Reinforcement Learning in Continuous State and Action Space

Thomas Strösslin and Wulfram Gerstner
Laboratory of Computational Neuroscience, EPFL, Lausanne, Switzerland

***Abstract*— To solve complex navigation tasks, autonomous agents such as rats or mobile robots often employ spatial representations. These "maps" can be used for localisation and navigation. We propose a model for spatial learning and navigation based on reinforcement learning. The state space is represented by a population of hippocampal place cells whereas a large number of locomotor neurons in nucleus accumbens forms the action space. Using overlapping receptive fields for both populations, state/action mappings rapidly generalise during learning. The population vector allows a continuous interpretation of both state and action spaces. An eligibility trace is used to propagate reward information back in time. It enables the modification of behaviours for recent states. We propose a biologically plausible mechanism for this trace of events where spike timing dependent plasticity triggers the storing of recent state/action pairs. These pairs, however, are forgotten in the absence of a reward-related signal such as dopamine. The model is validated on a simulated robot platform.**

I. Introduction

Animals show various behaviours when solving navigational tasks. The selection of an appropriate strategy for a given task depends on its complexity and on the available sensorial input. A stimulus-response behaviour, for instance, is sufficient for a rat to navigate to visibly marked food (taxon navigation [1]). In tasks where the goal is hidden, however, a representation of the environment is needed (locale navigation [2]).

The hippocampal formation of rats seems to contain a spatial representation which is important for complex navigation tasks. It receives highly processed multimodal sensory information and is a likely neural basis for spatial coding [2]–[4]. Hippocampal place cells (PCs) discharge selectively as a function of the position of the rat in the environment. Lesion studies show that the hippocampus is necessary for locale–but not for taxon navigation [5].

Reinforcement learning (RL) [6] has previously been used to solve navigation tasks for autonomous mobile agents [7]–[10]. Some models operate in continuous state and/or action spaces using function approximation [7], [10]–[12]. In most RL-based models, an eligibility trace [6] is used to speed-up learning artificially and no underlying biological mechanism is proposed.

PCs as well as dopaminergic neurons project onto the nucleus accumbens, an area which is related to motor control [9], [13], [14]. The output of dopamine neurons has been shown to code for errors in reward prediction. These errors are closely related to reinforcement learning [15]–[18].

Here we describe a spatial learning system based on reinforcement learning. In particular, we focus on a mechanism by which state – as well as action space become continuous. Learning quickly generalises in both spaces. We also propose a biologically plausible mechanism for eligibility traces which allow rewarding events to generalise back in time.

II. Model

In our model, a spatial representation similar to [7] serves as state space. It consists of a population of hippocampal place cells (PCs) with highly overlapping receptive fields. Here we focus on the use of this representation for navigation. PCs project onto a population of action cells (ACs). A navigation map is learnt using reinforcement learning and stored in PC→AC synapses.

Reinforcement learning has been used for problems where a small discrete set of actions is available to choose from at each state. The number of states usually is discrete and finite. The population vector of PCs, however, can be interpreted as the continuous state variable which represents the agent's location $\vec{x} \in \mathbb{R}^2$ in the environment. Similarly, the population vector of ACs stands for a continuous action. Although the population of ACs may be large, the learning speed is unaffected. The model is tested on a simulated mobile agent. Fig. 1 shows the architecture of the navigation system.

