UNIVERSITY COLLEGE LONDON

DOCTORAL THESIS

Hippocampal predictive maps of an uncertain world

Author:
Jesse Pepijn GEERTS

Supervisors:
Prof. Neil BURGESS
Dr. Marcus STEPHENSON-JONES

A thesis submitted in fulfillment of the requirements
for the degree of Doctor of Philosophy

in the

Institute of Cognitive Neuroscience
Sainsbury Wellcome Centre

Declaration of Authorship

I, Jesse Geerts, confirm that the work presented in this thesis is my own. Where information has been derived from other sources, I confirm that this has been indicated in the thesis.

Abstract

Humans and other animals can solve a wide variety of decision-making problems with remarkable flexibility. This flexibility is thought to derive from an internal model of the world, or 'cognitive map', used to predict the future and plan actions accordingly. A recent theoretical proposal suggests that the hippocampus houses a representation of long-run state expectancies. These "successor representations" (SRs) occupy a middle ground between model-free and model-based reinforcement learning strategies. However, it is not clear whether SRs can explain hippocampal contributions to spatial and model-based behaviour, nor how a putative hippocampal SR might interface with striatal learning mechanisms. More generally, it is not clear how the predictive map should encode uncertainty, and how an uncertainty-augmented predictive map modifies our experimental predictions for animal behaviour.

In the first part of this thesis, I investigated whether viewing the hippocampus as an SR can explain experiments contrasting hippocampal and dorsolateral striatal contributions to behaviour in spatial and non-spatial tasks. To do this, I modelled the hippocampus as an SR and DLS as model-free reinforcement learning, combining their outputs via their relative reliability as a proxy for uncertainty.

Current SR models do not formally address uncertainty. Therefore I extended the learning of SRs by temporal differences to include managing uncertainty in new observations versus existing knowledge. I generalise this approach to a multi-task setting using a Bayesian nonparametric switching Kalman Filter, allowing the model to learn and maintain multiple task-specific SR maps and infer which one to use at any moment based on the observations. I show that this Bayesian SR model captures animal behaviour in tasks which require contextual memory and generalisation.

In conclusion, I consider how the hippocampal contribution to behaviour can be considered as a predictive map when adapted to take account of uncertainty and combined with other behavioural controllers.

Impact statement

One of the most impressive features of cognitive function in humans and animals is its striking flexibility. Without explicit supervision, biological agents learn complex behaviours and flexibly adapt these behaviours to changing situations and environments. This stands in contrast to contemporary artificial systems: while these achieve impressive (even super-human) performance at particular tasks, they do not yet exhibit the type of domain-general *autonomous learning* that biological agents do. In this thesis, I study hypotheses for the representations and algorithms that the brain, particularly the hippocampus, employs for such autonomous learning while animals learn to navigate and to predict upcoming rewards.

Understanding these representations and algorithms will be key to understanding biological intelligence. The hippocampus, for example, is thought to play a key role in important aspects of intelligent behaviour such as memory, planning and contextual decision making. Knowing the computational problems that the hippocampus is faced with, and the algorithms it runs to solve these, will elucidate *how* the hippocampus performs this role and *why* the neural representations that we find there are there.

Such reverse-engineering of the brain could also benefit the parallel effort of developing novel methods in artificial intelligence (AI) and machine learning. At a minimum, careful behavioural characterisations of how and when biological agents outperform their machine counterparts can serve as a benchmark or goal formulation for AI systems. More speculatively, the brain's algorithmic solutions to these tasks – and their implementations in biological neural circuits – could inform designs of artificial neural networks that are able to match the brain's learning abilities.

Finally, the work presented here might have an impact in clinical psychiatry and neurology. A detailed understanding of specific cognitive processes underlying spatial learning may help us to understand in more detail how disease affects these processes as well as develop better diagnoses. Alzheimer's disease, for example, is known to develop in the hippocampal formation. Studying spatial processing in presymptomatic Alzheimer's patients has been demonstrated to provide a useful tool for early diagnosis.

Acknowledgements

This thesis was made possible by the immense support I received from the people around me during my time here at UCL. I started my PhD at a brand new institute, the SWC, and from year two onward I split my time between there and the ICN. I feel incredibly lucky to have been part of these two amazing research communities.

Thank you, Neil, for being a fantastic advisor and mentor. During the past three years I have learned an incredible amount from the manner in which you do science and the manner in which you distill fundamental scientific issues into clearly stated research questions. I try to emulate this way of thinking and it already makes me see things clearer. Equally importantly, I really enjoyed our personal interactions. You were always approachable for a chat with me or anyone, which quickly made me feel at home in the lab and created a really nice atmosphere in the group.

Thank you also to Kim Stachenfeld – who has been informally co-advising me – for inspiring me both as a scientist and as a person. You're very fun to work with and your enthusiasm for science is infectious: I left each of our supervisory meetings more motivated and inspired than before. On top of that you're extremely knowledgeable and you seemingly effortlessly come up with creative new ideas for both theories and experiments.

Neil and Kim have both been incredibly supportive this last year of my PhD, during which I (like many others) was not embedded in the normal academic work environment because of the Covid-19 pandemic. I feel very fortunate to have worked with this all-star supervisory team and I am excited to continue working together.

I am very grateful to Sam Gershman for contributing a lot to the ideas and experiments presented in Chapter 3. It has been fun and inspiring to discuss ideas with Sam, and I hope to continue collaborating in the future.

I would also like to thank my thesis committee – Caswell Barry, Tim Behrens and Marcus Stephenson-Jones – for taking the time to read through my thesis work and for their useful comments and suggestions.

A big thank you to the past and current members of the Space and Memory lab. Andrea, Andrej, Dan, Talfan, Laura, Robin, Alexa, Siti, Changmin, Bardur, Oliver, Ingrid, Eleanor, Luke, Sofie and Ewa made for a fun team at Queen Square; Henry, Tom and Mattias, it was great to finally have some Burgess lab vibes around the SWC too!

Thanks also to the many friends I made at the SWC and Gatsby Unit. When I first joined, the SWC PhD programme was still being developed, and I spent many fun hours learning about neuroscience methods with my PhD buddies Matt and Steve – it's not every day you get to build a two-photon microscope – and about theoretical neuroscience with Lea, Jorge and Kirsty. Thanks also to Adam Kampff and the whole lab, for making the first year of the PhD programme so fun and inspiring. Your no-black-boxes philosophy will stay with me.

A more personal acknowledgement also to the friends I made while in London. To my home office mates Annie, Quentin and Francesca, for the fun lunch-time discussions and for making me feel at home! And to my dual masters classmates, especially Eva, Cécile, Lindsay and Cassandra. It's been awesome discovering Paris and London with you.

Dankjewel ook aan mijn lieve vrienden thuis: David, Ilse, Henk, Theresa, Sjang, Dorien, Micha, Marie-Beth, Bas, Thomas, Roland, Thor. Ik ben blij dat ik jullie niet uit het oog verloren ben in al die tijd dat ik weg ben en ik heb veel zin jullie allemaal weer meer te zien!

Als laatst (but not least) dankjewel aan mijn lieve familie. Aan mijn ouders, voor teveel dingen om hier op te noemen; onder andere dat jullie me altijd ondersteunen bij wat ik ook wil doen (zelfs als ik daarvoor in een ander land wil zitten) en dat jullie me eraan blijven herinneren dat er altijd belangrijkere dingen zijn. En aan Yara en Jasper, ik ben blij en trots met jullie als broer en zus. Jasper, ik hoop wel dat je een keer stopt met titels vergaren – het kost me steeds meer tijd en moeite om je te evenaren! Dankjewel opa en oma voor al de ondersteuning, en ook oma Inge en opa Wim (het begon allemaal met opa's rekenpuzzels). Dankjewel Jolente, Vivian en Jarla; Jord, Jeske, Henriette en Walter; en Paul en Anneke.

Contents

List of Figures

List of Tables

List of Abbreviations

BOLD	**B**lood-**O**xygen-**L**evel-**D**ependent
BVC	**B**oundary **V**ector **C**ell
DLS	**D**orso-**L**ateral **S**triatum
DMS	**D**orso-**M**edial **S**triatum
fMRI	**f**unctional **M**agnetic **R**esonance **I**maging
GP	**G**aussian **P**rocess
HPC	**H**ippocampus
KF	**K**alman **F**ilter
KTD	**K**alman **T**emporal **D**ifferences
LDS	**L**inear-Gaussian **D**ynamical **S**ystem
MB	**M**odel-**B**ased
MDP	**M**arkov **D**ecision **P**rocess
mEC	**m**edial **E**ntorhinal **C**ortex
MF	**M**odel-**F**ree
PCA	**P**rincipal **C**omponent **A**nalysis
POMDP	**P**artially **O**bservable **M**arkov **D**ecision **P**rocess
RL	**R**einforcement **L**earning
RPE	**R**eward **P**rediction **E**rror
SPE	**S**ensory **P**rediction **E**rror
SMC	**S**equential **M**onte **C**arlo
SR	**S**uccessor **R**epresentation
TCM	**T**emporal **C**ontext **M**odel
TD	**T**emporal **D**ifference

List of Symbols

$\mathbb{I}$	indicator function
$\mathbf{x}$	continuous hidden state
$\mathbf{y}$	observations
K / $\boldsymbol{\kappa}$	Kalman gain
$\mathbf{w}$	(value) weights
M	successor representation matrix
s	state
a	action
V	state value
Q	state-action value
δ	prediction error
$\boldsymbol{\phi}$	state features
$\boldsymbol{\psi}$	successor features
$\boldsymbol{\xi}$	evolution noise
$\boldsymbol{\nu}$	observation noise
P_{ξ}	evolution noise covariance matrix
P_{ν}	observation noise covariance matrix

Chapter 1

Introduction

In this chapter, I introduce the necessary background for the work presented in this thesis. I will start, in Section 1.1, by discussing the cognitive neuroscience of spatial navigation, with a particular focus on the role of the hippocampal formation. Section 1.2 will comprise a discussion of reinforcement learning theory and its relevance to neuroscience and psychology. Finally, in Section 1.3 I will introduce uncertainty estimation using the Kalman Filter (Kalman, 1960) and related methods for recursive Bayesian estimation.

1.1 The neuroscience of navigation

1.1.1 Spatial navigation and the cognitive map

The ability to navigate to and from relevant goal locations is fundamental to the survival of all animals. This is a complex task involving many elements: animals need to localise themselves with respect to their environment, store memories of goal locations and learn or plan routes between them. A key question in neuroscience is how the brain achieves these.

A central idea within the literature on this spatial navigation problem is the notion that animals learn a rich internal representation, or "cognitive map" of the environment. These ideas can be traced back to early behavioural experiments by Tolman (1948), who trained rats to find a food reward on a maze. Tolman observed that animals learned to navigate to their goal *faster* if they had the opportunity to explore the environment (without a food reward) before the goal-directed navigation training began. This phenomenon, dubbed *latent learning*, suggests that animals learn to represent the environment even without reinforcement, and that this facilitates subsequent navigation abilities. Tolman saw further evidence for the cognitive

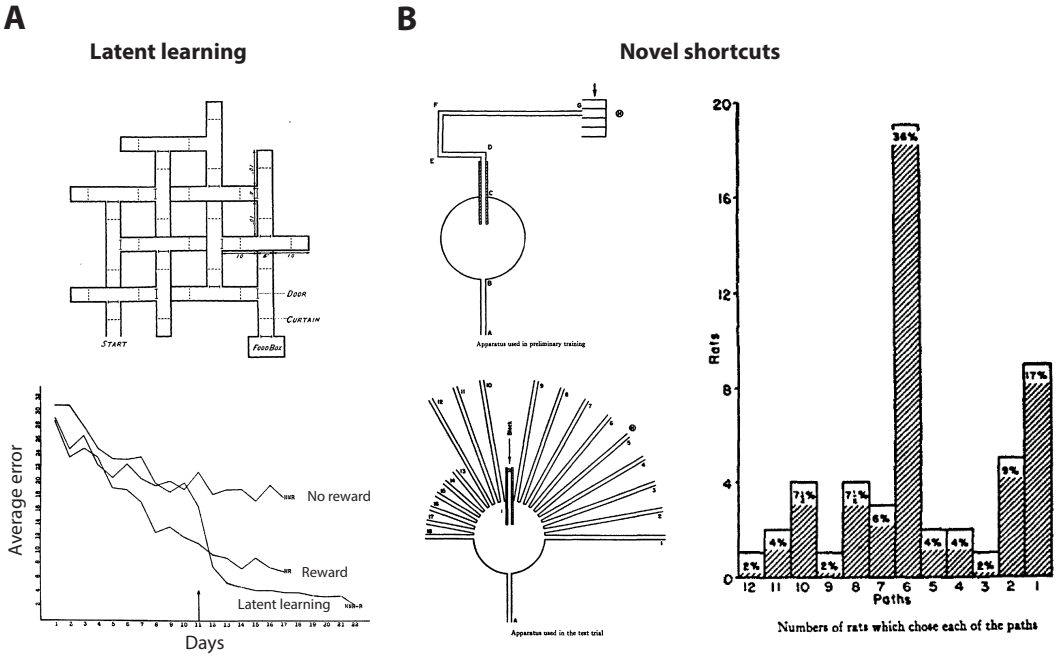


FIGURE 1.1: Tolman's experiments. **(A)** Rats learned to navigate from a start to a food box (top). Rats that received a reward in the food box learned better than rats that did not, but rats for which the reward was only presented on day 11 showed even higher performance, a phenomenon described as "latent learning" (bottom). **(B)** Rats were trained to take an indirect route in the maze shown in the top left panel. In a second phase, this route was blocked and rats were presented with multiple alternative options (bottom left). A majority of rats chose the route that led straight to the goal location, without having experienced this novel route before.

map come from a second experiment, in which animals were first trained to navigate an indirect route to a rewarded location. Then this long route was blocked, and animals were presented with multiple alternatives, all previously unexplored. The majority of animals directly chose the optimal route that took them directly to the goal, suggesting that the animals could compute the optimal heading direction (but see Grieves & Dudchenko, 2013).

Tolman argued that his experimental results suggest the existence of a cognitive map in the brain. This idea has become very influential, inspiring decades of neuroscience research aimed at finding where in the brain this map is stored, how it is learned and how it is used for navigation (Behrens et al., 2018). In the remainder of this section (1.1.2), I will give an overview of the hippocampal representations of space that have been found and are hypothesised to constitute the cognitive map.

In addition to the map-like navigation strategies described by Tolman,

there exists a more straightforward strategy for finding a goal, based on direct associations between local environmental stimuli and responses. In the spatial navigation literature, this is known as "response learning" (Chersi & Burgess, 2015). In section 1.2 I will describe how both stimulus-response learning and model-learning can be formally described in terms of Reinforcement Learning (RL) theories, and I will discuss previous work relating RL theory to hippocampal representations of space.

Finally, both stimulus-response and map-based learning strategies benefit from an optimal treatment of uncertainty. In section 1.3, I will introduce the Kalman filter (Kalman, 1960) and related methods as models of how the brain could achieve this.

1.1.2 Representations of space in the hippocampal formation

The hippocampus and surrounding areas (i.e. the hippocampal formation) have long been known to be important for spatial navigation. Experiments using various types of mazes demonstrated that certain types of navigation are particularly sensitive to hippocampal damage, including navigation to a hidden (unmarked) location from variable start locations and navigation that requires memory of previously visited locations (Cohen et al., 1971; Morris et al., 1982). Crucially, tasks that could be solved using local cues (allowing for simple response learning strategies) do *not* show a dependence on hippocampus, suggesting that the hippocampus is specifically necessary for navigation based on a cognitive map. This is further corroborated by the discovery of a plethora of map-like representations of space in and around the hippocampus. I will introduce the key spatial cell types that will be covered in this thesis below, as well as representations of non-spatial variables in the hippocampus. For more comprehensive discussions of spatial cell types see Hartley et al. (2014), Behrens et al. (2018) and Bicanski and Burgess (2020).

Place cells

O'Keefe and Dostrovsky (1971) discovered that single neurons in the hippocampus of freely moving rats fire only when the animal is in a particular location within an environment, but regardless of the animal's orientation (Figure 1.2A-B). The firing patterns of these cells are established very rapidly when an animal enters the environment for the first time and can remain stable for several days (Thompson & Best, 1990). Intriguingly, firing is robust to

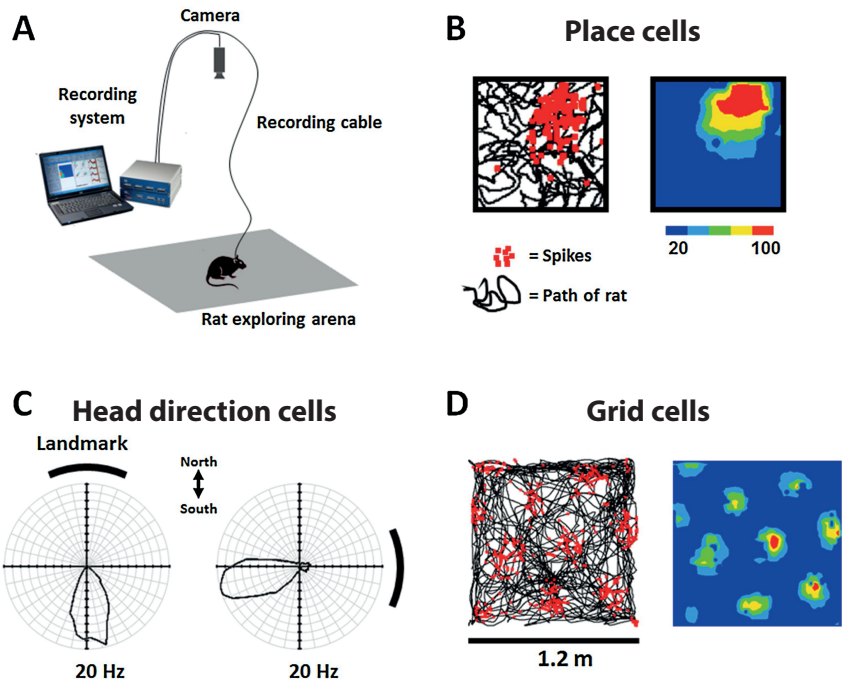


FIGURE 1.2: Representations of space in the brain. **(A)** Typical experimental setup. A rodent freely explores an arena while extracellular electrodes record neural activity. **(B)** Place cells fire preferentially at a particular spot in the environment. The left plot shows action potentials (spikes) superimposed on the animal's path. The right plot shows the firing rate at each location, adjusted for dwell time. **(C)** Head direction cells fire preferentially when the animal faces a certain direction. The polar plots show firing rate as a function of head direction. Head direction cell firing is anchored to landmarks, as shown by the rotation of the firing field after moving the landmark. **(D)** Grid cells fire in multiple locations, organised into a hexagonal lattice. Adapted from Grieves and Jeffery (2017)

the removal of subsets of cues (O'Keefe & Conway, 1978; Quirk et al., 1990) and not affected by changes in orientation (Muller et al., 1994), indicating that place cells do not code for a specific perceptual state but rather for a more holistic concept of location. The discovery of place cells marks the first evidence of encoding of a cognitive map in the brain (O'Keefe & Conway, 1978).

While place cell firing is robust to the removal of single cues, more drastic changes in environmental stimuli, geometry or context can induce changes in the firing rate maps, known as "remapping" (Lever et al., 2002; Mankin et al., 2012; Muller & Kubie, 1987; Ziv et al., 2013). Remapping can also be induced by changes in task (Markus et al., 1995) and has been related to behavioural performance in forming new spatial memories (Dupret et al., 2010).

The shape and location of place cell firing fields is mainly determined by the boundaries of the environment. Their size scales with the scale of the environment (Muller & Kubie, 1987; O'Keefe & Burgess, 1996). Furthermore, inserting a partial dividing boundary into the environment causes place fields to duplicate, such that there was a place field at the same distance and angle from each environmental boundary (Lever et al., 2002). Removing the dividing boundary caused the place fields to revert back to their original rate map.

Head direction cells

A key defining feature of place cells is that, at least in an open field, their representation is "allocentric" or world-centred. In other words, firing does not depend on the animals orientation or head direction. Head direction cells, discovered by (Taube et al., 1990), provide an explicit representation of head direction by firing preferentially when the animal faces a specific direction, independently of the animal's location (Figure 1.2C). Each cell has a preferred direction, such that it fires more strongly when the animal faces that direction. Head direction cells have been found in many brain areas, among which the dorsal presubiculum (Taube et al., 1990), entorhinal cortex (Sargolini et al., 2006) and outside the hippocampal formation in retrosplenial cortex (Taube et al., 1990).

Interestingly, the allocentric spatial map-like quality of place cells seems to depend on head direction cell function (Hartley et al., 2014). When salient cues in the environment are held constant, the preferred directions of head

direction cells remain stable, and the angle between the preferred directions of two given cells remains constant. However, the preferred tuning of the entire system can be rotated by rotating the salient cues. When this is done, the place cell map rotates too. Lesioning the head direction system, however, disrupts this ability of visual cues to control the orientation of place fields (Calton et al., 2003). Furthermore, place cells in an open field become more directional after head direction system lesions, further indicating that the allocentric properties of hippocampal place cells depend on direction information from head direction cells.

Grid cells

While a sufficient number of place cells to tile the environment could serve as an efficient representation for learning to navigate to a goal location (e.g. Foster et al., 2000), it is not clear how place cells could be used to directly compute the heading vector between the current location and a goal, as appeared to happen in Tolman's experiments (section 1.1.1). Furthermore, there seems to be no consistent relationship between the location of a place cell's firing field between one environment and the next (Dring & West, 1983, but see Whittington et al., 2020), which would imply that any relationship between place cells and goals would have to be relearned in every environment. These properties of place cells limit their utility for map-like navigation.

Grid cells, discovered in the medial entorhinal cortex (mEC) by Hafting et al. (2005), exhibit properties which more readily lend themselves to vector navigation (Figure 1.2D). Unlike place cells, grid cells have multiple firing fields distributed in a strikingly regular, hexagonally symmetric pattern across each environment visited by the animal (Figure 1.2D). Evidence for grid cells has also been found in humans in the form of a hexadirectional modulation of BOLD signal as human participants navigate a virtual reality environment while in an fMRI scanner (Doeller et al., 2010). Within mEC, grid cells are organised into functional modules: cells that are near to each other in the brain tend to show rate maps with the same scale and orientation, but with a different spatial phase such that the entire environment is covered by only a few grid cell firing patterns (Barry et al., 2007; Hafting et al., 2005; Stensola et al., 2012). The grid scale (i.e. the distance between fields in the grid) increases per module along the dorso-ventral axis of mEC. Crucially, the relative spatial phase of any two grid cells is conserved across

all environments. These properties of grid cells allow for the computations of vectors between start and goal locations (Bush et al., 2015), an important application of the cognitive map.

Important open questions remain about the relationship between place cells and grid cells. For example, why do place cells remap while grid cells do not, and how do place and grid cell activity depend on each other? Intriguingly, low-dimensional embeddings of place cell firing rate maps correspond to component basis functions that bear a striking resemblance to grid cell firing rate maps (**Sorscher2019AFormation**; **Mok2019ALearning**; **Franzius2007SlownessCells**; Dordek et al., 2016; Stachenfeld et al., 2017). This has led multiple authors to theorise that grid cells serve as a low-dimensional basis set, useful for denoising place cell maps or extracting hierarchical structure from an environment (Stachenfeld et al., 2017) or to generalise knowledge about the structure of an environment across environments that share similar structure (Whittington et al., 2020).

Boundary vector cells

A striking feature of place cells is that their firing seems mainly determined by the presence or absence of boundaries (see section 1.1.2). This led theorists to hypothesise early on that there should be cells upstream of place cells that fire at a fixed direction and angle of boundaries (Figure 1.3). In this theory, place cell firing is proportional to the thresholded sum of a number of such *boundary vector cells* (BVCs; Hartley et al., 2000; O'Keefe & Burgess, 1996). The BVC hypothesis was supported by more direct evidence when Lever et al. (2009) found such cells in the subiculum, an area adjacent to the hippocampus and entorhinal cortex.

Specifically, the theory holds that each BVC has a Gaussian tuned response to the presence of a boundary, with a a preferred distance d_i^* and angle θ_i^*. The firing to a boundary at distance d and direction θ, subtending at an angle $\delta\theta$ to the animal is then given by (see Barry et al., 2006):

$$\delta f_i = g_i(d, \theta)\delta\theta \tag{1.1}$$

where

$$g_i(d, \theta) \propto \mathcal{N}(d|d_i^*, \sigma_{rad}^2)\mathcal{N}(\theta|\theta_i^*, \sigma_{ang}^2). \tag{1.2}$$

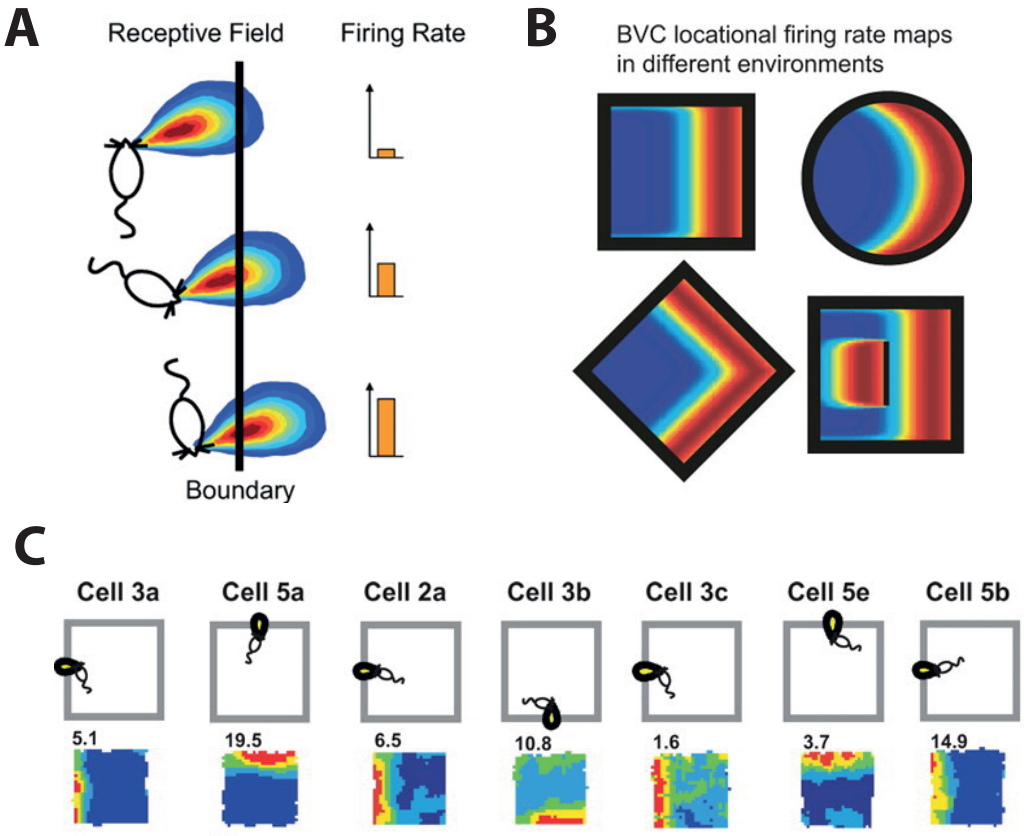


FIGURE 1.3: The boundary vector cell model. **(A)** The receptive field of a BVC tuned to respond to a barrier at a short distance east-northeast from the animal. **(B)** Predicted firing fields in different environment. **(C)** Example BVCs recorded from dorsal subiculum. Firing fields (bottom rows) and corresponding BVC receptive fields (top rows). Adapted from Lever et al. (2009).

The angular tuning width σ_{ang} is constant, while the radial tuning width increases with preferred tuning distance: $\sigma_{rad}(d_i^*) = d_i^* + c$ for a constant c. For each location x in the environment, the contribution of all boundaries to the firing of any BVC is obtained by integrating equation 1.1 over θ. Place cell firing $F_j(x)$ is then proportional to the thresholded sum of the N BVCs that have a feedforward connection to that place cell:

$$F_j(x) \propto H\left(\sum_{i=1}^{N} f_i(x) - \tau\right) \tag{1.3}$$

where the threshold τ is a constant, and $H(x) = x$ if $x > 0$ and $H(x) = 0$ otherwise. Importantly, firing does not depend on the rat's heading direction.

The BVC model has made several successful experimental predictions. For example, it successfully predicted doubling of place fields in response to the insertion of an extended barrier to an environment and crescent shaped place fields close to walls in circular environments (Lever et al., 2009). Similar cells with vector-coding to objects rather than to boundaries have more recently been found in mEC (Høydal et al., 2019).

Non-spatial representations

The most profound influence of Tolman's ideas about cognitive maps has been on the study of spatial navigation, given the striking evidence of map-like neural activity described above. Nevertheless, many researchers envisage this map as a more domain-general systematic organisation of knowledge that can be applied to any behavioural domain (Behrens et al., 2018). Consistent with this idea, representations of non-spatial variables have been found in the hippocampal formation of subjects performing tasks for which navigating physical space was not the relevant task.

A key example of this is work by Aronov et al. (2017), who trained rats to manipulate sound along a continuous pitch axis. Hippocampal neurons that formed place fields while navigating in an open environment displayed discrete firing fields at particular frequencies. In addition, entorhinal cells with grid fields in physical space showed multiple firing fields along the (1D) continuous axes, as is observed for grid cells recorded on a linear track. In humans, hexadirectional modulation of BOLD signal (suggestive of evidence

for grid cell firing; see Doeller et al., 2010) was found while participants "navigated" a completely abstract conceptual space defined by the neck and leg length of "stretchy birds" (Constantinescu et al., 2016).

These non-spatial representations suggest that the map-like codes in the hippocampal formation are used not only for navigation in physical space, but to organise knowledge more generally (Behrens et al., 2018). In that view, the hippocampal formation will represent those features of the animal's current *state* which are relevant for its upcoming decisions. We will now turn to Reinforcement Learning concepts that allow us think more formally about states, representations and decision making.

1.2 Reinforcement learning

Reinforcement learning (RL) formally describes the problem of learning which actions to take in order to maximise future reward (Sutton & Barto, 1998). In the general RL setup, an artificial agent learns an optimal control policy through interactions with an environment. The agent's goal is to find the *optimal* policy, i.e. the policy that maximises the agent's total future reward.

The environment is typically formalised as a Markov Decision Process (MDP), consisting of *states* $s \in S$, *transition probabilities* $T(s'|s,a)$ of moving from state s to states s' given the agent's actions $a \in A$, a reward function $R(s,a)$ specifying the expected immediate reward available when taking action a in state s and a discount factor $\gamma \in [0,1]$, down-weighting rewards that occur further away in the future. While traversing the state space S, the agent chooses actions according to a policy $\pi(a|s)$, collecting rewards along the way. RL methods specify how the agent should change this policy as a result of its experience, with the goal of finding the optimal policy π^* that maximises cumulative reward in the long term. This RL framework is deliberately defined in the most abstract terms, allowing it to be applied to a large number of problems in different ways. For instance, the actions can be low-level control signals such as the voltage signals to control a robot arm (e.g. Deisenroth & Rasmussen, 2011) or high-level decisions such as deciding whether to embark on a PhD. Similarly, states might correspond to low-level sensations, such as a robot's sensor readings, or they can correspond to the board configuration in a game of chess or Go (Silver et al., 2018).

The optimal policy maximises the discounted cumulative reward, or *value* of a given state or state-action pair. The value of a state s under policy π is defined as

$$V^{\pi}(s) = \mathbb{E}_{\pi}\left[\sum_{t=0}^{\infty} \gamma^t r_t | s_0 = s\right], \tag{1.4}$$

where $\mathbb{E}_{\pi}$ denotes the expectation given that the agent follows policy π, and r_t is the reward received at time t. Similarly, we define the value of taking action a in state s as

$$Q^{\pi}(s,a) = \mathbb{E}_{\pi}\left[\sum_{t=0}^{\infty} \gamma^t r_t | s_0 = s, a_0 = s\right], \tag{1.5}$$

where Q^{π} is known as the state-action value function.

