Dynamical Perceptual-Motor Primitives for Better Deep Reinforcement Learning Agents

Gaurav Patil, Patrick Nalepka, Lillian Rigoli, Rachel W. Kallen and Michael J. Richardson

July 1, 2021

Dynamical Perceptual-Motor Primitives for Better Deep Reinforcement Learning Agents

Gaurav Patil[1,2], Patrick Nalepka[1,2], Lillian Rigoli[1], Rachel W. Kallen[1,2], and Michael J. Richardson[1,2]

[1] Department of Psychology, [2] Centre for Elite Performance, Expertise and Training,
Macquarie University Sydney, NSW 2019, Australia

{gaurav.patil, patrick.nalepka, lillian.rigoli, rachel.kallen, michael.j.richardson}@mq.edu.au

Abstract. Recent innovations in Deep Reinforcement Learning (DRL) and Artificial Intelligence (AI) techniques have allowed for the development of artificial agents that can outperform human counterparts. But when it comes to multiagent task contexts, the behavioral patterning of AI agents is just as important as their performance. Indeed, successful multi-agent interaction requires that co-actors behave reciprocally, anticipate each other's behaviors, and readily perceive each other's behavioral intentions. Thus, developing AI agents that can produce behaviors compatible with human co-actors is of vital importance. Of particular relevance here, research exploring the dynamics of human behavior has demonstrated that many human behaviors and actions can be modeled using a small set of dynamical perceptual-motor primitives (DPMPs) and, moreover, that these primitives can also capture the complex behavior of humans in multiagent scenarios. Motived by this understanding, the current paper proposes methodologies which use DPMPs to augment the training and action dynamics of DRL agents to ensure that the agents inherit the essential pattering of human behavior while still allowing for optimal exploration of the task solution space during training. The feasibility of these methodologies is demonstrated by creating hybrid DPMP-DRL agents for a multiagent herding task. Overall, this approach leads to faster training of DRL agents while also exhibiting behavior characteristics of expert human actors.

Keywords: Deep Reinforcement Learning (DRL), Dynamical Motor Primitives (DMPs), Multiagent coordination.

1 Introduction

Rapid improvements in model-free Artificial Intelligence (AI) and Deep Reinforcement Learning (DRL) techniques [1–4] have resulted in the development of artificial agents capable of performing various tasks at levels equal to or better than human experts. In many cases, however, the success of these DRL agents requires a complex, highly tuned, and task-specific structure of DRL methodologies and neural-network architectures along with long and computationally intensive self-play training schemes [1, 2]. Moreover, even after constraining the action space of DRL agents to match human response limitations [1], the behavior of DRL agents is often qualitatively different from humans [5], such that, DRL agents often exhibit action sequences or behavioral strategies that are not readily performed by humans. Although this does not pose a problem if the goal is only to achieve optimal or near optimal performance, it poses a major challenge when the aim is to develop DRL agents capable of effective human-AI agent interaction. Indeed, effective human performance in multiagent contexts requires that co-actors behave reciprocally, are able to anticipate each other's behaviors, and can readily perceive each other's behavioral intentions [6] while maintaining the right interaction flexibility [7]. Thus, developing methods that produce DRL agents that are capable of human-like behavior leading to robust human-centered coordination is often essential.

One way to improve the "human-like" nature of DRL agents is to employ prerecorded human expert data or real-time human gameplay/interventions during the training process; e.g., behavior cloning [8], generative adversarial imitation learning (GAIL) [9], or oracle learning [10]. In addition to increasing the interactive effectiveness of DRL by exposing them to human actions and reciprocal patterns of coordination that are likely to be missed during self-play training [6], the use of human data to pre-train AI agents also helps to scaffold the essential "dynamics of gameplay" (e.g., basic action and coordination patterns that lead to preliminary levels of task success), both ensuring effective task learning and decreasing training time [11]. Unfortunately, these methods rely on the availability of large datasets of human gameplay, which are not readily available for most tasks (both real and computer based), and can suffer sharp performance declines when the expert data is sparse or imperfect [12].

However, despite the variability and complexity of human data within and across task contexts, research exploring the dynamics of human behavior has demonstrated that it typically reflects the context-specific realization of low-dimensional principles. Indeed, a growing body of research [13–17] has revealed that the spatiotemporal patterning of the behavioral actions that define human performance and decision making in both individual and multiagent task contexts can be modelled using a small, fundamental set of dynamical primitives (i.e., nonlinear dynamical functions) [15–18]. Moreover, that the task-specific structure and parameterization of such models can be achieved with small human datasets (i.e., 5 to 10 individuals/teams) and can readily generalize across various task contexts [19–23].