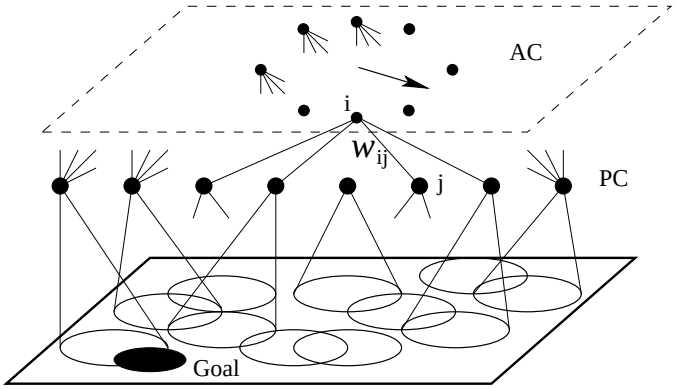


Fig. 1. *Architecture of our navigation system:* A layer of hippocampal place cells (PCs) represent the square environment. Each PC is active in a small portion of the environment and their receptive fields overlap. In order to learn to navigate to the goal, PCs are connected to action cells (ACs) which code for the direction of the next move. The population vectors of both layers allow a continuous interpretation of position (PCs) and direction of action (ACs).

The spatial representation is constructed in a separate exploration phase. The agent stores local views of each visited location and establishes a purely visual spatial map. This

visual place code projects to the PC layer shown in Fig. 1. Additionally, an internal map using path integration also converges onto PCs such as to reduce ambiguities in the visual map. The result are nicely tuned place cells with overlapping receptive fields. For details, see [7].

A. Action cells

A population of N^{AC} action cells (ACs) code for the agent's motor-commands. Each AC i represents a particular direction ϕ_i, which are uniformly distributed between 0 and 2π. The angle ϕ^{AC} of the AC population vector determines the direction of the next movement. An action consists of moving the agent in the desired direction for a fixed distance or until a wall is blocking the way. With r_k^{AC} being the firing rate of action cell k, the population vector can be written as:

$$\phi^{AC} = arctan\left(\frac{\sum_k r_k^{AC} \cdot sin(2\pi k/N^{AC})}{\sum_k r_k^{AC} \cdot cos(2\pi k/N^{AC})}\right)$$

The nucleus accumbens, situated in the ventral striatum, seems to be involved in goal-oriented navigation. It receives reward-related information from dopaminergic neurons of the ventral tegmental area as well as spatial information from the hippocampus. Its output is related to locomotion [9], [13], [14], [18]. We therefore assume that the biological locus of our modelled action cells may be the nucleus accumbens.

The activity of ACs can be divided in two phases which are separated in time. In each phase, ACs code for a different property:

Action-evaluation: First, ACs receive state information from PCs and learn to attribute a value to each action. This tells the agent which actions are good in the current situation. The firing rate ${}^1r_i^{AC}$ represents the estimated value $Q(s, a_i)$ for the current state s and action a_i. In contrast to most other models, we don't use the max-operator to determine the optimal action. Instead, we use the direction of the AC population vector. It represents the continuous action a^o (direction ϕ^o) which maximises the total future reward, given the current estimation of Q-values.

$${}^1r_i^{AC} = \sum_j w_{ij} \cdot r_j^{PC}$$

Generalisation: As soon as an action is selected, a Gaussian AC activity profile with variance σ_{AC}^2 is enforced around the selected action a^x (direction ϕ^x). The firing rates ${}^2r_i^{AC}$ then represent the action which was selected for execution. Biologically, this can be achieved using lateral connections between action cells. It is this activity profile which results in generalisation in action space. The width of the profile has a fixed physical meaning (an angle) and is independent of the number of ACs. If $\Delta\phi_i$ stands for the angular distance between ϕ^x and ϕ_i, the profile can be expressed as:

$${}^2r_i^{AC} = exp(-\Delta\phi_i^2/2\sigma_{AC}^2)$$

After learning, the AC population vector of the action-evaluation phase points in the direction of the goal for all locations in the environment and thus forms a navigational map.

In addition to excitatory space-related input, the hippocampus also receives rhythmic inhibitory input from the medial septum. This theta-rhythm could provide a separation of the two phases: First, low theta-activity would pass spatial information from hippocampus to nucleus accumbens. Later, when theta shuts-off place cell activity, lateral dynamics could shape the activity profile on action cells.