Given a state-action value function, acting optimally with respect to that value function is achieved by simply taking those (greedy) actions that maximise the value at each time step. The goal of many algorithms, therefore, is to find the value function Q^{π^*} corresponding to the optimal policy π^*. There are two main schemes for achieving this known as *policy iteration* and *value iteration*. Policy iteration methods iterate between evaluating the current policy (*on-policy learning*) and improving the policy with respect to that value function. These methods make use of the Bellman evaluation equation, which holds that the value of any state s and the value of its possible successor states must be consistent:

$$V^{\pi}(s) = \mathbb{E}_{\pi}\left[r_0 + \gamma V^{\pi}(s_1) | s_0 = s\right]. \tag{1.6}$$

Value iteration methods, on the other hand, aim to find the optimal value function iteratively by solving the Bellman optimality equation:

$$Q^{\pi^*}(s,a) = \mathbb{E}_{\pi}\left[r_0 + \gamma \max_{a'}\left(Q^{\pi^*}(s_1,a')\right) | s_0 = s, a_0 = a\right]. \tag{1.7}$$

Learning the optimal value function directly from data drawn from a different policy is referred to as *off-policy* learning, whereas learning the value function corresponding to the policy that transitions were drawn from is known as *on-policy* learning.

Another important subdivision is made between methods in which the

agent estimates the value function directly from trial and error in a model-free fashion, or whether it uses an explicit model of the transition probabilities $T(s'|s,a)$ to predict its future states. I will describe these approaches in the following sections.

1.2.1 Model-free RL

Value functions can be learned online from experience, in a model-free manner, using simple temporal difference (TD) methods that derive directly from the Bellman equations (1.6 and 1.7). The idea is to, at each time step, increment the value of a state or state-action pair by the difference between the observed and predicted reward, or the reward prediction error (RPE). When moving from state s_t to state s_{t+1}, the value estimate $\hat{V}^\pi(s_t)$ can be updated by:

$$\hat{V}^\pi(s_t) \leftarrow \hat{V}^\pi(s_t) + \alpha \underbrace{\left(r_t + \gamma \hat{V}^\pi(s_{t+1}) - \hat{V}^\pi(s_t)\right)}_{\delta_t} \tag{1.8}$$

where the RPE is given by δ_t. Off-policy model-free methods directly optimise the value function. A key example of this is Q-learning (Watkins & Dayan, 1992):

$$\hat{Q}(s_t,a_t) \leftarrow \hat{Q}(s_t,a_t) + \alpha \left(r_t + \gamma \max_a \hat{Q}(s_{t+1},a) - \hat{Q}(s_t,a_t)\right) \tag{1.9}$$

Computationally, the appeal of model-free algorithms like TD learning is their efficient action selection. This efficiency stems from the fact that model-free algorithms cache, for each state or state-action pair, a scalar value estimate. This means that evaluating a state (and therefore selecting a the best action), can be achieved by simply looking up the cached value for the relevant state or action. That efficiency comes at the cost of behavioural flexibility: if there are any changes in the reward function at one particular state, the value function must be re-estimated at all states, because the value at each particular state holds predictions for all rewards in the future (equation 1.6).

Model-free RL has been of particular interest in neuroscience because of the influential hypothesis that dopamine neurons in the midbrain encode the RPE term required for value updates (Schultz et al., 1997). Support for this hypothesis comes from findings showing that dopamine responses comply with the basic assumptions of TD learning (Waelti et al., 2001) and from a

causal role of dopamine in learning about reward (Chang et al., 2016; Pessiglione et al., 2006; Steinberg et al., 2013; Tsai et al., 2009). Value computation itself is thought to be carried out in the striatum (Cheer et al., 2007; Day et al., 2007).

1.2.2 Model-based RL

The model-free value learning methods described in equations 1.8 and 1.9 can be contrasted to model-based methods (e.g. Deisenroth & Rasmussen, 2011). This 'model' constitutes a knowledge structure that links actions to their likely outcomes. Specifically, the agent learns an estimate of the state transition model T, as well as an estimate of the reward function R. As an example, consider the following simple algorithm for learning the transition and reward models (c.f. Gardner et al., 2018):

$$\Delta T(s'|s,a) \propto \mathbb{I}(s_{t+1} = s') - T(s'|s,a) \tag{1.10}$$

$$\Delta R(s,a) \propto r_t - R(s,a) \tag{1.11}$$

where $\mathbb{I}(\cdot) = 1$ if its argument is true and 0 otherwise. These estimated models can be used to compute value, or the optimal policy, through simulated experience or dynamic programming.

Compared to model-free methods, model-based approaches are more efficient to learn, since the agent can substitute some actual interactions with the environment for simulated experience. They are also more flexible: local changes in the reward or transition distributions can be propagated to every state using offline simulation. This flexibility comes at the cost of expensive action selection: selecting the optimal action involves expensive computational methods such as dynamic programming or tree search. This process of using an 'action–outcome' model to guide decisions is equally referred to as 'planning'.

Psychological and neuroscientific research into the planning abilities of humans and animals focuses on tasks that require flexible re-evaluation after changes in the reward or transition structure. For example, outcome devaluation tasks test whether animals stop taking the action that was previously rewarded after the particular reward they received has been devalued through satiety or pairing it with illness. Since this type of task requires one step of action-outcome association, flexible changing of behaviour in this

type of paradigm has been taken as evidence for planning. The devaluation paradigm, and more sophisticated tasks that built on this (Daw et al., 2011), have generated important insights into the planning abilities of both humans and animals (Adams & Dickinson, 1981; Daw et al., 2005). However, the exact neural mechanisms underlying model-based planning are as of yet largely unknown (Daw & Dayan, 2014).

1.2.3 Brain areas involved in planning and control

While the focus of this thesis is on the hippocampus, many other brain areas have been shown to be implicated in planning and control. Most evidence for a role in MB or MF behaviour comes from lesion studies combined with a reward devaluation experiment. Animals tend to reliably behave in one of the two types of behaviour as a function of specific brain lesions, suggesting there is a relatively clean dissociation between networks that support MF control versus networks that support MB control (Daw & Dayan, 2014).

Within the striatum, MF and MB control appear to depend on the dorsolateral (DLS) and dorsomedial striatum (DMS), respectively (Balleine, 2005; Balleine & O'doherty, 2010; Devan et al., 2011; Thorn et al., 2010; Yin et al., 2008). These areas of the striatum are connected with cortical regions through topographic corticostriatal loops that run through the basial ganglia and thalamus (Frank & Claus, 2006). In the case of MB control, the whole loop is implicated in MB behaviour, with prelimbic cortical and mediodorsal thalamic lesions affecting MB control (Balleine & Dickinson, 1998; Corbit et al., 2003; Killcross & Coutureau, 2003). The orbitofrontal cortex (OFC) has also been implicated in MB behaviour, specifically the flexible assignment of credit (McDannald et al., 2014; McDannald et al., 2011; McDannald et al., 2012; Miller et al., 2017). Many of these brain areas have also been found in human neuroimaging experiments (Gläscher et al., 2010; Tricomi et al., 2009; Valentin et al., 2007; Wunderlich et al., 2012).

The ventral striatum, unlike the dorsal part, is implicated more in Pavlovian (prediction), rather than instrumental conditioning (control) (Daw & Dayan, 2014). As in the control setting, Pavlovian behaviours can be MB or MF (Dayan & Berridge, 2014), which are associated respectively with the shell region of the accumbens, OFC and the basolateral nucleus of the amygdala (in MB-like behaviours such as specific Pavlovian–instrumental transfer and identity unblocking), and with the core of the accumbens and the central

nucleus of the amygdala (in MF-like behaviours such as general Pavlovian instrumental transfer).

Taken together, these results seem suggest a clean dichotomy between the neural implementations of MF-like and MB-like control. However, it should be noted that the apparent independence after lesions does not rule out the possibility that MF and MB systems interact in the intact brain. For example, human neuroimaging experiments showed that the reward prediction errors associated with MF learning are sensitive to MB values (Daw et al., 2011). It has been hypothesised that this reflects an MB system training the MF value computation system (Daw & Dayan, 2014). For example, sequential hippocampal replay, which has been shown to be coordinated with the striatum (Jones & Wilson, 2005; Lansink et al., 2009), may reflect samples of simulated experience from a MB system (Foster & Wilson, 2006, 2007; Johnson & Redish, 2007; Pfeiffer & Foster, 2013). These sample trajectories may then be used to update values leveraging the brain's MF learning systems for MB evaluation. How MB and MF systems interact is a meta-control problem that will be discussed in this thesis.

1.2.4 Representation learning and function approximation

One of the most important questions in RL is how to best *represent* states to facilitate learning and planning. When the state space is discrete, finite and reasonably small, the value function can be represented exactly by a look-up table in which an entry is kept for the estimated value of each state or state-action pair. In most real-world problems, however, the state space is enormous. This means that almost every state that is visited will have never been visited before, requiring not only an enormous amount of storage space but also the time and data needed for filling it with value estimates. The solution to this problem is to forego learning about every state separately. Instead, we would like to *generalise* what we learn about the value of one state to all states that share some common features with that state. This is referred to as *function approximation* because we approximate the value function at every state using data acquired at some states.

A classical choice of function approximator, which we will often use in

this thesis, is the linear parameterisation. The value function is then approximated by:

$$\hat{V}_{\mathbf{w}}(s) = \sum_{i=1}^{d} w_i \phi_i(s) = \boldsymbol{\phi}(s)^T \mathbf{w} \tag{1.12}$$

where $\mathbf{w} \in \mathbb{R}^d$ is a vector of weights and $\boldsymbol{\phi}(s)$ is a vector of state features or basis functions, or in other words: the representation of the state. In models of animal learning, the entries of $\boldsymbol{\phi}(s)$ will typically signify the presence or absence of certain reward-predictive cues (e.g. Gardner et al., 2018; Gershman, 2015).

Methods for tabular RL such as TD learning (equation 1.8) can be easily adapted to update the weights in this function approximation case:

$$\Delta \mathbf{w} \propto \left(r_t + \gamma \hat{V}_{\mathbf{w}}(s_{t+1}) - \hat{V}_{\mathbf{w}}(s_t)\right) \boldsymbol{\phi}(s) \tag{1.13}$$

Linear function approximation is popular because of this simplicity. Indeed, for many algorithms, convergence guarantees are given only for this linear case (see Sutton & Barto, 1998). However, any dependence of the value function that depends on the *interaction* between different features cannot be captured (unless an additional feature coding for the product of these features is added to the feature vector). Non-linear function approximators such as (deep) neural networks tackle this problem (e.g. Mnih et al., 2015).

An additional question pertains to the choice of basis functions $\boldsymbol{\phi}(s)$: what constitutes a good representation? One answer to this question is that a good representation should facilitate the downstream RL process. In the next section, we will discuss the Successor Representation as one possible solution to this problem.

1.2.5 The Successor Representation

Although the difference between model-free and model-based RL methods has been set up as a dichotomy above, there are methods that occupy a space in between these extremes. One of these is based on the Successor Representation (SR), which aggregates statistics over the environmental structure instead of performing expensive forward simulations, settling for intermediate flexibility at a lower computational cost (Dayan, 1993). Specifically, the SR constitutes a cached estimate of future state visits, enabling an agent to jointly learn about states that predict similar futures.

A more formal definition of the SR is now given for the discrete, state-by-state case, as in its original formulation (Dayan, 1993)[1]. In this case, the SR is defined as the discounted sum of t-step transition matrices where the one-step transition matrix has entries $T_\pi(s, s') = P(s'|s, a \sim \pi)$:

$$M^\pi = \sum_{t=0}^{\infty} \gamma^t T_\pi^t \tag{1.14}$$

$$= (I - \gamma T_\pi)^{-1}. \tag{1.15}$$

where I is the identity matrix. Note that T^t corresponds to applying the one-step transition matrix t times, thus giving the probability of ending up in any particular state, exactly t steps into the future. By taking the discounted sum of these, each entry (s, s') of the SR matrix M^π gives the expected discounted sum of visits of state s' given a trajectory starting in s:

$$M^\pi(s, s') = \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} \gamma^t \mathbb{I}(s_t = s') | s_0 = s \right] \tag{1.16}$$

where $\mathbb{I}(s_t = s') = 1$ if $s_t = s'$ and 0 otherwise.

The SR satisfies a Bellman equation, meaning that any RL method can be used to learn M^π. For example, the SR can be updated using a TD prediction error, in the same way as was previously seen for value functions:

$$\Delta \hat{M}(s_t, s') = \alpha \delta_t^M = \alpha \left[\mathbb{I}(s_t = s') + \gamma \hat{M}(s_{t+1}, s') - \hat{M}(s_t, s') \right] \tag{1.17}$$

Here, the prediction error reflects errors in state predictions rather than reward.

The SR is useful as a representation because it allows us to express the value function in a particularly simple way:

$$V^\pi(s) = \sum_{s'} M^\pi \left(s, s'\right) R \left(s'\right) \tag{1.18}$$

where $R(s')$ is the immediate reward in state s'. This factorisation of value into the SR and reward confers more flexible behaviour than purely model-free value learning because if one term changes, it can be relearned while the

[1] Generalisations to SRs defined in terms of state-action pairs are straightforward: instead of counting state visits, we count visited state-action pairs (see e.g. Lehnert et al., 2017). Generalisations to the function approximation case will be given below.

other term remains intact (Dayan, 1993). This is particularly useful in the case of changes in the reward function: the SR allows for a quick (linear) computation of value under *any* reward function. However, since the SR involves caching future state occupancy estimates, a change in the transition function means that the SR will have to be re-estimated everywhere. Furthermore, even a change in the reward function only will also induce a change in the optimal policy. Since the SR is estimated with respect to a particular policy, such changes in the reward function limit the usefulness of an SR based on a previous behaviour policy (Lehnert et al., 2017; Russek et al., 2017). I discuss possible ways around this issue in section 4.2.1. Taken together, this semi-flexibility places the SR somewhere in the middle on a spectrum between classic model-free and model-based methods.

Successor features

The SR can be generalised to continuous states by using a set of features features $\boldsymbol{\phi}(s)$. In this linear function approximation case, the reward is given by the dot product

$$R(s) = \sum_j \phi_j(s) w_j \tag{1.19}$$

where $\boldsymbol{\phi}(s)$ are the state features and $\mathbf{w}$ are weights parameterising the reward function. The decomposition of value into the reward function R and the SR $\boldsymbol{\psi}(s)$ is then written as:

$$V(s) = \sum_j \psi_j(s) w_j \tag{1.20}$$

with:

$$\psi_j(s) = \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} \gamma^t \phi_j(s_t) | s_0 = s \right] \tag{1.21}$$

where $\boldsymbol{\psi}$ is defined such that each entry ψ_j gives the *expected discounted future occurrence of feature* j from starting state s, under the current policy. These are known as Successor Features (Barreto et al., 2016). In the particular case where the state space is finite and $\boldsymbol{\phi}$ is a tabular representation of the state (i.e. a one-hot vector), this definition is equivalent to the one given in Equation 1.16. We can use linear function approximation to parameterise the successor features:

$$\boldsymbol{\psi}^\pi(s) \approx M^T \boldsymbol{\phi}(s) \tag{1.22}$$

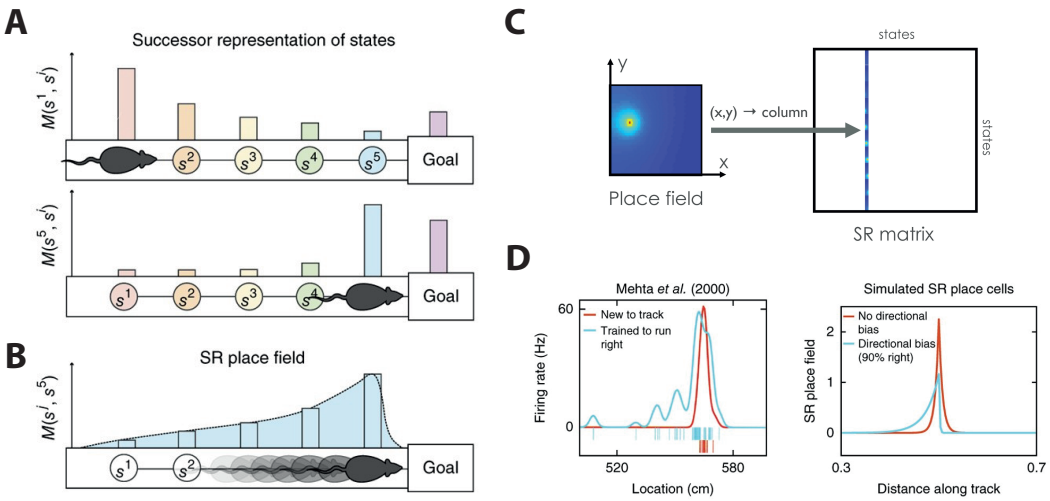


FIGURE 1.4: SR place cell model. (A) Example successor representations at states s^1 and s^5 on a discretised linear track environment. (B) Example SR place field. (C) Illustration of correspondence between a column of M and a 2D place field. (D) Experience-dependent skewing of place fields in data (Mehta et al., 2000) and in the model. Adapted from Stachenfeld et al. (2017).

where matrix M contains the weights parameterising the approximation. These weights can be updated using temporal difference learning with:

$$\Delta M_{i,j} \propto \delta_j \phi_i \tag{1.23}$$

$$\delta_j = \phi_j(s_t) + \gamma \psi_j(s_{t+1}) - \psi_j(s_t) \tag{1.24}$$

Analogous to the discrete case, the TD error here signals surprise about the state features, rather than reward.

The SR as a predictive map in the brain

There has recently been considerable attention in neuroscience for the hypothesis that the SR constitutes part of the decision making repertoire of humans and animals. For example, consistently with SR models, human decision making shows less flexibility after transition changes than after reward changes (Momennejad et al., 2017) and some dopamine signals that are not firing according to the canonical RPE theory are well explained by positing that they code for the 'successor prediction error' required for updating a successor representation over the reward-predictive features during conditioning experiments (Gardner et al., 2018).

Of particular relevance is work by Stachenfeld et al. (2017), who hypothesised that hippocampal place cells encode future state occupancy in the form of an SR. The theory holds that, in spatial contexts, the peak of a place field is determined by pure location input, while the manner in which the cell's firing rate falls off when moving out of the place field depends on the future state predictions under the current policy (and the discount factor under which these state predictions are learned). To see how this works, consider the SR for the discretised linear track environment depicted in Figure 1.4A, where the states correspond to locations on the track. In the model, each s^{th} *row* of the SR matrix M corresponds to the animal's current state representation, or to the firing of the population of place cells. This population codes for the extent to which, given that the current state (location) is s, each other location is expected to be visited in the future, given the discounted time horizon and the current policy. Each *column* s', on the other hand, encodes how much state s' is predicted to be visited from each of the other states. This corresponds to the firing of a particular place cell in each different state or location, or a firing rate map (Figure 1.4B). Figure 1.4C shows a cartoon illustration of how one column of the SR matrix for a random, exploratory policy corresponds to a place field in two dimensions. The SR place cell activity falls off exponentially because of the exponential discounting of state predictions further in the future, resembling real place cell activity.

The central prediction of the SR model is that place cell firing is predictive of where the animal is about to go. This means that, at any moment, cells corresponding to locations that are soon about to be visited (as expected by the animal's policy) will be more active than cells that are equally far away, but not expected to be visited. This prediction, and evidence for the prediction from Mehta et al. (2000), are depicted in Figure 1.4D: when animals are trained to run back and forth on a linear track, place cells have typical symmetric firing fields. However, when animals are trained to run preferentially to the right, place fields skew backward against the direction of travel. The SR interpretation of this result is that place cell firing is predictive: the cell starts firing because the location it corresponds to is a predicted successor state.

Stachenfeld et al. (2017) further hypothesised that entorhinal grid cells correspond to eigenvectors of the SR corresponding to a random-walk policy (or equivalently, eigenvectors of the transition matrix). As can be seen in Figure 1.5, these eigenvectors are spatially periodic, similarly to grid cells.

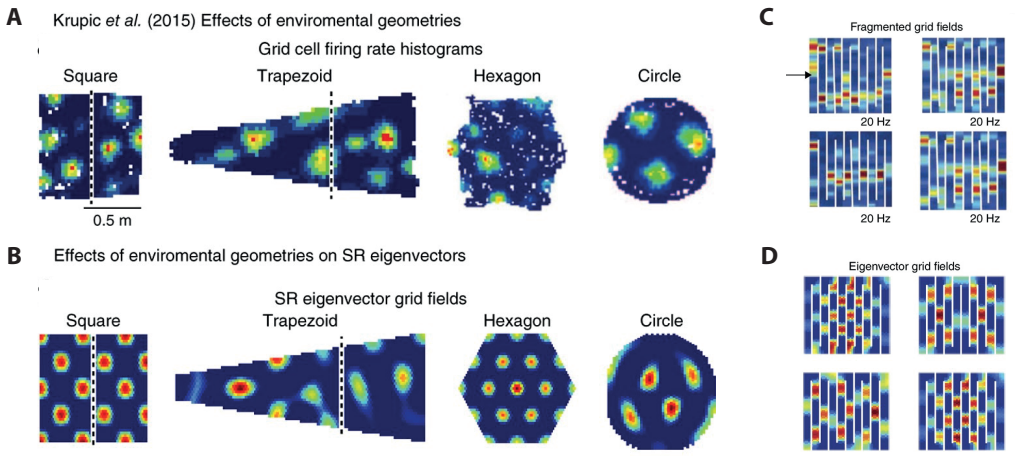


FIGURE 1.5: SR grid cell model. (A) Data from Krupic et al. (2015) showing how environmental geometry shapes grid cell firing. (B) Eigenvector grid cells show similar effects. (C) Data from Derdikman et al. (2009) showing fragmentation of grid cell firing in a hairpin maze. (D) Eigenvector grid cells in the hairpin maze. Adapted from Stachenfeld et al. (2017).

This spatial periodicity arises because these eigenvectors are, by their definition, vectors that are stable under multiplication by the transition matrix. Since applying the random-walk transition matrix corresponds to a step of random-walk diffusion, the optimal solution to this problem is a checkerboard pattern in 2D (Figure 1.5A-B), or a sinusoid wave in 1D, as in the hairpin maze shown in Figure 1.5C-D). Stachenfeld et al. (2017) suggested that these eigenvector grid cells can be useful for denoising the place cell map using spectral regularisation techniques and for finding a hierarchical structure in the environment in the shape of subgoals. This is useful for defining temporally extended actions or "options", an important topic in hierarchical reinforcement learning (see also Machado et al., 2017; Mahadevan & Maggioni, 2007).

1.2.6 Other areas involved in planning and control

While the focus of this thesis is on the hippocampus, many other brain areas have been shown to be implicated in planning and control. Most evidence for a role in MB or MF behaviour comes from lesion studies combined with a reward devaluation experiment. Animals tend to reliably behave in one of the two types of behaviour as a function of specific brain lesions, suggesting

there is a relatively clean dissociation between networks that support MF control versus networks that support MB control (Daw & Dayan, 2014).

Within the striatum, MF and MB control appear to depend on the dorsolateral (DLS) and dorsomedial striatum (DMS), respectively (Balleine, 2005; Balleine & O'doherty, 2010; Devan et al., 2011; Thorn et al., 2010; Yin et al., 2008). These areas of the striatum are connected with cortical regions through topographic corticostriatal loops that run through the basial ganglia and thalamus (Frank & Claus, 2006). In the case of MB control, the whole loop is implicated in MB behaviour, with prelimbic cortical and mediodorsal thalamic lesions affecting MB control (Balleine & Dickinson, 1998; Corbit et al., 2003; Killcross & Coutureau, 2003). The orbitofrontal cortex (OFC) has also been implicated in MB behaviour, specifically the flexible assignment of credit (McDannald et al., 2014; McDannald et al., 2011; McDannald et al., 2012; Miller et al., 2017). Many of these brain areas have also been found in human neuroimaging experiments (Gläscher et al., 2010; Tricomi et al., 2009; Valentin et al., 2007; Wunderlich et al., 2012).

The ventral striatum, unlike the dorsal part, is implicated more in Pavlovian (prediction), rather than instrumental conditioning (control) (Daw & Dayan, 2014). As in the control setting, Pavlovian behaviours can be MB or MF (Dayan & Berridge, 2014), which are associated respectively with the shell region of the accumbens, OFC and the basolateral nucleus of the amygdala (in MB-like behaviours such as specific Pavlovian–instrumental transfer and identity unblocking), and with the core of the accumbens and the central nucleus of the amygdala (in MF-like behaviours such as general Pavlovian instrumental transfer).

Taken together, these results seem suggest a clean dichotomy between the neural implementations of MF-like and MB-like control. However, it should be noted that the apparent independence after lesions does not rule out the possibility that MF and MB systems interact in the intact brain. For example, human neuroimaging experiments showed that the reward prediction errors associated with MF learning are sensitive to MB values (Daw et al., 2011). It has been hypothesised that this reflects an MB system training the MF value computation system (Daw & Dayan, 2014). For example, sequential hippocampal replay, which has been shown to be coordinated with the striatum (Jones & Wilson, 2005; Lansink et al., 2009), may reflect samples of simulated experience from a MB system (Foster & Wilson, 2006, 2007; Johnson & Redish, 2007; Pfeiffer & Foster, 2013). These sample trajectories may

then be used to update values leveraging the brain's MF learning systems for MB evaluation.

1.3 Handling uncertainty using the Kalman Filter

The RL framework outlined in the previous section speaks to a core idea in descriptions of associative learning theory: the idea that humans and animals learn to predict long-term cumulative reward. A second key idea is that animals estimate not only the strength of stimulus-reward associations, but also their uncertainty in these estimates. This has been formalised in Bayesian theories of learning (Dayan & Kakade, 2001; Kakade & Dayan, 2002; Kruschke, 2008) based on the Kalman filter (Kalman, 1960). In this section, I will first introduce the Kalman filter in its most general form (section 1.3.1). Then I will discuss a strategy for using the Kalman filter for value learning (section 1.3.2), as well as a "switching" Kalman filter that switches between multiple modes (section 1.3.3).

1.3.1 The Kalman Filter

The Kalman filter (KF) is an algorithm aimed at tracking a hidden state $\mathbf{x}$ of a dynamic system through indirect observations $\mathbf{y}_{1:t} = \{\mathbf{y}_1 \dots \mathbf{y}_t\}$. The goal of filtering will be to minimise a quadratic cost function:

$$J_t(\hat{\mathbf{x}}) = \mathbb{E}\left[||\mathbf{x}_t - \hat{\mathbf{x}}||^2 | \mathbf{y}_{1:t}\right] \tag{1.25}$$

To this end, at time $t-1$ the algorithm computes a prediction of the hidden state ($\hat{\mathbf{x}}_{t|t-1}$) and the observation ($\hat{\mathbf{y}}_{t|t-1}$) at time t. These predictions can be computed analytically assuming that the system's evolution and observation process are linear-Gaussian, as will be explained below.

Central to the KF problem is the state-space formulation of the system, consisting of an evolution or process equation and an observation equation. The evolution of the system is assumed to be governed by a known *evolution equation*:

$$\mathbf{x}_{t+1} = f_t(\mathbf{x}_t) + \boldsymbol{\xi}_t \tag{1.26}$$

where $\boldsymbol{\xi}$ is a random noise referred to as the evolution or process noise, which models uncertainty in the evolution process. Observations are linked

to states by an *observation equation*:

$$\mathbf{y}_t = g_t(\mathbf{x}_t) + \boldsymbol{\nu}_t \tag{1.27}$$

where $\boldsymbol{\nu}$ is another source of noise named the observation noise, which models uncertainty induced by noisy observations.

Together, the evolution (1.26) and observation (1.27) equations form the state-space description of the system. A primary goal is to recursively estimate the belief state $p(\mathbf{x}_t \mid \mathbf{y}_{1:t})$.

A special case of such a system is where all the dependencies are linear-Gaussian. In that case it is assumed that:

- The evolution model is linear:

$$\mathbf{x}_t = A_t \mathbf{x}_{t-1} + \boldsymbol{\xi}_t \tag{1.28}$$

- The observation model is a linear function:

$$\mathbf{y}_t = C_t \mathbf{x}_t + \boldsymbol{\nu}_t \tag{1.29}$$

- The evolution noise is Gaussian:

$$\boldsymbol{\xi}_t \sim \mathcal{N}(\mathbf{0}, \mathrm{P}_{\xi_t}) \tag{1.30}$$

- The observation noise is Gaussian:

$$\boldsymbol{\nu} \sim \mathcal{N}(\mathbf{0}, \mathrm{P}_{\nu t}) \tag{1.31}$$

In this special case, the model is called a linear-Gaussian dynamical system (LDS), and exact forward inference can be performed using the KF equations, as is shown below.

The KF algorithm performs exact Bayesian inference for LDS models. The posterior at time t will be represented by:

$$p(\mathbf{x} \mid \mathbf{y}_{1:t}) = \mathcal{N}(\mathbf{x} \mid \hat{\mathbf{x}}_t, \Sigma_t) \tag{1.32}$$

This will be achieved by alternating between prediction and update steps. The prediction step:

$$p(\mathbf{x}_t \mid \mathbf{y}_{1:t-1}) = \int \mathcal{N}(\mathbf{x}_t \mid A_t \mathbf{x}_{t-1}, \mathrm{P}_{\xi_t}) \mathcal{N}(\mathbf{x}_{t-1} \mid \hat{\mathbf{x}}_{t-1}, \Sigma_{t-1}) d\mathbf{x}_{t-1} \tag{1.33}$$

$$= \mathcal{N}(\mathbf{x}_t \mid \hat{\mathbf{x}}_{t|t-1}, \Sigma_{t|t-1}) \tag{1.34}$$

$$\hat{\mathbf{x}}_{t|t-1} = A_t \hat{\mathbf{x}}_{t-1} \tag{1.35}$$

$$\Sigma_{t|t-1} = A_t \Sigma_{t-1} A_t^T + \mathrm{P}_{\xi_t} \tag{1.36}$$

The measurement step can be computed using Bayes' rule:

$$p(\mathbf{x}_t \mid \mathbf{y}_t, \mathbf{y}_{1:t-1}) \propto p(\mathbf{y}_t \mid \mathbf{x}_t) p(\mathbf{x}_t \mid \mathbf{y}_{1:t-1}) \tag{1.37}$$

This is given by:

$$p(\mathbf{x}_t \mid \mathbf{y}_{1:t}) = \mathcal{N}(\mathbf{x}_t \mid \hat{\mathbf{x}}_t, \Sigma_t) \tag{1.38}$$

$$\hat{\mathbf{x}}_t = \hat{\mathbf{x}}_{t|t-1} + K_t \boldsymbol{\delta}_t \tag{1.39}$$

$$\Sigma_t = (I - K_t C_t) \Sigma_{t|t-1} \tag{1.40}$$

where $\boldsymbol{\delta}_t$ is the error or innovation. This is the difference between the predicted and actual observation:

$$\boldsymbol{\delta}_t = \mathbf{y}_t - \hat{\mathbf{y}}_t = \mathbf{y}_t - C_t \hat{\mathbf{x}}_{t|t-1} \tag{1.41}$$

and K_t is the Kalman gain, given by:

$$K_t = \Sigma_{t|t-1} C_t^T \Lambda_t^{-1} \tag{1.42}$$

where $\Lambda_t = C_t \Sigma_{t|t-1} C_t^T + \mathrm{P}_{\nu t}$ is the error covariance matrix.

These update equations make intuitive sense from a Bayesian perspective. The equation for the mean update (1.35) is driven by the prediction error $\boldsymbol{\delta}_t$, weighted by the Kalman gain. K_t (equation 1.42) is proportional to the ratio between the covariance of the prior, $\Sigma_{t|t-1}$ and the covariance of the measurement error, $\mathrm{P}_{\nu t}$. A strong prior, or high observation noise, will result in a small gain, placing little weight on the correction term. Conversely, a weak prior or high observation precision will result in a large gain and larger updates.

When the observation $\mathbf{y}$ is taken to be a scalar reward signal, and the

observation matrix C a vector of features, the KF can explain many associative learning phenomena seen in animals (see Gershman, 2015, for a review). For example, the fact that repeated exposure to a stimulus slows subsequent learning of stimulus-reward associations (a phenomenon known as *latent inhibition*) is explained by a reduction in posterior variance. Because the weights corresponding to concurrently presented cues develop negative covariance with each other, the KF also explains phenomena such as *backward blocking*, whereby learning about some cue-reward associations can "explain away" other hypothesised cue-reward associations (Dayan & Kakade, 2001).

These KF models of associative learning (Dayan & Kakade, 2001; Kakade & Dayan, 2002; Kruschke, 2008) can be seen as Bayesian generalisations of the seminal Rescorla-Wagner (1972) model, associating the instantaneous reward associated with input stimuli. As we will see in the next section (1.3.2), a simple modification of these models allows them to be used for the prediction of long-term cumulative reward or value.