The significant implication of the latter work is that human-inspired, dynamical models could be employed to (a) enhance DRL training across task contexts where large human datasets are not available or augment DRL models to inherit the low-dimensional dynamics of human behavior or decision making and (b) produce DRL agents that enact more human-like behavior, and thus, work more effectively in mixed human-AI multiagent task contexts. Thus, the aim of this paper is to provide a brief background of the application of dynamical primitives to model individual behavior in multiagent task contexts and to provide a methodology for using dynamical primitives to augment DRL agents trained to complete a complex multiagent herding task.

2 Modeling Perceptual-Motor Behaviors

2.1 Dynamical Perceptual-Motor Primitives in Individual Behavior

Research on perceptual-motor behavior [24] has revealed that human actions are composed of two fundamental movement types: (1) ***discrete movements***, as when one reaches for an object or target location, taps a key, or throws a dart; and (2) ***rhythmic movements***, as when one waves a hand, hammers a nail, or simply walks. Furthermore, previous research has demonstrated how task-defined human perceptual-motor behavior and decision making can be modelled using a relatively small set of nonlinear dynamical primitives: namely, environmentally coupled ***fixed-point*** (mass-spring) and ***limit cycle*** (self-sustained oscillator) equations, as well as multi-stable bifurcation functions [13–17]. For instance, research has shown these dynamical primitives can be employed to effectively model human reaching, object passing, rhythmic wiping, cranking tasks [25], goal-directed human navigation within an obstacle-ridden environment, including route selection [21], and drumming and racket ball tasks [20]. The dynamical primitives used to model human perceptual-motor behaviors can be termed as dynamical perceptual-motor primitives (DPMPs).

2.2 DPMPs in Multiagent Tasks

To succeed in human-human multiagent task contexts, individual agents have to plan their action in relation to both the desired goal and their partner's state and action [26]. This results in individuals coordinating their actions physically and temporally to collectively influence the environment [18, 27]. The stable patterns of such coordination, whether between a group of friends clearing a dinner table or teammates playing football, naturally emerge from the changing physical constraints and informational couplings that exist between the environmentally embedded co-acting individuals [28–30]. Thus, the same dynamical primitives used to model human perceptual-motor behavior can also be employed to model the task dynamics of numerous complex multiagent tasks, including cooperative object pick-and-place tasks [31] and goal-directed multiagent navigation and collision avoidance behaviors, as well as multiagent shepherding behavior [14, 19]. The latter research has also demonstrated how these DPMPs can be employed to control the behavior of artificial agents in human-AI agent contexts, with human-AI agent performance equivalent to and indistinguishable from human-human performance. It is also important to note that the DPMPs that underly these models can be readily generalized across a wide range of multiagent task contexts [19, 31–33].

2.3 Use of Deep Reinforcement Learning (DRL) in Conjunction with DPMPs

Importantly, DPMPs have the potential to provide a highly generative set of dynamical functions for developing low-dimensional models of synergistic human perceptual-motor behavior. Several researchers have demonstrated how DPMPs can significantly reduce the dimensionality of motor-skill training and control in artificial systems [34, 35]. For instance, Ijspeert and colleagues [23, 36] have shown how DPMPs can be employed to generatively train a virtual end-effector or multi-joint robotic arm to perform goal-directed reaching, obstacle avoidance movements, and racket swinging. It is important to note here that the use of DPMPs introduces the need for the selection of task-specific models and further optimization of the model parameters. Various machine learning techniques can be used for DPMP model selection and parameter optimization e.g., *imitation-* and *reinforcement*-based techniques [23], supervised learning [37], and search-based optimization techniques [38].