B. Eligibility trace

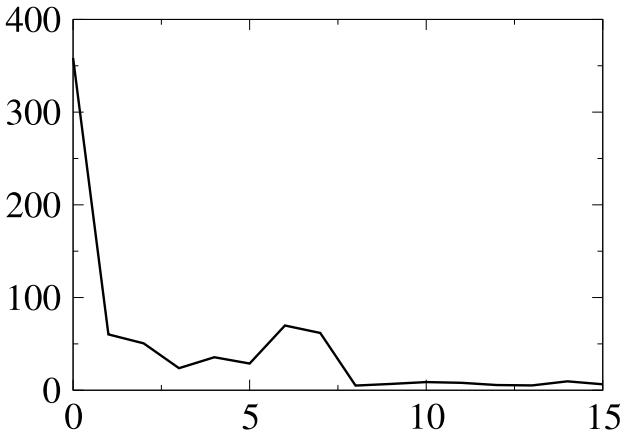
Rats quickly learn to navigate to a rewarding location from any place in the environment. If the reward is solely given in the goal state, learning propagates only slowly backwards to previously visited locations. Generalisation in state space, as provided by the large overlap of hippocampal PCs, is one improvement towards faster learning. It seems more natural, however, to propagate information back in *time* such as to optimise the behaviour in recent situations. An eligibility trace serves exactly this purpose. It is a fading memory device which stores past state/action pairs. In our model, the trace p_{ij} on the synapse from PC j to AC i decays exponentially with α in time. It is formally expressed as:

$$p_{ij}(t) = \alpha \cdot p_{ij}(t-1) + {}^2r_i^{AC}(t) \cdot r_j^{PC}(t)$$

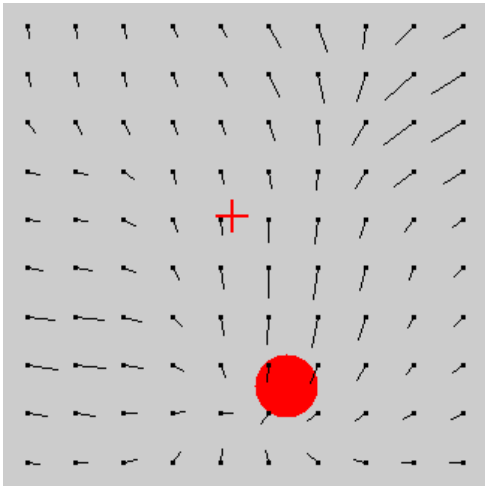
In animals it is still unclear, however, how a reward can influence learning in a synapse which was active in the past. Sustained neuronal activity, as seen in in primate striatum, is one possibility to implement a short term memory [19]. Other candidates are based on intra-synaptic concentration of agents related to plasticity which operate on a longer timescale than electrical activity. Examples include calcium concentration [20] or calmodulin-dependent protein kinase II [15], [21]. Yet another possibility is that rewards enhance the storage of a "replay", instead of the original event.

We propose that eligibility traces are implemented using dopamine-modulated spike timing dependent plasticity (STDP). Long term potentiation (LTP) on PC→AC synapses is triggered by STDP for each state/action pair which was tried during goal search [22]–[24]. We assume here that the process of action-selection is quick, such that most of the PC activity is paired with AC activity of the generalisation phase (${}^1r_i^{AC}$), and not the action-evaluation phase (${}^2r_i^{AC}$). However, LTP is modulated by a dopamine signal which codes for the reward prediction error [15]–[18], [21], [25], [26]. A decrease in dopamine due to punishment blocks LTP induction whereas an unexpected reward increases dopamine release and enhances LTP.

STDP triggers a complex chain of biochemical processes, possibly at both pre-and postsynaptic sites. Two conditions need to be fulfilled in order to make the synapse itself suitable for an eligibility trace. First, the changes in concentration of synaptic agents must be slow enough so that the memory



(a) *Escape latency vs. number of trials:* Escape latency is the number of timesteps it takes the agent to find the goal. After about 10 trials, the task has been learnt.