Algorithm 1: One step of the Kalman filter. Update the filtered posterior for a LDS with parameters $\theta = \{A, C, \mathrm{P}_\xi, \mathrm{P}_\nu\}$

Function `KFupdate`$(\hat{\mathbf{x}}_{t-1}, \Sigma_{t-1}, \mathbf{y}_t, \theta)$**:**

Prediction step ;

$\hat{\mathbf{x}}_{t|t-1} = A\hat{\mathbf{x}}_{t-1}$;

$\Sigma_{t|t-1} = A\Sigma_{t-1}A^T + \mathrm{P}_\xi$;

Update step ;

$K_t = \Sigma_{t|t-1}C_t^T\Lambda_t^{-1}$;

$\hat{\mathbf{x}}_t = \hat{\mathbf{x}}_{t|t-1} + K_t(\mathbf{y}_t - C\hat{\mathbf{x}}_{t|t-1})$;

$\Sigma_t = (I - K_tC_t)\Sigma_{t|t-1}$;

$L = p(\mathbf{y}_t \,|\, \mathbf{y}_{1:t-1}) = \mathcal{N}(\mathbf{y}_t \,|\, C\hat{\mathbf{x}}_{t|t-1}, C\Sigma_{t|t-1}C^T + \mathrm{P}_\nu)$;

return $\hat{\mathbf{x}}_t, \Sigma_t, L$;

1.3.2 Combining KF and RL: Kalman temporal differences

Geist and Pietquin (2010a) introduced Kalman Temporal Differences (KTD). The core idea is to cast value function evaluation as a hidden state tracking problem, where the parameters defining the value function are the to-be-tracked hidden variable, and the rewards are treated as noisy observations of this true value function. The main reason for doing so is that unlike traditional temporal difference methods, this KF approach provides some uncertainty information about the value function estimates.

In this section, I will introduce KTD in the case of state-value function evaluation, termed KTD-V. These same ideas apply, with some modification, to the evaluation (KTD-SARSA) or direct optimisation (KTD-Q) of state-action values. Furthermore, I will focus on the case of linear function approximation. Generalisations of KTD to non-linear function approximation, based on the unscented Kalman filter, are given in Geist and Pietquin (2010a).

In order to formulate the value function evaluation problem as a filtering problem, it must be turned into the state-space formulation[2] introduced in the previous section. This formulation consists of an evolution equation, describing how the parameters evolve over time, and an observation equation, describing how the parameters relate to the observations (rewards). Unfortunately, the dynamics of the value function are hard to model, as they depend on both the environment's transition statistics and the policy: even if the system itself is stationary, in a policy iteration scheme the value function will change every time the policy is improved. Therefore, a parsimonious evolution model that allows for non-stationarity is to model the evolution of the parameters as a random walk (in other words, the transition matrix A_t is simply the identity matrix):

$$\mathbf{w}_t = \mathbf{w}_{t-1} + \boldsymbol{\xi}_t \tag{1.43}$$

Here, $\mathbf{w}_t$ is the true parameter vector at time t and $\boldsymbol{\xi}$ is the evolution noise. The observation equation will express the relation between the observations (rewards) and the to-be-inferred variable (the value function parameter vector). This is given by the Bellman equation, plus some random noise:

$$r_t = V^{\pi}_{\mathbf{w}}(s_t) - \gamma V^{\pi}_{\mathbf{w}}(s_{t+1}) + \nu_t \tag{1.44}$$

In the case of a linear parameterisation, which I will assume here, the observation equation is given by:

$$\begin{aligned} r_t &= \boldsymbol{\phi}(s_t)^T \mathbf{w}_t - \gamma\, \boldsymbol{\phi}(s_{t+1})^T \mathbf{w}_t && (1.45)\\ &= (\boldsymbol{\phi}(s_t) - \gamma\, \boldsymbol{\phi}(s_{t+1}))^T \mathbf{w}_t && (1.46)\\ &= \mathbf{h}_t^T \mathbf{w}_t && (1.47) \end{aligned}$$

[2]Note that this terminology comes from the Kalman filtering (state-space models) literature and must not be confused with the state space of the MDP in RL terminology. The "hidden state" in Kalman filtering is the to-be-tracked variable, in this case the value function parameter vector.

where we have defined $\mathbf{h}_t = \boldsymbol{\phi}(s_t) - \gamma\,\boldsymbol{\phi}(s_{t+1})$ to be the temporal difference in features. This is what relates the observations to the rewards and it is KTD's equivalent to the observation matrix B_t in equation 1.29. Note that the linear parameterisation is chosen here just for simplicity, and not because this is necessarily a feature of KTD; there are versions of KTD for non-linear parameterisations based on the unscented KF (Julier & Uhlmann, 2004).

Given the generative model described above, the goal is to infer the parameters that gave rise to a set of observations. In the linear case, both the prediction and update steps of KTD can be derived analytically. The prediction step consists of computing $\hat{\mathbf{w}}_{t|t-1}$ and $\Sigma_{t|t-1}$:

$$\hat{\mathbf{w}}_{t|t-1} = \hat{\mathbf{w}}_{t-1|t-1} \tag{1.48}$$

$$\Sigma_{t|t-1} = \Sigma_{t-1|t-1} + \mathrm{P}_{\xi_t} \tag{1.49}$$

which gives the predicted observed reward as $\hat{r}_t = \mathbf{h}_t^T \hat{\mathbf{w}}_{t|t-1}$.

In the update step, the parameters are updated using the Kalman filter equations. The optimal gain is now a vector, given by:

$$\boldsymbol{\kappa}_t = \frac{\Sigma_{t|t-1}\mathbf{h}_t}{\lambda_t} \tag{1.50}$$

where $\lambda_t = \mathbf{h}_t^T \Sigma_{t|t-1} \mathbf{h}_t$ is the variance of the prediction error. The update equations are:

$$\hat{\mathbf{w}}_t = \hat{\mathbf{w}}_{t|t-1} + \boldsymbol{\kappa}_t (r_t - \hat{r}_t) \tag{1.51}$$

$$\Sigma_t = \Sigma_{t|t-1} - \boldsymbol{\kappa}_t \lambda_t \boldsymbol{\kappa}_t^T \tag{1.52}$$

Here, $\delta_t = r_t - \hat{r}_t$ corresponds to the (expectation of the) TD reward prediction error known from RL.

When used as models of animal learning, KTD and the KF for instantaneous rewards (which is simply KTD with a discount factor set to $\gamma = 0$) make many of the same predictions as TD learning and the Rescorla-Wagner rule, because the update to the mean of the weights is proportional to an RPE (equation 1.51). However, the crucial difference is that these methods estimate a *posterior distribution* over the parameters, whereas more classical methods estimate only a point estimate. Interestingly, this feature of KTD allows it to explain several more subtle effects in animal learning (Gershman, 2015) as well as dopamine firing during learning (Gershman, 2017a). In

Chapter 3, we will explore the merits and limitations of applying this method to estimating successor features.

1.3.3 Switching Kalman filter

The basic KF described in section 1.3.1 provides efficient inference in linear systems with Gaussian noise. Most real-world systems do not fall into this category, but some of these phenomena can be described using a model that can switch between multiple different linear dynamical systems (LDSs). Models that capture these switches are known as switching state-space models, switching LDSs, or switching Kalman filters (Murphy, 1998). They are of interest in cognitive neuroscience because the process of inferring when to make a new memory or adapt an old one has been likened to the inference process in an infinite capacity switching KF (Gershman et al., 2014).

Switching Kalman filter models exist in many forms, depending on which part of the model is switched between. For example, each different switch state or mode k might correspond to different transition dynamics A_t^k, allowing the model to capture different dynamical regimes that the system can switch between. Alternatively, each mode might correspond to a different observation matrix C_t^k, meaning each mode corresponds to a different way that the hidden state maps onto observations. Another alternative is a model with switching observation matrices and multiple hidden processes $\mathbf{x}_t^k$, as presented in Ghahramani and Hinton (1996) and Gershman (2015). In this case, the mode can be seen as selecting one of the hidden processes to pass through to the observation variable. As noted by Murphy (1998), this kind of factored model with K different hidden state vectors can always be converted into the more canonical form (i.e. a single hidden state vector with switching between parameters) by combining the separate $\mathbf{x}_t^k$ into a single block vector. In that case, the transition matrices A_t and P_{ξ_t} will become block diagonal matrices. For simplicity, we will present the canonical form here.

Inference in switching state-space models is generally intractable, because the number of possible mode assignments grows exponentially with time: if there are K possible modes, the posterior at time t will be a mixture of K^t Gaussians. Multiple solutions to this problem have been proposed, including approximating the exponentially large mixture of Gaussians with a smaller mixture of Gaussians (Barber, 2012), variational approximations (Ghahramani & Hinton, 1996) and sequential Monte Carlo (SMC) methods

that make use of the conditional linearity of the model to sample a discrete trajectory and then apply analytical filtering methods to the continuous variables in the model (Doucet et al., 2001). Here we describe inference in a switching LDS using particle filtering, an SMC method.

The idea of particle filtering is to approximate the belief state using a weighted set of particles. For each particle l, we can represent $p(\mathbf{x}_t \mid \mathbf{y}_{1:t}, \mathbf{z}_{1:t}^{\{l\}})$ using a mean and covariance matrix for each particle. Applied to the switching Kalman filter, the idea is to, for each particle, sample a mode assignment from a proposal distribution, such as the prior distribution over modes. Conditional on that sampled mode assignment, the posterior of the hidden state $\mathbf{x}_t$ is then computed using the Kalman filter equations. The Kalman filter's likelihood can then be used to update the particle's weights. Finally, to avoid particle degeneracy (after a few iterations, most particles will have close to zero weight, and updating these particles wastes computation), a resampling step is included. Pseudocode for the particle filter algorithm for switching LDS models is given in algorithm 2.

Algorithm 2: One step of particle filter for state estimation in switching LDS

for $l = 1 : L$ **do**
  $k \sim p(z_t | \mathbf{z}_{t-1}^{\{l\}})$;
  $z_t^{\{l\}} := k$;
  $(\hat{\mathbf{x}}_t^{\{l\}}, \Sigma_t^{\{l\}}, L_{tk}^{\{l\}}) = \text{KFupdate}(\hat{\mathbf{x}}_{t-1}^{\{l\}}, \Sigma_{t-1}^{\{l\}}, \mathbf{y}_t, \theta_k)$;
  $w_t^{\{l\}} = w_{t-1}^{\{l\}} L_{tk}^{\{l\}}$;
end
Normalize weights: $w_t^{\{l\}} = w_t^{\{l\}} / \sum_{l'} w_t^{\{l'\}}$;
Compute $\hat{L}_{\text{eff}} = 1/(\sum_{l=1}^{L} (w_t^l)^2)$;
if $\hat{L}_{eff} < L_{min}$ **then**
  Resample L indices $\boldsymbol{\pi} \sim \mathbf{w}_t$;
  $\mathbf{z}_t^i = \mathbf{z}_t^{\boldsymbol{\pi}}, \hat{\mathbf{x}}_t^i = \hat{\mathbf{x}}_t^{\boldsymbol{\pi}}, \boldsymbol{\Sigma}_t^i = \boldsymbol{\Sigma}_t^{\boldsymbol{\pi}}$;
  $w_t^{\{l\}} = 1/L$;

1.3.4 Summary

In summary, the KF and related approaches can serve as a useful model for how the brain might estimate and use uncertainty during associative learning (Dayan & Kakade, 2001; Gershman, 2015), while a switching KF successfully captures aspects of memory updating (Gershman et al., 2014). In Chapter 3, I will combine these models with SR learning to model context-dependent decision making.

Chapter 2

A model of hippocampal and dorsolateral striatal learning

The work presented in this chapter has been published in PNAS as Geerts et al. (2020)[1].

2.1 Introduction

As described in Chapter 1, animals can apply model-based (MB) and model-free (MF) methods for RL problem (Daw et al., 2005; Sutton & Barto, 1998). In MB RL, a model of the environment is used to simulate the future to plan optimal actions (Tolman, 1948), and the past for episodic memory (Bicanski & Burgess, 2018; Schacter et al., 2007; Tulving, 1972). MF RL, on the other hand, uses trial and error to estimate a direct mapping from the animal's state or sensory inputs to its expected future reward, which the agent caches and looks up at decision time (Rescorla & Wagner, 1972; Sutton, 1988), potentially supporting procedural memory (Squire & Zola-Morgan, 1991). In the brain, this computation is thought to be carried out in the brain through prediction errors signalled by phasic dopamine responses (Montague et al., 1996). These strategies are associated with different tradeoffs (Daw et al., 2005). The model-based (MB) approach is powerful and flexible but computationally expensive and therefore slow at decision time. MF methods, in contrast, enable rapid action selection, but these methods learn slowly and adapt poorly to

[1] Authorship declaration: Chersi's contribution in this paper refers to his work developing an instructive previous solution that was ultimately replaced by the model presented here. Stachenfeld and Burgess had supervisory roles.

changing environments. In addition to MF and MB methods, there are intermediate solutions that rely on learning useful representations that reduce burdens on the downstream RL process (Barreto et al., 2016; Dayan, 1993; Lehnert & Littman, 2018).

In the spatial memory literature (see section 1.1), a distinction has been observed between *"response learning"* and *"place learning"* (Chersi & Burgess, 2015; Dring & West, 1983; White, 2004). When navigating to a previously visited location, response learning involves learning a sequence of actions, each of which depends on the preceding action or sensory cue (expressed in egocentric terms). For example, one might remember a sequence of left and right turns starting from a specific landmark. Place learning, in contrast, involves learning a flexible internal representation of the spatial layout of the environment (expressed in allocentric terms). This *cognitive map* is thought to be supported by the hippocampal formation, where there are neurons tuned to place and heading direction (Hafting et al., 2005; O'Keefe & Dostrovsky, 1971; Taube et al., 1990). Spatial navigation using this map is flexible because it can be used with arbitrary starting locations and destinations which need not be marked by immediate sensory cues.

Here, I will posit that the distinction between place and response learning is analogous to that between MB and MF RL (Poldrack & Packard, 2003). Under this view, associative reinforcement is supported by the DLS (Yin & Knowlton, 2004; Yin et al., 2005). Indeed, there is evidence from both rodents (Mcdonald & White, 1994; Packard, 1999; Packard & McGaugh, 1996) and humans (Doeller & Burgess, 2008; Doeller et al., 2008) that spatial response learning relies on the same basal ganglia structures that support model-free RL. Evidence also suggests an analogy between MB reasoning and hippocampus-based place learning (Miller et al., 2017; Vikbladh et al., 2019). However, this equivalence is not completely straightforward. For example, in rodents, multiple hippocampal lesion and inactivation studies failed to elicit an effect on action-outcome learning, a hallmark of model-based planning (Corbit & Balleine, 2000; Corbit et al., 2002; Gaskin et al., 2005; Kimble & BreMiller, 1981; Kimble et al., 1982; Ward-Robinson et al., 2001). Nevertheless, there are indications that hippocampus might contribute to a different aspect of model-based RL: namely, the representation of relational structure. Tasks that require memory of the relationships between stimuli do show dependence on hippocampus (Bunsey & Eichenbaum, 1996; DeVito & Eichenbaum, 2011; Dusek & Eichenbaum, 1997; Garvert et al., 2017; Schapiro et al., 2016;

Scoville & Milner, 1957; Vargha-Khadem et al., 1997).

In this chapter, I formalise the perspective that hippocampal contributions to model-based learning and place learning are the same, as are the dorsolateral striatal contributions to model-free and response learning. In our model, hippocampus supports flexible behaviour by representing the relational structure among different allocentric states, while dorsolateral striatum (DLS) supports associative reinforcement over egocentric sensory features. The model arbitrates between the use of these systems by weighting each system's action values by the reliability of the system, as measured by a recent average of prediction errors, following Wan Lee et al. (2014). I show that hippocampus and DLS maintain these roles across multiple task domains, including a range of spatial and nonspatial tasks. Our model can quantitatively explain a range of seemingly disparate findings including the choice between place and response strategies in spatial navigation (Packard & McGaugh, 1996; Pearce & Hall, 1980) and choices on non-spatial multi-step decision tasks (Daw et al., 2011; Doll et al., 2015). Furthermore, it explains the puzzling finding that landmark-guided navigation is sensitive to the blocking effect, whereas boundary-guided navigation is not (Doeller & Burgess, 2008), and that these are supported by the DLS and hippocampus, respectively (Doeller et al., 2008). Thus, different RL strategies that manage competing tradeoffs can explain a longstanding body of spatial navigation and decision-making literature under a unified model.

2.2 Methods

I implemented a model of hippocampal and dorsolateral striatal contributions to learning, shown in Figure 2.1. Each system independently proposes an action and estimates its value. The value $Q(s,a)$ of taking action a while being in state s is the expected discounted cumulative return:

$$Q(s,a) = \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} \gamma^t r(s_t) | s_0 = s, a_0 = a \right], \tag{2.1}$$

where s_0 and a_0 are the starting state and action at time $t = 0$, r is a reward function specifying the instantaneous reward found in each state, $\gamma \in [0,1)$ is a discount factor that gives smaller weight to distal rewards and $\pi(a|s)$ is the policy specifying a distribution over available actions given the current

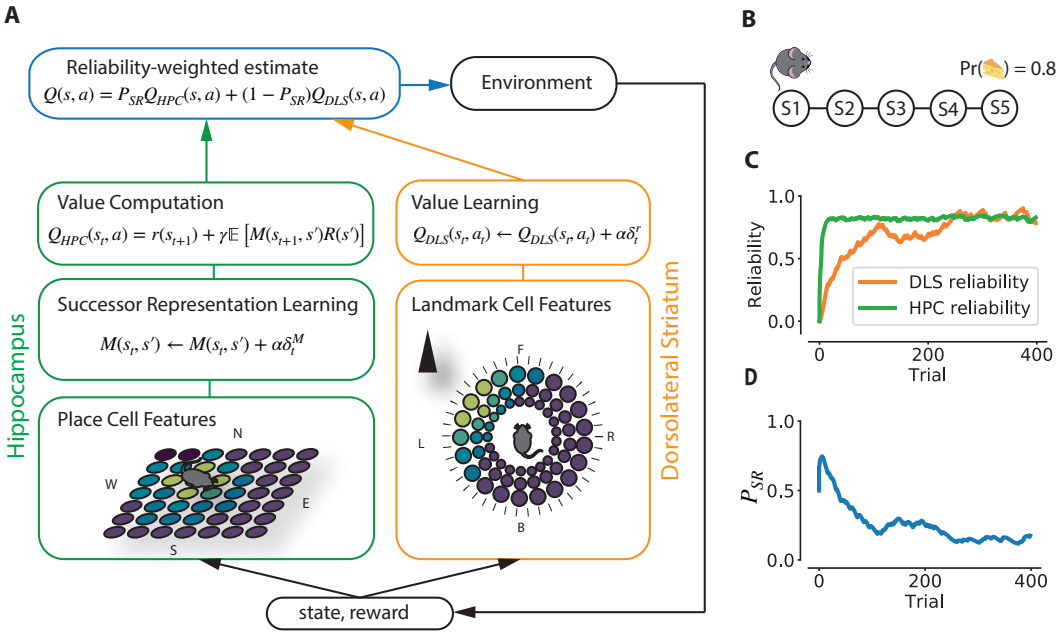


FIGURE 2.1: **(A)** Model architecture. Dorsolateral Striatum (DLS, orange) learns value directly from landmark features in egocentric directions with respect to the agent: L (left), R (right), F (front) or B (back). Hippocampus (HPC, green) learns a successor representation M over allocentric input features (North N, East E, South S or West W), which is subsequently used for value computation. An arbitrator (blue) computes an average of these values, weighted by each system's reliability (see Methods, section 2.2). Lighter colours mean higher firing rates. α: learning rate, δ^M: successor prediction error, δ^r: reward prediction error, P_{HPC}: proportion of influence of HPC component. **(B)** A linear track environment with five states. Terminal state S5 gives a reward with probability 0.8. **(C)** Reliability of the hippocampal SR system and the striatal model-free system over time as the agent navigates the linear track. Reliability is computed based on the recent average of successor prediction errors δ^M for the hippocampal system, and reward prediction errors δ^R for the striatal system. **(D)** The proportion of influence of the SR system on the value function, P_{SR} in the linear track environment across trials.

state. The objective of the RL agent is to discover an optimal policy π^* that will maximise value over all states.

Similarly to earlier work in spatial RL (Chersi & Burgess, 2015, 2016; Dollé et al., 2018; Dollé et al., 2010), the two systems in our model estimate value using qualitatively different strategies, which can cause them to generate divergent predictions for the optimal policy. The dorsal striatal component uses a model-free temporal difference (TD) method (Watkins & Dayan, 1992) to learn stimulus-response associations directly from egocentric sensory inputs given by landmark cells tuned to landmarks at given distances and egocentric directions from the agent (Figure 2.1A, see Methods, section 2.2).

The hippocampal component, in contrast, has access to state information provided by place cells that, in spatial tasks, fire when the agent occupies specific locations. I draw on previous work by Stachenfeld et al. (2017) and model hippocampal place cells as encoding the Successor Representation (SR, Dayan, 1993). The SR is a predictive representation, containing the discounted future occupancy of each state s' from current state s:

$$M^{\pi}(s, s') = \mathbb{E}_{\pi}\left[\sum_{t=0}^{\infty} \gamma^t \, \mathbb{I}(s_t = s') | s_0 = s\right] \tag{2.2}$$

where $\mathbb{I}(s_t = s') = 1$ if $s_t = s'$ and 0 otherwise. Each entry $M^{\pi}(s, s')$ of the SR estimates the exponentially discounted count of the number of times state s' is visited in the future, given that the current state is s, conditioned on the current policy $\pi(a|s)$. In addition to the SR, the hippocampal system learns a vector of rewards R associated to each state, which is multiplied with the SR to compute state values (see Equation 2.8). Crucially, the hippocampal SR algorithm learns aggregate statistics over the relational structure between states, which allows for some of the flexibility of fully model-based systems at lower computational cost. Specifically, SR-based systems decouple learning about transition dynamics from learning about reward, which allows for a quick re-computation of value under a new reward distribution.

Arbitration between the two systems was achieved by tracking their reliability in predicting states (HPC) and rewards (DLS), and weighting either systems' action values by this reliability, following Wan Lee et al. (2014). I operationalised this as the average recent reward prediction error for the MF system, and as the average successor state prediction error for the SR system. These reliability measures were then used to compute the proportion

of influence the SR system had on the value function, P_{SR} (see Equation 2.18 in Methods for details). Although not modelled in detail here, I suggest this arbitration is supported by the medial prefrontal cortex (mPFC), following previous theoretical and experimental work (Daw et al., 2005; Killcross & Coutureau, 2003). Figure 2.1B-D show an example of how the arbitrator functions. The agent was trained to find a reward (given with probability 0.8) at the end of a simple linear track, in which each state was uniquely identified by landmarks (Figure 2.1B). The agent was allowed to explore the environment randomly, so it started with a random-walk SR. Hence, the reliability of the HPC starts out higher than that of the DLS. As the average DLS reward prediction error goes down, and its reliability catches up with that of HPC, the proportion of HPC influence decreases.

In summary, our model combines a hippocampal reinforcement learning module based on the SR with a striatal model based on model-free value learning (see Figure 2.1A). It arbitrates between these modules based on their relative reliability, which can be computed using the average of recent prediction errors. Model details are outlined below.

2.2.1 Dorsal striatal system

The dorsolateral striatum module was implemented as a model-free RL system that learned direct associations between sensory stimuli and actions. Striatal neurons coded for the value of each action, where actions were expressed as egocentric heading directions in the spatial navigation tasks and left or right button presses in the non-spatial tasks. Sensory input was coded by a set of egocentric landmark vector cells coding for the presence or absence of a landmark in a particular egocentric direction, at a particular distance from the landmark to the agent, analogous to the egocentric boundary vector cells recently reported (Hinman et al., 2019). Specifically, the activation of each landmark cell (LC) was modelled as a bivariate Gaussian in a space defined by the egocentric angle θ and distance d of the landmark to the agent:

$$f^{LC}(d,\theta) \propto \mathcal{N}([d,\theta];[d^*,\theta^*],\Sigma) \tag{2.3}$$

where d^* and θ^* are the preferred distance and orientation of the landmark cell respectively, and $\Sigma = diag([\sigma_d,\sigma_\theta])$ is the covariance matrix with the tuning width and length of the receptive field on the diagonal entries. I assumed

that landmark cells are sensitive to the identity of the landmark, meaning that a different set of landmark cells will respond to a different landmark in our model. An example egocentric landmark cell is shown in Appendix A, Figure A.1. In the non-spatial tasks, states were encoded as "one-hot" vectors containing ones for their state indexes, reflecting the fact that states were uniquely identifiable as different images.

Landmark cells (LCs) in the sensory layer project to neurons in the dorsal striatum in an all-to-all connected way:

$$x_a^{\text{DLS}} = Q_{\text{DLS}}(s,a) = \sum_{i=1}^{N} w_{i,a} f_i^{\text{LC}}(s), \tag{2.4}$$

where f_i^{DLS} is the activity of landmark cell i, x_a^{DLS} is the firing rate of the dorsolateral striatal neuron corresponding to striatal estimated value Q^{DLS} of action a given state s, N the total number of sensory neurons, u_i^{LC} the firing rate of landmark cell i and $w_{i,a}$ the weight from sensory neuron i to striatal neuron a.

Learning in the striatal network is mediated by a Q-learning rule (Watkins & Dayan, 1992). This allows the model to compute a temporal difference (TD) reward prediction error δ_t^r:

$$\delta_t^r = r_{t+1} + \gamma \max_{a'} Q_{\text{DLS}}(s_{t+1}, a') - Q_{\text{DLS}}(s_t, a_t), \tag{2.5}$$

where r_{t+1} is the reward received at time $t+1$. This prediction error is then used to update the weights:

$$\Delta w_{i,a} = \alpha_Q \delta_t^r e_{i,a}, \tag{2.6}$$

with learning rate α_Q and eligibility trace $e_{i,a}$, which tracks which weights are eligible for updating based on recent activity. Every time step, the eligibility trace is updated according to the following rule:

$$e_{i,a}(t+1) = f_i^{LC} x_a^{DLS} + \lambda e_{i,a}(t), \tag{2.7}$$

where λ is the trace decay parameter, controlling for how long synapses stay eligible for updating. Eligibility traces enable faster learning by making it possible to update weights that were active in the recent past instead of only the very last time step (Sutton & Barto, 1998).

2.2.2 Hippocampal system

The hippocampal place cell system was modelled as encoding the SR, following work by Stachenfeld et al. (2017). The SR is a predictive representation employed in machine learning (Barreto et al., 2020; Barreto et al., 2016; Dayan, 1993; Kulkarni et al., 2016), containing the discounted future occupancy of each state s' from current state s (Equation 2.2). In the hippocampal SR model, a row of the SR, i.e. $M^{\pi}(s,:)$, constitutes the current population activity vector, i.e. the activity of every place cell in the current state. A column of M^{π} contains the activity of a single place cell in all possible locations (states), i.e. a rate map (see Appendix A, Figure A.1). In addition to the SR matrix, the agent will learn a vector with the expected reward $R(s)$ for each states. The agent combines these to compute state value:

$$V^{\pi}_{\mathrm{HPC}}(s) = \sum_{s'} M\left(s,s'\right) R\left(s'\right) \tag{2.8}$$

The factorisation of value into the SR and reward confers more flexible behaviour because if one term changes, it can be relearned while the other term remains intact (Dayan, 1993). The agent used one-step lookahead to compute the value of each action $Q(s,a)$, combining direct reward and the next state's value:

$$Q_{\mathrm{HPC}}(s_t,a_t) = r(s_t) + \gamma\, \mathbb{E}_{s_{t+1}|s_t,a_t}\left[V_{\mathrm{HPC}}(s_{t+1})\right] \tag{2.9}$$

The SR satisfies a Bellman equation, meaning that any RL method can be used to learn the SR. Here, learning was achieved using a temporal difference (TD) update:

$$\Delta\hat{M}(s_t,s') = \alpha_M \delta^M_t(s') \tag{2.10}$$

where $\delta^M_t(s') = \left[\mathbb{I}(s_t = s') + \gamma\hat{M}(s_{t+1},s') - \hat{M}(s_t,s')\right]$ is a temporal difference "successor prediction error" pertaining to state s' and α_M is a learning rate. For the spatial navigation studies modelled in this paper, animals were allowed to freely explore the environment without any reward before starting the task (Packard & McGaugh, 1996; Pearce et al., 1998). Hence, for these tasks, the SR was initialised as the SR associated to a random walk policy M^{RW} over a uniform spatial discretisation of the environment. This was not the case for the task graphs of the two-step decision tasks (Doll et al., 2015). Therefore, in these tasks I initialised the SR as the identity matrix I, encoding no other knowledge than the fact that every state predicts itself. Finally, the

reward vector $\hat{R}$ was learned using a simple delta rule:

$$\Delta\hat{R}(s_t) = \alpha_R \left(r_t - \hat{R}(s_t)\right) \tag{2.11}$$

Although the SR is often introduced as above (in terms of discrete state counts), accurately estimating the SR for every state is infeasible in very large state spaces. This is known as the *curse of dimensionality*, and it necessitates the use of function approximation (Sutton & Barto, 1998). The agent observes states through a vector of features $\mathbf{f}(s)$ which, if chosen rightly, will be of much smaller dimension than the number of states, allowing the agent to generalise to states that are nearby in feature space. The feature-based SR (also referred to as Successor Features Barreto et al., 2016) rather than encoding the discounted number of state visits, encodes the expected discounted future activity of each feature:

$$\boldsymbol{\psi}^{\pi}(s) = \mathbb{E}_{\pi}\left[\sum_{t=0}^{\infty} \gamma^t \mathbf{f}(s_t) | s_0 = s\right] \tag{2.12}$$

As in the tabular case, the feature-based SR can be used to compute value when multiplied with a vector of reward expectations per feature, $\mathbf{u}$: $V^{\pi}(s) = \psi^{\pi}(s)^T\mathbf{u}$. In the case of linear function approximation, these Successor Features - in equation 2.12 are approximated by a linear function of the features $\mathbf{f}$:

$$\hat{\boldsymbol{\psi}}(s) = W^T\mathbf{f}(s), \tag{2.13}$$

where W is a weight matrix which parameterises the approximation. Intuitively, W encodes how much each feature predicts every other feature. As in the tabular case, temporal difference learning can be used to update the SR weights (see Appendix A). Thus, at every state s (corresponding to a location) in the environment, the agent observed a population vector $\mathbf{f}(s)$ of BVC-driven place cells. It then computed its estimated Successor Features - using its current estimate of weights W and Equation 2.13, which encode the discounted sum of future population firing rate vectors $\mathbf{f}$ of the input place cells. In terms of circuitry, W might correspond to the Schaffer collaterals projecting from CA3 to CA1 neurons, corresponding to $\mathbf{f}$ and -, respectively.

In the context of hippocampus, the feature-based SR allows us to represent states as population vectors of place cells with overlapping firing fields (the features), rather than having a one-to-one correspondence between place

cells and states. Then we are free to model the dependence of the place cell firing on specific environmental features (boundaries). This dependence has been extensively characterised by computational models of boundary vector cells (BVCs) (Barry et al., 2006; Burgess et al., 2000; de Cothi & Barry, 2020; Grieves et al., 2018; Hartley et al., 2000), which were shown to exist in the subiculum (Lever et al., 2009). Accordingly, I modelled a set of hippocampal place cells whose activity $\mathbf{f}_i(s_t)$ was the thresholded sum of a set of BVC inputs (see Barry et al., 2006, for details on how BVC and place cell maps were calculated).

Crucially, modelling place cells as driven by BVCs allows us to explain the puzzling experimental finding by Doeller and Burgess (2008) that learning to navigate to a location relative to a landmark, but not relative to a boundary, is sensitive to the blocking effect (Kamin, 1967). In an accompanying neuroimaging paper, the authors showed that landmark-learning was associated to BOLD activity in the dorsal striatum, whereas boundary-related navigation was associated to activity in the hippocampus (Doeller et al., 2008).

2.2.3 Arbitration process

The agent has access to both its MF DLS component and its hippocampal component employing the SR. Both systems estimate the same value function but might make different types of errors and the agent has to arbitrate between them.