The advantage of reinforcement learning (RL) is that such machine learning approaches do not require the agents have a-priori knowledge of the dynamics of the environment nor the agent's action capabilities or consequences. In RL, agents learn via trial-and-error, modifying their behavior to maximize desired outcomes. Computationally, the goal of machine-based RL is to find the policy (state-action mapping) that results in an agent maximizing its reward within a complex dynamical environment [39]. Combined with deep-neural-network architectures and a "replay-memory", deep reinforcement learning (DRL) methods have gained wide notoriety for their ability to learn various tasks at or above human levels of performance [1–4]. This is in-part due to the powerful function approximation properties of deep neural networks which can learn low-dimensional feature representations from high-dimensional state-action spaces [40]. Most relevant here is the work demonstrating how DRL can be employed to map continuous action or parameter spaces [41, 42]. Interestingly, DRL applied within multiagent contexts can result in more robust behavioral policies than single actor RL [43]. However, although DRL methods have the advantage of generalizing over a wide set of state-action-reward scenarios and mapping high-dimensional states to actions, DRL methods are notoriously slow and computationally intensive resulting in researchers often relying on imitation learning methods to enhance the speed of novel task learning [44, 45].

The advantages and shortcomings of both DPMP and DRL methods necessitate the use of both methodologies in conjunction. Indeed, we propose that the DPMPs can be used in two ways to enhance the training and performance of DRL agents: 1) using DPMPs during DRL training and 2) augmenting DRL model architectures with DPMPs. The former approach is analogous to imitation learning approaches [8–10] while treating the DPMP model as an expert "human" demonstrator. The rest of the paper will however focus on the latter and will specifically present the application of the proposed methods to the multiagent herding problem.

3 The Herding Problem

3.1 Modeling Human Behavior using DPMPs

The herding problem is a widely studied multiagent paradigm wherein two or more herders (agents) have to corral multiple targets agents (e.g., sheep, autonomous agents) and either contain them or move them from one location to another [19]. The task is ideally suited for the investigation of human group and multiagent coordination and problem-solving behaviors, including task division, behavior-mode switching (corralling to containment), and adaptation to task perturbations (new targets) [46]. In the context of this paper, of particular interest is the recent research demonstrating how DPMPs can be employed to model the emergence of the coordinated perceptual-motor strategies of humans during successful task completion [13, 14, 19]. The task consists of 'herding agents' (HAs) successfully corralling and containing a set of 'target agents' (TAs), typically ranging from 3 to 7 targets, within a red containment region located on a game field. When left unperturbed, TAs exhibit Brownian motion, and thus naturally disperse if left alone. Importantly, however, the TAs are repelled away from the HAs, such that, when an HA is within a critical distance from a TA, the TA flees in the opposite direction. Thus, continuous action by both HAs is required to corral and keep the TAs contained within the containment region. Task trials are typically between 1 to 2 minutes, with a trial deemed successful if an HA dyad can contain the TAs within the containment area for a specified period or percentage of trial time (e.g., 70% of a 1-min trial or continuously for 10 s). An overview of the task layout is show in in Figure 1. An effective strategy to complete this task is to select and recover the TA that is farthest from the containment region, such that at each point in time each HA moves towards the farthest-TA closest to their current location (and not currently being corralled by another HA). This strategy, termed as Search and Recover (S&R), can be modelled by a DPMP based task dynamic model taking the form,

$$\ddot{r}_i + \alpha_r \dot{r}_i + \omega_\theta^2 \big(r_i - (r_{T,i} + r_{min})\big) = 0 \tag{1}$$

and

$$\ddot{\theta}_i + \alpha_\theta \dot{\theta}_i + \beta \dot{\theta}_i^3 + \gamma \theta_i \dot{\theta}_i + \omega_\theta^2 (\theta_i - \theta_{T,i}) = 0, \tag{2}$$

which model the radial distance and angle of each herder, respectively. More specifically, in Eq. (1), $\dot{r}$, and $\ddot{r}$ represent the velocity and acceleration of HA-i's radial distance, respectively, $r_{T,i}$ is the radial distance of the farthest TA that is being pursued, and r_{min} is a fixed parameter that specifies HA-i's minimum preferred radial distance from a TA during herding to ensure repulsion towards the goal. In Eq. (2), $\dot{\theta}_i$ and $\ddot{\theta}_i$ represent the velocity and acceleration of the radial angle, respectively, α_θ and ω_θ^2 represent the dampness and stiffness parameters, $\theta_{T,i}$ represents the radial angle of the TA pursued by HA-i, and $\beta \dot{\theta}_i^3$ and $\gamma \theta_i \dot{\theta}_i$ are the nonlinear Rayleigh and van der Pol terms. The inclusion of the nonlinear terms captures the amplitude-frequency and peak velocity-frequency relationship exhibited by human actors [25].

