face expression and saying positive sentences like "Thanks," or "I like it." The expression of a positive state, which follows an unconditioned stimulus, is always related to a burst of modulatory activity. While the iCub is constantly aware of a number of people in the room (as shown in **Figure 1**), from time to time the tutor enters the room and touches the hand of the iCub, thereby causing a positive smiling reaction.

In classical conditioning, if a stimulus predicts consistently the delivery of a reward, the learning process leads the agent (in this case the robot) to react immediately when the tutor enters the room, before any actual reward is given. The experiment in this scenario tests the learning capability of the proposed network model to associate a conditioned stimulus (CS) to a reward, also in the presence of a number of other disturbing stimuli.

3.3. LEARNING THE COLORS (OPERANT CONDITIONING)

A second scenario aims at testing operant conditioning, an experiment in which the iCub learns by trial and error to pronounce the correct word corresponding to the color of objects. The operant conditioning phase follows the classical conditioning only for practical reasons. When the iCub has learnt to recognize a tutor, it can easily follow his/her position and track colored objects. When the iCub detects a color object, it pronounces the name of a color. Initially, such an action is random because the iCub has no knowledge of which color corresponds to which name. If the color is correct, the tutor awards the iCub with a touch to the right hand, which delivers a reward to the network. If the iCub guesses the wrong color, the tutor ignores the answer and tries again after a few seconds. The cue (i.e., the colored object) and the action (i.e., the enunciation of a color) are not present anymore when the tutor gives the feedback. Thus, the neural mechanism that associates past actions with present rewards is tested in this scenario.

A scheme of the inputs and outputs in the robotic scenario is shown in **Figure 2**. The details of the learning network are explained in the next section.

3.4. THE LEARNING NETWORKS

The central controller comprises two neural networks, one for classical, and one for operant conditioning. The networks do not differ qualitatively because the modulated RCHP is capable of both operant and classical conditioning. However, due to the diverse type of inputs and outputs in the two tasks, the two networks represent effectively two separate areas of a neural system.

Each network has 800 excitatory neurons and 200 inhibitory neurons whose activity and outputs are governed by equations (2) and (3). Each neuron is connected to another neuron with probability 0.1. All excitatory neurons have plastic afferent connections that vary in the interval [0, 1] according to equation (3). Inhibitory neurons have fixed afferent connections. The network has therefore a random connectivity and random initial weights.

Figure 3 is a graphical representation of the two networks with the inputs and outputs. Each person-stimulus ($S1..S9$) is conveyed to the network by increasing the neuron state u by 10 for each neuron in a group of 60 randomly selected excitatory neurons ($G_{S1}..G_{S9}$). The activity of one group of neurons (G_{A0}), composed of 60 randomly selected excitatory neurons, triggers the conditioned response, i.e., it becomes active when the tutor is recognized after conditioning. The activity of a group is computed as the sum of the output of all neurons in the group, normalized by their number. Both networks receive a modulatory signal when the unconditioned stimulus is given by touching the iCub's hand. The haptic sensor conveys a modulatory signal that acts in the network as the signal m in equation (3).

Neurons in input groups do not receive connections from the rest of the network. Such a topology is devised in the current study to cope with real-world persistent and simultaneous input signals. In fact, as opposed to Izhikevich (2007) and Soltoggio and Steil (2013), in which stimuli were brief and impulse-like in nature, the network in the current experiments may receive continuous stimuli for long periods and simultaneously. Such input regimes, combined with Hebbian-driven growth of recurrent loops, might induce self-sustained activity, an unwanted regime in which neural dynamics do not respond to input anymore. This topology assumption prevents such a problem and is compatible with the role of input neurons.

The color trackers send inputs to the operant conditioning network. These binary signals are injected raw and unprocessed in the network through the groups of neurons $G_{S10..S14}$. As opposed to the classical conditioning network, which has only one output, the operant conditioning network has eight different outputs, corresponding to eight possible actions, i.e., the enunciation of the name of eight different colors. Neurons in the output groups do not project recurrent connections to the network. Such a topology is important to prevent that high neural activity generated by actions is feed unnecessarily back to the network. When a color-stimulus is present, the activity levels of the output groups are monitored for 1 s. If none of the groups reaches 30% of the maximum activity at the end of the waiting period of 1 s, many groups might have nearly equivalent levels of activity. In other words, when weights are low, the network may not be able to express a clear decision on what action to perform. To overcome this situation, the group with the highest activity, even by a small margin, triggers the action, which in turn increases the activity of its group and lower those of the other groups (u is increased/decreased by 10). This change

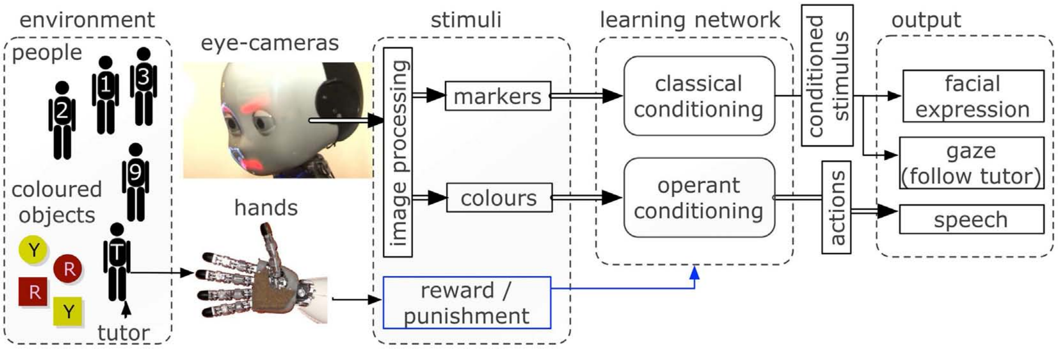


FIGURE 2 | The robotic software and hardware architectures used to interface the environment with the learning neural network. Inputs are captured via cameras and haptic sensors and processed to provide a vectorized representation to the network. The control network triggers facial expressions, gazing behavior, and speech output depending on the neural activity.

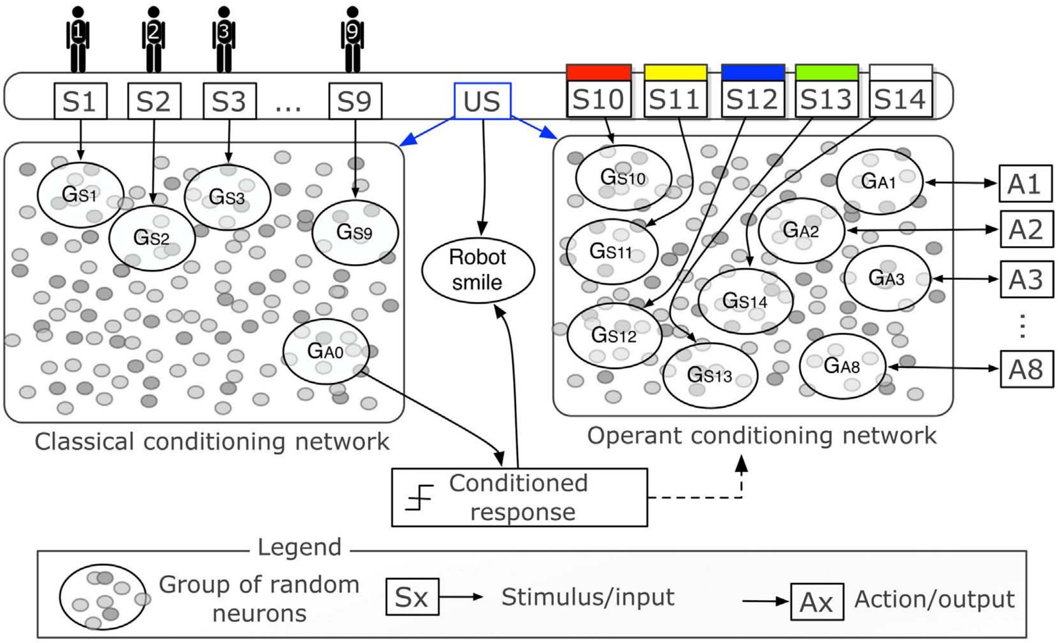


FIGURE 3 | Graphical representation of the control networks (expansion of central part of Figure 2). Two networks of 1,000 neurons each represent two distinct neural areas to perform classical and operant conditioning. The two networks differ only in the inputs and outputs, and in the initial random connectivity. The binary stimuli $S1..S9$ indicate the presence of different people in the visual field of the iCub and are delivered to their respective groups of random neurons $G_{S1}..G_{S9}$. The binary stimuli $S10..S14$ indicate the presence of objects of five different colors (all five colors were tested in simulation, only two, S10 and S11, with the real robot). The actions $A1..A8$ correspond to the enunciation of one particular color. The haptic sensor delivers a reward that represents the unconditioned stimulus (US). Both the US and high activity of G_{A0} cause the robot to smile.

in the neural activity is in effect an action-to-network feedback meant to inform the network of which action was performed. These dynamics are similar to winner-take-all policies (Kaski and Kohonen, 1994). In this way, the network can correlate correctly the input group with the action group that corresponds to the action performed.