Rational arbitration should reflect the relative uncertainty (Daw et al., 2005), requiring the posterior distribution over values rather than just the values themselves. Here, we use a convenient proxy for uncertainty, introduced by Wan Lee et al. (2014), namely, the recent average of prediction errors: the reward prediction error for the MF component and the successor prediction error for the SR component. If the successor prediction error is low, this means that the SR system has a good estimate of the world. Similarly, if reward prediction errors are low, this means the MF system has a reliable estimate of the value function. The reliability can be tracked using a Pearce-Hall like update rule (Hall, 1991), computing the recent average of absolute prediction errors Ω:

$$\Delta\Omega = \eta(|\delta| - \Omega) \tag{2.14}$$

where $|\delta|$ is the absolute reward prediction error and η is a learning rate. The reliability is defined as:

$$\chi = (\delta_{MAX} - \Omega)/\delta_{MAX} \tag{2.15}$$

with δ_{MAX} being the upper bound of the prediction error, which was set to 1. Since in the model both systems are trained by a prediction error, this applies to both the MF and SR systems. Following Wan Lee et al. (2014), I used the reliability measure for arbitration. These authors computed transition rates α and β for transitioning from MF to MB states and vice versa as follows. Here I use the same terms but for transitions between MF and SR. These transition rates are functions of the reliability of the respective systems:

$$\alpha(\chi_{MF}) = \frac{A_\alpha}{1 + \exp(B_\alpha \chi_{MF})} \tag{2.16}$$

$$\beta(\chi_{SR}) = \frac{A_\beta}{1 + \exp(B_\beta \chi_{SR})} \tag{2.17}$$

where the A and B parameters in both equations determine transition rate and the steepness of these curves, respectively. These parameters were fitted to behavioural data Wan Lee et al. (2014) and I matched their parameter values (see Appendix A, Table A.1). At each time step, the rate of change of the proportion of influence of the SR system P_{SR} was computed using the following differential equation, generating a push-pull mechanism between HPC and DLS influence over behaviour:

$$\frac{dP_{SR}}{dt} = \alpha(\chi_{MF})(1 - P_{SR}) - \beta(\chi_{SR})P_{SR} \tag{2.18}$$

Note that, consistent with behavioural data from human subjects (Wan Lee et al., 2014), this arbitration mechanism results in a weighted influence of both systems in the final value estimates (Figure 2.1), rather than a discrete choice. Note also that the arbitrator combines the action values, not the actions. Thus, the agent will not end up with a mid-way action when the two systems encode different preferences. Lesions or partial inactivations of either the DLS or the hippocampus were achieved by setting limits on P_{SR} (see Appendix A for more details).

2.3 Results

To test the validity of the model, I applied it to spatial and non-spatial decision making tasks and compared its behaviour to that of humans and rodents.

2.3.1 Hippocampal lesions and Water Maze navigation

An adaptation to the classic Morris Water Maze task - in which rodents swim in opaque water to find an invisible platform - involves putting an intra-maze landmark into the pool at a fixed offset from the platform, and moving both platform and landmark to a different location within the tank at the start of each block of four trials (Pearce et al., 1998, see Figure 2.2A). In this version of the task, hippocampally lesioned animals perform *better* than intact animals on the first trial of each session, because intact animals initially linger at the previous goal location (see Figure 2.2B). However, these animals show little intra-session learning while learning across sessions is relatively unimpaired, indicating that they are learning to navigate to the goal location relative to the landmark, since this relationship remains constant across sessions.

In the model, the session-by-session displacement of landmark and platform means that the value function will have to change when using allocentric place cell features, but not when using egocentric landmark cell features. Hence, when I simulated this task by comparing the performance of the full model to a model with a silenced hippocampal component, the model shows the same effects as in the original experiments (Figure2.2C). Fast within-session learning, which relies on the SR's capacity for quick re-evaluation of rewards, is impaired after a hippocampal lesion. Between-session learning, which depends on learning the landmark-platform relations, is unimpaired. Finally, control agents perform worse than hippocampally lesioned agents on the first trial after the platform has been moved, because the value function has changed in allocentric but not egocentric coordinate frames. An inspection of the occupancy maps (Figure 2.2D-F) reveals that equivalent errors are made by the agents and by the rats, i.e. lingering at the previous platform location. The hippocampal predictive map guides the agent to the previous platform location because of its allocentric place representation. Only when it reaches that location and the platform is not there does it start unlearning the hippocampal reward representation, see Equation 2.11.

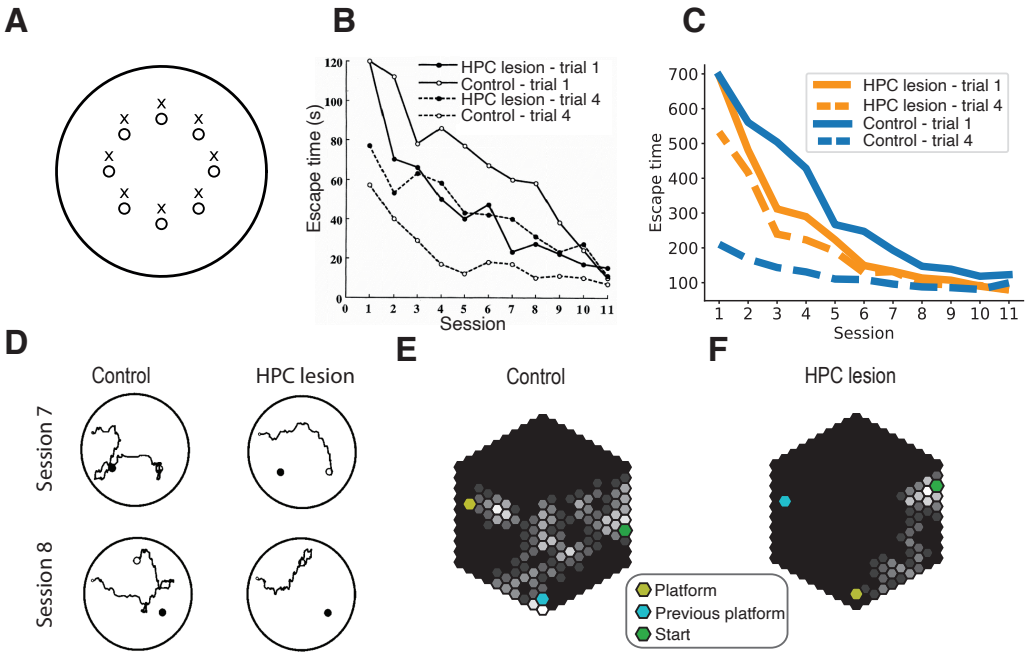


FIGURE 2.2: Results and simulations of the experiment described in Pearce et al. (1998). Sessions lasted 4 trials, and platform and landmark were moved at the beginning of each session. **(A)** Possible locations of the hidden platform (o) and the corresponding landmark (x) in each session. **(B)** Escape latency in the water maze for hippocampal lesioned and control animals on trial 1 (solid lines) and 4 (dashed line) of each session. Hippocampal damage impairs intra-session learning but preserves learning across sessions. Because animals with hippocampal damage follow a response strategy based on egocentric visual input, they perform better on the first trial of each session than control animals (adapted from Pearce et al., 1998). **(C)** Equivalent plot for the full model (blue) and the model without a hippocampal component, relying solely on model-free mechanisms. **(D)** Example trajectories from the first trials of sessions 7 and 8. Animals using a hippocampal place strategy tend to wander around the previous platform location (filled circle) before finding the new platform location (empty circles; adapted from Pearce et al., 1998). **(E)** Occupancy maps show a similar effect for simulated agents. Control agents (left) linger around the previous platform location, whereas agents that cannot use map-based navigation take a more direct path to the new platform location.

Simulating DLS lesions in the task used by Pearce et al. (1998) shows the emergence of the opposite pattern to that of HPC lesions: there is little to no learning across sessions for the first trials, while fourth-trial performance is not significantly worse than control performance (see Appendix A, Figure A.2A). This is consistent with previous findings showing that lesions of the DLS induced a preference for place-guided navigation (Devan et al., 1999) and that dopamine depletion in the DLS impairs egocentric but not allocentric water maze navigation (Braun et al., 2015). Our model also accurately captures results from Miyoshi et al. (2012), who classified navigation behaviours as cue-guided or place-guided in the cued water maze task after lesions to both the HPC and the DLS (see Appendix A, Figure A.2B-C).

These results show that our model captures both landmark-guided and place memory-guided behaviour on the Water Maze. Furthermore, our model gives a normative perspective on why the animals switch to a landmark-based strategy: since the striatal system learns about the rewarded location with respect to landmarks, it can use the landmark to navigate directly to the correct location on the first trial of a given session. This gives an advantage to using the striatal system for decision making, which agents learn to exploit. Over the course of multiple sessions, the average prediction error of the striatal system will decrease, causing the reliability-based arbitration mechanism to favour the striatal system, driving lower escape times on first trials of later sessions.

2.3.2 Animals switch to a response strategy on the Plus Maze

The distinct roles of the hippocampus and dorsal striatum have also been investigated using the place/response learning task (Packard, 1999; Packard & McGaugh, 1996). In this task, rats were trained to find a food reward on one arm of a Plus Maze, starting in the same arm every time, while the opposite arm is blocked (Figure 2.3). After training, a probe trial is performed in which the animal starts at the opposite end of the maze. If animals take the same egocentric turning direction as before, thus ending up at the opposite goal arm, their strategy is interpreted as response learning (relying on a remembered egocentric turn). If they take the opposite turn to end up in the same goal arm, their strategy is interpreted as flexible place learning (relying on an allocentric representation of space).

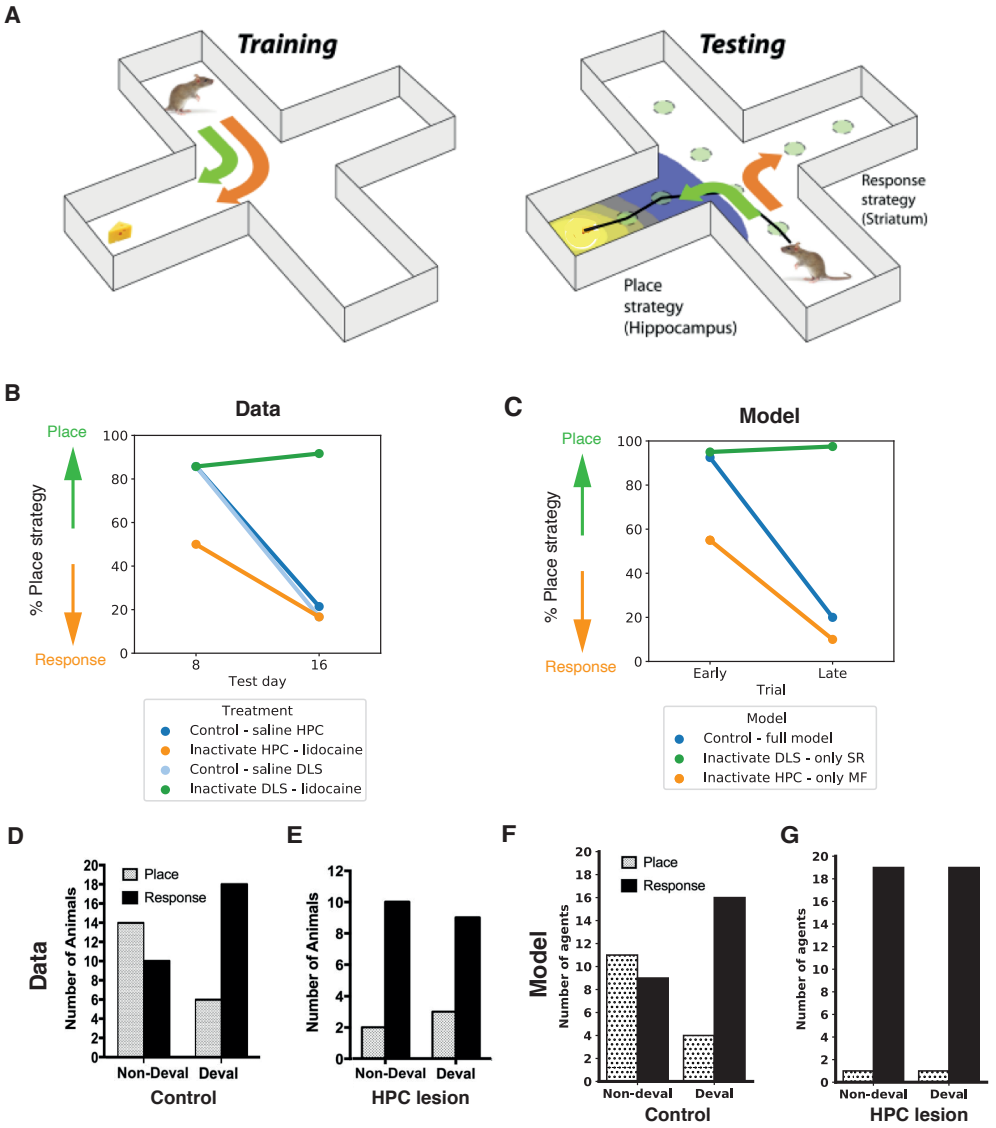


FIGURE 2.3: Navigation in the Plus Maze. **(A)** Experimental setup used by Packard and McGaugh, 1996. During training, animals were trained to run from the same starting place to a baited goal arm. During probe trials (on day 8 and day 16), the animal started in the opposite arm. If the animal ran to the same allocentric location as during training, this was labelled as a place strategy (green). Taking the same egocentric turn to end up in the opposite goal arm was classified as a response learning strategy (orange). **(B)** Behavioural data from Packard and McGaugh, 1996. Control animals (blue) showed a shift to response learning over the course of training. This was prevented by the inactivation of DLS using lidocaine. The inactivation of HPC using lidocaine caused animals to use a response strategy early on. **(C)** Model results recapitulate these findings. **(D-E)** Behavioural data from Kosaki et al., 2018 showing probe trial behaviour before and after the outcome was devalued by pre-feeding the animal with the food reward, for control (D) and hippocampally lesioned animals (E). **(F-G)** Model simulation results recapitulate these findings.

Figure 2.3 shows the results of the original experiment and our simulations. Early in training most control rats (injected with saline) use a place strategy, but switch to a response strategy after extensive training. Inactivation of the dorsal striatum with lidocaine prevented this switch. Inactivation of the hippocampus, by contrast, caused the response strategy to be used more often even early in training. These results indicate that the dorsal striatum supports response learning, while the hippocampus supports place learning. I simulated the lidocaine inactivation of hippocampus and dorsal striatum by partly deactivating the SR and MF components of our model, respectively. Early in training, the control agent shows a preference for actions proposed by the hippocampus, leading the agent to follow a place strategy. This is because the SR reliability is higher than the MF reliability at the start of training, reflecting the fact that animals have explored the environment without rewards before training. Over the course of training, reward prediction errors in the striatum decrease, causing the reliability of the MF system to increase, at which point the model switches to the MF strategy because of a bias to use the more computationally efficient system. Inactivation of the dorsal striatal and hippocampal components of the model biases the agent to follow a place or response strategy, respectively.

While the results described above show that the DLS and HPC are involved in ego- and allocentric navigation, respectively, the navigational strategy alone does not speak to an important aspect of MB learning: flexibility in the face of reward devaluation. In devaluation studies, the value of a reinforcer is decreased by pairing it with an aversive event such as illness, or by inducing satiety by pre-feeding the animal with the reinforcer (Adams & Dickinson, 1981). Since MF algorithms need to re-experience the state/action leading to the devalued reward to update its value, MF behaviour (also referred to as stimulus-response learning) is insensitive to devaluation. MB algorithms, in contrast, can estimate that state/action transitions will lead to a devalued reward without having to re-experience them. This goal-directed, devaluation-sensitive behaviour is a hallmark of MB planning (Daw et al., 2005).

To investigate the relationship between place and response learning on one hand, and goal-directed and stimulus-response learning on the other, I simulated results from Kosaki et al. (2018), who studied devaluation on the Plus Maze. Specifically, they trained rats on the same task as described in

Figure 2.3A (see De Leonibus et al., 2011, for a similar study in mice). Subsequently, they devalued the food reinforcer by prefeeding the animals. The results of this devaluation procedure are depicted in Figure 2.3D. Consistent with the idea that the place strategy is sensitive to the expected value of the outcome, while the response strategy is not, the procedure resulted in a switch from place to response strategies. Furthermore, rats with hippocampal lesions displayed a reliance on the response strategy, regardless of outcome devaluation (Figure 2.3E), further indicating that the response strategy is insensitive to devaluation. Since sensitivity to reward devaluation is also a property of SR-based learning (see e.g. Gardner et al., 2018), our model naturally accommodates these results.

2.3.3 Blocking in landmark but not boundary related navigation

A signature of learning stimulus-reward associations using reward prediction errors is the blocking phenomenon (Kamin, 1967). Learning one stimulus-reward association hinders learning of a subsequent association between a different stimulus and the same reward because the prediction error becomes small, reducing further weight updates. In humans, spatial blocking has been shown to occur when learning locations relative to discrete landmarks, but not relative to boundaries (Doeller & Burgess, 2008). Furthermore, learning with respect to landmarks corresponds to increased BOLD signal in the dorsal striatum, whereas learning with respect to boundaries corresponds to activity in the posterior hippocampus (Doeller et al., 2008).

I aimed to capture these effects by examining the behaviour of our agent, following a paradigm similar to (Doeller & Burgess, 2008) (see Figure 2.4): the agent navigated through an open field to find an unmarked reward location. In order to investigate blocking with respect to boundaries, I explicitly modelled the effect of boundaries on hippocampal place cells, given their dominant role in determining place cell firing fields (cf. Cressant et al., 1997; O'Keefe & Burgess, 1996). Rather than learning an SR over a punctate state representation the agent learned a matrix of successor features provided by the firing rates of a set of place cells driven by boundary vector cells (BVCs) (see section 1.1.2; Barry et al., 2006; Bicanski & Burgess, 2020; Hartley et al., 2000; Lever et al., 2009).

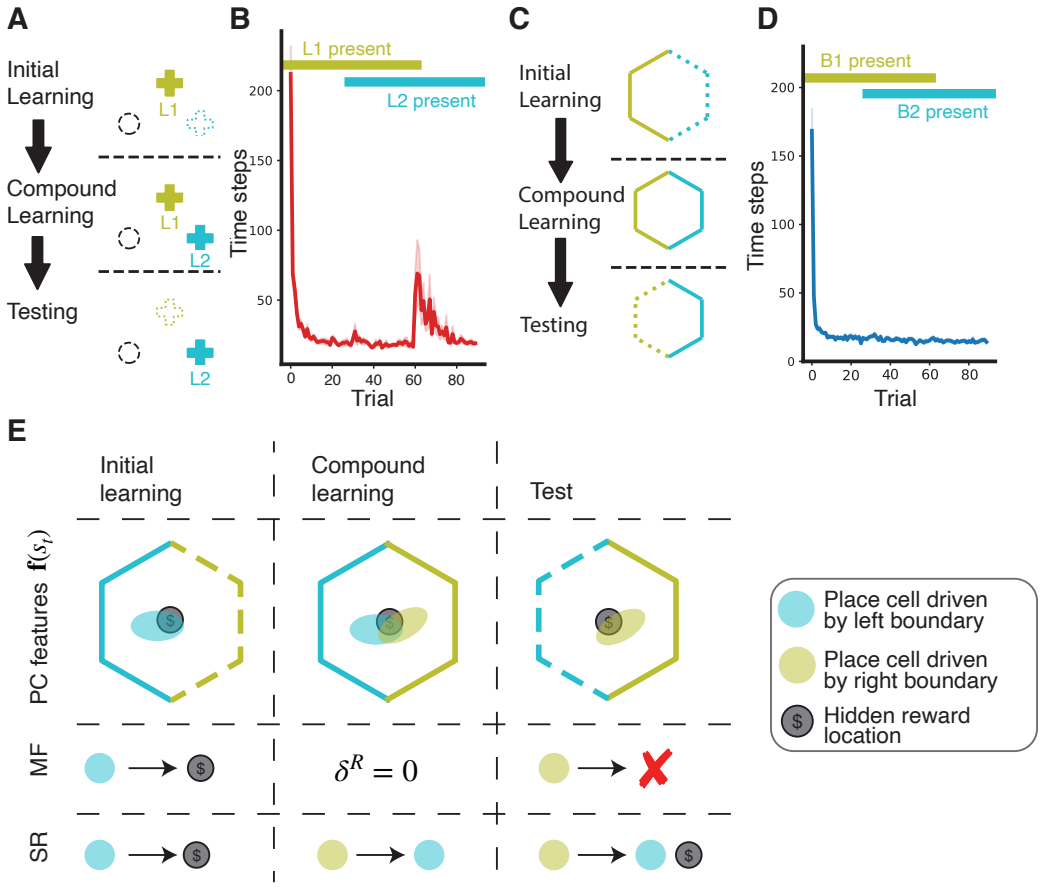


FIGURE 2.4: Boundary versus landmark blocking experiments, similar to Doeller and Burgess (2008). **(A)** Landmark blocking experiment. Agents navigate a virtual water maze to find a hidden platform (dashed circle). During initial learning, one landmark is present (L1). During compound learning, a second landmark is added (L2), after which L1 is removed. **(B)** Average time to find the platform per trial. Increased escape times on removal of L1 indicates blocking of learning about platform location relative to L2 by the prior learning relative to L1. **(C)** Boundary blocking experiment, following (A) but with two boundaries (solid green and blue lines). **(D)** Average escape time shows no effect of blocking of learning platform location relative to right boundary (blue) when left boundary (green) is removed. **(E)** Illustration of the lack of blocking in boundary-related learning under the SR system, in contrast to a MF system.

In the landmark blocking condition (Figure 2.4A-B), the agent used a landmark to guide navigation. After 10 trials, a second landmark was added, and after 20 trials the first landmark was removed. Importantly, in this experiment there were no boundaries and only one or two landmarks visible at any time. A single landmark has little effect on place cell firing (Cressant et al., 1997) and indeed, the presence of a single or two landmarks does not support a reliable place cell map (Barry et al., 2006). Therefore, and consistently with BOLD activation results (Doeller et al., 2008), I assume that behaviour was controlled by the DLS in this experiment.

As predicted by the TD learning rule, and consistent with the findings of Doeller et al. (Doeller & Burgess, 2008), learning about the second landmark was blocked by the prior learning about landmark 1, as evidenced by the drop in performance after its removal.

In the boundary blocking condition (Figure 2.4C-D), there were no landmarks, meaning that the agent had to rely on its hippocampal system for navigation. The hippocampal system learns a predictive map over boundary-related place cell activations using successor prediction errors (SPEs, see Appendix A). Prediction error-based learning like that is susceptible to the blocking effect, and the SR has indeed been used as an explanation for the occurence of blocking, when learning stimulus-stimulus associations (Gardner et al., 2018). However, when we subject the agent to a boundary-related blocking paradigm, no blocking occurs (Figure 2.4C-D).

To understand why this happens, consider the situation in Figure 2.4E, in which one example place cell is active at the rewarded location, driven by the left boundary. During initial learning, an association between that place cell and the reward is learned. During compound learning, a second boundary drives the activity of another place cell at the rewarded location. In a MF system, the learned value associated to the previous place cell means there is zero prediction error, preventing learning of an association between the second place cell and the reward. In a SR system, however, the agent learns a predictive relationship between the two place cells. Thus, while there is no reward prediction error, and the reward vector remains unchanged, the newly firing place cell comes to predict the firing of the first place cell (that is associated with reward), mitigating its reduction in firing when the first boundary is removed. This means that, when the first boundary and its associated firing are removed, the agent still predicts reward at the correct location. Thus, consistent with behavioural evidence (Doeller & Burgess,

2008; Doeller et al., 2008), our model shows no blocking effect during the boundary-related navigation paradigm. This result speaks to the utility of structure learning: the hippocampal SR system learns a multitude of relations, such that its policies are more robust to change in cues and rewards.

Boundary-learning blocks landmark learning but not vice versa

In addition to boundary-to-boundary and landmark-to-landmark blocking, Doeller et al. (2008) also asked whether learning about a goal location relative to a landmark can block learning relative to a boundary (LB) and, vice versa, whether learning relative to a boundary blocks learning relative to a landmark (BL). In the BL condition, a boundary is indicating the goal location during the learning phase, with an intra-maze landmark being added during the compound learning phase, before the boundary is then removed during the test trial. Conversely, in the LB condition, an intra-maze landmark is initially indicating the location of the goal location, with a boundary being added during the compound learning phase, before the landmark is then removed during the test trial. Strikingly, the authors found that prior boundary-guided learning blocked subsequent landmark-guided learning (B blocks L), while the opposite setup did not result in a blocking effect (L does not block B).

Can this more complicated pattern of results be explained by the same model? Following the same logic that explained the BB and LL experiments, the lack of blocking observed in the LB condition can be explained by the hippocampal component being in control of behaviour during the test phase. This is because the removal of a single landmark during the test phase has only little effect on place cell firing (Barry et al., 2006), resulting in a low average feature prediction error (i.e. high reliability) in the HPC system. The boundary included in the compound phase allows much more spatially reliable place cells firing, so the Q value function improves and is not much disrupted by removal of the landmark. By contrast, there is high reward prediction error in the DLS because a discrete landmark is a more important cue than the extended boundary in terms of the egocentric sensory input that the DLS relies on for value prediction (see also Figure 2.6). Hence, consistent with the experimental results observed in Doeller et al. (2008), no L-to-B blocking is observed in the model (Figure 2.5B).

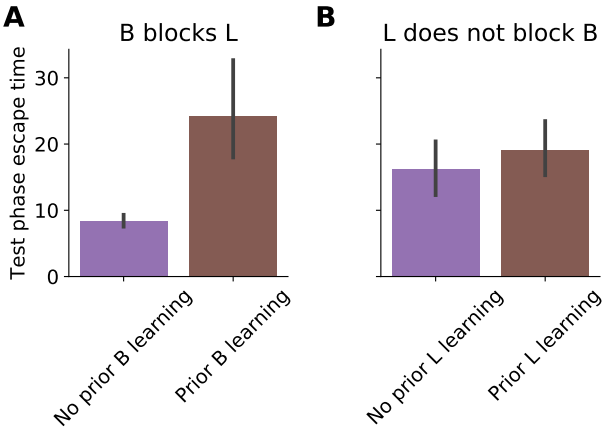


FIGURE 2.5: Boundaries block landmarks but landmarks do not block boundaries. **(A)** Mean escape time (number of steps until the goal state is reached) during the test phase probe trial with only the intra-maze landmark present when the agent first learned with a boundary present (brown) or without prior learning with a boundary (purple). **(B)** Mean escape time during the test phase probe trial with only the boundary present with (brown) or without (purple) prior learning relative to a landmark.

Conversely, in the BL condition, the fact that blocking is observed suggests that (i) the DLS is in control of behaviour during the test phase and (ii) that during initial learning with respect to the boundary, prediction errors are reduced sufficiently to block some learning when a landmark is then added during the compound phase. If DLS reward prediction error were reduced completely to zero by learning about the boundary, the DLS would take over control during boundary-related learning, meaning that blocking would also be expected in the BB condition. However, if DLS prediction errors were only partly reduced, one might still see a blocking effect in the BL condition, even though HPC is in control during boundary-related navigation. Such intermediate-level prediction error might be expected during the compound phase after boundary-related navigation because extended boundaries are simply a less informative landmark for the egocentric system, compared to pillar-like landmarks (see also Sheynikhovich et al., 2009).

In the egocentric landmark system, extended boundaries activate more cells at the same time, yielding a less precise relationship between landmark cell activation and optimal heading direction. This can be observed in the model by reduced performance when the striatal component alone is in charge of behaviour while there are only boundaries (and no intra-maze landmarks) available for orientation (Figure 2.6). Hence, when learning to

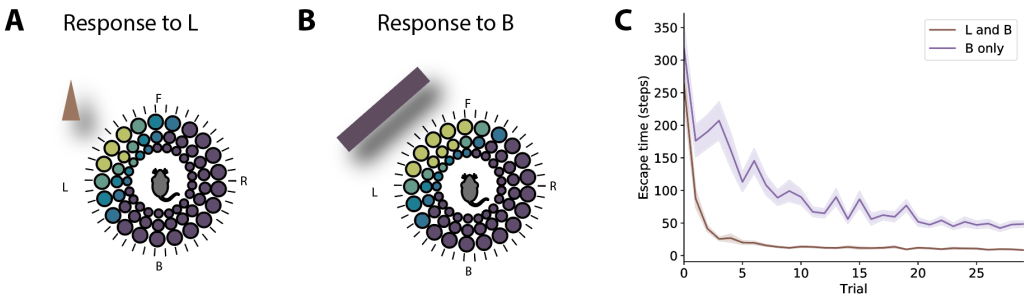


FIGURE 2.6: Striatal learning relative to boundaries is less effective than relative to intra-maze landmarks. **(A)** Illustration of striatal landmark cell responses to a pillar-like landmark. **(B)** Illustration of striatal landmark cell respones to an extended boundary. **(C)** Escape time in steps is shown per trial as the agent with only DLS active learns to navigate to a set goal location while both boundaries and a landmark are present (brown) or when only a boundary is present (purple).

navigate relative to boundaries, striatal prediction errors are expected to be reduced to an intermediate level, with an associated intermediate level performance. During the compound phase, this reduction in prediction error means there is reduced landmark-related learning, compared to a situation without prior learning relative to a boundary (Figure 2.5).

In the context of blocking, several experimental predictions can be drawn from this model. Firstly, boundary related navigation should be less effective in animals with hippocampal lesions. Secondly, because pillar-like landmarks are a superior input for the egocentric navigation system, the striatal dopamine signal that occurs when the hidden goal location is reached should be more reduced during landmark-related navigation the second time the goal is found than during boundary-related navigation. Thirdly, if the HPC were to be inactivated during the LB condition, or indeed during the BB condition, blocking should still be observed, since it was only because the hippocampus was in control of behaviour that blocking was not observed.

2.3.4 Two step task

Outside of the spatial domain, the distinction between model-free and model-based reinforcement learning has been heavily investigated using sequential decision tasks. Here I describe how our model solves a cognitive decision task of this type – the task of Daw et al. (2011) (see Figure 2.7A).

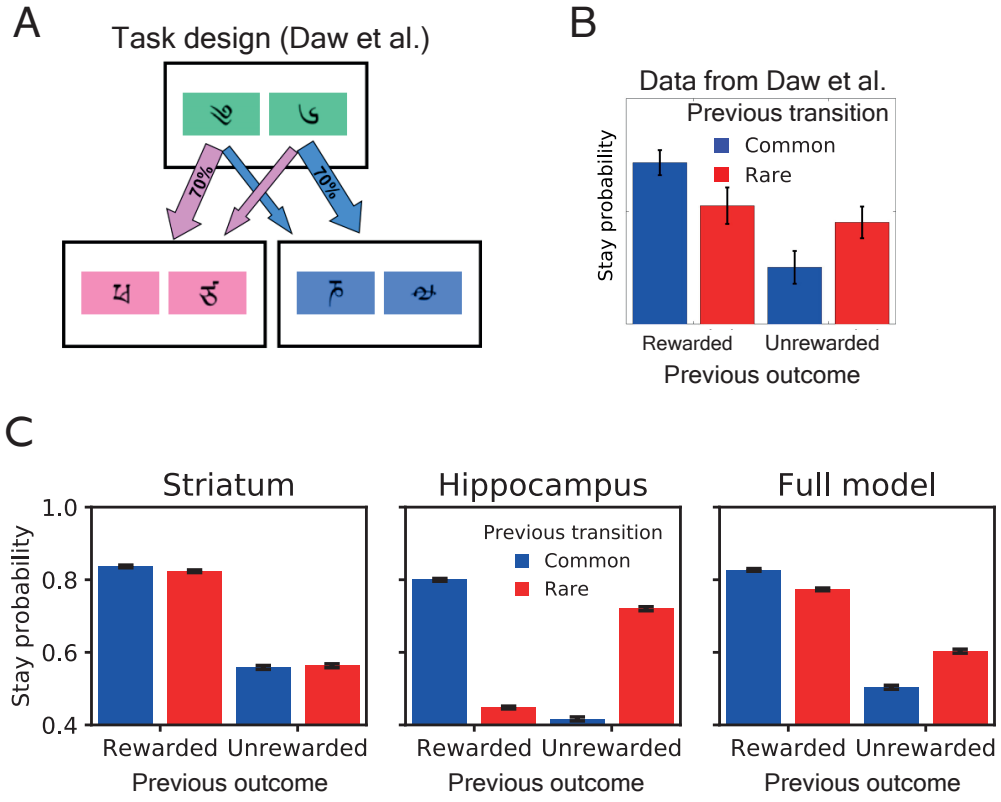


FIGURE 2.7: A non-spatial two step task. **(A)** Task employed by Daw et al. (2011). Here, a single start state leaded probabilistically to one of either two second states, depending on the action chosen and whether by chance a rare (70%) or common (30%) transition was made. **(B)** Data from Daw et al. (2011) showing that human performance lies in between MF and MB. **(C)** Simulation results for the striatal, hippocampal and full model, respectively.