The S&R strategy is effective in corralling TAs into a containment region, but when tasked with continuously containing more than four TAs within a containment region for extended periods of time, the S&R strategy becomes unstable (ineffective) and a more robust strategy is adopted by experienced herders. This latter containment strategy involves the HAs performing oscillatory movements that together encircle the entire TA herd and has been termed as coupled oscillatory containment (COC) [19]. A more complex and robust DPMP model can be used to model both S&R and COC behaviors with additional terms for coupling between HAs (see [46] for more details). However, for the scope of this paper, which is concerned with demonstrating the feasibility of using DPMP models to augment DRL agents, the simplified model approximated by Eqs. (1) and (2) is sufficient.

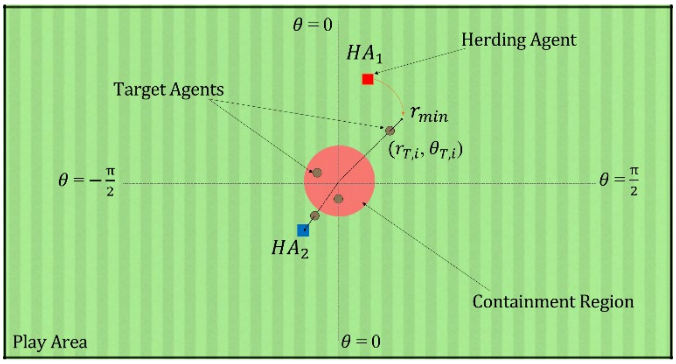

Fig. 1. Top view of the herding environment

3.2 Target Selection (Decision Dynamics)

It is also important to note here that, in general, the DPMP model for the S&R herding approximated by the above equations defines the *action dynamics* of a herder (i.e., the movement dynamics when moving towards and corralling a TA). However, the effectiveness of Eqs. (1) and (2), is dependent on the decision dynamics of *target selection,* which determines the TA to be pursued (i.e., $(r_{T,i}, \theta_{T,i})$). Indeed, research has demonstrated how the specifics of the target selection rule (dynamics) can significantly influence task performance [47]. Auletta et al. [48], for example, showed that the TA selection strategies derived from expert human players can be significantly different and lead to better task performance than those derived from novice human players, while resulting in the same number of task successes.

Of particular importance here, Nalepka et al. [19] demonstrated how human TA selection can be modeled heuristically as: select the TA that was (i) closer to their HA than the other HA and (ii) was furthest from the containment area. Rigoli et al. [49] further demonstrated that this TA selection rule results in robust novice human-AI agent interaction while also providing training equivalent to a human expert.

3.3 Hybrid DRL Agents for Herding

A classical approach to applying DRL techniques to a multiagent problem like herding would be to use a single deep neural network to approximate the target selection and action dynamics policy for each HA where the states of all the TAs and HAs are provided as an input and the network outputs the HA's action. This approach can be further decentralized by using separate networks for target selection and action dynamics which can be trained independently. To draw upon the advantages of DPMP models, the decentralized DRL architectures can be augmented by creating hybrid models such that either the target selection or the action dynamics of the DPMP model described in the previous sub-section is used with a deep neural network which is trained by DRL. Here we will refer to these hybrid models as *DRL-target selection* and *DRL-action dynamic* models, respectively, and their schematic is shown in Figure 2.

The DRL-target selection model for each HA uses a neural network to observe the states of all the TAs and HAs and outputs the TA to be pursued. The position of this selected TA is used with the DPMP model described by Eqs. (1) and (2) to determine the action of each HA. It is expected that by using this hybrid model and training it by DRL, the HAs would be able to exhibit better task division and the neural network trained for TA selection can compensate for the absence of the oscillatory and coupling behaviors in the action dynamics. This should further result in a better performance as compared to the simplified DPMP model, which only models the S&R behavior, when the goal containment time is higher (>5s).

On the other hand, the DRL-action dynamics model for each HA uses the heuristic TA selection rule from the DPMP model to select the TA to pursue and the neural network takes the state of that selected TA with the states of all the HAs and outputs the change in position in radial distance and radial angle. In this case, it is expected that the neural network trained to approximate the action dynamics will exhibit the oscillatory and coupling behaviors observed in human experts and thus result in better task performance than the simplified DPMP model.