The two networks are independent and can be tested independently. Nevertheless, the conditioned stimulus in classical conditioning, i.e., the tutor, is used to start the second learning phase that tests operant conditioning. When the group G_{A0} responds with high activity, signaling the presence of the tutor, the robot switches to operant conditioning with a probability 0.1/s. This behavioral sequence is not a central feature of the experiments but creates a natural interactive sequence of actions, which allows the participants and the tutor to observe both classical and operant conditioning taking place.

4. EXPERIMENTAL RESULTS

The experiments in this section test the learning capabilities of the control network both with the iCub robot and in simulation. The control network is simulated with the Matlab scripts provided as support material. The experiments were also video recorded. Both Matlab scripts and the illustrative video can be downloaded at the author's associate website http://andrea.soltoggio.net/icub. The robotic experiments require a real robot, or a robot simulator. The Matlab code can be also used as a stand-alone script with simulated input/output flow. The simulation without a real robot is used to test precisely controlled input-output regimes and timing which are difficult to achieve in a real-life human-robot interaction.

4.1. CLASSICAL CONDITIONING

The experiments in this section test the classical conditioning scenario previously described in Section 3.2. The experiments are conducted with the iCub. Further tests in simulation are also presented.

4.1.1. Real robot conditioning

The experiment was conducted by instructing nine people to approach the iCub and remain in its visual field for a random amount of time between a few seconds and approximately 1 min[1]. The participants did not follow a particular pattern in coming and leaving, and simply approached the robot, like visitors could do in an open exhibition, fair, or museum. Each person was uniquely identified by a marker as in **Figure 1** and corresponded to one stimulus in the range $S1..S9$. The participants could freely move in front of the robot and were not instructed to perform particular actions. The tutor also entered and left the robot's field of view at random times. As opposed to other people, the tutor also touched the iCub's hand each time he approached the robot, thereby delivering a reward. Such rewards were delivered at random times

[1] In effect, it is not easy to impose an exact time to people entering and exiting the iCub's field of view. The variability of such timing and overlapping of stimuli are characteristics of human-robot interactions.

by the tutor without a precise pattern. Other people, beside the tutor, could be present at the time of reward, making it difficult to establish the correct association between the tutor and the reward.

Over time, the pathway that connected G_{S*} (the neuron group that receives stimuli when the tutor is present) to the group G_{A0} grew consistently stronger. The pathways connecting the other groups G_{Sx} grew only marginally and not consistently as shown in **Figure 4A**. The growth of the pathway G_{S*} to G_{A0} led to an increased response of the group G_{A0} to the stimulus S^* as shown **Figure 4B**. While the stimulus S^* initially did not elicit a particular response, with time and more rewarding episodes, the network started responding with significant peaks in the activity when the stimulus S^* was perceived. Between the 7th and the 9th reward episode, and approximately after 20 min, the activity of G_{A0} presents distinct peaks in response to S^*. When the activity of the output group reached a preset threshold of 0.5, it caused a conditioned response. The response consisted in a smiling expression and a phrase like "Hello, it's nice to see you again," or "Hello, you are my friend." These sentences were so structured to manifest the conditioned response, representing effectively a reward prediction. As with the unconditioned response, the iCub smiled. The robot was also pre-programmed to follow the tutor's position with head movements to express clearly that the recognition had occurred.

Repeated experiments showed that the learning is manifested in three phases. An initial phase in which the tutor is not being recognized, an intermediate phase in which the tutor is recognized at times, or with a delay, and a final phase in which the tutor is recognized consistently and without delay. The intermediate phase is caused by the noisy fluctuations in the neural activity. When the pathway from G_{S*} to G_{A0} is not yet strong, such fluctuations result in inconsistent or delayed responses.

The activity of G_{A0}, after learning takes place, becomes a predictor of a reward delivery. The conditioning occurs despite two potential obstacles that derive from the real-life robotic scenario, and namely, (1) the noisy and unreliable perception of cues, and (2) the presence of many cues at the same time. In particular, the detection of markers is not 100% reliable for a number of reasons. Affecting the reliability of the detection are varying light conditions, different orientation of the markers due to the free movement and orientation of the participants, the obstruction of markers and noise in the camera. The slow decay of eligibility traces however ensures that the presence of a stimulus, in the present or in the immediate past, is represented at the synaptic level by the traces themselves. As a result, imprecise, unreliable, and noisy perception does not compromise the neural learning dynamics. The simultaneous presence of the reward-predicting stimulus and other disturbing stimuli is a potential obstacle in learning. **Figure 4C** shows that many stimuli are often present simultaneously. This situation induces occasional reinforcement of disturbing stimuli, as can be observed in **Figure 4A**. Nevertheless, the network reinforces consistently only the reward-predicting stimulus. **Figure 4D** shows the time of arrival of all nine stimuli and the correspondence of S^* with the intense network responses in **Figure 4C**. The experimental results in this section show that the control network, embedded within the robotic platform and exposed to human-robot interaction, modifies the connection weights to implement classical conditioning.

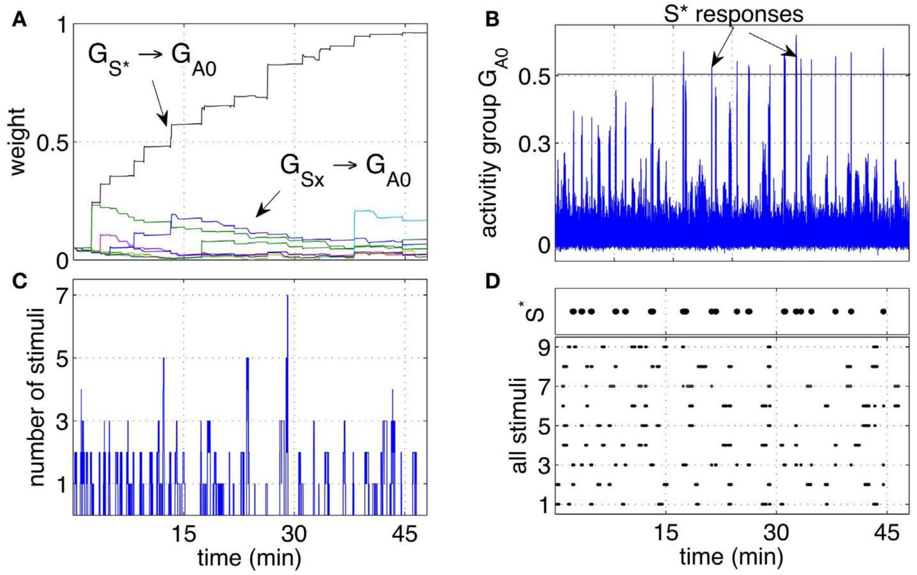


FIGURE 4 | Classical conditioning with disturbing stimuli. (A) The average strength of all weights connecting neurons in the groups $S1..S9$ to neurons in the output group G_{A0} are shown. The pathway $S^* \rightarrow G_{A0}$ grows to reach the saturation value. The other pathways remain at low values with only occasional increments. **(B)** The activity of the output group G_{A0} is characterized by increasingly high peaks. Those peaks are the learnt responses to the stimulus S^*. **(C)** Number of simultaneous stimuli present at any time. The plot shows that at times the network receives many stimuli simultaneously, making it difficult to detect which stimulus is causing a reward. **(D)** The presence of all the stimuli $S1..S9$ and S^* is plotted to show the correspondence of the arrival of S^* with the peaks of the neural activity **(B)**.