In the two-step decision task designed by Daw et al. (2011), human participants were shown a pair of symbols and asked to choose one (Figure 2.7A). Left or right choices lead to different corresponding second-stage states with high probability (common transitions), but there was a small probability (rare transitions) that the agent transitions to the opposite state. For example, in Figure 2.7A, the left icon in the first (green) state usually leads to the choice in the pink state (common transition), but occasionally leads to the choice in the blue state (rare transition). During the second stage, participants made another left-or-right choice, resulting in either receiving a reward or not, before starting the next trial. Each of the four outcomes was associated with a reward probability that varied over time as a Gaussian random walk limited between 0.25 and 0.75.

The rewards received or not received on a given trial modify the participants' value estimates for the different actions taken during the two stages, but different RL strategies lead to different behaviours on the next trial. Model-free learners increase the likelihood of repeating their first-stage action following a reward, regardless of whether a common or rare transition was made. In contrast, model-based learners use knowledge of the task's transition structure, such that rewards obtained after a rare transition lead to the opposite choice on the next trial (to maximise the likelihood of reaching the same second state). The key finding of Daw et al. (2011) was that human choices reflect both model-based and model-free influences (Figure 2.7B).

Our model recapitulates these findings and suggests the HPC could support MB choice in this task, as well as another two-step decision task with deterministic transitions (see Appendix A, Figure A.3; Doll et al., 2015). The model DLS, implementing a model-free RL system, increases stay probability after rewards regardless of whether a rare or common transition was made (Figure 2.7C). In contrast, the HPC uses the SR to generalise value over the graph. When a goal state is reached and a reward is obtained, value is generalised over the graph according to the degree to which states predict each other. Therefore, on the next trial, the actions are taken that will most likely lead to the recent goal state. Separating transition dynamics from reward estimates thus recapitulates true model-based behaviour. Combining the two systems results in behaviour that is similar to that of human participants in this task.

It has been shown previously that other, simpler models than pure MB systems can look like MB agents on the two-step task (Akam et al., 2015).

Here, I show that the SR can mimic MB behaviour. Because the transition structure is unchanging, caching future state predictions is sufficient for flexible behaviour.

2.3.5 Relationship between spatial and two-step tasks

A central principle of our model is that model-based reasoning and allocentric navigation strategies both rely on the same hippocampal structures. The most direct evidence for this comes from Vikbladh et al. (2019), in which both healthy participants and patients with hippocampal damage performed the two-step planning task (Daw et al., 2011) as well as a Landmark versus Boundary spatial memory task (Doeller et al., 2008). This allowed the authors to show that, in healthy participants, the degree of MB planning on the sequential decision task correlates with the contribution of allocentric, boundary-driven place memory on the spatial task (reflected in smaller errors from the location predicted by the boundary; Figure 2.8A). Notably, this correlation cannot be accounted for by variation in general intelligence (IQ). In patients with hippocampal damage, however, this relationship was significantly reduced.

To test for this effect in our model, I sampled a set of 20 agents with different values for the parameters governing the hippocampal-striatal trade-off, as well as 20 agents with a partially lesioned hippocampal component (see Appendix A). Each agent performed the two-step decision task (Daw et al., 2011) and the Water Maze task of Pearce et al. (1998), depicted in Figure 2.2. MB planning was quantified as the interaction between effects of reward and transition type in the previous trial on staying with the same action or switching in the next trial (see Appendix A, c.f. Daw et al., 2011; Vikbladh et al., 2019). I quantified the degree of allocentric place memory as the average distance between the previous platform location and the location of the maximum of the agent's value function at the start of the next session. This is akin to the boundary distance error employed by Vikbladh et al. (2019). I found a significant correlation ($z = 1.89, p < 0.001$) between model based and allocentric planning (Figure 2.8B). Agents with hippocampal lesions did not show a significant correlation ($z = -0.02, p = 0.97$), and the difference between these correlation coefficients was significant ($z = 5.44, p < 0.001$), recapitulating the result found by Vikbladh et al. (2019).

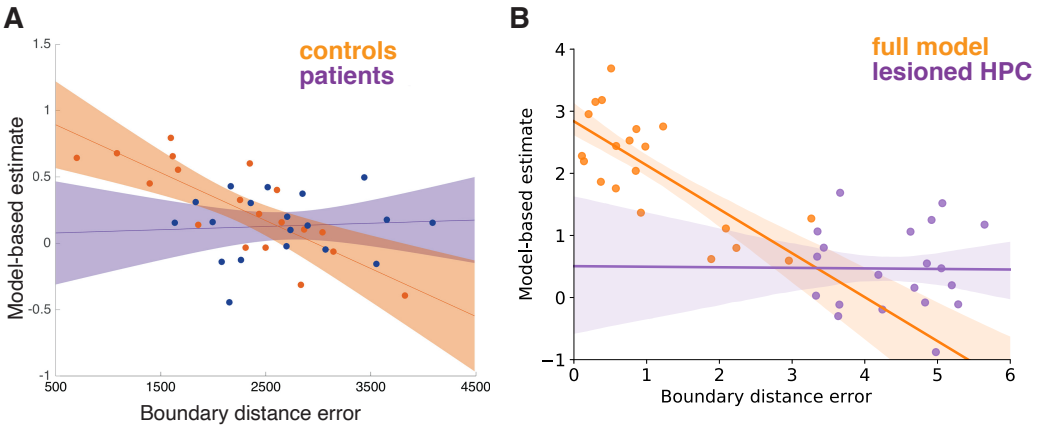


FIGURE 2.8: Relationship between model-based planning and allocentric spatial memory. Error bars indicate 80% confidence intervals of the regression in both panels. (A) Data from healthy control participants and anterior temporal lobectomy patients, reprinted from Vikbladh et al. (2019). Allocentric place memory is reflected by responses close to the boundary-predicted location after the landmark has moved (i.e., smaller boundary distance errors). Dots indicate model-based estimates for individual participants, calculated from a mixed-effects logistic regression. (B) Simulation data for the full model and agents for which the hippocampal (HPC) component was turned off. Here, allocentric place memory is reflected by the average distance between the previous platform location and the location of the maximum of the agent's value function at the start of the next session. Dots represent estimates for individual agents, estimated by a mixed-effects logistic regression.

2.4 Discussion

I presented a model of hippocampal and dorsolateral striatal contributions to learning across both spatial navigation and non-spatial decision making. Our simulations support the view that the hippocampus serves both allocentric place learning and flexible decision making by supplying a predictive map of the underlying structure of the task or environment, whereas the dorsolateral striatum underlies model-free learning based on (egocentric) sensory features and actions, and that these systems combine weighted by their relative reliability in predicting outcomes.

The involvement of the hippocampus in abstract non-spatial tasks raises questions about its role throughout evolution. Did the system evolve initially in the spatial domain but become recruited more generally (Dring & West, 1983), or was spatial decision making always part of a more general ability (Eichenbaum et al., 1992)? The role of the hippocampus in MB decision making is much debated. On one hand, lesions of the hippocampus have not affected hallmarks of MB planning such as outcome devaluation in lever-pressing tasks (Corbit & Balleine, 2000; Corbit et al., 2002), although a recent study showed that hippocampus is involved in devaluation-sensitivity of lever pressing immediately after acquisition (when pressing is context dependent, Bradfield et al., 2020). On the other hand, hippocampal lesions led to a loss of devaluation-sensitivity on the Plus Maze (Figure 2.3, Kosaki et al., 2018) and impair MB behaviour on the two-step task (Figure 2.7, Miller et al., 2017; Vikbladh et al., 2019). One crucial difference between the lever-pressing tasks and the tasks simulated here is that the lever-pressing tasks required only one action-outcome association, whereas solving the two-step task and many spatial tasks require chaining multiple action-outcome associations together. Perhaps then, as suggested by Miller et al. (2017), the hippocampus is specifically required when planning requires linking actions to outcomes over multiple steps. By storing temporal abstractions of future states separately from a representation of reward, the SR is particularly well-suited for this task of rapidly propagating novel reward information to distant states. That property of the SR has previously inspired models of temporal context memory (Gershman et al., 2012) and might also relate to the role of relational memory tasks more broadly, as they require chaining multiple stimulus-stimulus associations together (Bunsey & Eichenbaum, 1996; Dusek

& Eichenbaum, 1997). In line with this role, our simulations showed the hippocampal SR as driving a correlation between spatial memory performance and MB behaviour (Figure 2.8, Vikbladh et al., 2019).

Consistent with our model, dorsal striatal neurons show a great degree of spatial coding in spatial tasks (van der Meer et al., 2010), but not in tasks where reward locations were explicitly dissociated from space (Schmitzer-Torbert & Redish, 2008), or where multiple locations were equivalently associated with rewards (Berke et al., 2009). Indeed, dorsal striatum selectively represents those task aspects which computational accounts suggest are important for gradual, model-free learning (van der Meer et al., 2010).

The striatal controller in the work presented here is specifically associated with the dorsolateral striatum. Lesion and inactivation studies have shown that the dorsal striatum is functionally very heterogeneous (Yin & Knowlton, 2006). Lesions of the dorsomedial striatum (DMS) result in a switch to response strategies on the Plus Maze (Yin & Knowlton, 2004), and to cue based responding in the Water Maze, while the DLS underlies response learning (Devan et al., 1999). Furthermore, the DMS has been implicated in learning action-outcome contingencies outside the spatial domain (Yin & Knowlton, 2006; Yin et al., 2005). Anatomical connectivity supports this functional dissociation in the dorsal striatum (Devan et al., 1999; Yin & Knowlton, 2006). Whereas the DLS receives inputs mostly from sensorimotor cortex and dopaminergic input from the substantia nigra, the DMS receives input from several meso and allocortical areas including the hippocampus. Indeed, cells encoding route and heading direction have been found in the DMS (Mulder et al., 2004; Ragozzino et al., 2001). It is therefore likely that the dorsal hippocampus and the DMS are part of a single circuit involved in flexible goal-directed decision making, whereby the hippocampus provides map-based information, and the DMS is involved in action selection.

Our work follows several models of spatial decision making by hippocampal and striatal systems (Chersi & Burgess, 2015; Dollé et al., 2018; Dollé et al., 2010; Foster et al., 2000; Gustafson & Daw, 2011). Dollé and colleagues used a similar hippocampo-striatal model to explain behaviour on the adapted Water Maze task (Pearce et al., 1998) presented in Figure 2.2 (Dollé et al., 2018; Dollé et al., 2010). Our model differs in two important ways. Firstly, in their model place cells connected to "graph cells" that formed an explicit topological graph of the spatial environment, used to explicitly plan a path to the goal. In the present model, by contrast, the topological structure of

the environment is implied in the predictive "successor representation", following a theoretical proposal by Stachenfeld et al. (2017) and neuroimaging (Garvert et al., 2017; Schapiro et al., 2016) and behavioural findings (Bellmund et al., 2019). Thus, our agent mimicked true model-based behaviour (explicit graph search) by using an intermediate SR-based strategy. Secondly, their model used another expert network that learned whether to take striatal or hippocampal outputs using TD learning. In contrast, our model arbitrates between systems based on their reliability. This arbitration mechanism predicts that on trials with high reward prediction error, control should shift away from the MF system. In contrast, a low predictability of state transitions leads to higher average errors in the SR system and should therefore lead to a higher degree of MF control. Evidence for this comes from Wan Lee et al. (2014), who furthermore showed that the prefrontal cortex encodes neural correlates of arbitration based on reliability.

As noted above, the hippocampal results I simulated are also consistent with a fuly MB system, which is strictly more flexible. An interesting question is how to disambiguate between animals using a MB strategy versus the SR? One weakness of the temporal-difference SR model used here is that it cannot respond flexibly when the transition structure changes. Momennejad et al. (2017) have shown that humans are better at revaluating when the reward function changes than when the transition structure changes, consistent with use of an SR. In addition, hippocampal replay has been suggested to perform offline updates of the hippocampal predictive map to incorporate these kinds of transition changes (Evans & Burgess, 2019; Russek et al., 2017). As an alternative, tracking input covariances and using these for updating the SR, allows it to solve certain kinds of transition revaluation problems without requiring explicit forward simulation (Geerts et al., 2019). A second weakness of the SR, compared to MB systems, is that the SR is policy-dependent. This means that the SR corresponding to an optimal policy for one reward setting is of limited use for problems with a different reward function (Lehnert et al., 2017). Piray and Daw (2019a) have recently proposed that the hippocampal system might resolve this latter weakness using a *default representation*, corresponding to a default policy. Alternatively, the hippocampus might represent a set of multiple distinct SR maps corresponding to different policies (Madarasz, 2019). Taken together, these two failure modes of the SR provide interesting avenues for experiments probing animals' behavioural strategies and for theoretical work on computational

tradeoffs between these strategies.

In addition to the HPC, the orbitofrontal cortex (OFC) has been hypothesised to be important for representing states in RL problems. Wilson, Niv & colleagues introduced a model in which OFC plays a critical role in identifying states that are perceptually similar (Wilson et al., 2014). This corresponds to data showing that OFC is specifically necessary for decision making in partially observable environments (Bradfield et al., 2015). Evidence for this theory comes from human fMRI research showing that unobservable task states can be decoded from OFC, and that this relates to task performance (Schuck et al., 2016). This proposed role of the OFC is distinct from, and possibly complementary to, our proposed role for the HPC. In our model, the HPC encodes a predictive map based on observable features that can be used for rapid, flexible decision making. The OFC, on the other hand, is crucial for a general state representation that can be used for downstream MB or MF processes. Whether and how the OFC and the HPC can interact to allow SR learning in partially observable environments is an interesting avenue for further research (see also Vertes & Sahani, 2019).

Our explanation for the absence of boundary-related blocking (Figure 2.4) relies on boundary vector cell inputs to hippocampal place cells. BVCs can respond to intra-maze landmarks as well as to boundaries (although, in contrast to DLS landmark cells, BVCs fire irrespective of object identity; Bicanski & Burgess, 2020). This means that a sufficient number of landmarks could drive a reliable place cell representation of space, allowing hippocampal control and the prevention of blocking. However, in the experiments simulated here, there were only one or two landmarks present. Single landmarks have little influence on firing relative to extended boundaries (Cressant et al., 1997), consistent with the BVC model. Because BVCs fire proportionally to the angle subtended by the stimulus (Burgess & Hartley, 2002), place cells do not provide a reliable representation of space when there is only a single landmark (Barry et al., 2006). Thus, I predict that the addition of greater numbers of landmarks should allow construction of a reliable place cell map thereby leading to increased hippocampal influence and a reduction of blocking effects.

Our model reflects the assumption, driven by our knowledge of the neural representations, that in spatial tasks the hippocampal SR system uses allocentric representations, while the MF system uses egocentric representations. This allowed us to fit the behavioural data well, and raises the question

of why the goal-directed system is allocentric, while the stimulus-response system is egocentric? Perhaps an answer lies in the time-scale of learning: the allocentric layout of a large environment is stable irrespective of your changes in location or direction, making it suitable for learning long-term relationships between stimuli. Consistent with this idea, "slow feature analysis" produces grid and place cell representations from visual inputs because they vary slowly (Schoenfeld & Wiskott, 2015). On the other hand, egocentric representations are more suited to mapping sensory inputs to physical actions, both of which are specified egocentrically.

In conclusion, dorsal hippocampus and DLS support qualitatively different strategies for learning about reward in spatial as well as non-spatial contexts, as captured by the model presented here. The fact that the same model explains behaviour in both types of task implies that the hippocampal-striatal system is a general purpose learning device that adaptively combines MB and MF mechanisms.

Chapter 3

Uncertainty and the predictive map

Parts of the work presented in this chapter have been previously published as conference papers in Computational Cognitive Neuroscience (CCN) as Geerts et al. (2019) and as a short paper at the Bridging AI and Cognitive Science workshop at ICLR 2020.

3.1 Introduction

Humans and other animals are able to solve a wide variety of decision-making problems with remarkable flexibility. This flexibility is thought to derive from an internal model of the world, or 'cognitive map', used to predict the future and plan actions accordingly. A recent theoretical proposal suggests that the hippocampus houses a model in the form of a representation of long-run state expectancies (Stachenfeld et al., 2017). These "Successor Representations" (SRs; Dayan, 1993) occupy a middle ground between classical model-free (MF) and model-based (MB) Reinforcement Learning (RL) strategies, and resemble aspects of hippocampal place cell activity.

A second source of behavioural flexibility in computational models derives from an optimal treatment of uncertainty in the environment. Previous work has demonstrated that a range of animal learning phenomena can be explained by Bayesian generalizations of simple model-free learning algorithms (Dayan & Kakade, 2001; Dayan & Yu, 2003; Gershman, 2015). These theories posit that, rather than learning a point estimate of the expected value, animals track a posterior distribution over expected value. These distributions contain additional information about the variance of parameters of the value function, which reflect uncertainty, as well as information about

the covariance between inderdependent parameters. These uncertainty and interdependency terms can explain why animals learn more slowly in situations of low uncertainty (as evidenced by phenomena such as latent inhibition) and why they can learn about stimuli that are not currently present (as evidenced by phenomena such as backward blocking) (Dayan & Kakade, 2001; Gershman, 2015).

In addition to uncertainty about value, animals might also be uncertain about which previously experienced task or context the current observations belong to. This type of uncertainty is of interest because the hippocampus is also implicated in context-dependent behaviour (Holland & Bouton, 1999). Previous theoretical work has suggested that the hippocampus performs this role in context-dependent behaviour by clustering observations as belonging to different latent causes (Gershman et al., 2010), with implications for memory updating (Gershman et al., 2014) and hippocampal remapping (Sanders et al., 2020).

In this chapter, I introduce an extension to the SR model by augmenting it with an ability to track and manage uncertainty using Kalman filtering. A probabilistic interpretation will allow an optimal, Bayesian treatment of uncertainty. This can be useful for, for example, balancing prior beliefs and new sensory evidence, and finding a solution to the explore-exploit dilemma (Geist & Pietquin, 2010a; Gershman, 2015). As in the model-free case described by Gershman (2015), this allows for tracking uncertainty and covariance, explaining a set of animal learning phenomena that require learning about stimuli that are not currently present. I then generalise this approach to a multiple task or multiple context setting using a Bayesian nonparametric switching Kalman filter (Gershman et al., 2014), allowing the model to learn and maintain multiple task-specific SR maps and infer which one to use at any moment based on its sensory observations. I show that this Bayesian SR model captures animal behaviour in tasks which require contextual memory and generalisation.

3.2 Model description

This paper addresses the problem of how to deal with uncertainty when learning a predictive map. As in previous work (Dayan, 1993; Stachenfeld

et al., 2017), this predictive map takes the form of a Successor Representation (SR). Our first contribution is to introduce a probabilistic SR, in which the agent's belief about the parameters of the SR is expressed in terms of a distribution over possible Successor Representations. This enables efficient learning by making use of the second-order statistics of predictions about future states or features, and can be used to understand a range of animal learning phenomena. Our second contribution is to extend to the probabilistic SR to a probabilistic *hierarchical* SR in which the agent can switch between multiple SR maps when the environment, task, or context changes. In this section I describe the pieces of the model in sequence. First, I describe how to handle uncertainty when learning the SR in a single environment using Kalman Temporal Differences (KTD) (Geist & Pietquin, 2010a; Gershman, 2015) applied to the SR. Next, I describe how to generalise this for multiple environments and context-dependent SR maps by using a non-parametric Switching Linear Dynamical System (SLDS) (Fox et al., 2011; Gershman et al., 2014; Murphy, 1998), that infers new maps when observations change drastically over time. Simulations using these models are presented in sections 3.3.1 and 3.3.2, respectively.

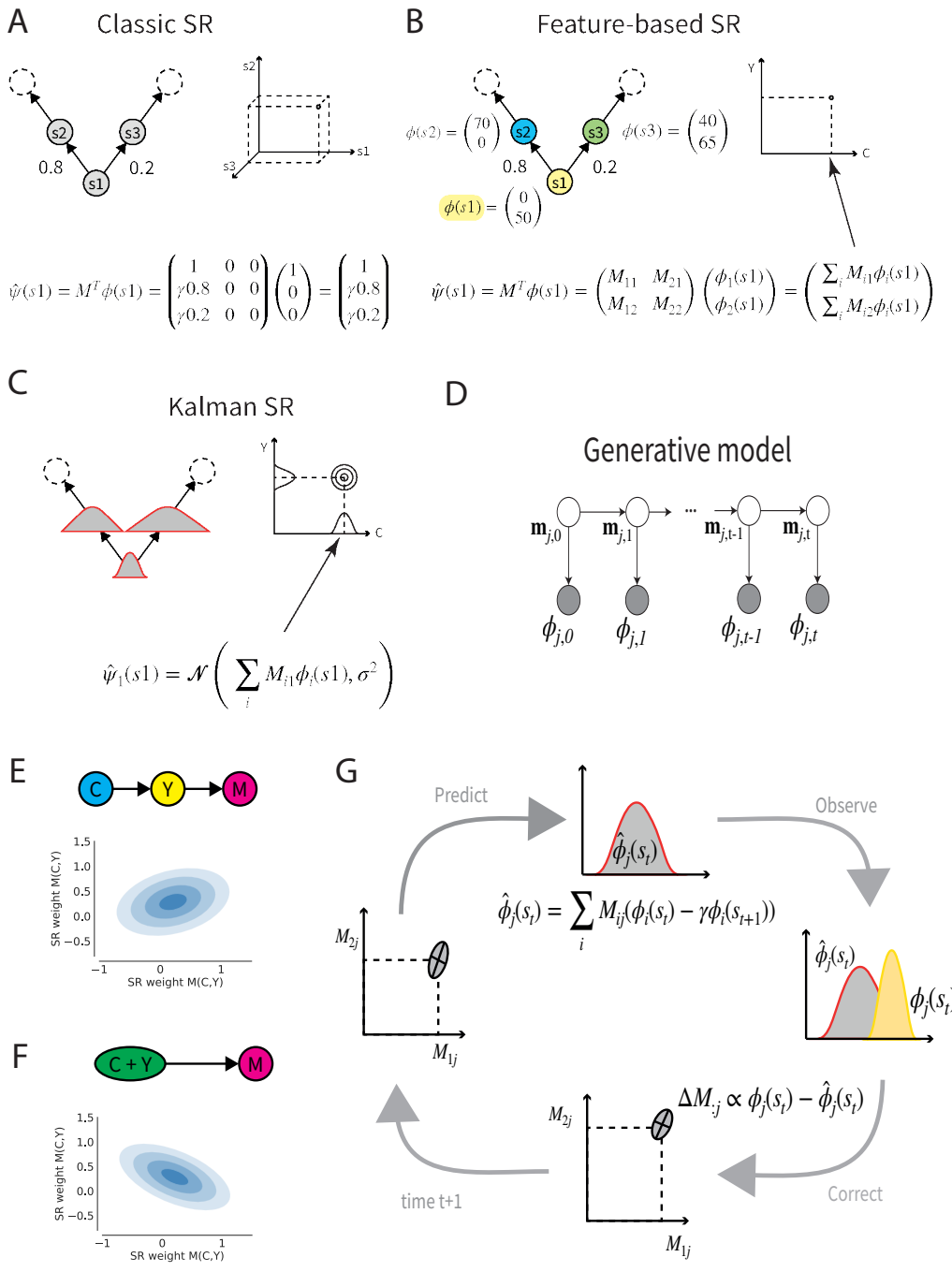

A
Classic SR
s2
s3
s1
0.8
0.2
s2
s3
s1
$\hat{\psi}(s1) = M^T\phi(s1) = \begin{pmatrix} 1 & 0 & 0 \\ \gamma 0.8 & 0 & 0 \\ \gamma 0.2 & 0 & 0 \end{pmatrix}\begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ \gamma 0.8 \\ \gamma 0.2 \end{pmatrix}$
B
Feature-based SR
$\phi(s2) = \begin{pmatrix} 70 \\ 0 \end{pmatrix}$
$\phi(s3) = \begin{pmatrix} 40 \\ 65 \end{pmatrix}$
s2
s3
s1
0.8
0.2
$\phi(s1) = \begin{pmatrix} 0 \\ 50 \end{pmatrix}$
Y
C
$\hat{\psi}(s1) = M^T\phi(s1) = \begin{pmatrix} M_{11} & M_{21} \\ M_{12} & M_{22} \end{pmatrix}\begin{pmatrix} \phi_1(s1) \\ \phi_2(s1) \end{pmatrix} = \begin{pmatrix} \sum_i M_{i1}\phi_i(s1) \\ \sum_i M_{i2}\phi_i(s1) \end{pmatrix}$
C
Kalman SR
Y
C
$\hat{\psi}_1(s1) = \mathcal{N}\left(\sum_i M_{i1}\phi_i(s1), \sigma^2\right)$
D
Generative model
$\mathbf{m}_{j,0}$
$\mathbf{m}_{j,1}$
...
$\mathbf{m}_{j,t-1}$
$\mathbf{m}_{j,t}$
$\phi_{j,0}$
$\phi_{j,1}$
$\phi_{j,t-1}$
$\phi_{j,t}$
E
C
Y
M
SR weight M(C,Y)
SR weight M(C,Y)
F
C + Y
M
SR weight M(C,Y)
SR weight M(C,Y)
G
Predict
Observe
$\hat{\phi}_j(s_t)$
$\hat{\phi}_j(s_t) = \sum_i M_{ij}(\phi_i(s_t) - \gamma\phi_i(s_{t+1}))$
M_{2j}
M_{1j}
$\hat{\phi}_j(s_t)$
$\phi_j(s_t)$
$\Delta M_{:j} \propto \phi_j(s_t) - \hat{\phi}_j(s_t)$
M_{2j}
M_{1j}
time t+1
Correct

FIGURE 3.1

FIGURE 3.1 *(previous page)*: Model overview. **(A)** The Classic SR encodes the discounted future state occupancy. The state representation is given by a one-hot vector. Since the transitions from $s1$ to $s2$ are more probable than transitions from $s1$ to $s3$, the SR ψ for $s1$ will be closer to that of $s2$ than that of $s3$ ($\|\psi(s1) - \psi(s2)\| < \|\psi(s1) - \psi(s3)\|$). The states framed by dashed lines represent "absorbing states" at which the episode terminates. **(B)** In the case of function approximation, the instantaneous state representation is given by a feature vector $\boldsymbol{\phi}$. In this toy example, the encoded features are Cyan and Yellow dimensions of a CMYK colour space. The vector $\boldsymbol{\psi}(s)$ encodes the total future occurence of each of these features, given the current state s. **(C)** In the Kalman SR model, a distribution over feature predictions is estimated. **(D)** The Kalman SR generative model's graphical structure. **(E)** When features are presented in sequence, the model builds positive covariance between the weights (see Gershman, 2015). **(F)** When features are presented together, a negative covariance is learned (see Gershman, 2015). **(G)** The Kalman filter iterates between predicting new observations based on the current weights and updating the weight estimate based on the new observations, weighted by uncertainty.

3.2.1 Background

We define an RL environment to be a Markov Decision Process consisting of *states* s the agent can occupy, *transition probabilities* $T_\pi(s'|s)$ of moving from state s to states s' given the agent's policy $\pi(a|s)$ over actions a, and the reward available at each state, for which $R(s)$ denotes the expectation. An RL agent is tasked with finding a policy that maximises its expected discounted total future reward, or *value*:

$$V(s) = \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} \gamma^t R\left(s_t\right) | s_0 = s \right] \tag{3.1}$$

where t indexes time step and γ, where $0 \leq \gamma < 1$, is a discount factor that down-weights distal rewards. In classical model-free learning algorithms (Sutton & Barto, 1998), V is stored and updated directly using temporal difference reward prediction errors. However, such algorithms suffer from a lack of flexibility: when the mapping from states to rewards changes, model-free learners are slow to re-learn the appropriate new value function. Dayan (1993) proposed one solution to this problem, made possible by the fact that V decomposes into a dot product of the direct rewards R and a predictive representation $\boldsymbol{\psi}$:

$$V(s) = \sum_{s'} \psi_{s'}^{\text{state}}(s) R(s') \tag{3.2}$$

where $\boldsymbol{\psi}^{\text{state}}(s)$ is a vector with entries $\psi^{\text{state}}_{s'}(s)$ containing the expected discounted future occupancy of state s' along trajectories started in state s (see Figure 3.1A for a simple example):

$$\psi^{\text{state}}_{s'}(s) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t \mathbb{I}(s_t = s') | s_0 = s\right]. \tag{3.3}$$

Factorising value into an SR term and a reward term permits greater flexibility because if one term changes, it can be relearned while the other remains intact (Barreto et al., 2016; Dayan, 1993; Gershman, 2018). Since long term expectancies about state occupancy can be slow to estimate, this lends particular robustness when reward is changing and transition dynamics are not.

The SR assumes each state is represented as a onehot vector, s_i. However, this can be generalized to the cases where the onehot vectors are replaced by any arbitrary feature vector $\boldsymbol{\phi}$, and "Successor Features" (SF) $\boldsymbol{\psi}(s)$ capture the expected discounted future amount of each feature (Barreto et al., 2016). In this linear function approximation case, the reward is given by the dot product

$$R(s) = \sum_j \phi_j(s) w_j \tag{3.4}$$

where $\boldsymbol{\phi}(s)$ are the state features and $\mathbf{w}$ are weights parameterising the reward function. The decomposition of value into the reward function R and the SR $\boldsymbol{\psi}(s)$ is then written as:

$$V(s) = \sum_j \psi_j(s) \mathbf{w}_j \tag{3.5}$$

with:

$$\psi_j(s) = \mathbb{E}_\pi\left[\sum_{t=0}^{\infty} \gamma^t \phi_j(s_t) | s_0 = s\right] \tag{3.6}$$

where $\boldsymbol{\psi}$ is now defined such that each entry ψ_j gives the *expected discounted future occurrence of feature* j from starting state s, under the current policy (see Figure 3.1B for an example). In the particular case where the state space is finite and $\boldsymbol{\phi}$ is a tabular representation of the state (i.e. a one-hot vector, Figure 3.1A), this definition is equivalent to the one given in Equation 3.3. The contents of the feature vector can be arbitrary, and will in this paper be dependent on the particular task being modelled. In either case, we model this feature-based SR as $\hat{\boldsymbol{\psi}}(s) = M^T \boldsymbol{\phi}(s)$, where M is a weight matrix where each

entry M_{ij} indicates the extent to which feature i predicts feature j. Seen as a single layer of a biological neural network, each column of M can be seen as a vector of input weights of one SR-encoding neuron ψ_j (and, by analogy, the vector $\boldsymbol{\psi}(s)$ gives the population activity of SR-encoding neurons). Henceforth, to avoid cluttered notation, I denote a column by $\mathbf{m}_j = M_{:,j}$ so that $\hat{\psi}_j(s) = \mathbf{m}_j^T \boldsymbol{\phi}(s)$. Thus, each column of the SR serves as weights to predict the future occurrence of one particular feature, given the current features.

$$M = \begin{bmatrix} M_{1,1} & \cdots & \underbrace{M_{1,j}}_{\mathbf{m_j}} & \cdots & M_{1,n} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ M_{i,1} & \cdots & M_{i,j} & \cdots & M_{i,n} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ M_{n,1} & \cdots & M_{n,j} & \cdots & M_{n,n} \end{bmatrix}$$

$$\psi_j(s) = \begin{bmatrix} M_{1,j} & \cdots & M_{i,j} & \cdots & M_{n,j} \end{bmatrix} \begin{bmatrix} \boldsymbol{\phi}_1 \\ \vdots \\ \boldsymbol{\phi}_i \\ \vdots \\ \boldsymbol{\phi}_n \end{bmatrix}$$

3.2.2 Probabilistic Successor Features

The first key contribution of this work is to replace the point estimate of the expected value of ψ_j from the values of the SR weight matrix M with a probabilistic interpretation of the SR. This probabilistic SR involves explicitly representing uncertainty. Adopting a statistical view, each column $\mathbf{m}_j$ of the SR weight matrix M will be modelled as a set of random variables. In this interpretation, the animal implicitly assumes there is a true, hidden set of SR parameters $\mathbf{m}_j$, which predict each new noisy observation via a generative model. The animal's goal is to invert this generative model in order to infer a distribution over the SR weights from observations. More precisely, from a sequence of observations $\boldsymbol{\phi}_{1:t}$, the agent can infer information about the hidden SR weights using Bayes' rule:

$$p(\mathbf{m}_{j,t} | \boldsymbol{\phi}_{1:t}) \propto p(\boldsymbol{\phi}_{1:t} | \mathbf{m}_{j,t}) p(\mathbf{m}_{j,t}) \tag{3.7}$$

This idea has previously been applied to learning value functions (Geist & Pietquin, 2010a; Gershman, 2015) and readily applies to the SR. Indeed, learning the j^{th} component of the SR, $\psi_j(s)$ is equivalent to estimating the value function with the j^{th} feature $\phi_j(s)$ as the reward function. Notably, the probabilistic interpretation introduces additional terms to describe the shape of this distribution. In this case, because of Gaussian assumptions, that is a "covariance" term to describe the distribution. Thus, each column $\mathbf{m}_j$ of M is modelled as a vector-valued random variable, with dimensionality N_ϕ. The covariance matrix captures how variation in future occurrence of feature j depends on observations of all the other features jointly. As we will see, taking into account variance and covariance means that the agent can learn about features that are not currently present, as long as they show nonzero covariance with the current features.