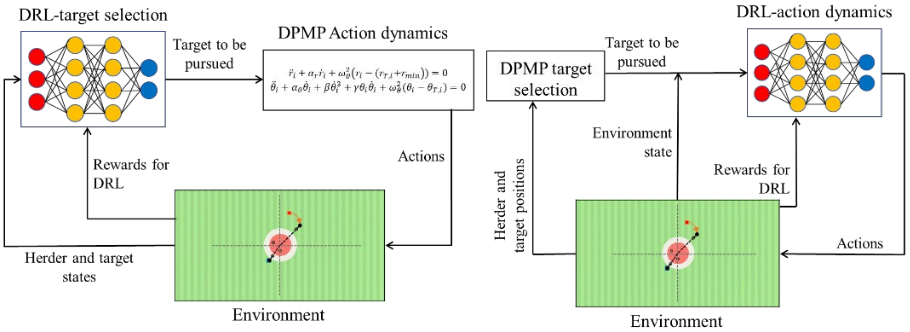
DRL-target selection
Target to be pursued
DPMP Action dynamics
$\ddot{r}_i + \alpha_r \dot{r}_i + \omega_\theta^2(r_i - (r_{T,i} + r_{min})) = 0$
$\ddot{\theta}_i + \alpha_\theta \dot{\theta}_i + \beta \dot{\theta}_i^3 + \gamma \theta_i \dot{\theta}_i + \omega_\theta^2(\theta_i - \theta_{T,i}) = 0$
Actions
Rewards for DRL
Herder and target states
Environment
DPMP target selection
Target to be pursued
DRL-action dynamics
Environment state
Herder and target positions
Rewards for DRL
Actions
Environment

Fig. 2. Schematic of Hybrid DRL agents. (Left) DRL-target selection agent and (Right) DRL-action dynamics agent

4 Simulation Experiments

4.1 Task Environment

The herding environment was developed using the Unity game engine (Unity Technologies, San Francisco, USA) and the DRL agents were implemented using the Unity ML-Agents package [50]. The environment size was set to 1m x 1.8m with two HAs corralling four TAs which spawned randomly in a ±0.3m x ±0.6m rectangle at the center of the field. The task goal was for the HAs to contain the TAs continuously for 10 s while each trial lasted 90 s. The velocity of the HAs was limited to 1 m/s in each direction and the TA behavior and DPMP parameters for equations (1) and (2) were set according to a model tested to approximate human-like behavior (see Nalepka et al. [14] for more details).

4.2 DRL models and Training

The DRL-target selection model for each HA used a neural network with 2 densely connected hidden layers with 128 neurons each and took the states (position and velocity) of all TAs and HAs as inputs (24 inputs) and outputted a one-hot vector of the TA to pursue. The same neural network was used to approximate the policy of both HAs in any given environment, but the actions and observations were transformed such that each HA observed the playing field from the bottom. The neural network was trained according to the Proximal Policy Optimization (PPO) algorithm for reinforcement learning (RL) for 10 million training steps with observations collected every 15th frame while the environment updated at 50 Hz. A curriculum learning was implemented such that, during the first 3 million training steps, the TA spawn area increased in steps linearly from ±0.15m x ±0.3m to ±0.3m x ±0.6m.

For the DRL-action dynamics model, the state of the selected TA and the HAs was used as an input (12 inputs) to a neural network with 2 densely connected hidden layers with 128 neurons each and outputted a continuous action vector (2 outputs) of change in radial distance and radial angle for each HA. The neural network was trained according to the Proximal Policy Optimization (PPO) algorithm for reinforcement learning (RL) for 25 million training steps with observations collected every 5th frame while the environment updated at 50 Hz. Between 2 and 10 million training steps, the TA spawn area increased in steps linearly from ±0.15m x ±0.3m to ±0.3m x ±0.45m while the area of the field increased in steps linearly from 0.8m x 1.2m to 1m x 1.8m. During training, the environment started with 2 TAs and an additional TA was added at 10 and 15 million training steps. Finally, at 5, 7.5, and 10 million training steps the distance within which the HA influenced a TA was stepped from 20cm to 16cm to 12cm and the random motion of TAs when not influenced by HAs proportional to the experimental value (used in [14]) was stepped from 0.25 to 0.5 to 1 times, respectively.

During training, the reward for both hybrid DRL agents was calculated in each environment update such that each HA received a negative (0.01 x distance of TA from center of environment) reward for every TA outside the containment area and positive 0.01 reward for every TA in the containment area.