4.1.2. Simulated input/output flow

The previous experiment can be run as a stand-alone script in Matlab without the interface with the robot. In the simulated version, the signals representing the people are generated by means of a Poisson process that ensures random patterns in the sequence of stimuli. Thus, the experiments in this section eliminate possible bias in the pattern of appearance of people and tests rigorously the neural learning. The stand-alone experiments offer the possibility of reproducing the results with the provided Matlab scripts without a robot.

Each stimulus (representing one person) has a probability 0.15%/s of appearing, i.e., all stimuli are independent and may be present at any time. Once present, one stimulus lasts for a variable interval in the range [3, 30] s. As before, one particular stimulus $S^* \in (S1..S9)$ is designated to be the rewarding stimulus. When S^* is present, it causes a reward to be delivered in a random interval [0, 5] s. The simulation was run extensively for 2 h to test the stability of the learning, and to observe in particular that the pathways from the disturbing stimuli remained low. To assess further the robustness of learning, 10 independent runs were executed. **Figure 5A**, shows the statistical analysis of the pathways of all 10 independent runs. **Figures 5B–D** show respectively the weight changes, the number of stimuli and the network activity for one particular run. The results are qualitatively similar to the robotic experiment that was conducted with human subjects interacting with the robot. This indicates that differences in timing of the reward, duration, and frequency of stimuli between robot and simulation are not affecting the learning dynamics. It can be concluded that, as hypothesized, uncertain timing of the stimuli and variable delays are successfully processed by the neural network to discover the correct cue-reward sequence.

4.1.3. Delayed rewards after stimuli occurrence

In the previous experiments, the delivery of the reward occurs with a variable delay up to 5 s, but the causing stimulus S^* is likely to be present at the moment of reward delivery, except for the flickering and view obstruction of the marker. This fact derives from the intrinsic nature of the scenario in which a person is visible to the robot while pressing its hand (**Figure 1**). However, the capability of solving the distal reward problem is demonstrated when the reward occurs with a delay after the stimulus has ceased. This is the scenario in which, for example, a brief noise or sound predicts the delivery of the reward seconds later (e.g., the bell in Pavlov's experiment). To simulate this condition, in a variation of the original experiment, each stimulus remains present only for 1–2 s. The network receives a reward with a delay up to 5 s after the responsible stimulus has ceased. This experiment was run only in simulation. The equivalent version with the robot involves, for example, the recognition of a distinctive noise that predicts the arrival of each different participant.

Also in this scenario, the network learns to respond to the CS S^* despite S^* is not present anymore at the moment of reward delivery, and other disturbing stimuli may be present instead. Similarly to the previous experiment, throughout the simulation

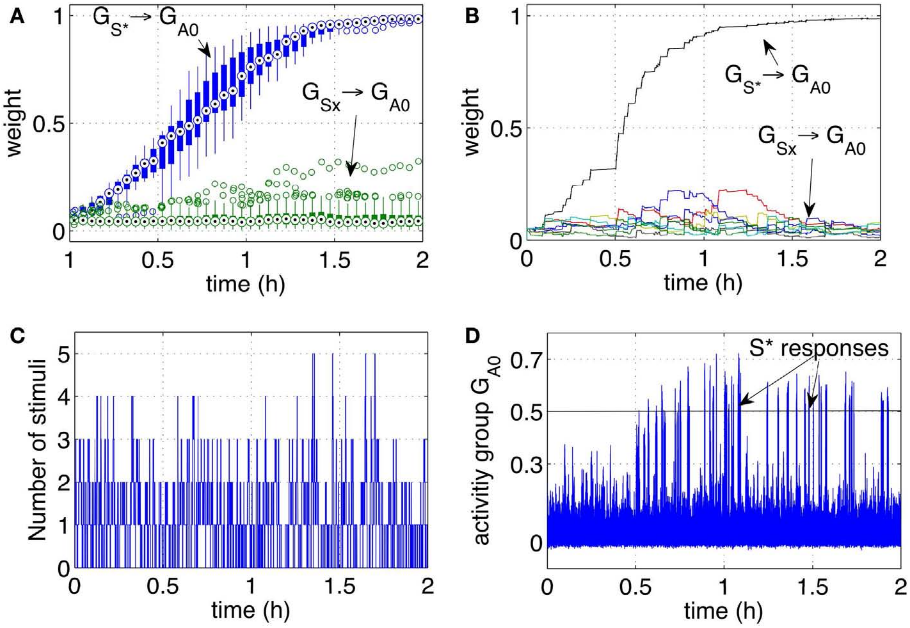


FIGURE 5 | Classical conditioning with disturbing stimuli and simulated input sequence. (A) The connection strengths of the pathways from the input groups $G_{S1..S9}$ to the output group G_{A0} are shown here. The statistics include a set of 10 experiments and are represented by box plots indicating the median (central point), 25th and 75th percentiles (thick lines), most extreme data points (thin lines), and outliers (circles) (McGill et al., 1978). The box plots are computed and drawn over 3-min intervals. The strength of the pathway $G_{S^*} \rightarrow G_{A0}$ (10 lines from 10 runs) grows consistently during the learning process and across all independent runs. The pathways from disturbing stimuli (box plots from eight lines for each run, i.e., 80 lines) remain at low values. **(B)** The strength of the pathways as in **(A)** are shown in one particular run. **(C)** Number of stimuli present at any time during one particular run. **(D)** The activity of the output group G_{A0} during one particular run. The network increases its responses as the stimulus S^* becomes progressively associated with the reward.

the response of the output group G_{A0} grows stronger. **Figure 6A** shows that the strength of the pathways from S^* to G_{A0} grows consistently in all the 10 independent simulations.

To observe the network behavior during a specific occurrence of the conditioned stimulus, **Figure 6B** shows the response of the output group to stimulus S^* before and after learning. The graphs show that a reward is delivered when the stimulus S^* is no longer present, and that disturbing stimuli may occur in between S^* and the reward delivery. While S^* initially does not elicit a response in the network, after learning, the neural activity of the neurons in the group G_{A0} is significantly higher than average. The peaks of activity in the right plot are a consequence of S^* and occur before the reward is actually delivered (right plots). Note that the activity alternates between high and low values due to the effect of inhibitory neurons.

4.1.4. The role of rare correlations and traces

The results in the previous sections showed robust learning dynamics in the classical conditioning scenario. How do rare correlations, eligibility traces, and delayed reward cooperate in the learning algorithm to achieve such a result?

This section looks at the small time-scale in which the weight changes occur. In particular, the neural dynamics are monitored and analyzed during a single cue-reward sequence. **Figure 7** shows the arrival of a stimulus S1 (first row). Such an event is registered by the network with an increase of correlating activity (second row). Such correlations are concentrated mainly on connections from the group G_{S1} and generate a significant increase of the eligibility traces of those synapses (third row). Those eligibility traces then decay with a time constant of 4 s. When a reward is delivered a few seconds later, it multiplies the traces to produce a net weight increment. Note that the presence of traces causes a very small decrement of the pathway (bottom plot) before the reward is delivered. This decrement is due to the small negative baseline modulation given by the term b in equation (4). This setting causes a pathway to decrease its strength if repeated stimuli are never followed by a reward. It is important to note that all synapses in the network are active and transmit signals at all times. Nevertheless, because correlations are rare, other synapses in the network are affected by minor changes, resulting in negligible variations of the weights. The robustness to disturbances is ensured by the principle that on average only the reward-predicting stimulus consistently creates traces that are later converted to weight changes. Other stimuli cause also correlations and generate traces, but their values are not converted to weight changes.