(b) *Navigation map:* At each sample location, the line points in the direction of the optimal action. The shaded area is the goal-location and the cross marks the initial location of the last trial

Fig. 2. Results in a $80 \times 80cm$ simulated robot environment

does not fade too quickly. Secondly, a reward signal such as dopamine must be capable of changing the course of plasticity *after* the induction of a change in synaptic strength. However, it seems that dopamine needs to be present during or even before [26], or at most hundreds of milliseconds after induction [15] in order to modulate LTP. An alternative explanation is that dopamine inhibits depotentiation after LTP induction. Synapses are vulnerable to depotentiation after LTP induction. Depotentiation has been found to occur when low-frequency stimulation is applied after LTP induction. However, if dopamine is present at the synapse, depotentiation is completely blocked [25]. This mechanism operates at a timescale of seconds or even minutes after induction, which seems an appropriate range for eligibility traces.

C. Learning algorithm

To build a navigation map, the synaptic strengths w_{ij} from PCs j to ACs i need to be learnt. This section describes how this is achieved.

At each timestep t, all action values $Q(s(t), a_i)$ are calculated. Next, a continuous action $a^x(t)$ is selected. Most of the time, the optimal action $a^o(t)$ is chosen. In contrast to most other models, we don't use the max-operator to determine the optimal action. Instead, we use the direction of AC population vector of the action-evaluation phase ${}^1\phi^{AC}$ as the optimal action. Sometimes, however, an ϵ-greedy mechanism selects a non-optimal direction. This ensures sufficient exploration of the action space [6]. Then, the AC activity profile ${}^2r^{AC}(t)$ is enforced around the selected action and the eligibility trace $p_{ij}(t)$ updated. After taking action $a^x(t)$, the immediate reward $R(t+1)$ is inspected. Finally, the synaptic weights w_{ij} are updated. The following list briefly illustrates these steps:

1) Calculate action values: $Q(s(t), a_i) = {}^1r_i^{AC}(t)$.
2) Select action: $a^x(t) = a^o(t)$ with probability $1 - \epsilon$ (exploitation) or randomly select action with probability ϵ (exploration).
3) Generalise in action space: Lateral connections impose activity profile ${}^2r_i^{AC}(t)$ around the selected action $a^x(t)$.
4) Update eligibility trace $p_{ij}(t)$.
5) Execute action $a^x(t)$ and advance time ($t = t + 1$).
6) Calculate reward prediction error: $\delta(t) = R(t) + \gamma \cdot Q(s(t), a^o(t)) - Q(s(t-1), a^x(t-1))$. Note that the PC and AC activities for the new location have to be updated to calculate $Q(s(t), a^o(t))$.
7) Update synaptic strengths (η is the learning rate): $\Delta w_{ij}(t) = \eta \cdot \delta(t) \cdot p_{ij}(t-1)$.

The reward prediction error $\delta(t)$ can be interpreted as the output of dopaminergic neurons in the ventral tegmental area. One problem which is not addressed here is how dopaminergic neurons can process information coming from different timesteps. We assume that dopamine neurons receive action values via two separate pathways, one of which has a delay-line [15].

As there is a discrete set of ACs coding for a continuous direction $\phi \in [0..2\pi]$, action values need to be generalised. For sake of simplicity, we use linear interpolation between the Q-values of the two neighbouring ACs to calculate $Q(s, a^x)$ and $Q(s, a^o)$.