4.3 Comparison between modeling methodologies

Twenty hybrid DRL agents were trained by each methodology and the 3 top agents of each type were selected by ranking them by the average episode length in the last 0.25 million training steps. 20 simulation trials were carried out for each selected agent (60 trials per condition) and 60 simulation trials were carried out using the DPMP model (parameters set to values specified in [14]) while they completed the 2-HA, 4-TA herding task where both HAs were controlled by the same model. The trial data (states of HAs and TAs) recorded from all trials was used to discern basic performance outcomes for the three agent types.

The analysis revealed that the DPMP, DRL-action dynamics, and DRL-target selection models were found to be successful in 26.67%, 98.33%, and 100% of the simulation trails, respectively. Further, a similar procedure employed by Nalepka et al. [19, 46] was used to classify oscillatory (COC) behaviour during containment for each agent

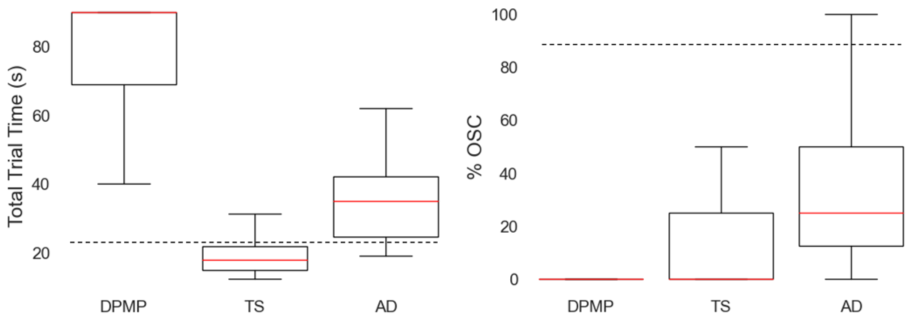

Fig. 3. Boxplots displaying the total time taken to complete a trial (left) and proportion of time spent oscillating during containment (right) for the three agent types, where TS refers to the DRL-target selection model and AD refers to the DRL-action dynamics model. Dotted lines indicate human expert performance for reference.

and proportion of time spent oscillating (%OSC) was then averaged for each agent type. This measure is displayed in Fig. 3 (right).

5 Discussion

The analysis of the performance measures from the simulated trials of the agents modeled by DPMP, DRL-target selection, and DRL-action dynamics methods is in line with the expectation of the hybrid agents performing better than the agent modeled by the simplified DPMP. Indeed, agents trained by both hybrid DRL methods outperform the DPMP agent in terms of task success and total time required for task completion. It is again important to note that the DPMP model used for comparison was a simplified model without the oscillatory and coupling behavior which are characteristic of expert human behavior during successful TA containment [19]. The better performance of the hybrid DRL agents can be attributed to the differences in strategies approximated by the simplified DPMP model and the corresponding deep neural networks. In the case of the DRL-target selection agent, it was observed that the policy diverges from the heuristic policy of the simplified DPMP once the TAs are in the containment region – resulting in higher task success. On the other hand, the policy of the DRL-action dynamics agent when pursuing the TA which is inside the containment region results in oscillatory behavior. This may be due to the fact that the TA selection heuristic encodes information regarding whether all TAs are within the containment region, and thus whether oscillatory behaviors are appropriate. This change in policy is also reflected by the higher proportion of time spent oscillating during containment by the DRL-action dynamics model. Finally, from the box plots in Figure 3, it can be seen that the total trial time taken by the hybrid DRL agents is comparable to expert human pairs. Further, although the time spent by the hybrid DRL agents oscillating during containment is not even close to the expert human level, it is sufficient for task success and supports the occurrence of a bifurcation in human behavior with the increased skill level [13].

In this paper, we successfully demonstrated the usage of DPMP models for creating better hybrid DRL agents. Although not presented here, an alternative approach of using a single deep neural network, or two separate deep neural networks to approximate both target selection and action dynamics, was unsuccessful in learning the task with similar curriculum learning steps and even longer training times (> 100 million steps). If required, the networks from the hybrid DRL agents can be detached and combined to create a completely neural network-based agent for further training using DRL. Finally, as highlighted at the end of section 2, DPMP models can also be used to supplement methods that use expert data (imitation learning) or expert models (oracle learning) for DRL and will need further exploration and testing. Given that the DPMPs capture the essence of human movement behaviors, their use for creating DRL agents can allow for creating DRL agents for a much wider range of tasks without being limited by the complexity of state-action-reward structures and lack of expert datasets. Finally, more research is required to create DRL agents which can exhibit adaptive behavior based on the human teammate's skill level such that DRL agents can be used as a trainer or synthetic teammate for skill-learning.