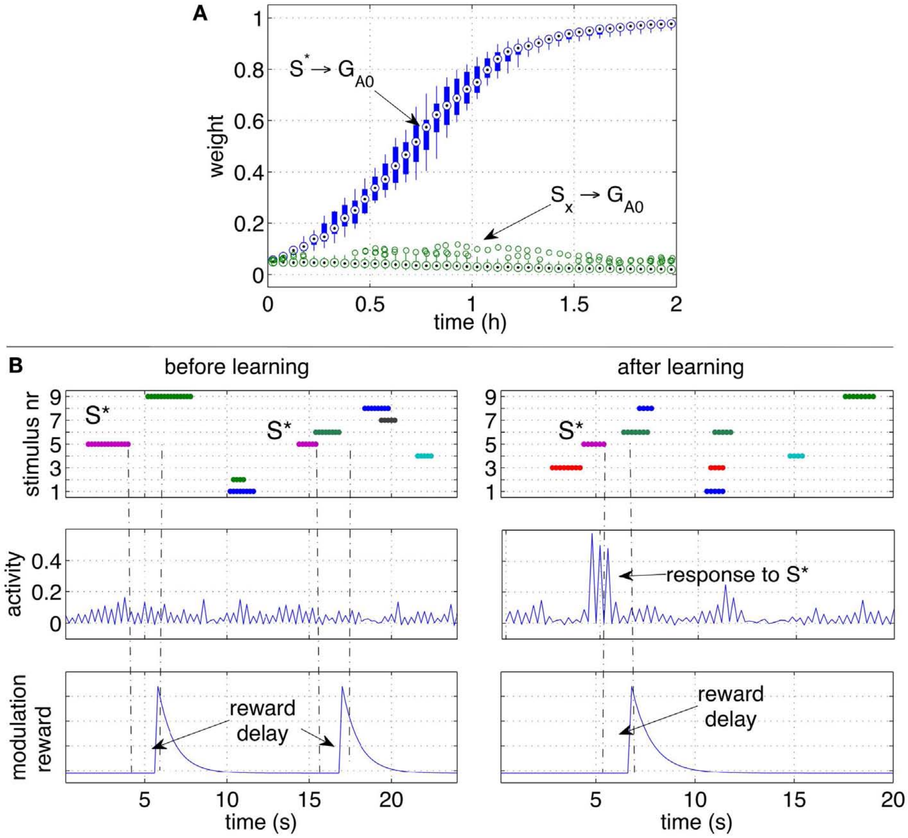


FIGURE 6 | Classical conditioning with delayed rewards after stimuli occurrence. (A) The strengths of the pathways are shown with box plots over a set of 10 independent runs. Similarly to **Figure 5A**, the pathway from the conditioned stimulus to the output group increases consistently, while the other pathways from the other stimuli remain at low values. **(B)** A close-up over a brief simulation interval during a particular run. The stimuli (top row), activity of group G_{A0} (middle row) and the modulatory signal (bottom row) are plotted before (left) and after (right) learning. While S^* does not elicit a response before learning, after learning S^* causes a clear increase of the activity of the output group before the reward is delivered.

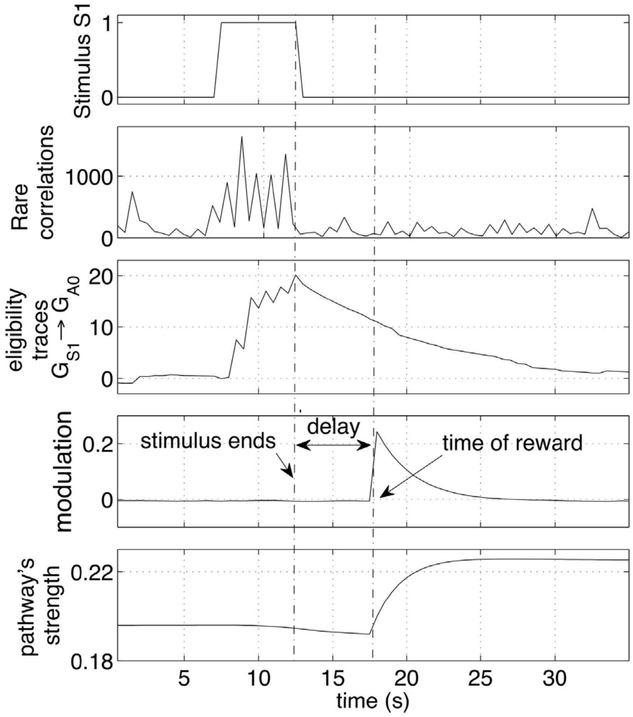


FIGURE 7 | Effect of a stimulus followed by a reward. The sequence of stimulus, rare correlations, eligibility traces, reward and weight increase is shown during a 30 s interval. The network preserve a memory of a recent stimulus by selectively creating eligibility traces along the pathways that transmit signals. The traces are later converted to weight increase if a reward is delivered. The strength of the pathway is represented as the average weight.

4.2. OPERANT CONDITIONING

Operant learning is triggered with probability 0.1/s when the iCub recognizes the tutor as a conditioned stimulus (CS) (after the robot was conditioned to recognize one person). At this point, the tutor presented different objects of different colors. Red and yellow colored objects were used with the robot. Up to five input colors were tested in simulation. Both real robot and simulation had eight actions available, i.e., eight output groups ($A1..A8$) triggered the enunciation of eight colors.

Once the iCub detected a colored object, it enunciated the name of a color. If the color pronounced by the iCub correspond to that of the object, the tutor touched the right hand of the iCub, thereby providing positive feedback. If the iCub answered by enunciating another color, the tutor ignored the answer and waited for the next trial. Between each trial, the tutor waited a random amount of time, generally varying between 5 and 20 s. On an average trial, between a correct answer and the time the tutor touched the hand, a time between 1 and 3 s elapsed.

Initially the robot displayed an exploratory behavior. The exploration is due to neural noise and to the fact that none of the pathways is significantly stronger than the others. During the exploratory phase, the iCub answered with different colors each time the same object was presented, occasionally repeating the same color. The robot switched to choosing the correct answer after a few correct guesses. A higher level of reward, or a longer touch to the iCub's hand, could be used to achieve a one-shot learning in which one single positive reward episode led to the repetition of that action, i.e., no further exploration. **Figure 8A** shows the strengths of the pathways from the two inputs $S10$ and $S11$ (representing two colors) to the actions (representing the enunciation of those colors). Each reward episode was caused by pressure on the iCub's arm causing $r(t)$ to be 1 during the touch.

In a variation of this experiment, the tutor could induce a small negative reward $[r(t) = -0.5]$ by touching the left hand of the robot whenever a wrong answer was given. When that happened, the corresponding pathway registered a reduction in strength. At the next trial, the previous erroneous choice was therefore less likely to be selected, because the other pathways were stronger. These dynamics resulted in a faster exploration in which colors were not randomly selected: colors that resulted in negative reward were less likely to be named subsequently. The data from this experiment is not shown, but the simulated version described following adopts a similar rewarding policy.

The experiment with the iCub was extended in simulation to include five different colored objects ($S10..S14$). The automated process produced one stimulus (corresponding to one colored object) every 20 s. Every stimulus was presented sequentially and circularly, i.e., in the sequence 1, 2, 3, 4, 5, 1, 2, ..., etc. If the answer was correct, a reward $r(t) = 5$ was given with a delay in the interval [0, 5] s, otherwise a small negative reward $[r(t) = -0.5)]$was given. The weights of the pathways, statistically analyzed over 10 independent runs, are presented in **Figure 8B**. The plot indicates that within 30 min of simulated time, all objects during all runs were correctly associated with their respective colors.