III. Results and Conclusions

The model is tested with a simulated agent in a square environment ($80 \times 80cm$). An ϵ-greedy policy with varying ϵ is used to balance exploration vs. exploitation: In each trial, the agent first tries to find the goal using its current knowledge (exploitation, low ϵ). The probability for exploration increases exponentially with the trial time, up to some fixed maximum. The agent then mainly explores for a fixed period, before ϵ is reset to its initial value and the cycle restarts. This ensures that the agent uses its knowledge instantly at the beginning of each trial, resorting to exploration only when the goal can't be found. Each time a wall is hit, a negative reward is given. This results in an obstacle-avoidance behaviour. When the goal is reached, a positive reward is given and the trial ends. The agent is then put on a random location in the environment and the next trial begins. Eligibility traces are cleared whenever a direct reward is given.

Fig. 2(a) presents the mean number of timesteps to find the goal vs. the number of learning trials. After about 10 trials, the task is learnt. Fig. 2(b) shows the navigation map after 10 trials. At each sample location, the line points in the direction of the AC population vector. The shaded area represents the goal. The agent has indeed learnt to navigate to the goal from all locations in the environment.

By qualitative analysis, the navigation map and escape latency are stable long before the Q-values have reached their final values. Navigation maps after around 100 trials look much the same than after 10 trials. Nevertheless, a systematic study of the convergence properties in the case of overlapping state and action values should be performed. Its dependence on the policy might be very useful for designing optimal learning strategies.

The problem of how time-constants of the order of seconds are produced in the brain is still unsolved. For the eligibility trace to be efficient, however, long time-constants are needed. Most of the protocols to assess the influence of dopamine on LTP neglect the importance of timing. However, it is crucial for reinforcement learning systems that a reward can alter learning *after* the induction of LTP.

Here we show that reinforcement learning in continuous state and action spaces can be solved efficiently. The performance of our learning mechanism does not depend on the number of neurons because tuning-widths are attached to physical units (positions for states and angles for actions). Although we don't model eligibility traces in detail, we show how they could be implemented in the brain. We will explore these models in future work.

References

[1] H. Schöne, *Spatial Orientation: Spatial Control of Behavior in Animals and Man*. NJ: Princeton, 1984.

[2] J. O'Keefe and L. Nadel, *The Hippocampus as a Cognitive Map*. Oxford: Clarendon Press, 1978.

[3] J. O'Keefe and J. Dostrovsky, "The hippocampus as a spatial map. preliminary evidence from unit activity in the freely-moving rat," *Brain Research*, vol. 34, pp. 171–175, 1971.

[4] E. Save, L. Nerad, and B. Poucet, "Contribution of multiple sensory information to place field stability in hippocampal place cells," *Hippocampus*, vol. 10, pp. 64–76, 2000.

[5] J. M. Pearce, A. D. L. Roberts, and M. Good, "Hippocampal lesions disrupt navigation based on cognitive maps but not heading vectors," *Nature*, vol. 396, pp. 75–77, 1998.

[6] R. Sutton and A. G. Barto, *Reinforcement Learning - An Introduction*. MIT Press, 1998.

[7] A. Arleo and W. Gerstner, "Spatial cognition and neuro-mimetic navigation: A model of hippocampal place cell activity," *Biological Cybernetics, Special Issue on Navigation in Biological and Artificial Systems*, vol. 83, pp. 287–299, 2000.

[8] D. J. Foster, R. G. M. Morris, and P. Dayan, "A model of hippocampally dependent navigation, using the temporal difference learning rule," *Hippocampus*, vol. 10(1), pp. 1–16, 2000.

[9] M. A. Brown and P. E. Sharp, "Simulation of spatial-learning in the Morris water maze by a neural network model of the hippocampal-formation and nucleus accumbens," *Hippocampus*, vol. 5, pp. 171–188, 1995.

[10] J. del R. Millán, D. Posenato, and E. Dedieu, "Continuous-action q-learning," *Machine Learning*, vol. 49, pp. 247–265, 2002.

[11] J. C. Santamaría, R. S. Sutton, and A. Ram, "Experiments with reinforcement learning in problems with continuous state and action spaces," *Adaptive Behavior*, vol. 6, no. 2, pp. 163–218, 1998.