Acknowledgments. This work was supported by ARC Future Fellowship (Richardson, FT180 100447), Macquarie University Research Fellowship (Nalepka), and Human Performance Research Network (HPRnet, partnership grant ID9024).

References

1. Berner, C. et al.: Dota 2 with Large Scale Deep Reinforcement Learning. arXiv. 1912.06680, (2019).
2. Vinyals, O. et al.: Grandmaster level in StarCraft II using multi-agent reinforcement learning. Nature. 575, 350–354 (2019).
3. Pohlen, T. et al.: Observe and Look Further: Achieving Consistent Performance on Atari. arXiv. 1805.11593, (2018).
4. Mnih, V. et al.: Human-level control through deep reinforcement learning. Nature. 518, 529–533 (2015).
5. Shek, J.: Takeaways from OpenAI Five (2019) [AI/ML, Dota Summary], https://senrigan.io/blog/takeaways-from-openai-5/.
6. Carroll, M. et al.: On the Utility of Learning about Humans for Human-AI Coordination. In: Advances in Neural Information Processing Systems 32 (NeurIPS 2019) (2019).
7. Nalepka, P. et al.: Interaction Flexibility in Artificial Agents Teaming with Humans. In: CogSci 2021 [In press] (2021).
8. Bain, M., Sammut, C.: A Framework for Behavioural Cloning. In: Machine Intelligence 15, Intelligent Agents [St. Catherine's College, Oxford, July 1995]. pp. 103–129. Oxford University, GBR (1999).
9. Ho, J., Ermon, S.: Generative Adversarial Imitation Learning. arXiv. 1606.03476, (2016).

10. Maclin, R. et al.: Giving Advice about Preferred Actions to Reinforcement Learners Via Knowledge-Based Kernel Regression. In: AAAI (2005).
11. Amodei, D. et al.: Concrete Problems in AI Safety. arXiv. 1606.06565, (2016).
12. Osa, T. et al.: An Algorithmic Perspective on Imitation Learning. Found. Trends Robot. 7, 1–179 (2018).
13. Patil, G. et al.: Hopf Bifurcations in Complex Multiagent Activity: The Signature of Discrete to Rhythmic Behavioral Transitions. Brain Sci. 10, 536 (2020).
14. Nalepka, P. et al.: Human social motor solutions for human–machine interaction in dynamical task contexts. Proc. Natl. Acad. Sci. U. S. A. 116, 1437–1446 (2019).
15. Richardson, M.J. et al.: Modeling Embedded Interpersonal and Multiagent Coordination. In: Proceedings of the 1st International Conference on Complex Information Systems. pp. 155–164. SCITEPRESS - Science and and Technology Publications (2016).
16. Warren, W.H.: The Dynamics of Perception and Action. Psychol. Rev. 113, 358–389 (2006).
17. Kelso, J.A.S.: Dynamic Patterns: The Self-organization of Brain and Behavior. MIT Press (1997).
18. Schmidt, R., Richardson, M.: Dynamics of interpersonal coordination. Coord. Neural, Behav. Soc. Dyn. 281–308 (2008).
19. Nalepka, P. et al.: Herd Those Sheep: Emergent Multiagent Coordination and Behavioral-Mode Switching. Psychol. Sci. 28, 630–650 (2017).
20. Sternad, D. et al.: Bouncing a ball: Tuning into dynamic stability. J. Exp. Psychol. Hum. Percept. Perform. 27, 1163–1184 (2001).
21. Fajen, B.R. et al.: A dynamical model of visually-guided steering, obstacle avoidance, and route selection. Int. J. Comput. Vis. 54, 13–34 (2003).
22. Lamb, M. et al.: To Pass or Not to Pass: Modeling the Movement and Affordance Dynamics of a Pick and Place Task. Front. Psychol. 8, 1061 (2017).
23. Ijspeert, A.J. et al.: Dynamical Movement Primitives: Learning Attractor Models for Motor Behaviors. Neural Comput. 25, 328–373 (2013). https://doi.org/10.1162/NECO_a_00393.
24. Hogan, N., Sternad, D.: On rhythmic and discrete movements: Reflections, definitions and implications for motor control. Exp. Brain Res. 181, 13–30 (2007).
25. Kay, B.A. et al.: Space-time behavior of single and bimanual rhythmical movements: data and limit cycle model. J. Exp. Psychol. Hum. Percept. Perform. 13, 178–192 (1987).
26. Vesper, C. et al.: Joint Action: Mental Representations, Shared Information and General Mechanisms for Coordinating with Others. Front. Psychol. 07, 2039 (2017).
27. Repp, B.H., Keller, P.E.: Adaptation to tempo changes in sensorimotor synchronization: Effects of intention, attention, and awareness. Q. J. Exp. Psychol. Sect. A Hum. Exp. Psychol. 57, 499–521 (2004).
28. Lagarde, J.: Challenges for the understanding of the dynamics of social coordination. Front. Neurorobot. 7, (2013).
29. Richardson, M.J. et al.: Challenging the egocentric view of coordinated perceiving, acting, and knowing. mind Context. 307–333 (2010).
30. Schmidt, R.C., O'Brien, B.: Evaluating the Dynamics of Unintended Interpersonal Coordination. Ecol. Psychol. 9, 189–206 (1997).
31. Lamb, M. et al.: A Hierarchical Behavioral Dynamic Approach for Naturally Adaptive Human-Agent Pick-and-Place Interactions. Complexity. 2019, (2019).

