Autonomous Reinforcement of Behavioral Sequences in Neural Dynamics

Sohrob Kazerounian*‡, Matthew Luciw*‡, Yulia Sandamirskaya†, Mathis Richter†,
Jürgen Schmidhuber*, and Gregor Schöner†
*IDSIA, Galleria 2, Manno CH-6928, Switzerland
†Ruhr-Universität Bochum, Institut für Neuroinformatik Universitätstr, Bochum, Germany
*‡ The first two authors should be considered as first authors.

Introduction. Computational approaches to reinforcement learning (RL) often formalize the learning problem in terms of discrete state and action spaces, on which the learning agent operates in discrete time [1]. The problem of how these discrete representations emerge from the continuous in time and in space sensory-motor representations is often not addressed in the RL literature. On the other hand, some RL models in neuroscience include the continuous neural representations of states and actions, but they do not address the problem of learning *sequences* of behaviors through reinforcement, as well as how these sequences may be generated using real sensors and motors [2], [3]. Here, we present a neural-dynamic model that implements a RL agent, which is able to acquire action sequences based on a reward signal and uses the state and action representations that may be continuously linked to raw perceptual inputs and motor dynamics. In the neural-dynamic RL architecture, the behavioral decisions are modeled as instabilities of continuous in time and graded in space dynamics of neural fields. These instabilities demarcate transitions between stable states that represent the agent's actions, which unfold continuously in physical time and environment. The stable states, emerging from the continuous dynamics, provide the basis for building neural-dynamic representations of previously selected state-action pairs, their eligibility traces, and value function of the reinforcement learner. The model uses the neural-dynamic framework of Dynamic Field Theory (DFT) [4] to represent the behaviors of the agent that form the state-action space, on which learning operates. The RL algorithm known as State-Action-Reward-State-Action (SARSA) is used to implement the reinforcement learning of action sequences using the neural-dynamics representations. We provide a neurally grounded method for autonomously discretizing behaviors occurring in continuous time and show how the neural-dynamic framework enables grounding of RL in continuous sensory-motor processes. We implement the model, which we call DN-SARSA(λ), in a simple color-search scenario and demonstrate its functioning on a real robot.

Architecture. The DN-SARSA(λ) model consists of a neural-dynamic architecture for generation of behavioral sequences and a dynamical reinforcement learner. A number of coupled dynamic neural fields (DNFs) [5] and neural nodes form a representation of the elementary behaviors (EBs) of the agent's behavioral repertory. Each EB has a DNF representation of the intention and of the condition-of-satisfaction (CoS) of the respective behavior. Both these representations are graded in space and continuous in time attractor dynamics, which may be coupled to perceptual and motor systems of a robotic agent. The intention DNF interacts with bottom-up sensory inputs to drive low-level motor commands. Activation of the CoS DNF indicates that the currently active behavior has completed [6].

For the reinforcement learner, an active CoS field represents the *state*, in which the agent decides, which *action* to activate next (represented by the intention DNF of the next EB). A state/action DNF of the reinforcement learner receives inputs from CoS fields and the intention fields of the EBs and builds a peak of positive activation in each transition phase between EBs, when the CoS field of the previous EB is still active and the intention field of the next EB is already activated. The positive activation in the state/action DNF ultimately serves as input to an Item and Order working memory system [7], [8]. Activity in this system represents an eligibility trace, since the more recently occurring state/action transitions result in higher levels of activity than those state/action transitions having occurred further in the past. This pattern of activity excites a value opposition (VO) field, which sets input to a dynamical array performing calculation of a Temporal Difference (TD)-error. The calculated value of the TD-error modulates learning, which is implemented as a Hebbian learning process, whose long-term memory values represent the stored Q-values of the reinforcement learner. The Q-values are updated in the learning process and are utilized during sequence production to select the next EBs.

Simulation Results. The model is tested on a robotic vehicle simulated in the Webots simulator, performing a search for rewarding sequences of colored blocks, as illustrated in Fig. 1(a). The E-Puck robot is surrounded by 16 blocks of four different colors (red(R), green(G), blue(B), yellow(Y)), which are picked up by the robot's camera and are represented as localized color-space distributions in the perceptual DNF. The robot "finds" a particular color, as determined by the currently active intention node, by rotating on the spot so that an object of the given color falls onto the center of the image of the vehicle's camera. Once centered, activation in the CoS node of the particular EB initiates a new EB to be performed (i.e., a new color to be searched for). If the robot finds the correct five color sequence, a positive reward is provided for a few time steps.