It is important to note that the amount of weight increase depends on how much time elapses between the action and the reward. In the current study, exponentially decaying traces [equation (2)] were employed, making the trace decay over time as e^{-t}. Because the modulation $m(t)$ multiplies the traces to achieve a weight increment [equation (3)], the weight increase is also related to such a decay.

Interestingly, several tests showed that the answers became reliable when one pathway became approximately 20% stronger than the other pathways (measure only visually estimated). For smaller differences, stronger pathways were still more likely to drive the output, but the neural noise and random fluctuations in the neural activity meant that weaker pathways could at times prevail. When one pathway became at least 20% stronger than the others, the answer became reliable. Any further increase of such a pathway did not appear to manifest in a behavioral change. However, each increase in the rewarded pathways represents in effect a further consolidation of a behavior, which can be seen as a *belief* that stimulus S10, for example, is the color "red." It can be inferred that in the phase of exploitation, the strength of the strongest pathway is an index of how *sure* or *confident* the robot is that the answer is the correct one. Although two or three correct and rewarded answers were sufficient to establish an immediate correct behavior, further trials

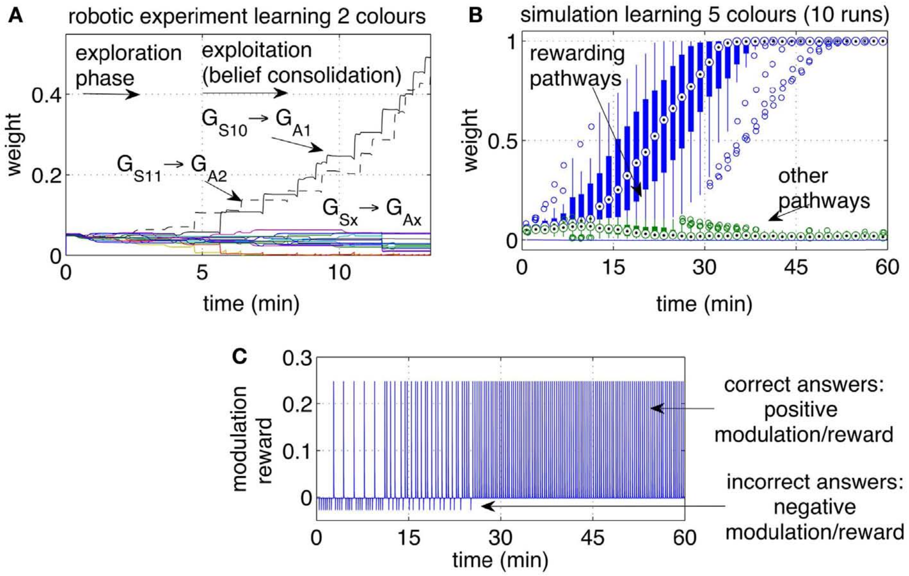


FIGURE 8 | Operant conditioning: learning stimulus-action associations. (A) Example of one run with the robot learning two colors. The weights of the rewarding pathways ($G_{S10} \rightarrow G_{A1}$ and $G_{S11} \rightarrow G_{A2}$) and from the same inputs to all outputs are shown. After a period of exploration, the robot discovered the rewarding actions and repeated them consistently, causing the corresponding pathways to grow, while the other pathways remained at low values. **(B)** Weight dynamics during the simulated learning of five colors. The statistical analysis includes 10 independent runs. All correct associations (five colors for each of the 10 runs) were discovered within 30 min. All pathways that did not lead to a reward, i.e., the incorrect associations, remained at low values. **(C)** Modulation while learning five colors. Initially the modulation signal is negative due to negative rewards, a consequence of wrong answers. With time, each of the five color is associated with the correct response. When all answers are correct, all rewarding episodes become positive.

provided confirmation, resulting in what can be named as *belief consolidation*. The effect of the weight strengths on behavioral properties such as exploration, exploitation, and belief consolidation is further investigated in the next section on behavior reversal.

As it is mentioned above, the operant conditioning phase was started conventionally by the recognition of the tutor. Nevertheless, the pathways in the network to the right of **Figure 3**, i.e., those that learn the colors, are learnt independently of the classical conditioning experiment. Once the colors are learnt, a new person may be introduced to the iCub as a new tutor. The iCub will be able to answer correctly to the new person because the recognition of the tutor is independent from the object-color associations.

4.3. BEHAVIOR REVERSAL

In the previous section it was mentioned that the tutor could provide a negative reward touching the left hand of the robot. In effect, a negative reward (negative modulation signal) can be interpreted as a punishment. In this section, the use of punishment to implement behavioral reversal is tested.

In this new experiment, the tutor conditioned the iCub to learn one association between one color and the name of a color, as it was also done in the previous experiment. After the association was established, the tutor attempted to reverse this association by providing negative feedback. Each time the iCub was presented with the yellow object, and responded "yellow," the tutor gave a punishment touching the left hand. A punishment was set to be equivalent to a reward but with opposite sign. The purpose of the tutor was to remove the previous association in favor of a new one. In this particular case, a whimsical tutor attempted to cancel the correct association "yellow" in favor of the enunciation "orange."

Figure 9 shows the pathways from the group G_{S10} to the action groups. The graph shows the same initial phases of exploration and exploitation as in **Figure 8**. When the tutor starts giving negative feedback (marked in the graph with *policy switch*), the weights of the yellow-pathway decrease progressively. The reversal of the previously acquired behavior is gradual. The amount of negative modulation was in effect equal to the amount of positive modulation. Each punishment resulted in a decrement of the pathway comparable to the increment that was previously obtained by one rewarding episode. If the robot was previously rewarded many times and had established a strong association between a cue and one action, it was consequently more adamant to changes. As anticipated, it can be said that the strength of a pathway reflects a level of *belief*. A strong pathway, reflecting a strong belief, also resulted in a robust behavior in front of false or misleading, but occasional, input cues. Even if the robot received a punishment from a correct answer, for example due to an error or a whim of the tutor, the single episode did not reverse the robot belief unless the tutor insisted on the new policy.

The repeated punishments led the network to reduce progressively the difference in weights among the pathways. When all pathways reached similar values, the answers started to vary among colors, i.e., the robot resumed an exploratory phase. A new association was now possible. When the robot, seeing a yellow object, pronounced the correct color (orange, according to the new tutor's

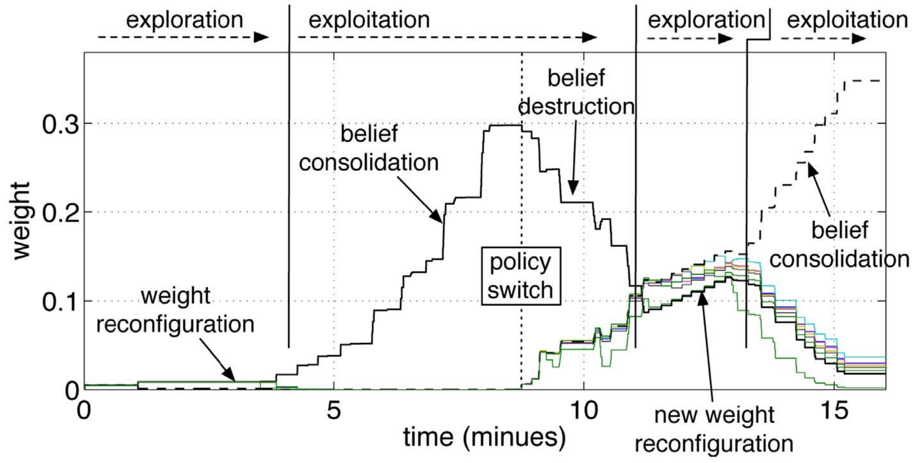


FIGURE 9 | Operant conditioning and behavior reversal. The pathways from the input group G_{S10} to the action groups $G_{A1}..G_{A9}$ are shown. As in the previous experiment, the robot attempts to guess the color during an initial exploratory phase. When the robot guesses the color correctly, and the tutor rewards it, the correct stimulus-to-action pathway is reinforced. Subsequently, the tutor changes his policy and gives a punishment if the robot perseverates with the previous answer. This leads to a convergence of all weights and to forgetting the previous association. A new exploratory phase then starts, which results in a new exploitation once the new correct color is guessed and the tutor rewards the robot.

policy), the tutor gave rewards and led the robot to build the new association, as reflected by the growth shown in **Figure 9** at the end of the experiment. The length of time that is necessary to achieve the behavior reversal depends on the strength of the pathway (also indicating the strength of the belief) and the plasticity rate. Strong pathways and slow plasticity rates result in robust and slow-changing behaviors, while weak pathways and fast plasticity rate result in quick behavior reversal.