[12] K. Doya, "Reinforcement learning in continuous time and space," *Neural Computation*, vol. 12, pp. 219–245, 2000.

[13] I. Q. Whishaw and G. Mittleman, "Hippocampal modulation of nucleus accumbens: Behavioral evidence from amphetamine-induced activity profiles," *Behavioral and Neural Biology*, vol. 55, pp. 289–306, 1991.

[14] M. Legault, P.-P. Rompré, and R. A. Wise, "Chemical stimulation of the ventral hippocampus elevates nucleus accumbens dopamine by activating dopaminergic neurons of the ventral tegmental area," *Journal of Neuroscience*, vol. 20, no. 4, pp. 1635–1642, 2000.

[15] J. C. Houk, J. L. Adams, and A. G. Barto, "A model of how the basal ganglia generate and use neural signals that predict reinforcement," in *Models of Information Processing in the Basal Ganglia*, J. C. Houk, J. L. Davis, and D. G. Beiser, Eds. Cambridge, Massachusetts, USA: MIT Press, 1995, ch. 13, pp. 249–270.

[16] W. Schultz, P. Dayan, and P. R. Montague, "A neural substrate of prediction and reward," *Science*, vol. 275, pp. 1593–1599, 1997.

[17] P. R. Montague, P. Dayan, and T. J. Sejnowski, "A framework for mesencephalic dopamine systems based on predictive hebbian learning," *Journal of Neuroscience*, vol. 16, no. 5, pp. 1936–1947, 1996. [Online]. Available: http://www.jneurosci.org/cgi/content/abstract/16/5/1936

[18] W. Schultz, "Predictive Reward Signal of Dopamine Neurons," *Journal of Neurophysiology*, vol. 80, pp. 1–27, 1998.

[19] W. Schultz, P. Apicella, R. Romo, and E. Scarnati, "Context-dependent activity in primate striatum reflecting past and future behavioral events," in *Models of Information Processing in the Basal Ganglia*, J. C. Houk, J. L. Davis, and D. G. Beiser, Eds. Cambridge, Massachusetts, USA: MIT Press, 1995, ch. 2, pp. 11–27.

[20] J. Wickens and R. Kötter, "Cellular models of reinforcement," in *Models of Information Processing in the Basal Ganglia*, J. C. Houk, J. L. Davis, and D. G. Beiser, Eds. Cambridge, Massachusetts, USA: MIT Press, 1995, ch. 10, pp. 187–214.

[21] N. Otmakhov, L. C. Griffith, and J. E. Lisman, "Postsynaptic inhibitors of calcium/calmodulin-dependent protein kinase type II block induction but not maintenance of pairing-induced long-term potentiation," *Journal of Neuroscience*, vol. 17, no. 14, pp. 5357–5365, 1997.

[22] W. Gerstner and W. Kistler, *Spiking Neuron Models: Single Neurons, Populations, Plasticity*. Cambridge University Press, 2002.

[23] W. Senn, H. Markram, and M. Tsodyks, "An algorithm for modifying neurotransmitter release probability based on pre- and post-synaptic spike timing," *Neural Computation*, vol. 13, no. 1, pp. 35–68, 2001.

[24] R. P. Rao and T. J. Sejnowski, "Spike-timing-dependent hebbian plasticity as temporal difference learning," *Neural Computation*, vol. 13, pp. 2221–2237, 2001.

[25] N. A. Otmakhova and J. E. Lisman, "D1/D5 dopamine receptors inhibit depotentiation at CA1 synapses via cAMP-dependent mechanism," *Journal of Neuroscience*, vol. 18, no. 4, pp. 1270–1279, 1998.

[26] ——, "D1/D5 dopamine receptor activation increases the magnitude of early long-term potentiation at CA1 hippocampal synapses," *Journal of Neuroscience*, vol. 16, no. 23, pp. 7478–7486, 1996.