A complete report on DN-SARSA(λ) can be found at http://www.idsia.ch/~luciw/icdl2012.html

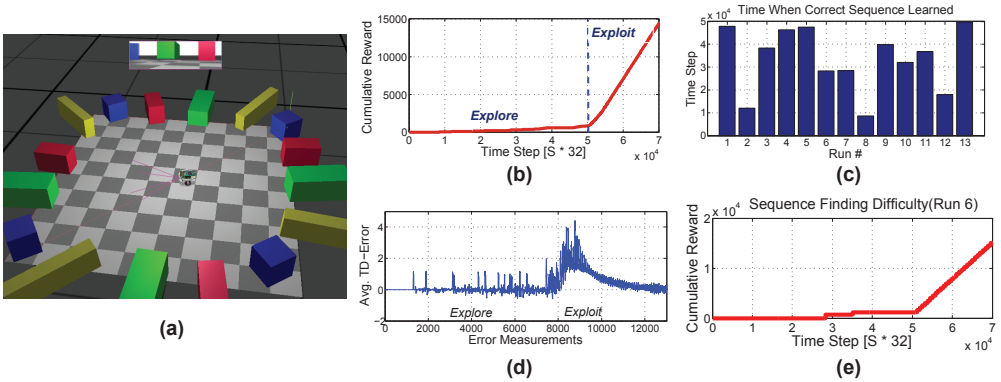

Fig. 1. (a) Simulation environment in which a e-puck vehicle at the center rotates on the spot to direct its camera at colored objects and is rewarded for doing so in a particular order of colors. (b) Cumulative reward as a function of time averaged across 13 runs. In the first $50,000$ time steps (32 time steps per second), the system randomly selects intended colors; thereafter it selects the most valuable intended color. (c) Time needed to learn the rewarded sequence in each of the 13 runs. (d) Average TD-error. (e) The cumulative reward from example run (6)

Results of the robot's learning performance are illustrated in Fig. 1. In all trials in which the robot uncovered the rewarding sequence in exploration mode, it was able to eventually execute the optimal policy in exploitation mode. In some trials, the optimal policy was attained only in the exploitation phase, which showed that it is useful to maintain learning both during exploration and exploitation. Learning in the exploitation phase consists primarily of *unlearning* incorrect "shortcuts" inherited from the exploration phase. This occurs, for example, when the robot finds the sequence, and correctly values the transition from $R \rightarrow G$ the most, but incorrectly also values the transition from any other color than Y to R. This occurs since we handle the POMDP difficulty with the eligibility trace (rather than e.g., memory-based state estimation), where the value can fill in for the reward, enabling shortcuts. However, during exploitation the robot realizes that shortcuts do not lead to reward (by executing them and not receiving any reward). Their values are diminished until the true rewarding sequence remains.

Fig. 1(c) shows the times when the sequence was first uncovered and Fig 1(e) illustrates the reward from one run, in which the robot finds the target sequence a first time after about $30,000$ steps. When the system enters exploitation mode its starts maximizing reward by doing the correct thing over and over again until the simulation ends. Fig. 1(d) shows the averaged TD-error, illustrating that the neural system learns to predict discounted future reward. The detection of reward acts as an instability for the reinforcement learner, and the learning mechanism is simply a constant drive towards stability.

Transfer to Real Robot. To show that our system can deal with real sensory information and real motor system, we transferred a set of weights learned from a successful run of simulation to a real E-puck. A video of the robot successfully moving through two iterations of the sequence is at http://www.idsia.ch/~luciw/videos/DFTBot.mp4. The success of transfer onto a real robotic system shows that the DN-SARSA reinforcement learner brings about a representation that is capable to produce behavior in the robot based on continuous (raw) visual input and physical motors, driven by continuous-time dynamics.

Conclusion. The DN-SARSA(λ) model provides a framework which shows how computational learning algorithms can be incorporated into a continuous neural-dynamical model. This enables autonomous learning and acting in continuous and dynamic environments, a challenge that is easily overlooked when formalizing the learning problem in discretized spaces without accounting for their coupling to sensory-motor dynamics.

Acknowledgement. The authors gratefully acknowledge the financial support of the European Union Seventh Framework Program FP7-ICT-2009-6 under Grant Agreement no. 270247 – NeuralDynamics.

REFERENCES

[1] R. Sutton and A. Barto, *Reinforcement learning: An introduction*. Cambridge Univ Press, 1998, vol. 1, no. 1.
[2] W. Schultz, P. Dayan, and P. Montague, "A neural substrate of prediction and reward," *Science*, vol. 275, no. 5306, pp. 1593–1599, 1997.
[3] G. Berns and T. Sejnowski, "A computational model of how the basal ganglia produce sequences," *Journal of Cognitive Neuroscience*, vol. 10, no. 1, pp. 108–121, 1998.
[4] G. Schöner, "Dynamical systems approaches to cognition," *Campridge Handbook of Computational Modelling*.
[5] S. Amari, "Dynamics of pattern formation in lateral-inhibition type neural fields," *Biological Cybernetics*, vol. 27, pp. 77–87, 1977.
[6] Y. Sandamirskaya, M. Richter, and G. Schoner, "A neural-dynamic architecture for behavioral organization of an embodied agent," in *Development and Learning (ICDL), 2011 IEEE International Conference on*, vol. 2. IEEE, 2011, pp. 1–7.
[7] S. Grossberg and S. Kazerounian, "Neural dynamics of speech perception: Phonemic restoration in noise using subsequent context." *Journal of the Acoustical Society of America*, vol. 125, no. 1, 2011.
[8] S. Grossberg, "Behavioral contrast in short-term memory: Serial binary memory models or parallel continuous memory models?" *Journal of Mathematical Psychology*, vol. 3, pp. 199–219, 1978.