5. DISCUSSION

The human-robot interactions presented in this study allowed human operators to explore the dynamics of learning in a natural scenario. The tests revealed a number of significant aspects of the neural model that can be compared to biological counterparts.

The generation of eligibility traces by means of rare correlations is a mechanism that selects synapses that may reflect relationships between stimuli or stimuli/actions. The event of a subsequent reward reinforces synapses that are even more likely related to a reward. The presence of disturbing stimuli and delays means that one reward episode is not sufficient to determine uniquely the stimulus that predicts a reward, or the action that causes it. Accordingly, the plasticity rule increases significantly the weights only over many consecutive rewards episodes, suggesting that a correct rate of learning is fundamental in conditioning experiments. A comparison of different learning rates was not rigorously conducted in the present study. Nevertheless, preliminary experiments confirmed the intuitive notion that fast plasticity rates result in a belief being established in fewer episodes. Fast plasticity rates, also possible in the proposed algorithm[2], can be used to observe the accidental response-contingency hypothesis of Skinner (1948). Thus, superstitious behavior can be reproduced with the current model if weights are highly plastic, confirming that high learning rates may results sometimes in establishing wrong associations. However, while this position is a common assumption in machine learning, the proposed neural model attributes the causes of erroneous wrong associations to precise weight dynamics. The process of selecting synapses for weight update must be highly selective and the update must be moderate to endow the network with the necessary prudence before establishing an association. Further research in biology could ascertain whether, similarly to the present computational model, traces, and modulatory episodes in biological brains could be regulated parsimoniously to prevent runaway synapses (Hasselmo, 1994), forgetting (Wixted, 2004), or preserve learning capabilities (Anlezark et al., 1973; Hasselmo, 1999; Bailey et al., 2000; Reynolds and Wickens, 2002).

The decay rate of traces determines how long the network remembers a stimulus. Assume for example that the tutor shows the iCub a yellow object, to which the robot erroneously answers "blue." The tutor ignores the incorrect response, but immediately, i.e., 1 or 2 s later, presents a red object to the robot that answers "red." If now the tutor gives a reward, such a reward reinforces the association of the red stimulus to the red enunciation, but it reinforces to a small extent also the immediately preceding wrong association of the previous trial. If tutoring is enforced with insufficient time between trials, a correct learning is disturbed by interference with previous episodes. Interestingly, this interference is dealt with by the learning rule the same way as disturbing stimuli are, i.e., over the long term they are not reinforced as the reward-causing action. Such a consideration leads once more to the rate of learning: with slower learning rates, the learning is more robust to interferences. Unfortunately, even if in the long

[2] More plastic weights can be implemented in the current model with higher modulation, higher parameters α and β of the RCHP, or higher percentages of correlations. These factors are sometimes referred to in the literatures as "learning rate."

term slow learning rates guarantee better results, this behavior is generally not appreciated by the human tutor who might not display sufficient patience or perseverance toward a slow learning robot.

The test on behavior reversal showed that the weight dynamics in this experiment follow the *reconfigure-and-saturate* rule in Soltoggio and Stanley (2012), which describes the alternation of exploration and exploitation as a consequence of noisy anti-Hebbian plasticity (due to negative modulation and noise) and Hebbian growth (due to positive modulation). In that study, the strength of pathways also represented the probabilities of performing certain action. The growth and decrease of weights was not a consequence of weight tuning or memory decay, but, similarly to the present study, represented the consolidation or forgetting of behaviors. Whilst in Soltoggio and Stanley (2012) the reward was simultaneous with the actions, in the experiments of the current study the alternation of exploration and exploitation emerges from *delayed* negative and positive modulation. This confirms that the reconfigure-and-saturate dynamics in Soltoggio and Stanley (2012) can be reproduced also with delayed rewards as in the realistic robotic scenarios presented in this paper. In particular, the feedback-driven alternation of exploring and exploiting behaviors can be observed even with time gaps between causally related cues, actions, and rewards.

A behavior reversal can be induced, as in the presented case, by applying a negative reward, or punishment. However, the absence of a reward (or unconditioned stimulus) may also induce the extinction of actions (Gallistel, 1993). The absence of a reward is particularly relevant when there is an expectation after conditioning, e.g., food comes after pressing a lever. In the current experiments, expected reward is not modeled and the reward signal is used without pre-processing. A form of extinction is present in the current experiments because a small negative baseline modulation is present at all times [parameter b in equation (4)]. When a strong stimulus propagates through the network, it generates eligibility traces which make those pathways sensitive to modulatory signals for weight update. If no reward occurs in the following interval, the small baseline negative modulation causes also a small decrement of those synapses with high positive traces. Thus, extinction occurs if cues and actions are never followed by rewards. A fully fledged model of behavior extinction, including the modeling of an expected reward, was not the focus of the current study. A number of aspects must be clarified to introduce the notion of unexpected reward, or surprise. In particular, for each stimulus, an average value associated with previous rewards must be memorized in the network. Subsequently, a difference between expected and actual reward must be computed. However, if the timing of the reward is uncertain, it is also unclear when such a difference is to be computed. Moreover, the learning of a correct association may not require further reinforcement later on. In summary, the questions that emerge in scenarios with both delayed rewards and expected rewards make the topic a promising venue for extensions of the current model.

The current model does not implement blocking (Kamin, 1969). Blocking is a phenomenon in which, once a conditioned stimulus CS1 is associated with an unconditioned stimulus, a second conditioned stimulus CS2, occurring simultaneously to CS1, is not associated anymore. Simulations (not shown) indicated that a second stimulus (CS2) is also paired to the US. This characteristic, although different from some observations in animal learning (Kamin, 1969), shows the ability of the model of continuous learning and to discover new associations even after initial associations are established.

Finally, it is worth noting that the success in bridging temporal gaps emerges from the balanced equilibrium between the production rate of traces (by means of rare correlations) and their duration. In the current study, a time constant of 4 s for the eligibility traces was used. With such a constant, associations between cues and rewards can be discovered if a reward is delayed by a maximum of 10–12 s. Longer delays mean that the responsible stimuli and actions are forgotten. Making traces more durable, i.e., having a slower decay, is a way to empower a network to bridge even more distal rewards. To preserve the selectivity of the RCHP rule, longer-lasting traces must be compensated with a lower rate of production, i.e., they must be generated even more parsimoniously. Such a position suggests that long gaps between cues, actions, and rewards can be handled by a learning neural network only if the creation and destruction of traces is particularly rare (Soltoggio and Steil, 2013). For biological brains, which are notoriously subject to a considerably higher level of inputs and outputs, the current model predicts that particularly selective mechanisms could be responsible for filtering relevant information to be integrated later in time upon reward delivery.

6. CONCLUSION

This study demonstrates neural robotic conditioning in human-robot interactive scenarios with delayed rewards, disturbing stimuli, and uncertain timing. The neural dynamics employ rare neural correlations, eligibility traces, and delayed modulation to learn solutions in conditioning problems with realistic timing. The plasticity rule extracts rare correlations, generates eligibility traces, and uses them with Hebbian and anti-Hebbian plasticity according to environmental cues and human feedback. The result is robust classical and operant conditioning with delayed rewards and disturbances. The robotic experimentation proves the robustness and suitability of the proposed neural mechanism in learning with uncertain timing, unreliable inputs, delayed rewards, and variable human-robot reaction times and feedback.