Generative model

The Bayesian treatment of the SR requires specifying a probabilistic generative model relating the hidden SR weights to the animal's observations. This probabilistic model consists of a *prior* on each column of the SR matrix $\mathbf{m}_{j,0}$, an *evolution* equation describing how these hidden SR vectors evolve over time and an *observation* equation describing how the hidden SR relates to observations. The observation equation follows from the Bellman equation, with additive Gaussian observation noise $\nu \sim \mathcal{N}(0, \sigma_\phi^2)$:

$$\boldsymbol{\phi}_j(s_t) = \boldsymbol{\psi}_j(s_t) - \gamma\, \boldsymbol{\psi}_j(s_{t+1}) + \nu \tag{3.8}$$

$$= \mathbf{m}_j^T\, \boldsymbol{\phi}(s_t) - \gamma \mathbf{m}_j^T\, \boldsymbol{\phi}(s_{t+1}) + \nu \tag{3.9}$$

$$= \mathbf{m}_j^T\, \mathbf{h}_t + \nu \tag{3.10}$$

where I have defined $\mathbf{h}_t = \boldsymbol{\phi}(s_t) - \gamma\, \boldsymbol{\phi}(s_{t+1})$ to be the discounted temporal difference in feature observations. I assume, in other words, that each successor feature $\psi_j(s_t)$ is a linear function of the current features, which means in turn that the estimated current feature $\phi_j(s_t)$ are a linear function of the discounted time derivatives of the features, h_t. For the evolution equation, our generative model follows a Gaussian random walk allowing the weights to change incrementally over time. I also assume a Gaussian prior on the weights. Together, these form the following probabilistic generative model

(shown in Figure 3.1D):

$$\mathbf{m}_{j,0} \sim \mathcal{N}(\boldsymbol{\mu}_0, \Sigma_0) \tag{3.11}$$

$$\mathbf{m}_{j,t} \sim \mathcal{N}(\mathbf{m}_{j,t-1}, \mathrm{P}_{\boldsymbol{\xi}}) \tag{3.12}$$

$$\phi_{j,t} \sim \mathcal{N}(\mathbf{m}_{j,t}^T \mathbf{h}_t, \sigma_\phi^2) \tag{3.13}$$

where $\boldsymbol{\mu}_0$ is the prior mean, Σ_0 is the prior covariance matrix, $\mathrm{P}_{\boldsymbol{\xi}}$ is the (diagonal) transition noise covariance matrix and σ_ϕ^2 is the observation noise variance. Intuitively, this generative model states that the weights for a particular "successor feature" ψ_j tend to change slowly and independently over time, and that the future feature predictions are a noisy linear function of the current features.

Inference

Our goal is to estimate the parameters $\mathbf{m}_j$ such that they satisfy $\psi_j = \mathbf{m}_j^T \boldsymbol{\phi}$, for each successor feature j. Since the generative model described above is a linear-Gaussian dynamical system (LDS), we can perform exact inference on these SR weights by combining the Kalman filter equations with temporal difference learning. This method has been previously derived for inference on value function parameters (Geist & Pietquin, 2010a), and readily applies to estimating each column of the SR. As with value, estimating a distribution over SR weights involves adjusting the mean estimate $\mathbf{m}_j$ using a temporal difference learning rule, but now taking into account the relative covariances Σ via the Kalman gain $\boldsymbol{\kappa}$, an adaptive, feature-specific learning rate. This allows for a closed-form update of a *posterior distribution* over the weights (Figure 3.1C):

$$\mathbf{m}_{j,t+1} = \mathbf{m}_{j,t} + \boldsymbol{\kappa}_t \delta_{j,t} \tag{3.14}$$

$$\Sigma_{t+1} = \Sigma_t + \mathrm{P}_{\boldsymbol{\xi}} - \lambda_t \boldsymbol{\kappa}_t \boldsymbol{\kappa}_t^T \tag{3.15}$$

where $\delta_{j,t} = \phi_j(s_t) - \hat{\phi}_j(s_t) = \phi_j(s_t) + \gamma \hat{\psi}_j(s_{t+1}) - \hat{\psi}_j(s_t)$ is the successor prediction error for feature j, $\lambda_t = \mathbf{h}_t^T(\Sigma_t + \mathrm{P}_{\boldsymbol{\xi}})\mathbf{h}_t + \sigma_\phi^2$ is the residual variance, and k_t is the Kalman gain is given by:

$$\boldsymbol{\kappa}_t = \frac{(\Sigma_t + \mathrm{P}_{\boldsymbol{\xi}})\mathbf{h}_t}{\lambda_t}. \tag{3.16}$$

Importantly, this learning rate is feature-specific and dependent on the covariance.

The Kalman Filter's covariance-dependent learning rate gives rise to several learning phenomena that have been previously explored in the literature. For example, under high uncertainty, newly observed data has increased influenced when it is combined with the hidden state estimate, expressed by a higher learning rate. When the uncertainty about the hidden state is low compared to the uncertainty of the observation, the posterior should be close to the prior, resulting in a lower learning rate. Furthermore, when there is non-zero covariance between a set of weights, these weights are updated together. This permits nonlocal updating of parameters; that is, parameters for features not present in the current observation may be updated if these parameters have a known covariance with parameters in the current observation. In Kalman temporal differences, weights corresponding to features that are presented concurrently with each other develop negative covariance (Figure 3.1E). The intuition behind this is that the predicted features are weighted sums of the input features. If multiple input features, are present, that time step will only provide information about the sum of their corresponding weights. The mean values of the weights will share value equally, but the weights will be anticorrelated, because the more one feature was the actual predictor, the less the second feature should be associated. This explains the "backward blocking" effect, in which pairing a single stimulus A with reward after that stimulus was paired with reward in compound with another stimulus B, reduces responding to stimulus B (Dayan & Kakade, 2001). By contrast, features that are shown in sequence will result in positively covarying weights (Figure 3.1F). Previous work has used both these effects to explain various aspects of animal behaviour during model-free value learning (Gershman, 2015). Here, we will explore analogous effects in learning the SR.

In summary, the Kalman filter algorithm can be seen as a cycle of predicting the occurrence of a feature using the weights $\hat{\boldsymbol{\phi}}$, observing an actual feature $\boldsymbol{\phi}$, and using the difference to correct the weights (Figure 3.1G). Finally, since in the Kalman filter the covariance only depends on the input features and not on the outcome, the covariance matrix of the weights $\mathbf{m}_j$ is the same for each column j. Therefore, only a single covariance matrix needs to be stored.

3.2.3 Inferring SR and context simultaneously

One of the assumptions of the Kalman filter model described above is that changes in the environment are given by a linear function plus some Gaussian noise. This means that, by design, Kalman filter models do not capture situations where the hidden variable undergoes large sudden changes or jumps (Figure 3.2). In the context of reinforcement learning, such sudden changes in the environment might occur when the animal switches to a different task, environment or context, or returns to an old one. A key principle to weave in to our model is to integrate both fast and slow changes in the same model.

We can account for these jumps by positing that there is a collection of different modes or contexts, in which each context is associated with its own linear-Gaussian dynamical system (LDS, see section 1.3.1). A generative process that switches between these modes is known as a switching LDS[1]. In the context of the Kalman SR model described above, this means that the model switches between different SR maps M^k that correspond to different contexts k. Since there are infinitely many possible contexts, I use a non-parametric switching LDS (Fox et al., 2011; Gershman, 2014) which allows the number of inferred contexts to grow as more observations are made.

Generative model

In the generative process this model assumes, a context z_t is first drawn from a sticky Chinese restaurant process (sCRP) prior:

$$p(z_t = k|\mathbf{z}_{1:t-1}) = \begin{cases} \frac{N_k+\beta\delta[z_{t-1},k]}{\alpha+\beta+t-1}, & \text{if } k \text{ is previously sampled context} \\ \frac{\alpha}{\alpha+\beta+t-1}, & \text{otherwise} \end{cases} \tag{3.17}$$

where N_k is the number of observations previously assigned to context k and $\delta[x, y] = 1$ if $x = y$ and 0 otherwise. The concentration parameter α controls the propensity to create new modes and the "stickiness" parameter β determines how likely the model will stay with the current context. The CRP prior allows for a potentially infinite number of contexts, while placing high probability for a small number of contexts through its "rich get richer" dynamics,

[1]Note that another way of accounting for large as well as small jumps would be to assume heavy-tailed noise distributions. However, that kind of model does not have a "memory for contexts", the property that after a jump, we can go back to the original mode.

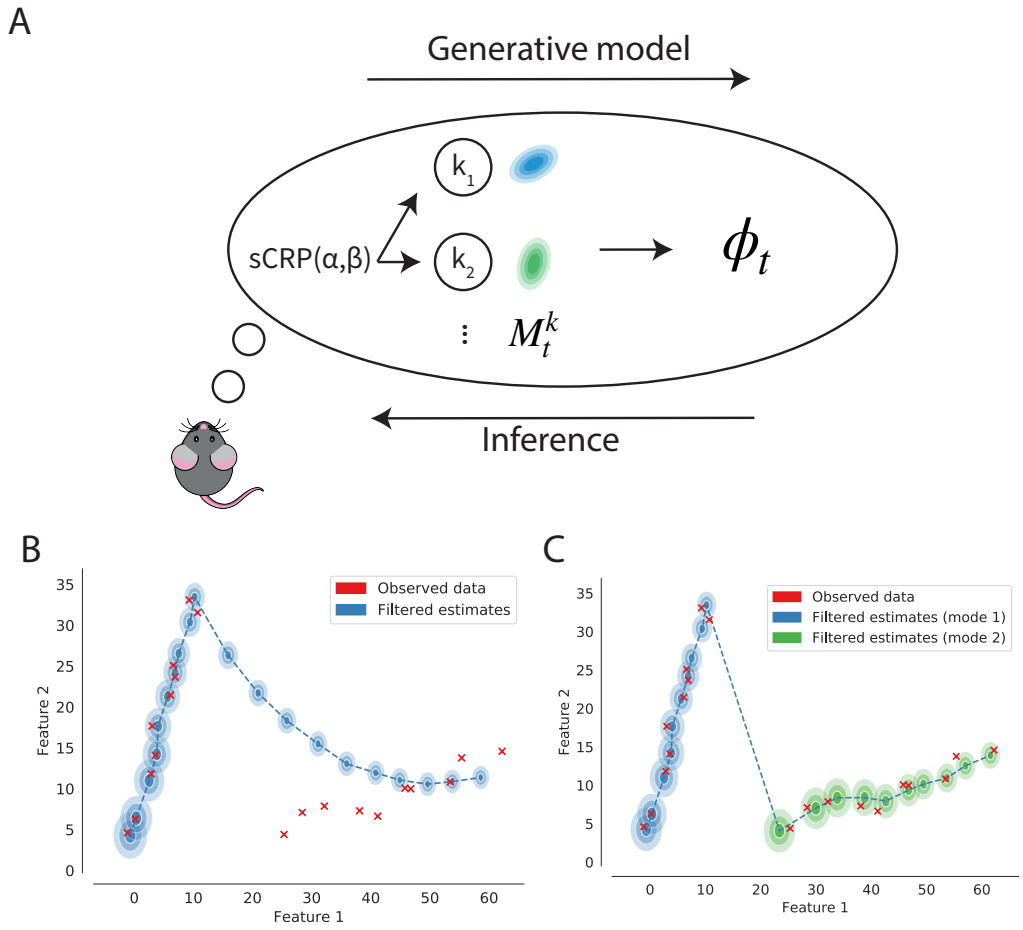


FIGURE 3.2: Switching Kalman filter model illustration. **(A)** In the infinite switching Kalman filter generative model, a context k is drawn from a sticky Chinese Restaurant Process (sCRP) prior. The currently active context selects one of infinitely many possible linear-Gaussian models to pass through to the observations, ϕ_t. Given this generative model, the animal's goal is to infer both the SR parameters and the discrete context variable. **(B)** A single Kalman filter does not account for large jumps in the hidden variable that is tracked (ellipses show the posterior distribution at each time step). **(C)** A switching Kalman filter deals with large prediction errors by assigning them to a new mode or context (posterior distributions are colour-coded with the inferred context).

whereby the probability of choosing a mode is proportional to the number of observations already assigned to that mode. The sticky CRP additionally has a bias in favor of continuing the most recent context.

Recall that each context corresponds to its own LDS. Therefore, after choosing a context, the generative model proceeds by evolving the state variable for each previously active mode k according to the evolution equation of the LDS: $\mathbf{m}^k_{j,t} \sim \mathcal{N}(\mathbf{m}^k_{j,t-1}, Q)$. If z_t is a new context, a new SR is first drawn with columns $\mathbf{m}^{z_t}_{j,0}$ drawn from a Gaussian prior: $\mathbf{m}^{z_t}_{j,0} \sim \mathcal{N}(\boldsymbol{\mu}_0, \sigma^2_\mu I)$. Finally, a sensory observation $\boldsymbol{\phi}_t$ is emitted from the currently active context z_t using the observation equation: $\boldsymbol{\phi}_{j,t} \sim \mathcal{N}((\mathbf{m}^{z_t}_{j,t})^T \mathbf{h}_t, \sigma^2_\phi)$.

In summary, the hidden SR diffuses gradually until it jumps to either a previously activated context, or to a new one. This generative model corresponds to that used in Gershman et al. (2014) to model memory updating, with the difference that here the continuous hidden state is the SR.

Inference

Given the generative model described above, the agent's goal is to infer both the context assignments and the SR parameters. When the context is given, the conditional distribution over each SR column for that mode is Gaussian. When there is uncertainty about the contexts, this computation requires marginalising over all possible context histories $\mathbf{z}_{1:t}$:

$$p(\mathbf{m}^k_{j,t} | \, \boldsymbol{\phi}_{1:t+1}) = \sum_{\mathbf{z}_{1:t}} p(\mathbf{m}^k_{j,t} | \, \boldsymbol{\phi}_{1:t+1}, \mathbf{z_{1:t}}) p(\mathbf{z}_{1:t}) \tag{3.18}$$

This summation over mode histories grows exponentially with time: if there are K modes, the posterior at time t will be a mixture of K^t Gaussians, one for every possible history $z_1, ...z_t$. This exponential increase renders inference intractable, so a reasonable approximation must be found (Murphy, 1998). Several approximation schemes have been developed, such as a Gaussian sum approximation (Barber, 2012), particle filtering (Fearnhead & Clifford, 2003; Murphy, 1998), Markov Chain Monte Carlo methods (Fox et al., 2011) and a "local" maximum a posteriori (MAP) approximation that keeps only a single high probability path in the tree of histories (Gershman et al., 2014). Here, we use the particle filter, which is a sequential Monte Carlo method that approximates the posterior distribution at each time step using a set of

weighted particles, which are updated sequentially as new observations are obtained.

Conditional on knowing the actual contexts $\mathbf{z}_{1:t}$, the different state space models k are separate linear-Gaussian systems, for which exact inference can be performed using the Kalman filter equations as described in the previous section. The key idea of the particle filter approach to this problem is to sample a mode assignment z_t for each particle $\{l\}$ from the sCRP prior (equation 3.17), and to use the Kalman filter equations to estimate each $\mathbf{m}_{j,t}^k$, with Kalman gain:

$$\mathbf{k_t} = \begin{cases} \frac{(\Sigma_t + Q)\mathbf{h}_t}{\lambda_t} & \text{if } k \text{ is a previously sampled context} \\ \mathbf{0} & \text{otherwise} \end{cases} \tag{3.19}$$

Thus, each particle will represent a possible history of context assignments, and the posterior over contexts is obtained by averaging. The particles' weights are updated according to $w_t^{\{l\}} \propto p(\boldsymbol{\phi}_t | z_t = k, \mathbf{z}_{1:t-1}^{\{l\}}, \boldsymbol{\phi}_{1:t-1})$, which corresponds to the one-step ahead predictive density, or the likelihood of the next observation, given by:

$$p(\phi_{j,t} | \boldsymbol{\phi}_{1:t-1}, z_t = k) = \begin{cases} \mathcal{N}(\phi_{j,t}; \mathbf{h}_t^T \mathbf{m}_{j,t}^k, \lambda_t) & \text{if } k \text{ previously sampled} \\ \mathcal{N}(\phi_{j,t}; \mathbf{h}_t^T \boldsymbol{\mu}_0, \mathbf{h}_t^T \Sigma_0 \mathbf{h}_t + \sigma_\phi^2) & \text{otherwise.} \end{cases} \tag{3.20}$$

A key characteristic of this model is that large prediction errors will likely lead to the inference of a new context. This can be seen in equation 3.20: the log-likelihood of any existing context will be inversely proportional to the magnitude of the SR prediction error for that context, $||\boldsymbol{\phi}_t - M^T \mathbf{h}||^2$. This means that any context for which the cached SR produces a large prediction error is unlikely to be chosen. If none of the existing contexts have a high likelihood, given a broad prior, the model will be likely to infer a new mode. Furthermore, since the variance of a mode grows with the amount of time since its last occurrence, older modes will be more tolerant to prediction errors. The intuitive explanation for this is that if the animal has not seen a context for a long time, its certainty about the details of the events will have deteriorated.

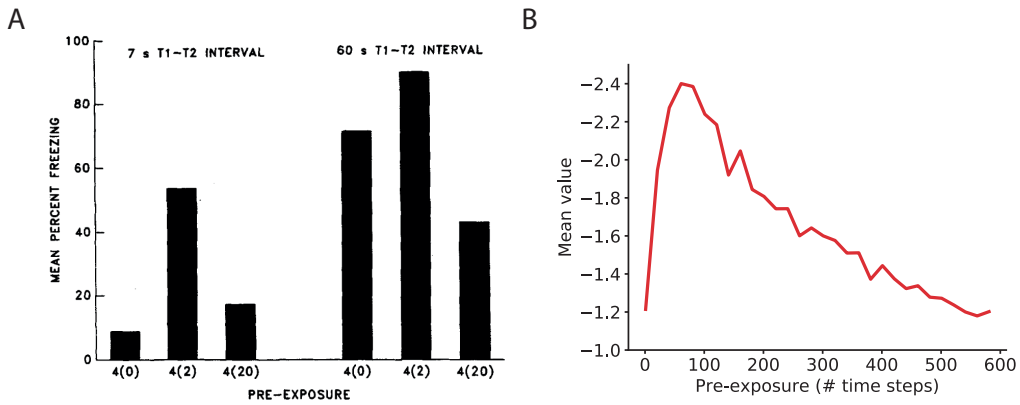


FIGURE 3.3: A brief amount of pre-exposure to an environment facilitates subsequent learning but extended pre-exposure impairs learning. **(A)** Behavioural data from Kiernan and Westbrook (1993). Mean percentage freezing scores in the shocked environment E1 for the groups receiving 0, 2 or 20 pre-exposures to the environment and exposed to a T1-T2 interval of either 7 sec or 60 sec. **(B)** Under the Kalman SR model interpretation, exploring the environment during pre-exposure allows a predictive representation to be learned. Since value is computed by multiplying the SR by the reward function, this means that longer pre-exposure initially facilitates learning the negative value in the environment. Prolonged pre-exposure, however, causes a decrease in Kalman gain, inhibiting further learning. Simulation results showing the mean value estimated by the model.

3.3 Results

The Bayesian view of the SR outlined above allows us to reconcile and reinterpret some results in the animal learning literature, which we will describe in this section. In the first part of this section, we will discuss results that follow from the single Kalman filter interpretation of the SR described in Section 3.2.2. In the second part, we will discuss experimental predictions relating to the switching Kalman filter model described in Section 3.2.3.

3.3.1 Kalman SR simulations

Facilitation and latent inhibition in contextual fear conditioning

Contextual fear conditioning is acquired faster if animals explore the environment for several minutes before a first shock, a finding known as the 'context pre-exposure facilitation effect' (Fanselow, 2010). As pointed out by Stachenfeld et al. (2017), a predictive model such as the SR can account for this effect: during pre-exposure, the animal explores and learns a predictive

representation of the context such that subsequent value learning is rapidly propagated across the environment. However, context pre-exposure facilitation stands in apparent contrast to 'latent inhibition', which refers to the finding that pre-exposure to a conditioned stimulus (CS) typically *impairs* the acquisition of a conditioned response. This latent inhibition effect has been taken as evidence for the assertion that animals are Bayesian learners: as the pre-exposed cue is presented repeatedly, the animal's uncertainty about the expected reward associated with that cue decreases, resulting in slower subsequent learning (Gershman, 2015).

How can pre-exposure facilitation and inhibition be reconciled under a single model? According to the Kalman SR model described above, both facilitation (driven by the SR) and inhibition (driven by the Kalman filter) are expected to occur when animals are pre-exposed to an environment. The model predicts that, as the animal explores the environment (a tabular grid world), there should be facilitation early on because of the SR. However, this should be followed by inhibition after extensive training because reduced variance in the estimates results in a decrease in Kalman gain (Equation 3.16), as shown in Figure 3.3B. Consistent with this, Kiernan and Westbrook (1993) showed that the amount of freezing after fear conditioning depends non-monotonically on the amount of pre-exposure to the to-be-shocked environment (Figure 3.3A).

Transition revaluation

A key prediction of standard temporal difference SR learning is that "reward revaluation" (changes in the reward function) should be easier to acquire than "transition revaluation" (changes in the transition dynamics), since the latter requires propagating state occupancy predictions to distal states. Updating the SR locally after a transition, such as is the case in temporal difference learning, will not affect the SR in non-local states. Momennejad et al. (2017) tested whether or not this is the case in human learning. In the first phase of their experiment, participants learned two different sequences of states terminating in different reward amounts: 2→4→6→\$1 and 1→3→5→\$10 (see Figure 3.4A). In the next stage, half of the participants

were exposed to the transition revaluation condition, observing novel transitions: 4→**5**→\$10 and 3→**6**→\$1. The other half experienced "reward revaluation" in the form of novel reward amounts 6→**\$10** and 5→**\$1**. Importantly, the novel experiences start from intermediate states such that transitions from 1 or 2 are not seen following phase 1. While participants were significantly better at reward revaluation than transition revaluation, they were capable of some transition revaluation as well (Figure 3.4B). Accordingly, the authors proposed a hybrid SR model: an SR-TD agent that is also endowed with capacity for replaying experienced transitions (Figure 3.4E). This permits updating of the SR vectors of states 1 and 2 through simulated experience.

Simulating this experiment with Kalman SR shows that the model can account for the partial transition revaluation without explicit simulated experience. Kalman SR correctly learns the SR matrix after phase 1 as well as an estimate of the covariance between features, Σ. Unlike standard temporal difference methods, Kalman TD uses the covariance matrix to estimate the Kalman gain and uses that to update the SR non-locally. This means that after seeing $3 \rightarrow 6$, it updates not just $\boldsymbol{\psi}(3)$ but also $\boldsymbol{\psi}(1)$ because these entries have historically covaried (and similarly for $\boldsymbol{\psi}(4)$ and $\boldsymbol{\psi}(2)$).

The fact that the standard temporal difference SR cannot acquire state transitions that are not directly experienced can also impair behaviour in the context of associative learning. To illustrate this, consider the experiment shown in Figure 3.5A, designed by Sharpe et al. (2017) to show that the sensitivity to reward devaluation, a hallmark of model-based learning, is dependent on dopamine transients. This experiment started with a preconditioning phase, during which associations were learned between pairs of neutral (a.k.a. nonrewarding) stimuli. A key finding was that animals' responding to the preconditioned cues was sensitive to the subsequent devaluation of the food reward (Figure 3.5B, see also Hart et al., 2020).

The key feature of this experiment is that the food reward was paired with illness in the absence of any of the neutral stimuli introduced in the preconditioning stage. Thus, unlike the animals, a standard SR agent is not sensitive to the reward devaluation (Figure 3.5B) (Gardner et al., 2018). This is because in the temporal difference SR, only stimuli that directly predict reward will change value after devaluation. In this paradigm, however, C was never directly associated with food. Hence, any algorithm that only updates associations with the currently active features will be insensitive to

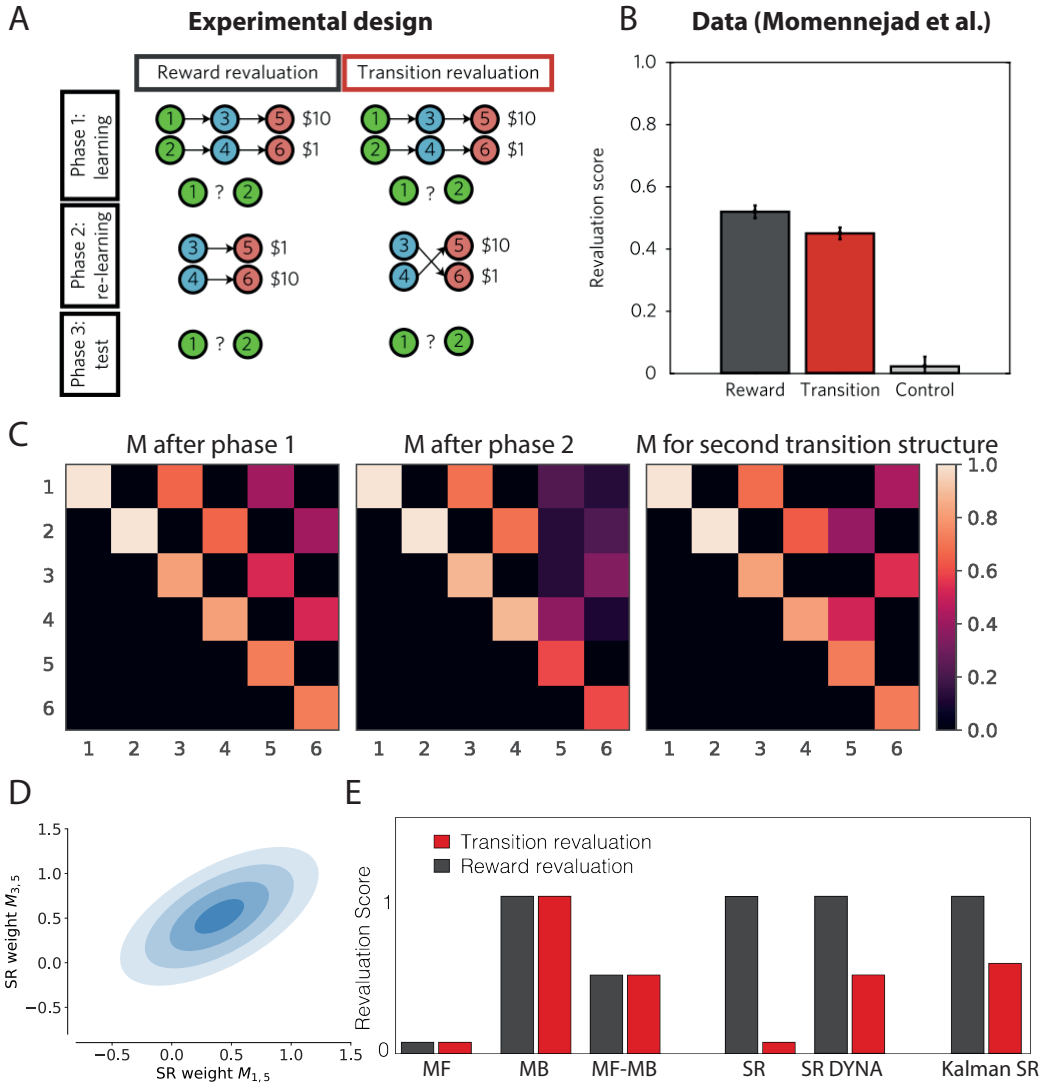
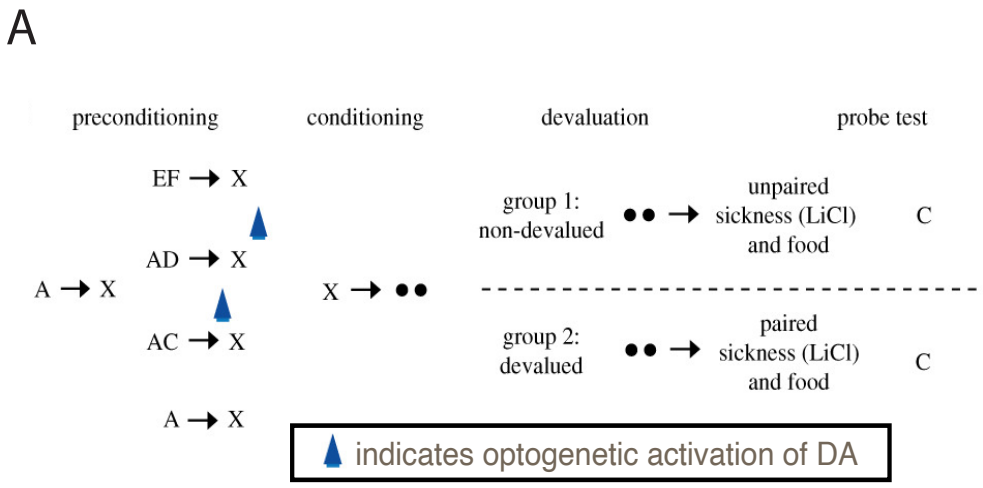


FIGURE 3.4: Revaluation experiment of Momennejad et al. (2017). **(A)** Experimental design. In an initial learning phase, participants learned sequences of states, associated with high ($10) or low ($1) rewards. During a second re-learning phase, either the rewards associated to the two terminal (red) states were swapped (reward revaluation) or the transitions from the middle (blue) to the terminal states were swapped (transition revaluation). **(B)** Human participants' revaluation scores (Momennejad et al., 2017). **(C)** Kalman SR mean estimates of weight matrix M after phase 1, phase 2, and in the case where the transition structure is as in phase 2 from the start. **(D)** The joint distribution over weights $M_{1,5}$ and $M_{3,5}$ shows a positive covariance induced by the first phase of learning, which explains the revaluation from state 1 to 5. **(E)** Predicted revaluation scores (change in rating $(V(1) - V(2))$ between phase 1 and 3 for different algorithms.



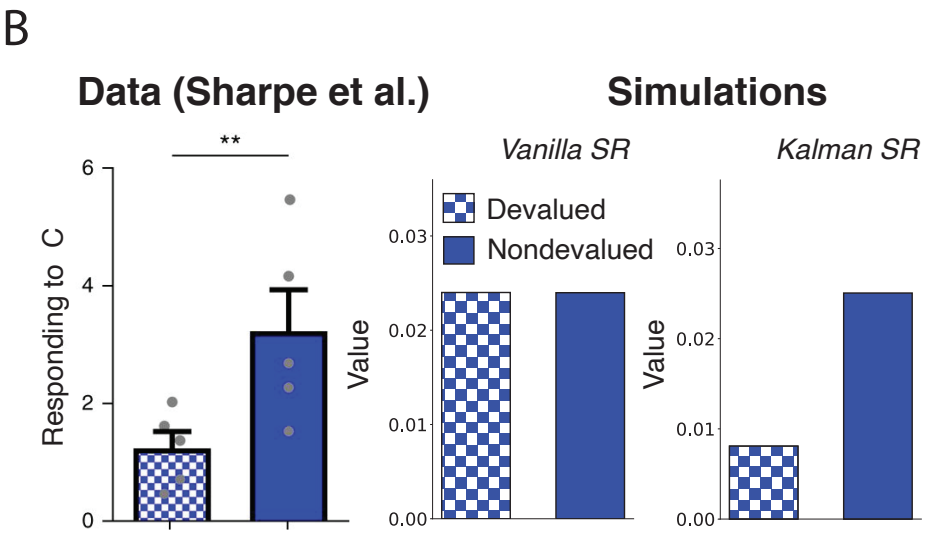


FIGURE 3.5: Devaluation experiment by Sharpe et al. (2017). **(A)** Experimental design. **(B)** Data and simulation results show that, like animals and unlike TD-SR, Kalman SR shows sensitivity to devaluation in this paradigm.

devaluation.