32. Yokoyama, K., Yamamoto, Y.: Three People Can Synchronize as Coupled Oscillators during Sports Activities. PLoS Comput. Biol. 7, e1002181 (2011).

33. Zhang, M. et al.: Critical diversity: Divided or United States of social coordination. PLoS One. 13, (2018).

34. Schaal, S. et al.: Learning movement primitives. Robot. Res. 15, 1–10 (2005).

35. Schaal, S. et al.: Nonlinear Dynamical Systems as Movement Primitives. Int. Conf. Humanoid Robot. Cambridge MA. 38, 117–124 (2001).

36. Ijspeert, a. J. et al.: Movement imitation with nonlinear dynamical systems in humanoid robots. Proc. 2002 IEEE Int. Conf. Robot. Autom. (Cat. No.02CH37292). 2, 1–6 (2002).

37. Mukovskiy, A. et al.: Modeling of coordinated human body motion by learning of structured dynamic representations. In: Springer Tracts in Advanced Robotics. pp. 237–267. Springer Verlag (2017).

38. Nalepka, P. et al.: "Human-like" emergent behavior in an evolved agent for a cooperative shepherding task. In: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2017), Vancouver, Canada. (2017).

39. Sutton, R.S., Barto, A.G.: Reinforcement learning: an introduction, Second Edition. MIT Press (2017).

40. Arulkumaran, K. et al.: Deep reinforcement learning: A brief survey. IEEE Signal Process. Mag. 34, 26–38 (2017).

41. Mnih, V. et al.: Asynchronous Methods for Deep Reinforcement Learning. Mach. Learn. (2016).

42. Lillicrap, T.P. et al.: Continuous control with deep reinforcement learning. In: 4th International Conference on Learning Representations, ICLR 2016 - Conference Track Proceedings. International Conference on Learning Representations, ICLR (2016).

43. Tampuu, A. et al.: Multiagent cooperation and competition with deep reinforcement learning. PLoS One. 12, (2017).

44. Hester, T. et al.: Learning from Demonstrations for Real World Reinforcement Learning. arXiv. 1704.03732, (2017).

45. Hussein, A. et al.: Imitation Learning: A Survey of Learning Methods. ACM Comput. Surv. 50, 1–35 (2017).

46. Nalepka, P. et al.: Practical Applications of Multiagent Shepherding for Human-Machine Interaction. In: PAAMS 2019: Advances in Practical Applications of Survivable Agents and Multi-Agent Systems. pp. 168–179. Springer, Cham (2019).

47. Auletta, F. et al.: Herding stochastic autonomous agents via local control rules and online global target selection strategies. arXiv. 2010.00386, (2020).

48. Auletta, F. et al.: Human-inspired strategies to solve complexjoint tasks in multi agent systems. (2021).

49. Rigoli, L.M. et al.: Employing Models of Human Social Motor Behavior for Artificial Agent Trainers. In: An, B. et al. (eds.) Proceedings of the 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2020). p. 9. International Foundation for Autonomous Agents and Multiagent Systems, Auckland, New Zealand (2020).

50. Juliani, A. et al.: Unity: A General Platform for Intelligent Agents. arXiv. (2018).