This study also further promotes the idea that differences in the strength of neural pathways may reflect the tendency toward exploration or exploitation. Smaller differences cause the neural dynamics to be driven mainly by neural noise, which leads to exploration. Greater differences cause the network to exploit particular behaviors that were previously reinforced.

Finally, decaying eligibility traces model important learning dynamics with potential implications and predictions in biology. The model lends itself to predictions on how long and how many past events can be traced by a small network. Additionally, the plasticity rate and the strength of the pathways represent the rapidity with which a behavior (or a belief) is established, and the strength and robustness of such behaviors. Once a behavior is established, further confirmations and rewards continuously reinforce the involved pathways, thereby imprinting such a behavior that becomes later more difficult to eradicate. Such types of simulated behaviors are of interest in cognitive developmental robotics, an area in which delayed rewards and human interaction are used in learning processes. In conclusion, the proposed neuro-robotic model displays strongly bio-inspired synaptic and behavioral dynamics that are therefore relevant not only for robotics, but also for biology, neuroscience, and psychology.

ACKNOWLEDGMENTS

This work was supported by the European Community's Seventh Framework Programme FP7/2007-2013, Challenge 2 Cognitive Systems, Interaction, Robotics under grant agreement No 248311 - AMARSi.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at http://www.frontiersin.org/Neurorobotics/10.3389/fnbot.2013.00006/abstract

REFERENCES

Alexander, W. H., and Sporns, O. (2002). An embodied model of learning, plasticity, and reward. *Adapt. Behav.* 10, 143–159.

Anlezark, G. M., Crow, T. J., and Greenway, A. P. (1973). Impaired learning and decreased cortical norepinephrine after bilateral locus coeruleus lesions. *Science* 181, 682–684.

Asada, M., MacDormanb, K. F., Ishigurob, H., and Kuniyoshic, Y. (2001). Cognitive developmental robotics as a new paradigm for the design of humanoid robots. *Rob. Auton. Syst.* 37, 185–193.

Aston-Jones, G., and Cohen, J. D. (2005). Adaptive gain and the role of the locus coeruleus-norepinephrine system in optimal performance. *J. Comp. Neurol.* 493, 99–110.

Avery, M. C., Nitz, D. A., Chiba, A. A., and Krichmar, J. L. (2012). Simulation of cholinergic and noradrenergic modulation of behavior in uncertain environments. *Front. Comput. Neurosci.* 6:5. doi:10.3389/fncom.2012.00005

Bailey, C. H., Giustetto, M., Zhu, H., Chen, M., and Kandel, E. R. (2000). A novel function for serotonin-mediated short-term facilitation in *Aplysia*: conversion of a transient, cell-wide homosynaptic Hebbian plasticity into a persistent, protein synthesis-independent synapse-specific enhancement. *Proc. Natl. Acad. Sci. U.S.A.* 97, 11581–11586.

Brembs, B., Lorenzetti, F. D., Reyes, F. D., Baxter, D. A., and Byrne, J. H. (2002). Operant reward learning in aplysia: neuronal correlates and mechanisms. *Science* 296, 1706–1709.

Carew, T. J., Walters, E. T., and Kandel, E. R. (1981). Classical conditioning in a simple withdrawal reflex in *Aplysia californica*. *J. Neurosci.* 1, 1426–1437.

Cox, R. B., and Krichmar, J. L. (2009). Neuromodulation as a robot controller: a brain inspired strategy for controlling autonomous robots. *IEEE Robot. Autom. Mag.* 16, 72–80.

Doya, K. (1999). What are the computations of the cerebellum, the basal ganglia and the cerebral cortex? *Neural. Netw.* 12, 961–974.

Doya, K. (2002). Metalearning and neuromodulation. *Neural. Netw.* 15, 495–506.

Farries, M. A., and Fairhall, A. L. (2007). Reinforcement learning with modulated spike timing-dependent synaptic plasticity. *J. Neurophysiol.* 98, 3648–3665.

Fellous, J.-M., and Linster, C. (1998). Computational models of neuromodulation. *Neural. Comput.* 10, 771–805.

Florian, R. V. (2007). Reinforcement learning through modulation of spike-timing-dependent synaptic plasticity. *Neural. Comput.*19, 1468–1502.

Frey, U., and Morris, R. G. M. (1997). Synaptic tagging and long-term potentiation. *Nature* 385, 533–536.

Gallistel, C. R. (1993). *The Organization of Learning*. Cambridge: MIT Press.

Hammer, M. (1993). An identified neuron mediates the unconditioned stimulus in associative olfactory learning in honeybees. *Nature* 366, 59–63.

Hasselmo, M. E. (1994). Runaway synaptic modification in models of cortex: implications for Alzheimer's disease. *Neural. Netw.* 7, 13–40.

Hasselmo, M. E. (1999). Neuromodulation: acetylcholine and memory consolidation. *Trends Cogn. Sci. (Regul. Ed.)* 3, 351–359.

Hasselmo, M. E. (1995). Neuromodulation and cortical function: modeling the physiological basis of behavior. *Behav. Brain Res.* 67, 1–27.

Hull, C. L. (1943). *Principles of Behavior*. New-York: Appleton Century.

Izhikevich, E. M. (2007). Solving the distal reward problem through linkage of STDP and dopamine signaling. *Cereb. Cortex* 17, 2443–2452.

Kamin, L. J. (1969). *Punishment and Aversive Behavior, Chapter Predictability, Surprise, Attention and Conditioning*. New York: Appleton-Century-Crofts, 279–296.

Kandel, E. R., and Tauc, L. (1965). Heterosynaptic facilitation in neurones of the abdominal ganglion of *Aplysia depilans*. *J. Physiol. (Lond.)* 181, 1–27.

Kaski, S., and Kohonen, T. (1994). Winner-take-all networks for physiological models of competitive learning. *Neural. Netw.* 7, 973–984.

Krichmar, J. L. (2008). The neuromodulatory system: a framework for survival and adaptive behavior in a challenging world. *Adapt. Behav.* 16, 385–399.

Legenstein, R., Pecevski, D., and Maass, W. (2008). A learning theory for reward-modulated spike-timing-dependent plasticity with application to biofeedback. *PLoS Comput. Biol.* 4:e1000180. doi:10.1371/journal.pcbi.1000180

Marder, E., and Thirumalai, V. (2002). Cellular, synaptic and network effects of neuromodulation. *Neural. Netw.* 15, 479–493.

McGill, R., Turkey, J. W., and Larsen, W. A. (1978). Variations of box plots. *Am. Stat.* 32, 12–16.

Menzel, R., and Müller, U. (1996). Learning and memory in honeybees: from behavior to natural substrates. *Annu. Rev. Neurosci.* 19, 379–404.

Montague, P. R., Dayan, P., and Sejnowski, T. J. (1996). A framework for mesencephalic dopamine systems based on predictive Hebbian learning. *J. Neurosci.* 16, 1936–1947.

Osherson, D., Stob, M., and Weinstein, S. (1990). *Systems that Learn: An Introduction to Learning Theory for Cognitive and Computer Scientists (Learning, Development, and Conceptual Change)*. Cambridge: MIT Press.

Päpper, M., Kempter, R., and Leibold, C. (2011). Synaptic tagging, evaluation of memories, and the distal reward problem. *Learn. Mem.* 18, 58–70.

Pavlov, I. P. (1927). *Conditioned Reflexes*. Oxford: Oxford University Press.

Pfeiffer, M., Nessler, B., Douglas, R. J., and Maass, W. (2010). Reward-modulated Hebbian learning of decision making. *Neural. Comput.* 22, 1–46.

Porr, B., and Wörgötter, F. (2007). Learning with relevance: using a third factor to stabilize Hebbian learning. *Neural. Comput.* 19, 2694–2719.

Potjans, W., Diesmann, M., and Morrison, A. (2011). An imperfect dopaminergic error signal can drive temporal-difference learning. *PLoS Comput. Biol.* 7:e1001133. doi:10.1371/journal.pcbi.1001133

Potjans, W., Morrison, A., and Diesmann, M. (2009). A spiking neural network model of an actor-critic learning agent. *Neural. Comput.* 21, 301–339.