I simulated this task using Kalman SR and found that, like the animals in Sharpe et al. (2017), Kalman SR was sensitive to the reward devaluation paradigm. Like in the transition revaluation task, this is because Kalman SR estimates the covariance between features, and uses this for a non-local update of successor feature weights corresponding to features that are not currently active. This permits long range temporal credit assignment without explicitly necessitating hand engineered features or simulated sequential experience. Specifically, during the pre-conditioning phase, a positive covariance between C and X is learned, which means that during conditioning, C becomes directly associated to the food. Subsequent devaluation thus directly affects C as well as X.

3.3.2 Switching model simulations

The Kalman SR model described above describes behaviour of animals learning in a single environment or context. When faced with sudden changes to the true underlying parameters that govern the SR, a single Kalman filter will be slow to adapt (Figure 3.2). To model animals' ability to learn in multiple contexts, I implemented a switching version of the Kalman SR. In this generative model, there are multiple hidden SR maps that are associated to a hidden context, as well as a discrete hidden variable z_t that indicates which of the hidden SR maps currently gives rise to the observations. The animal is then faced with the tasks to infer the SR parameters as well as the context, as described in section 3.2.3. In this section, we discuss experimental predictions relating to this switching model.

Contextual memory

Taken as a model of switching between different contexts, equation 3.20 reveals some of its predicted behaviour: when the distance between the observation and the predicted observation (successor prediction error: $\boldsymbol{\phi}_{j,t} - \mathbf{h}_t^T \mathbf{m}_{j,t}^k$) is large, the model will assign a low likelihood to that context. If the posterior probability of every currently active mode is low, the model will be likely to assign the observation to a new cluster, initiating the use of a new, separate predictive map. Furthermore, since the variance of clusters that have not been visited for a while keeps growing (as can be seen in the Kalman filter updates above), old clusters will be more 'tolerant' to prediction errors, i.e.

their likelihood will be larger, even for larger distances (Figure 3.6B). A re-exposure to the original context will reduce the variance again, restoring the sensitivity to prediction errors (Figure 3.6C).

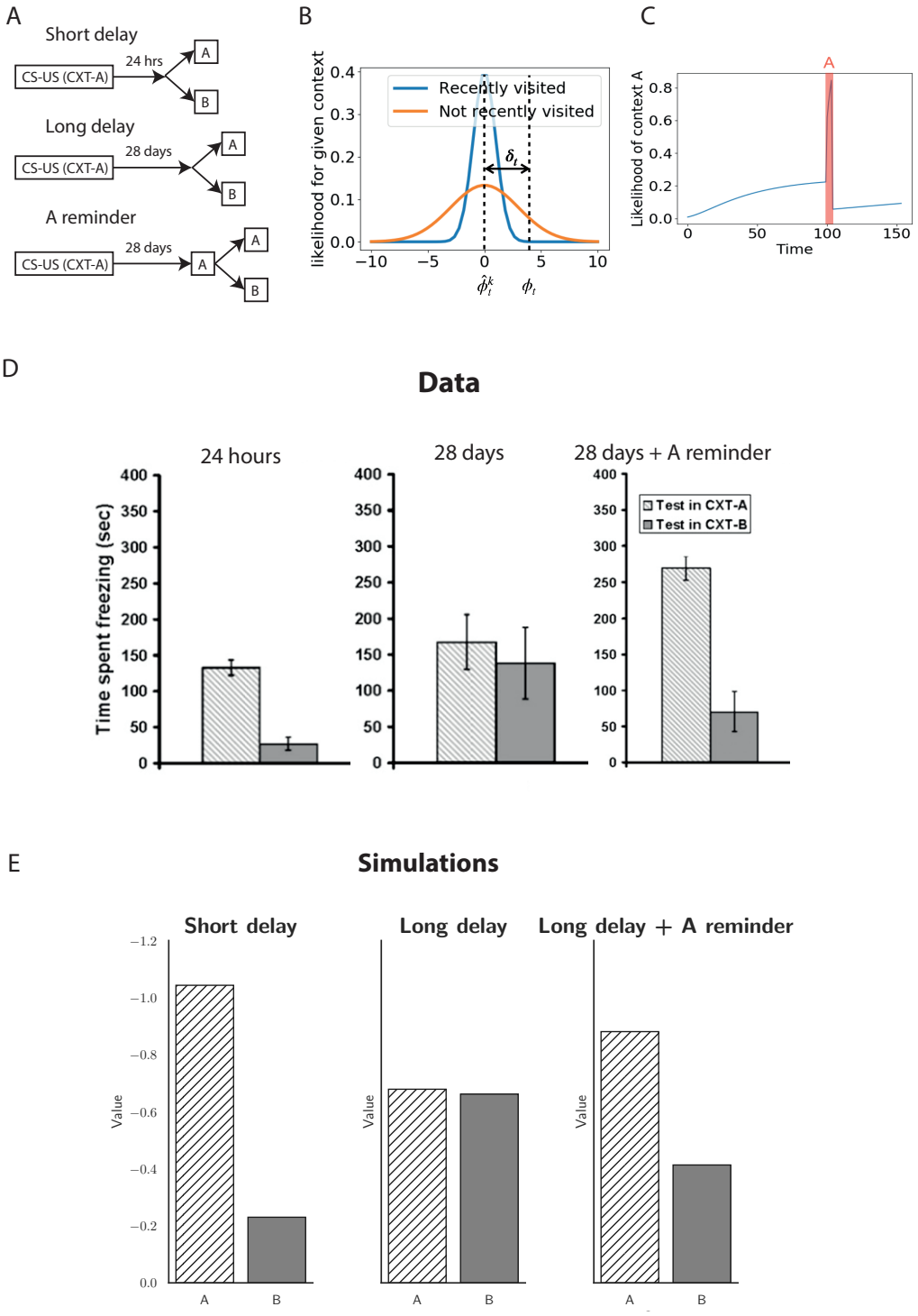

A
Short delay
24 hrs
CS-US (CXT-A)
A
B
Long delay
28 days
CS-US (CXT-A)
A reminder
28 days
CS-US (CXT-A)
B
likelihood for given context
Recently visited
Not recently visited
δ_t
$\hat{\phi}_t^k$
ϕ_t
C
Likelihood of context A
Time
D
Data
24 hours
28 days
28 days + A reminder
Time spent freezing (sec)
Test in CXT-A
Test in CXT-B
E
Simulations
Short delay
Long delay
Long delay + A reminder
Value

FIGURE 3.6

FIGURE 3.6 *(previous page)*: Contextual memory experiment by Winocur et al. (2009). **(A)** Experimental design. In the short delay condition, animals were conditioned in context A, and tested in context A and a different context B, 24 hours later. In the long delay condition, there was a 28 day delay between conditioning and testing. In the reminder condition, animals were briefly reintroduced to context A, without administering the CS or US, before testing. **(B)** In the model, each context's likelihood is a Gaussian centered on that context's predicted observation $\hat{\boldsymbol{\phi}}$. The larger the SR prediction error δ_t for that mode, the lower the likelihood. Sub-models for contexts that have not been active for a long time will have higher variance around the predicted mean and be more tolerant to prediction errors. **(C)** A reintroduction to the original context (A, red zone) should reduce that context's model's variance, and hence it should reduce the likelihood of inferring the context given a large prediction error. **(D)** Data from Winocur et al. (2009) showing the time spent freezing in response to the CS in different conditions. **(E)** Simulation results showing the state value estimate when the CS is shown in different conditions.

In summary, the model predicts that learning is highly context-specific early on but will lose context-specificity with time because of the growing uncertainty in the predicted successor features for that environment. Furthermore, because the Kalman filter's covariance updates do not depend on the outcomes, mere re-exposure to the features of context A should restore the context-specificity of the learned predictions. Evidence for this comes from Winocur et al. (2009), who trained animals on a contextual fear-conditioning task. Animals were first exposed to a CS-US pair in context A, and subsequently tested in either context A or B, after either short (24 hr) or long (28 day) delays. Consistent with our model, there is little generalization to context B after a short delay, but the level of generalization increases with the delay interval. Furthermore, when animals were briefly exposed to the training context prior to test in the second context after the long delay, the generalization decreased (Figure 3.6D). Our model recapitulates these results (Figure 3.6E).

Contextual generalization

Contextual generalization was also studied by Kiernan and Westbrook (1993). Recall from the previous section (Figure 3.3) that these authors showed a non-monotonic effect of pre-exposure duration *within* a context, whereby pre-exposure to a context first facilitates, then inhibits learning. In contrast,

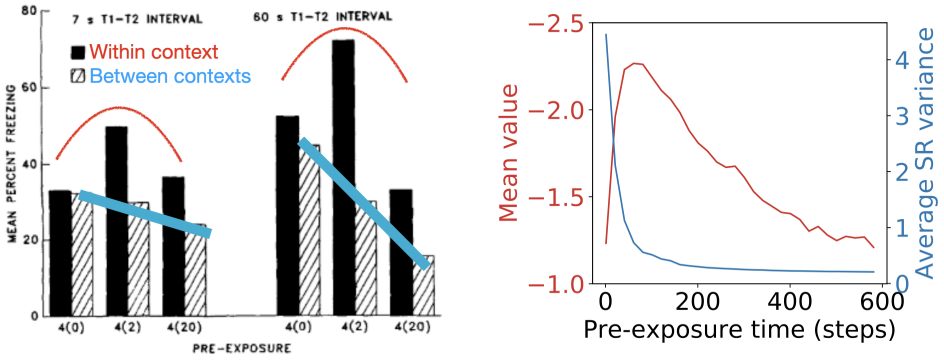


FIGURE 3.7: **(A)** Contextual discrimination data from Kiernan and Westbrook (1993, left). Black bars show conditioned freezing responses after conditioning for animals pre-exposed 0, 2 and 20 times to environment 1. White striped bars show the conditioned response to the same cue in a different environmental context. The same effect is shown for two experimental conditions in which the T1-T2 interval (i.e. the time between entering the environment and the presentation of the CS) was varied. **(B)** Model simulation results showing the negative value estimated by the model after conditioning as a function of pre-exposure time in red. In blue, the average variance of the Kalman SR model is shown.

increasing pre-exposure duration monotonically *decreases* the amount of generalization of the fear response to a second context. Under our model, increased exposure to the context results in a sharper posterior over the SR and reward weight parameters (Figure 3.7). This *reduces* contextual generalization because the likelihood of the original context 1 will now be low in context 2: because context 1's SR is represented with a high precision, even small differences in context 2 will distinguish it from context 1.

3.4 Discussion

The SR constitutes a middle ground between model-based and model-free RL algorithms by separating reward representations from cached long-run state predictions. Here I introduce a probabilistic SR model using Kalman temporal differences that supports principled handling of uncertainty about state feature predictions and inter-dependencies between these predictions. This model is extended to a switching Kalman filter that switches between different modes or contexts. I show that these models capture human and animal behaviour in settings of context preexposure, transition revaluation and contextual generalisation and memory.

3.4.1 Potential roles for replay

An attractive feature of models such as the Kalman filter that track the covariance between different weights is that this allows for retrospective revaluation. This feature has previously been used to explain learning phenomena such as backward blocking (Dayan & Yu, 2003; Gershman, 2015). Applied to SR learning, I have shown that this can extend to re-evaluating states after a change in the transition structure (Figure 3.4 and Figure 3.5). These effects have been explained in the past by positing that agents augment their SR learning with a replay buffer that can replay experienced transitions to update the SR offline (Gardner et al., 2018; Momennejad et al., 2017). In fact, these two explanations might be closely related: In the neural network implementation of the Kalman filter introduced by Dayan and Kakade (2001) and applied to Kalman TD by Gershman (2017a), the covariance matrix is approximated by a recurrent layer. Given a feature vector, the network activates other features whose weights positively covary with the weights of the currently activated features and it deactivates features whose weights negatively covary. This process can be seen as a covariance-based memory retrieval process similar to an attractor network.

A further reduction in uncertainty about the SR could be achieved using offline inference or smoothing. The uncertainty of the Kalman filter could be an interesting measure for determining which states should be replayed (Evans & Burgess, 2019). An alternative metric for the utility of replaying a specific state, suggested by Mattar and Daw (2017), is the product of a gain and need term, where the need term corresponds to the SR and the gain term quantifies the net increase in value expected after a policy change in a given state. This latter measure does not explicitly take into account uncertainty, but such a term might be approximated using the *value of information*, which can be computed from uncertainty estimates (Dearden et al., 1998).

Another interesting avenue for further research is to investigate whether we can understand replay as offline inference (i.e. smoothing) in the case of multiple maps. In switching Kalman filters, smoothing does not only sharpen the posterior of the within-mode continuous latent variable, it also makes the posterior over modes more precise (Barber, 2012). In this context, replay could serve the function of better separating out different maps from each other, or alternatively, to merge maps where this is appropriate. Guo et al. (2020) found evidence that a single coherent map is being built during

sleep. In Lever et al. (2002), maps for environments of different geometries differentiate within trials but they get more similar again between trials.

3.4.2 Effect of shock on context inference

In the proposed model, inferred context changes are induced by large magnitude state prediction errors. This context allocation mechanism allowed the agent to separate associations made in different environments, or indeed in the same environment with different future behaviour, such as in the fear conditioning experiments of Winocur et al. (2009). However, if context changes are induced by large prediction errors, one would expect the negative reward prediction error associated to the shock in fear conditioning itself to be a driver of context switching. This raises the question how this type of prediction error would affect the model presented here.

Consistent with the idea that negative reward prediction error should be a driver of context change, it has been observed that electrical shocks in fear conditioning experiments can induce hippocampal place cell remapping (Moita et al., 2004). Intriguingly, however, this remapping was much stronger when animals learned that the environment itself was predictive of the negative reinforcement (context condition). By contrast, when the shock was paired with a specific auditory cue (cue condition, as was also the case in Winocur et al., 2009), only a small subset of the place cells showed remapping. As noted by the authors, one explanation for that disparity could be that the cells that remap are more involved in acquiring the new contextual associations than non-remapping cells. If this were the case, the cue condition would result in a re-coding only of the cue itself because the cue is what requires a rapid new association. The small amount of remapping that was still observed in this condition can be explained by the animals still acquiring some association between the environment and the shock. In the context condition, the new association is made with the place itself, requiring the local place cells to remap.

What would this look like in the model? When the negative reinforcement is applied, the prediction error drives a switch of context. In this new contextual representation, the features that are active at that moment acquire the association with the shock. In the cue condition, this association will mostly be with the cue, because this most salient cue will overshadow associations with active place cells. In the context condition, when no cue is

present, this association will be with the place cells directly. On subsequent arrivals to the environment, the agent will have to infer which context applies. In the cue condition, the fearful context will have lower likelihood until the cue appears. In the context condition, any location that predicts the location where the shock arrived is consistent with the fearful context, driving a higher likelihood for the fearful condition (and thus remapping) for the whole environment. This interpretation means that if there are non-cued shocks that happen in very specific locations of a maze environment (with only low predicted occupancy from other parts of the environment), remapping should mainly be observed in that part of the environment. Regions of the environment where the predicted occupancy of the shocked state is low should not show remapping.

3.4.3 Limitations and alternative models

I make several assumptions in order to make this model tractable. Following the value estimation method described by (Geist & Pietquin, 2010a), we chose a random-walk model for describing the evolution process on the SR parameters. With this identity evolution model, all inference burden is put on the observation process. This means that Kalman TD is simply a reinterpretation of TD learning, i.e. a model-free way to estimate the SR. Given this evolution model, and assuming independent noise, we could make the assumption that the parameters for each successor feature (i.e. each column of the weight matrix) were independent such that, effectively, the Kalman SR model consists of N independent filters. Furthermore, since the evolution of the covariance matrix is independent of the prediction errors, the covariance matrix corresponding to each column was the same.

Of course, in reality there do exist dependencies between the different columns of M. For example, in the tabular case, visiting any particular state more than expected means that all other states will be visited less than expected. A cleverer evolution model could exploit these dependencies. This would break the independence assumptions and thereby increase the computational burden.

Inference in the switching Kalman filter is generally intractable, and I have chosen for a particle filter based approximation in our simulations. The

experiments I modelled here do not speak to one or another form of approximate inference, but this is an interesting avenue for further research. Interestingly, trial-by-trial fluctuations and sudden changes are well-described by particle filter algorithms with a low number of particles, suggesting that the brain may indeed use sequential Monte-Carlo sampling (Daw & Courville, 2007).

A different way to combine the SR with uncertainty was proposed by Janz et al. (2018). In their approach, the SR $\boldsymbol{\psi}$ is approximated using temporal difference methods, without taking into account uncertainty, but the reward weight vector $\mathbf{w}$ is estimated using Bayesian linear regression. The authors use these reward uncertainty estimates to balance exploitation and exploration and show that their method is effective on a set of exploration benchmarks. Since the uncertainty estimates are not used to alter the updates of the SR itself, this model would not display behaviours modelled here such as transition revaluation. To allow more naturally for non-stationary reward functions, it would be interesting to swap the Bayesian linear regression for a Kalman filter.

3.4.4 Conclusions

In conclusion, combining SR theory with uncertainty estimation can explain many learning phenomena that the SR alone cannot. This chapter demonstrates that several hitherto unconnected themes in animal learning can be unified under a single model.

Chapter 4

Discussion

4.1 Summary

4.1.1 A general model of HPC and DLS

In Chapter 2, I presented a model of hippocampal and dorsolateral striatal (DLS) contributions to learning across both spatial navigation and nonspatial decision making. In this model, the hippocampus facilitates flexible decision making by learning a successor representation (SR), while the DLS learns model-free value in an egocentric feature space. An arbitration mechanism computes the relevant state-action values as a weighted combination of the outputs of the two systems, where the weights are determined by the average recent prediction error associated to the SR and model-free systems. I showed that this model reproduces a range of behavioural findings in spatial and nonspatial decision tasks (Daw et al., 2011; Packard & McGaugh, 1996; Pearce et al., 1998), in accordance with effects of lesions to DLS and hippocampus on these tasks. Modelling place cells as driven by boundaries furthermore explained the observation that, unlike navigation guided by landmarks, navigation guided by boundaries is robust to the blocking effect (Doeller & Burgess, 2008; Doeller et al., 2008).

4.1.2 Uncertainty and the SR

In Chapter 3, I first considered a Bayesian reinterpretation of temporal difference (TD) learning known as Kalman TD (Geist & Pietquin, 2010a) and applied this to estimating successor features. This reinterpretation involved explicitly representing the uncertainty the agent has about the successor feature parameters. This model predicted a temporal pattern of context preexposure induced facilitation and inhibition observed in rodents (Kiernan &

Westbrook, 1993), as well as retrospective re-evaluation of stimulus-stimulus predictions.

In addition, I considered an infinite capacity switching Kalman filter (KF) to study context-dependent decision-making. In this version, the model could maintain and update multiple task-specific SR maps, inferring which one to use for value computation based on the current sensory observations and the successor prediction error these observations caused under each SR map. I showed that this model qualitatively captured context-dependent memory and generalisation effects observed in rodents.

4.2 Outlook

Much of the work presented in this thesis built on the theoretical proposal that hippocampal place cells predict future state occupancy in the form of an SR (Stachenfeld et al., 2017). This theory is attractive because it offers an elegant explanation for neural firing phenomena in place and grid cells (e.g. Mehta et al., 2000) and behavioural phenomena associated with hippocampal lesions (e.g. Fanselow, 2010). The appeal of this theory further lies in the fact that it provides a straightforward answer to the complicated question: *"what constitutes a good state representation?"* The SR's answer to this question is that a good representation should facilitate value learning. Unlike many previous theories of the hippocampal formation that focus on physical space (e.g. Bush et al., 2015; Hartley et al., 2000; Penny et al., 2013), this RL interpretation gives us a normative framework by which to understand both spatial and non-spatial functions of the hippocampus. However, the SR place cell model has significant weaknesses that limit its use in flexible decision making and thereby its potential to explain empirical data from animals exhibiting such flexible behaviour. Here, I will discuss how these limitations relate to the advantages, and I will discuss alternative models that aim to address these issues. In addition, I will discuss another aspect of SR theory that has been largely neglected in this thesis: that of grid cells representing a basis set for representing transition functions. Finally, I will discuss the role that uncertainty plays in this type of RL theory.

4.2.1 The SR as a model of hippocampus

Planning

The most crucial drawback of using the SR as a state representation for RL is that, by its definition, the representation is dependent on the agent's policy. This means that, even though value associated to a specific policy can be recomputed for different reward functions, most reward function changes will change the optimal policy, meaning the SR will have to be re-learned (Lehnert et al., 2017; Russek et al., 2017). The policy-dependence of the SR means that it is incapable of explaining many planning phenomena observed in biological agents (although there is also evidence suggesting that humans exhibit biases expected from an SR; Momennejad et al., 2017; Tomov et al., 2019) and it highlights a key desirable feature for map-like representations: representations must be useful for *transfer* of knowledge to novel tasks.

Several alternatives have been proposed to remedy this issue. Barreto et al. (2016) and Madarasz (2019) achieved an improvement in transfer learning by learning multiple SRs corresponding to different policies and associating each novel task with (a combination of) previously learned representations. Universal successor feature approximators (USFA) improve on this further by modelling SRs as a function $\boldsymbol{\psi}(s, a; \mathbf{w})$ that takes the reward weights $\mathbf{w}$ as a description of the current task as input (Borsa et al., 2018). This eliminates the need to store multiple SRs for each policy. However, all of these SR-based approaches are limited to transfer between tasks that share the same transition structure.

Finding representations that transfer across tasks with different transition functions provides an additional challenge. Lehnert et al. (2020) suggested learning *reward-predictive state representations*. These are state representations $\boldsymbol{\phi}$ that allow the agent to optimally predict which reward will be observed after taking a sequence of actions starting at a specific state. This is opposed to *reward-maximising* state representations, that merely allow the agent to optimally maximise reward in a specific environment. Unlike the SR, reward-predictive state abstractions are not tied to a particular policy or transition function, allowing them to be more useful for transfer between tasks with different reward and transition functions. In fact, there is an interesting relationship between these abstractions and the SR: one way of learning reward-predictive representations, suggested by the authors, is to construct a state abstraction so as to most accurately predict the SR at each state-action pair.

One could speculate that, in the brain, SR-like representations in the hippocampus could serve to train more abstract reward-predictive representations elsewhere.

Policy-independent representations with a close link to the SR also feature in an efficient approach to RL based on substituting the Bellman optimality equation with a *linear* expression (Piray & Daw, 2019a; Todorov, 2009). This linear RL approach redefines the value function to not only include instantaneous rewards from the environment but also a penalty for diverging from a default policy π^d given by the Kullback–Leibler divergence $\mathrm{KL}(\pi||\pi^d)$ between the chosen and the default policy. Substituting these penalised rewards into the Bellman equation (1.7) results in a linear, non-recursive expression for the optimal value function:

$$\exp(\mathbf{v}^*) = DP\exp(\mathbf{r}) \tag{4.1}$$

where $\mathbf{v}^*$ is the vector of optimal values for each state, $\mathbf{r}$ is a vector of rewards at a set of terminal (goal) states and P encodes the one-step probability of reaching each goal state from each other state. Of particular interest is the *default representation* D, which measures how close all non-terminal states are to each other under the default policy. This is similar to the SR, but *independent of the current policy*: D is defined with respect to the default policy and can be used to compute optimal values irrespective of what the current goal state is.

In the context of navigation, the limitation of SRs in depending on prior experience and policy is illustrated by the issue of taking a novel shortcut. If the animal has not made a particular transition before, the SR is unable to provide that solution. There is some controversy about the question whether animals can do this, and if so which animals. Tolman's (1948) result on the sunburst maze (Figure 1.1), which demonstrated rodents' ability to take shortcuts (36% of animals chose the optimal novel path), has proven difficult to replicate, with some studies corroborating Tolman's findings (Harley, 1979) while other studies found rats unable to take novel shortcuts (e.g. Gentry et al., 1948; Grieves & Dudchenko, 2013). This has lead some authors (e.g. Bennett, 1996; Grieves & Dudchenko, 2013) to suggest that perhaps the purported use of a cognitive map by rodents in studies like Tolman's was in fact due to conspicuous landmarks or beacons indicating the goal location being visible to the animals. Bats, on the other hand, have convincingly

been shown to take novel shortcuts without any beacon or landmark (Fenton, 2020), while humans show large individual differences (Hartley et al., 2003; Pazzaglia et al., 2018). How the brain computes the sense of direction required for taking such shortcuts is unknown. Interestingly, Yu et al. (2020) recently showed that a "sense of direction" (defined as the angle of the transitions that maximise the future probability of reaching a target state given an initial state) can be computed using the eigendecomposition of an action-conditional transition matrix or SR, so long as the effects of these actions are translation invariant (as is the case for Euclidean space). This suggests that shortcut finding could be achieved from the eigenvectors of a transition matrix, which is hypothesised to be represented by grid cells in mEC (see next section 4.2.2).

In summary, the policy-dependence of the SR limits its usefulness for flexible decision making. This has led several authors to propose alternative models that are better suited to transfer across different reward and transition functions (Lehnert et al., 2020; Piray & Daw, 2019a). On the other hand, it is exactly this feature of the model that is crucial for explaining the experience-dependent asymmetric skewing effects observed in place cells (Stachenfeld et al., 2017). An important open question, therefore, is why these skewing effects occur. Interestingly, Mehta et al. (2000) observed that place cells that were skewing on one day would reset to be symmetric again on the next day. The authors hypothesised that this might be due to reactivation of the place cells during sleep, in a different order to that experienced on the track. One could speculate that the hippocampus aims to learn a representation for some target default policy π^d, with skewing arising because experienced transitions are drawn from a different behaviour policy π. Under this hypothesis, offline reactivations (which has been observed to follow a random-walk diffusive pattern; Stella et al., 2019) would serve as an off-policy correction to this mismatch in distributions, similarly to importance sampling techniques that have been considered in RL (Precup et al., 2000). An interesting experiment would be to disrupt replay after skewing occurs, to observe whether the learned skewness in place cells remains after sleep.

Memory

Besides planning, another important hypothesised function of the hippocampus is episodic memory (Burgess & Hartley, 2002). An exciting open question, therefore, is how the memory and planning capacities of the hippocampus relate. Some authors have suggested the SR could play a role in both functions (Gershman, 2017b): when the SR is learned using TD methods with eligibility traces (see Chapter 2), it can be shown to be in some cases equivalent to the temporal context model (TCM) of episodic memory (Gershman et al., 2012). Under this model, the eligibility traces correspond to exponentially decaying memory traces while the SR stores associations between items. The model successfully predicts the *context repetition effect* (Smith et al., 2013): the finding that repeating the temporal context of a particular item will strengthen memory for that item (even when the item itself is not repeated).

In contrast, *episodic RL* models (Botvinick et al., 2019; Gershman & Daw, 2017; Lengyel & Dayan, 2007) explicitly store in memory individual experiences, along with their associated returns. When a familiar state is then encountered, the set of trajectories that have followed each action in that state (or similar states) are retrieved, and the value of each action can be computed by averaging. The key advantage of this approach is that it works well in the extremely low data limit, when even model-based approaches can struggle (Lengyel & Dayan, 2007). Note that this is a very different sort of process than storing a transition matrix or SR, which is sometimes referred to as *semantic memory* (Gershman & Daw, 2017). In those cases, predictions about the value of states or actions are built up as an average over many episodes. In episodic RL, on the other hand, memory for individual episodes can significantly drive behaviour. This is often hard to distinguish from non-episodic RL strategies because tasks in behavioural neuroscience tend to involve many repeated trials. In many such experiments, episodic RL might make the same behavioural predictions as model-based RL.

When applied to large or continuous state spaces, episodic RL requires a generalisation mechanism to decide which memories apply to which states. This can be achieved by allowing value estimates to be smooth interpolations of remembered episodes, where how much each remembered episode counts towards the value estimate is weighted by a kernel function measuring the similarity between the current and retrieved state. As pointed out by Gershman and Daw (2017), the SR would be a useful choice of kernel,

since it groups those states as similar that predict similar futures (and that, therefore, will be of similar value). This raises the additional possibility that SR-like representations in the hippocampus exist not (just) to approximate value directly, but as a kernel for assessing how previous episodes should be weighted in decision making.

4.2.2 Grid cells as a low-dimensional basis set

The linear RL theory discussed in section 4.2.1 (Piray & Daw, 2019a) also speaks to an apparent inconsistency in the SR model: while the dependence on policy is required for explaining asymmetric skewing in place cells (and indeed for correctly computing value under that policy), the grid cell simulations assume that an eigendecomposition is taken of an SR matrix for a random-walk policy (Stachenfeld et al., 2017). The default representation does not suffer from this issue, since it is defined with respect to the default policy. Thus, its eigendecomposition will resemble a stable grid cell map regardless of the policy (although conversely it is unclear how experience-dependent changes in place cell firing would be explained under this theory). The default representation *does* need updating when the transition structure is changed, for example when a barrier is introduced in a navigation domain. For those cases, the authors suggest using the Woodbury matrix inversion identity to update the representation as a sum of the original matrix plus a low-rank correction matrix that reflects the change due to the barrier. Under this model, grid cells represent a low-dimensional basis representation for a baseline map while other spatial cells such as entorhinal border cells (Solstad et al., 2008) represent basis functions corresponding to components that can be used to alter the map.

The idea of entorhinal cells as a basis describing environmental structure features in several other models. For example, Baram et al. (2018) showed that the shortest path to any goal can be computed using weighted sums of eigenvector grid cells, reducing the need for explicit planning by repeated multiplication of the transition matrix (see also Corneil & Gerstner, 2015; Yu et al., 2020). Whittington et al. (2020) presented the Tolman-Eichenbaum machine (TEM), a model of generalisation in the hippocampal-entorhinal system. In TEM, the agent learns to predict the next observations using a generative model in which latent variables are separated between variables that

are grounded in sensory experience and codes of abstract locations that generalise across different maps. These more abstract representations encode "structural knowledge" such as the understanding that, in 2D physical space, a sequence of east-south-west-north movements will bring you back to the same place as you started. After training, abstract location representations in TEM resemble grid cells, border cells and object vector cells, depending on the agent's behaviour. Importantly, these representations generalise across different environments, consistent with data from entorhinal representations. In contrast, the units coding for conjunctions between locations and sensory experiences resemble place cells and are different in each environment, consistent with place cell remapping observed in hippocampus.

The overarching picture that emerges from the SR and the related theories described above is of a hippocampal formation in which the hippocampus proper encodes a precise representation of the current environment while the entorhinal cortex encodes a low-dimensional basis set useful for finding subgoals (Stachenfeld et al., 2017), planning (Baram et al., 2018; Yu et al., 2020) or transferring structural knowledge (Whittington et al., 2020). To test these theories further, more experiments are needed, which will be carried out by our laboratory and others. For example, a strong prediction of the basic eigenvector grid cell model is that grid cell firing reflects topological structure rather than euclidean distance. This means, for example, that there is only one connection difference between a linear track structure and a loop. Accordingly, when a "broken loop" environment is joined together, half of the eigenvector grid cells should change their firing. More generally, the basic model where grid cells are the eigenvectors of the transition matrix learned for one particular environment (e.g. Stachenfeld et al., 2017), predicts that environmental topology, rather than Euclidean distance, should directly affect grid cell firing. In contrast, Yu et al. (2020) assume a transition matrix reflecting translation-invariant experience in all previous environments (i.e. ignoring any local barriers), which would resemble Euclidean distance in spatial tasks. TEM (Whittington et al., 2020) assumes that grid cells abstract the structure that is common across tasks, which might also resemble the basic rules of space rather than the connectivity of a specific environment.