Redondo, R. L., and Morris, R. G. M. (2011). Making memories last: the synaptic tagging and capture hypothesis. *Nat. Rev. Neurosci.* 12, 17–30.

Reynolds, J. N., and Wickens, J. R. (2002). Dopamine-dependent plasticity of corticostriatal synapses. *Neural. Netw.* 15, 507–521.

Sarkisov, D. V., and Wang, S. S. H. (2008). Order-dependent coincidence detection in cerebellar Purkinje neurons at the inositol trisphosphate receptor. *J. Neurosci.* 28, 133–142.

Schultz, W., Apicella, P., and Ljungberg, T. (1993). Responses of monkey dopamine neurons to reward and conditioned stimuli during successive steps of learning a delayed response task. *J. Neurosci.* 13, 900–913.

Schultz, W., Dayan, P., and Montague, P. R. (1997). A neural substrate for prediction and reward. *Science* 275, 1593–1598.

Skinner, B. F. (1948). "Superstition" in the pigeon. *J. Exp. Psychol.* 38, 168–172.

Skinner, B. F. (1953). *Science and Human Behavior*. New York: MacMillan.

Soltoggio, A., Bullinaria, J. A., Mattiussi, C., Dürr, P., and Floreano, D. (2008). "Evolutionary advantages of neuromodulated plasticity in dynamic, reward-based scenarios," in *Artificial Life XI: Proceedings of the Eleventh International Conference on the Simulation and Synthesis of Living Systems*. Cambridge: MIT Press.

Soltoggio, A., and Stanley, K. O. (2012). From modulated Hebbian plasticity to simple behavior learning through noise and weight saturation. *Neural. Netw.* 34, 28–41.

Soltoggio, A., and Steil, J. J. (2013). Solving the distal reward problem with rare correlations. *Neural. Comput.* 25, 940–978.

Soula, H., Alwan, A., and Beslon, G. (2005). "Learning at the edge of chaos: temporal coupling of spiking neurons controller for autonomous robotic," in *Proceedings of the AAAI Spring Symposia on Developmental Robotics*. Stanford, CA: AAAI Spring Symposium Series.

Sporns, O., and Alexander, W. H. (2002). Neuromodulation and plasticity in an autonomous robot. *Neural Netw.* 15, 761–774.

Sutton, R. S., and Barto, A. G. (1998). *Reinforcement Learning: An Introduction*. Cambridge, MA: MIT Press.

Thorndike, E. L. (1911). *Animal Intelligence*. New York: Macmillan.

Tsakarakis, N., Metta, G., Sandini, G., Vernon, D., Beira, R., Becchi, F., et al. (2007). iCub – the design and realization of an open humanoid platform for cognitive and neuroscience research. *Adv. Robot.* 21, 1151–1175.

Vasilaki, E., Frémaux, N., Urbanczik, R., Senn, W., and Gerstner, W. (2009). Spike-based reinforcement learning in continuous state and action space: when policy gradient methods fail. *PLoS Comput. Biol.* 5:e1000586. doi:10.1371/journal.pcbi.1000586

Wang, S. S. H., Denk, W., and Häusser, M. (2000). Coincidence detection in single dendritic spines mediated by calcium release. *Nat. Neurosci.* 3, 1266–1273.

Wixted, J. T. (2004). The psychology and neuroscience of forgetting. *Annu. Rev. Psychol.* 55, 235–269.

Ziemke, T., and Thieme, M. (2002). Neuromodulation of reactive sensorimotor mappings as short-term memory mechanism in delayed response tasks. *Adapt. Behav.* 10, 185–199.

Conflict of Interest Statement: 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.

Received: 01 December 2012; accepted: 06 March 2013; published online: 02 April 2013.

Citation: Soltoggio A, Lemme A, Reinhart F and Steil JJ (2013) Rare neural correlations implement robotic conditioning with delayed rewards and disturbances. Front. Neurorobot. ***7****:6. doi: 10.3389/fnbot.2013.00006*

APPENDIX

DETAILS OF THE NEURAL MODEL

The plasticity rule (RCHP) described by equations (1, 2) and (4) is fully specified by the parameters in **Table A1** in Appendix. The neural model described by equations (2–4) is fully specified by the values in **Table A2** in Appendix. The integration of equations (2) and (4) with a sampling time Δt of 200 ms is implemented step-wise by

$$c_{ji}\left(t+\Delta t\right) = c_{ji}\left(t\right) \cdot e^{\frac{-\Delta t}{\tau_c}} + \mathrm{RCHP}_{ji}\left(t\right) \tag{A1}$$

$$m\left(t+\Delta t\right) = m\left(t\right) \cdot e^{\frac{-\Delta t}{\tau_m}} + \lambda r\left(t\right) + b \tag{A2}$$

The measured rates of correlations $\rho_c(t)$ and decorrelations $\rho_d(t)$ are computed over a sliding time window of 10 s summing all correlations and decorrelations buffered in $cq(t)$ and $dq(t)$

$$\rho_c\left(t\right) = \Delta t \frac{\sum_{0}^{t-10} cq\left(t\right)}{10} \quad , \tag{A3}$$

and similarly for $\rho_d(t)$. The adaptive thresholds θ_{hi} and θ_{lo} in equation (1) are estimated as follows.

$$\theta_{hi}\left(t+\Delta t\right) = \begin{cases} \theta_{hi} + \eta \cdot \Delta t & \text{if} \rho_c(t) > 5\mu \\ \theta_{hi} - \eta \cdot \Delta t & \text{if} \rho_c(t) < \mu/5 \\ \theta_{hi}(t) & \text{otherwise} \end{cases} \tag{A4}$$

and

$$\theta_{lo}(t+\Delta t) = \begin{cases} \theta_{lo} - \eta \cdot \Delta t & \text{if} \rho_d(t) > 5\mu \\ \theta_{lo} + \eta \cdot \Delta t & \text{if} \rho_d(t) < \mu/5 \\ \theta_{lo}(t) & \text{otherwise} \end{cases} \tag{A5}$$

with $\eta = 0.002$. If correlations are lower than a fifth of the target or are greater than five times the target, the thresholds are adapted to the new increased or reduced activity. This heuristic has the purpose of maintaining the thresholds relatively constant and perform adaptation only when correlations are too high or too low for a long period of time.

Table A1 | Parameters of the plasticity rule (RCHP) and modulation.

Time constant of eligibility traces [τ_c, equation (2)]	4 s
α [Equation (1)]	0.1
β [Equation (1)]	0.1
λ [Equation (4)]	0.05 (0.07*)
b [Equation (4)]	−0.002/s
Target rate of rare correlations μ	0.5%

(*) The higher value 0.07 is effectively a slight increase in the learning rate that was used in the classical conditioning experiment with brief stimuli (Section 4.1.3): this experiment set-up resulted in fewer rewarding episodes and so the higher value of λ led to convergence within the 2 h of simulated time.

Table A2 | Parameters of the neural model.

Excitatory neurons	800
Inhibitory neurons	200
Connection probability	0.1
Weight range	[0, 1]
Inhibitory weights	Fixed in [0, 1]
Excitatory weights	Plastic
Noise on neural transmission [$\xi_i(t)$, equation (6)]	Uniform [−0.1, 0.1]
Target rate of rare correlations μ	0.5%
Sampling time step [Δt, equation (6)]	200 ms
Time constant of modulation [τ_m, equation (4)]	1 s
Neural gain [γ, equation (6)]	0.25

The reward signal $r(t)$ was impulse-like in nature for the simulated classical and operant conditioning experiments, i.e., lasting one computational step (200 ms). In the robotic experiments, the duration of the touch to the iCub's hand/arm effectively determined the magnitude of the reward episode simply by making this signal last longer. The magnitude of $r(t)$, in this study set in the range [1, 5], can be used to achieve different learning rates (data not shown).

The complete scripts for reproducing the experiment in simulation can be downloaded from the author's associate website http://andrea.soltoggio.net/icub