In conclusion, while the SR's policy-dependence is problematic for its use as a cognitive map, the model has sparked a wealth of research into related biologically realistic planning methods and into the intruiging possibility that cognitive maps in the brain can be built by composing basis functions

represented by grid cells. Further experimental research is needed to adjudicate between the different hypotheses outlined above.

4.2.3 The estimation and use of uncertainty in RL

Another main theme in this thesis concerned the estimation and use of uncertainty during decision making. In Chapter 2, I considered the question of arbitrating between hippocampal and striatal controllers underlying allocentric and egocentric navigation strategies. Experimental results from animals shifting between these strategies were well described by modelling this arbitration based on an average of (unsigned) recent prediction errors as an uncertainty estimate. In Chapter 3, on the other hand, uncertainty was tracked by estimating a distribution over successor features using a KF. These two estimates correspond to different types of uncertainty: in the KF, it is assumed that the true associations fluctuate at a constant rate, with the uncertainty determined by the variances of the (often known) evolution and observation processes. The error-based uncertainty estimate, on the other hand, measures the speed at which these associations themselves might change. This is often referred to as the *volatility* of the environment. Volatility estimates have been a key feature in classical learning theories such as the Pearce-Hall model (Hall, 1991), which posits that the learning rate should increase under higher volatility (see Roesch et al., 2012, for a review of evidence for this model in brain and behaviour).

Multiple models have been proposed that combine the Bayesian approach of KFs with volatility estimation. One approach is to not assume that the evolution noise covariance is known, but rather that it is learned and updated online so that the variance (and learning rate) scales with the volatility of the environment. For example, a stochastic gradient descent update can be derived by differentiating the KF's log-likelihood with respect to the diffusion variance, resulting in an update proportional to the squared prediction error (Gershman, 2017a). Another class of theories explicitly builds volatility into the generative model. In these models, a higher-level variable is added which controls the speed of the hidden variable's random walk diffusion, and the animal is assumed to approximate inference both on the original and the higher-level hidden variable (Behrens et al., 2007; Dayan & Yu, 2003; Mathys et al., 2011; Piray & Daw, 2019b). Either of these approaches could

be combined with the model presented in Chapter 3 to learn SFs in volatile environments.

In this thesis, the focus has been on modelling *uncertainty about parameters* governing the value function or SR, respectively. Another important source of uncertainty in RL is *uncertainty about which state* the agent is in (e.g. Daw et al., 2006). When there is such uncertainty, this can be modelled as a *partially observable* Markov Decision Process (POMDP; Kaelbling et al., 1998), which comprises, in addition to the parameters of an MDP, an observation function $O(s, x)$ specifying the probability of seeing sensory data x in state s. Such an environment is not necessarily Markovian in the sensory data, but it is Markovian in the posterior distribution over states or *belief state* $b(s) = P(s|x)$. These belief states can be computed from the sensory observations using Bayes' rule:

$$b(s) \propto O(s, x)P(s) \tag{4.2}$$

where $P(s)$ is a prior over states. TD learning methods can be applied directly over representations of belief states to learn value or an SR. Vertes and Sahani (2019) introduced an alternative, neurally plausible method for learning SRs in POMDPs based on distributed distributional population codes.

Another interesting question is how uncertainty about value should affect the agent's policy directly, specifically in the exploration-exploitation dilemma. Traditional exploration strategies, such as the softmax exploration used in Chapter 2, choose actions based on value estimates alone, thus ignoring uncertainty. A more sophisticated random exploration strategy is Thompson sampling (Thompson, 1933), in which an action value is sampled from a distribution over values $\tilde{Q}_t(s, a) \sim p(Q_t(s, a))$, and the agent acts greedily with respect to the sampled value: $a_t = \arg\max_a \tilde{Q}(s, a)$. Geist and Pietquin (2010a) explored Thompson sampling in the context of KTD. Directed exploration strategies, on the other hand, explicitly direct the agent's choices towards uncertain states. For example, the upper confidence bound algorithm adds an uncertainty bonus to action values: $a_t = \arg\max_a [Q_t(s, a) + U_t(s, a)]$, where $U_t(s, a)$ is proportional to the posterior variance (Srinivas et al., 2009). Recent evidence shows that human choice behaviour shows a mixture of random and directed exploration strategies (Gershman, 2019; Tomov et al., 2020). If uncertainty about the SR is propagated to uncertainty in the

value function, the model presented in Chapter 3 could be fruitfully combined with any of these exploration strategies.

Finally, uncertainty estimates play a key role in any hybrid model (such as the one presented in Chapter 2) that must choose between different strategies that trade off efficiency and flexibility. This is an important issue in systems that arbitrate between model-based and model-free strategies (e.g. Daw et al., 2005; Keramati et al., 2011), as well as for systems that combine model-free learning with an ability to replay past experiences in order to train the model-free system (Mattar & Daw, 2018; Sutton, 1991). In either case, metalevel decisions need to be made about when to invoke the more computationally expensive system and when to rely on simple cached values. The SR can be seen as a third system with intermediate computational cost and intermediate flexibility, which might compete for control with traditional model-free and model-based systems.

4.3 Conclusion

Predictive map theories provide a promising step towards a biologically plausible mechanism for approximating model-based planning, capture aspects of neural and behavioural data in humans and animals, and give a normative explanation for spatial and non-spatial signals in the hippocampus. However, animals live in an uncertain world. Tasks and environments are subject to change, which limits the usefulness of a single predictive map. Context-specific maps in the hippocampus, as well as trade-offs between different RL systems, are a likely cause of some of the behavioural flexibility exhibited by animals.

Bibliography

Adams, C. D., & Dickinson, A. (1981). Instrumental responding following reinforcer devaluation. *The Quarterly Journal of Experimental Psychology Section B*, *33*(2b), 109–121.

Akam, T., Costa, R., & Dayan, P. (2015). Simple Plans or Sophisticated Habits? State, Transition and Learning Interactions in the Two-Step Task. *PLoS Computational Biology*, *11*(12), 1–25.

Aronov, D., Nevers, R., & Tank, D. W. (2017). Mapping of a non-spatial dimension by the hippocampal–entorhinal circuit. *Nature*, *543*(7647), 719–722.

Baird, L. (1995). Residual algorithms: Reinforcement learning with function approximation. *Machine learning proceedings 1995* (pp. 30–37). Elsevier.

Balleine, B. W. (2005). Neural bases of food-seeking: affect, arousal and reward in corticostriatolimbic circuits. *Physiology & behavior*, *86*(5), 717–730.

Balleine, B. W., & Dickinson, A. (1998). Goal-directed instrumental action: contingency and incentive learning and their cortical substrates. *Neuropharmacology*, *37*(4-5), 407–419.

Balleine, B. W., & O'doherty, J. P. (2010). Human and rodent homologies in action control: corticostriatal determinants of goal-directed and habitual action. *Neuropsychopharmacology*, *35*(1), 48–69.

Baram, A. B., Muller, T. H., Whittington, J. C. R., & Behrens, T. E. (2018). Intuitive planning: global navigation through cognitive maps based on grid-like codes. *bioRxiv*, 421461.

Barber, D. (2012). *Bayesian Reasoning and Machine Learning*. Cambridge Univ Press.

Barreto, A., Hou, S., Borsa, D., Silver, D., & Precup, D. (2020). Fast reinforcement learning with generalized policy updates. *Proceedings of the National Academy of Sciences*, 201907370.

Barreto, A., Munos, R., Schaul, T., & Silver, D. (2016). Successor Features for Transfer in Reinforcement Learning. *arXiv*, 1–13.

Barry, C., Hayman, R., Burgess, N., & Jeffery, K. J. (2007). Experience-dependent rescaling of entorhinal grids. *Nature neuroscience*, *10*(6), 682–684.

Barry, C., Lever, C., Hayman, R., Hartley, T., Burton, S., O'Keefe, J., Jeffery, K., & Burgess, N. (2006). The boundary vector cell model of place cell firing and spatial memory. *Reviews in the Neurosciences*, *17*(1-2), 71–98.

Behrens, T. E., Muller, T. H., Whittington, J. C., Mark, S., Baram, A. B., Stachenfeld, K. L., & Kurth-Nelson, Z. (2018). What Is a Cognitive Map? Organizing Knowledge for Flexible Behavior. *Neuron*, *100*(2), 490–509.

Behrens, T. E., Woolrich, M. W., Walton, M. E., & Rushworth, M. F. (2007). Learning the value of information in an uncertain world. *Nature Neuroscience*, *10*(9), 1214–1221.

Bellmund, J. L. S., de Cothi, W., Ruiter, T. A., Nau, M., Barry, C., & Doeller, C. F. (2019). Deforming the metric of cognitive maps distorts memory. *Nature Human Behaviour*.

Bennett, A. T. (1996). Do animals have cognitive maps? *Journal of Experimental Biology*, *199*(1), 219–224.

Berke, J. D., Breck, J. T., & Eichenbaum, H. (2009). Striatal versus hippocampal representations during win-stay maze performance. *Journal of Neurophysiology*, *101*(3), 1575–1587.

Bicanski, A., & Burgess, N. (2018). A neural-level model of spatial memory and imagery. *eLife*, *7*(7052), 1–3.

Bicanski, A., & Burgess, N. (2020). Neuronal vector coding in spatial cognition. *Nature Reviews Neuroscience*, *21*(9).

Borsa, D., Barreto, A., Quan, J., Mankowitz, D., Munos, R., van Hasselt, H., Silver, D., & Schaul, T. (2018). Universal Successor Features Approximators. (2015), 1–24.

Botvinick, M., Ritter, S., Wang, J. X., Kurth-Nelson, Z., Blundell, C., & Hassabis, D. (2019). Reinforcement Learning, Fast and Slow. *Trends in Cognitive Sciences*.

Bradfield, L. A., Dezfouli, A., Van Holstein, M., Chieng, B., & Balleine, B. W. (2015). Medial Orbitofrontal Cortex Mediates Outcome Retrieval in Partially Observable Task Situations. *Neuron*, *88*(6), 1268–1280.

Bradfield, L. A., Leung, B. K., Boldt, S., Liang, S., & Balleine, B. W. (2020). Goal-directed actions transiently depend on dorsal hippocampus. *Nature Neuroscience*, *23*(10), 1194–1197.

Braun, A. A., Amos-Kroohs, R. M., Gutierrez, A., Lundgren, K. H., Serology, K. B., Skelton, M. R., Vorhees, C. V., & Williams, M. T. (2015).

Dopamine depletion in either the dorsomedial or dorsolateral striatum impairs egocentric Cincinnati water maze performance while sparing allocentric Morris water maze learning. *Neurobiology of Learning and Memory*, *118*, 55–63.

Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J., & Zaremba, W. (2016). Openai gym. *arXiv preprint arXiv:1606.01540*.

Bunsey, M., & Eichenbaum, H. (1996). Conservation of hippocampal memory function in rats and humans. *Nature*, *379*(6562), 255.

Burgess, N., & Hartley, T. (2002). Orientational and geometric determinants place and head-direction. *Advances in Neural Information Processing Systems*.

Burgess, N., Jackson, A., Hartley, T., & O'keefe, J. (2000). Predictions derived from modelling the hippocampal role in navigation. *Biological cybernetics*, *83*(3), 301–312.

Bush, D., Barry, C., Manson, D., & Burgess, N. (2015). Using Grid Cells for Navigation. *Neuron*, *87*(3), 507–520.

Calton, J. L., Stackman, R. W., Goodridge, J. P., Archey, W. B., Dudchenko, P. A., & Taube, J. S. (2003). Hippocampal place cell instability after lesions of the head direction cell network. *Journal of Neuroscience*, *23*(30), 9719–9731.

Chang, C. Y., Esber, G. R., Marrero-Garcia, Y., Yau, H.-J., Bonci, A., & Schoenbaum, G. (2016). Brief optogenetic inhibition of dopamine neurons mimics endogenous negative reward prediction errors. *Nature neuroscience*, *19*(1), 111–116.

Cheer, J. F., Aragona, B. J., Heien, M. L. A. V., Seipel, A. T., Carelli, R. M., & Wightman, R. M. (2007). Coordinated accumbal dopamine release and neural activity drive goal-directed behavior. *Neuron*, *54*(2), 237–244.

Chersi, F., & Burgess, N. (2015). The Cognitive Architecture of Spatial Navigation: Hippocampal and Striatal Contributions. *Neuron*, *88*(1), 64–77.

Chersi, F., & Burgess, N. (2016). Hippocampal and striatal involvement in cognitive tasks : a computational model. *Proceedings of the 6th International Conference on Memory, ICOM16*, 24–28.

Cohen, J. S., LaRòche, J. P., & Beharry, E. (1971). Response perseveration in the hippocampal lesioned rat. *Psychonomic Science*, *23*(3), 221–223.

Constantinescu, A. O., O'Reilly, J. X., & Behrens, T. E. J. (2016). Organizing conceptual knowledge in humans with a gridlike code. *Science*, *352*(6292), 1464–1468.

Corbit, L. H., & Balleine, B. W. (2000). The role of the hippocampus in instrumental conditioning. *Journal of Neuroscience*, *20*(11), 4233–4239.

Corbit, L. H., Muir, J. L., & Balleine, B. W. (2003). Lesions of mediodorsal thalamus and anterior thalamic nuclei produce dissociable effects on instrumental conditioning in rats. *European Journal of Neuroscience*, *18*(5), 1286–1294.

Corbit, L. H., Ostlund, S. B., & Balleine, B. W. (2002). Sensitivity to instrumental contingency degradation is mediated by the entorhinal cortex and its efferents via the dorsal hippocampus. *Journal of Neuroscience*, *22*(24), 10976–10984.

Corneil, D. S., & Gerstner, W. (2015). Attractor Network Dynamics Enable Preplay and Rapid Path Planning in Maze–like Environments. *Advances in Neural Information Processing Systems*, 1675–1683.

Cressant, A., Muller, R. U., & Poucet, B. (1997). Failure of centrally placed objects to control the firing fields of hippocampal place cells. *Journal of Neuroscience*, *17*(7), 2531–2542.

Daw, N., & Courville, A. (2007). The pigeon as a particle filter. *Advances in neural information processing systems*, (20), 369–376.

Daw, N. D., Courville, A. C., & Tourtezky, D. S. (2006). Erratum: Representation and timing in theories of the dopamine system (Neural Computation (July 2006) 7, (1637)). *Neural Computation*, *18*(10), 2582.

Daw, N. D., & Dayan, P. (2014). The algorithmic anatomy of model-based evaluation. *Phil. Trans. R. Soc. B*, *369*(1655), 20130478.

Daw, N. D., Gershman, S. J., Seymour, B., Dayan, P., & Dolan, R. J. (2011). Model-based influences on humans' choices and striatal prediction errors. *Neuron*, *69*(6), 1204–1215.

Daw, N. D., Niv, Y., & Dayan, P. (2005). Uncertainty-based competition between prefrontal and dorsolateral striatal systems for behavioral control. *Nature Neuroscience*, *8*(12), 1704–1711.

Day, J. J., Roitman, M. F., Wightman, R. M., & Carelli, R. M. (2007). Associative learning mediates dynamic shifts in dopamine signaling in the nucleus accumbens. *Nature neuroscience*, *10*(8), 1020–1028.

Dayan, P. (1993). Improving Generalisation for Temporal Difference Learning: The Successor Representation. *Neural Computation*, *5*(4), 613–624.

Dayan, P., & Berridge, K. C. (2014). Model-based and model-free Pavlovian reward learning: revaluation, revision, and revelation. *Cognitive, Affective, & Behavioral Neuroscience*, *14*(2), 473–492.

Dayan, P., & Kakade, S. (2001). Explaining away in weight space. *Advances in Neural Information Processing Systems*, *13*, 451–457.

Dayan, P., & Yu, A. (2003). Uncertainty and learning. *IETE Journal of Research*, *49*(2-3), 171–181.

De Leonibus, E., Costantini, V. J., Massaro, A., Mandolesi, G., Vanni, V., Luvisetto, S., Pavone, F., Oliverio, A., & Mele, A. (2011). Cognitive and neural determinants of response strategy in the dual-solution plus-maze task. *Learning and Memory*, *18*(4), 241–244.

Dearden, R., Friedman, N., & Russell, S. (1998). Bayesian Q-Learning. *AAAI/IAAI*.

de Cothi, W., & Barry, C. (2020). Neurobiological successor features for spatial navigation. *Hippocampus*, (June), 1–9.

Deisenroth, M. P., & Rasmussen, C. E. (2011). PILCO: A model-based and data-efficient approach to policy search. *Proceedings of the 28th International Conference on Machine Learning, ICML 2011*, 465–472.

Derdikman, D., Whitlock, J. R., Tsao, A., Fyhn, M., Hafting, T., Moser, M.-B., & Moser, E. I. (2009). Fragmentation of grid cell maps in a multicompartment environment. *Nature neuroscience*, *12*(10), 1325–1332.

Devan, B. D., McDonald, R. J., & White, N. M. (1999). Effects of medial and lateral caudate-putamen lesions on place- and cue- guided behaviors in the water maze: Relation to thigmotaxis. *Behavioural Brain Research*, *100*(1-2), 5–14.

Devan, B. D., Hong, N. S., & McDonald, R. J. (2011). Parallel associative processing in the dorsal striatum: segregation of stimulus–response and cognitive control subregions. *Neurobiology of learning and memory*, *96*(2), 95–120.

DeVito, L. M., & Eichenbaum, H. (2011). Memory for the order of events in specific sequences: contributions of the hippocampus and medial prefrontal cortex. *Journal of Neuroscience*, *31*(9), 3169–3175.

Doeller, C. F., & Burgess, N. (2008). Distinct error-correcting and incidental learning of location relative to landmarks and boundaries. *Proceedings of the National Academy of Sciences*, *105*(15), 5909–5914.

Doeller, C. F., King, J. A., & Burgess, N. (2008). Parallel striatal and hippocampal systems for landmarks and boundaries in spatial memory. *Proceedings of the National Academy of Sciences*, *105*(15), 5915–5920.

Doeller, C. F., Barry, C., & Burgess, N. (2010). Evidence for grid cells in a human memory network. *Nature*, *463*(7281), 657–661.

Doll, B. B., Duncan, K. D., Simon, D. A., Shohamy, D., & Daw, N. D. (2015). Model-based choices involve prospective neural activity. *Nature Neuroscience*, *18*(5), 767–772.

Dollé, L., Chavarriaga, R., Guillot, A., & Khamassi, M. (2018). Interactions of spatial strategies producing generalization gradient and blocking: A computational approach. *PLoS Computational Biology*, *14*(4), 1–35.

Dollé, L., Sheynikhovich, D., Girard, B., Chavarriaga, R., & Guillot, A. (2010). Path planning versus cue responding: a bio-inspired model of switching between navigation strategies. *Biological Cybernetics*, *103*(4), 299–317.

Dordek, Y., Soudry, D., Meir, R., & Derdikman, D. (2016). Extracting grid cell characteristics from place cell inputs using non-negative principal component analysis. *eLife*, *5*(MARCH2016), 1–36.

Doucet, A., Gordon, N. J., & Krishnamurthy, V. (2001). Particle filters for state estimation of jump Markov linear systems. *IEEE Transactions on signal processing*, *49*(3), 613–624.

Dring, M. J., & West, J. A. (1983). Photoperiodic control of tetrasporangium formation in the red alga Rhodochorton purpureum. *Planta*, *159*(2), 143–150.

Dupret, D., O'neill, J., Pleydell-Bouverie, B., & Csicsvari, J. (2010). The reorganization and reactivation of hippocampal maps predict spatial memory performance. *Nature neuroscience*, *13*(8), 995.

Dusek, J. A., & Eichenbaum, H. (1997). The hippocampus and memory for orderly stimulus relations. *Proceedings of the National Academy of Sciences*, *94*(13), 7109–7114.

Eichenbaum, H., Otto, T., & Cohen, N. J. (1992). The hippocampus: What does it do? *Behavioral & Neural Biology*, *57*(1), 2–36.

Engel, Y., Mannor, S., & Meir, R. (2005). Reinforcement learning with Gaussian processes. *Proceedings of the 22nd international conference on Machine learning*, 201–208.

Evans, T., & Burgess, N. (2019). Coordinated hippocampal-entorhinal replay as structural inference. *Advances in Neural Information Processing Systems*, 1729–1741.

Fanselow, M. S. (2010). From contextual fear to a dynamic view of memory systems. *Trends in cognitive sciences*, *14*(1), 7–15.

Fearnhead, P., & Clifford, P. (2003). On-line inference for hidden Markov models via particle filters. *Journal of the Royal Statistical Society. Series B: Statistical Methodology*, *65*(4), 887–899.

Fenton, M. B. (2020). Bats navigate with cognitive maps. *Science*, *369*(6500), 142 LP –142.

Foster, D. J., Morris, R. G., & Dayan, P. (2000). A model of hippocampally dependent navigation, using the temporal difference learning rule. *Hippocampus*, *10*(1), 1–16.

Foster, D. J., & Wilson, M. A. (2006). Reverse replay of behavioural sequences in hippocampal place cells during the awake state. *Nature*, *440*(7084), 680–683.

Foster, D. J., & Wilson, M. A. (2007). Hippocampal theta sequences. *Hippocampus*, *17*(11), 1093–1099.

Fox, E., Sudderth, E. B., Jordan, M. I., & Willsky, A. S. (2011). Bayesian Nonparametric Inference of Switching Dynamic Linear Models. *IEEE Transactions on Signal Processing*, *59*(4), 1569–1585.

Frank, M. J., & Claus, E. D. (2006). Anatomy of a decision: striato-orbitofrontal interactions in reinforcement learning, decision making, and reversal. *Psychological review*, *113*(2), 300.

Gardner, M. P. H., Schoenbaum, G., & Gershman, S. J. (2018). Rethinking dopamine as generalized prediction error. *Proceedings of the Royal Society B: Biological Sciences*, *285*(1891), 20181645.

Garvert, M. M., Dolan, R. J., & Behrens, T. E. (2017). A Map of Abstract Relational Knowledge in the Human Hippocampal-Entorhinal Cortex. *eLife*, *6*, 1–20.

Gaskin, S., Chai, S.-C., & White, N. M. (2005). Inactivation of the dorsal hippocampus does not affect learning during exploration of a novel environment. *Hippocampus*, *15*(8), 1085–1093.

Geerts, J., Stachenfeld, K., & Burgess, N. (2019). Probabilistic Successor Representations with Kalman Temporal Differences. *2019 Conference on Cognitive Computational Neuroscience*.

Geerts, J. P., Chersi, F., Stachenfeld, K. L., & Burgess, N. (2020). A general model of hippocampal and dorsal striatal learning and decision making. *Proceedings of the National Academy of Sciences*, 202007981.

Geist, M., & Pietquin, O. (2010a). Kalman Temporal Differences. *Journal of Artificial Intelligence Research*, *39*, 483–532.

Geist, M., & Pietquin, O. (2010b). Kalman temporal differences. *Journal of Artificial Intelligence Research*, *39*, 483–532.

Gentry, G., Brown, W. L., & Lee, H. (1948). Spatial location in the learning of a multiple-T maze. *Journal of Comparative and Physiological Psychology*, *41*(5), 312.

Gershman, S. J. (2014). The penumbra of learning: A statistical theory of synaptic tagging and capture. *Network: Computation in Neural Systems*, *25*(3), 97–115.

Gershman, S. J. (2015). A Unifying Probabilistic View of Associative Learning. *PLOS Computational Biology*, *11*(11), e1004567.

Gershman, S. J. (2017a). Dopamine, Inference, and Uncertainty. *Neural Computation*, *29*(12), 3311–3326.

Gershman, S. J. (2017b). Predicting the past, remembering the future. *Current Opinion in Behavioral Sciences*, *17*, 7–13.

Gershman, S. J. (2018). The Successor Representation: Its Computational Logic and Neural Substrates. *The Journal of Neuroscience*, *38*(33), 7193–7200.

Gershman, S. J. (2019). The rational analysis of memory, 1–13.

Gershman, S. J., Blei, D. M., & Niv, Y. (2010). Context, Learning, and Extinction. *Psychological Review*, *117*(1), 197–209.

Gershman, S. J., & Daw, N. D. (2017). Reinforcement Learning and Episodic Memory in Humans and Animals: An Integrative Framework. *Annual Review of Psychology*, *68*(1), 101–128.

Gershman, S. J., Moore, C. D., Todd, M. T., Norman, K. A., & Sederberg, P. B. (2012). The Successor Representation and Temporal Context. *Neural Computation*, *24*(6), 1553–1568.

Gershman, S. J., Radulescu, A., Norman, K. A., & Niv, Y. (2014). Statistical Computations Underlying the Dynamics of Memory Updating. *PLoS Computational Biology*, *10*(11), e1003939.

Ghahramani, Z., & Hinton, G. E. (1996). *Switching state-space models* (tech. rep.). Citeseer.

Gläscher, J., Daw, N., Dayan, P., & O'Doherty, J. P. (2010). States versus rewards: Dissociable neural prediction error signals underlying model-based and model-free reinforcement learning. *Neuron*, *66*(4), 585–595.

Grieves, R. M., Duvelle, É., & Dudchenko, P. A. (2018). A boundary vector cell model of place field repetition. *Spatial Cognition & Computation*, *00*(00), 1–40.

Grieves, R. M., & Jeffery, K. J. (2017). The representation of space in the brain. *Behavioural Processes*, *135*, 113–131.

Grieves, R. M., & Dudchenko, P. A. (2013). Cognitive maps and spatial inference in animals: Rats fail to take a novel shortcut, but can take a previously experienced one. *Learning and Motivation*, *44*(2), 81–92.

Guo, W., Zhang, J., Newman, J., & Wilson, M. (2020). Latent learning drives sleep-dependent plasticity in distinct CA1 subpopulations. *bioRxiv*.

Gustafson, N. J., & Daw, N. D. (2011). Grid cells, place cells, and geodesic generalization for spatial reinforcement learning. *PLoS Computational Biology*, *7*(10).

Hafting, T., Fyhn, M., Molden, S., Moser, M., & Moser, E. I. (2005). Microstructure of a spatial map in the entorhinal cortex. *Nature*, *436*(7052), 801–806.

Hall, G. (1991). Perceptual and associative learning. *Oxford psychology series*, *18*, 29–107.

Harley, C. W. (1979). Arm choices in a sunburst maze: Effects of hippocampectomy in the rat. *Physiology & Behavior*, *23*(2), 283–290.

Hart, E. E., Sharpe, M. J., Gardner, M. P. H., & Schoenbaum, G. (2020). Responding to preconditioned cues is devaluation sensitive and requires orbitofrontal cortex during cue-cue learning. *Elife*, *9*, e59998.

Hartley, T., Burgess, N., Lever, C., Cacucci, F., & O'Keefe, J. (2000). Modeling place fields in terms of the cortical inputs to the hippocampus. *Hippocampus*, *10*(4), 369–379.

Hartley, T., Lever, C., Burgess, N., & O'Keefe, J. (2014). Space in the brain: how the hippocampal formation supports spatial cognition. *Philosophical Transactions of the Royal Society B: Biological Sciences*, *369*(1635), 20120510.

Hartley, T., Maguire, E. A., Spiers, H. J., & Burgess, N. (2003). The Well-Worn Route and the Path Less Traveled: Distinct Neural Bases of Route Following and Wayfinding in Humans. *Neuron*, *37*(5), 877–888.

Hinman, J. R., Chapman, G. W., & Hasselmo, M. E. (2019). Neuronal representation of environmental boundaries in egocentric coordinates. *Nature communications*, *10*(1), 1–8.

Holland, P. C., & Bouton, M. E. (1999). Hippocampus and context in classical conditioning. *Current opinion in neurobiology*, *9*(2), 195–202.

Høydal, Ø. A., Skytøen, E. R., Andersson, S. O., Moser, M.-B., & Moser, E. I. (2019). Object-vector coding in the medial entorhinal cortex. *Nature*, *568*(7752), 400–404.

Janz, D., Hron, J., Mazur, P., Hofmann, K., Hernández-Lobato, J. M., & Tschiatschek, S. (2018). Successor Uncertainties: Exploration and Uncertainty in Temporal Difference Learning. (NeurIPS), 1–10.

Johnson, A., & Redish, A. D. (2007). Neural Ensembles in CA3 Transiently Encode Paths Forward of the Animal at a Decision Point. *Journal of Neuroscience*, *27*(45), 12176–12189.

Jones, M. W., & Wilson, M. A. (2005). Theta rhythms coordinate hippocampal–prefrontal interactions in a spatial memory task. *PLoS biology*, *3*(12), e402.

Julier, S. J., & Uhlmann, J. K. (2004). Correction to "Unscented Filtering and Nonlinear Estimation". *Review Literature And Arts Of The Americas*, *92*(3), 1–2.

Kaelbling, L. P., Littman, M. L., & Cassandra, A. R. (1998). Planning and acting in partially observable stochastic domains. *Artificial intelligence*, *101*(1-2), 99–134.

Kakade, S., & Dayan, P. (2002). Acquisition and extinction in autoshaping. *Psychological review*, *109*(3), 533.

Kalman, R. E. (1960). A New Approach to Linear Filtering and Prediction Problems 1. *Transactions of the ASME - Journal of Basic Engineering*, *82*(Series D), 35–45.

Kamin, L. J. (1967). Predictability, surprise, attention, and conditioning.

Keramati, M., Dezfouli, A., & Piray, P. (2011). Speed/accuracy trade-off between the habitual and the goal-directed processes. *PLoS computational biology*, *7*(5), e1002055.

Kiernan, M. J., & Westbrook, R. F. (1993). Effects of Exposure to a To-Be-Shocked Environment upon the Rat's Freezing Response: Evidence for Facilitation, Latent Inhibition, and Perceptual Learning. *The Quarterly Journal of Experimental Psychology Section B*, *46*(3b), 271–288.

Killcross, S., & Coutureau, E. (2003). Coordination of actions and habits in the medial prefrontal cortex of rats. *Cerebral Cortex*, *13*(4), 400–408.

Kimble, D. P., & BreMiller, R. (1981). Latent learning in hippocampal-lesioned rats. *Physiology & behavior*, *26*(6), 1055–1059.

Kimble, D. P., Jordan, W. P., & BreMiller, R. (1982). Further evidence for latent learning in hippocampal-lesioned rats. *Physiology & behavior*, *29*(3), 401–407.

Kosaki, Y., Pearce, J. M., & McGregor, A. (2018). The response strategy and the place strategy in a plus-maze have different sensitivities to devaluation of expected outcome. *Hippocampus*, *28*(7), 484–496.

Krupic, J., Bauza, M., Burton, S., Barry, C., & O'Keefe, J. (2015). Grid cell symmetry is shaped by environmental geometry. *Nature*, *518*(7538), 232–235.

Kruschke, J. K. (2008). Bayesian Approaches to Associative Learning: From Passive to Active Learning. *Learning and Behavior*, *36*(3), 210–226.

Kulkarni, T. D., Saeedi, A., Gautam, S., & Gershman, S. J. (2016). Deep Successor Reinforcement Learning.

Lansink, C. S., Goltstein, P. M., Lankelma, J. V., Mcnaughton, B. L., & Pennartz, C. M. A. (2009). Hippocampus Leads Ventral Striatum in Replay of Place- Reward Information. *7*(8).

Lehnert, L., Frank, M. J., & Littman, M. L. (2020). Reward Predictive Representations Generalize Across Tasks in Reinforcement Learning. *PLoS Computational Biology*, *16*(10).

Lehnert, L., & Littman, M. L. (2018). Transfer with Model Features in Reinforcement Learning.

Lehnert, L., Tellex, S., & Littman, M. L. (2017). Advantages and Limitations of using Successor Features for Transfer in Reinforcement Learning.

Lengyel, M., & Dayan, P. (2007). Hippocampal Contributions to Control: The Third Way. *Advances in Neural Information Processing Systems 2007.*, 889–896.

Lever, C., Burton, S., Jeewajee, A., O'Keefe, J., & Burgess, N. (2009). Boundary vector cells in the subiculum of the hippocampal formation. *Journal of Neuroscience*, *29*(31), 9771–9777.

Lever, C., Wills, T., Cacucci, F., Burgess, N., & O'Keefe, J. (2002). Long-term plasticity in hippocampal place-cell representation of environmental geometry. *Nature*, *416*(6876), 90–94.

Machado, M. C., Bellemare, M. G., & Bowling, M. (2017). A Laplacian Framework for Option Discovery in Reinforcement Learning. *International Conference on Machine Learning*, 2295–2304.

Madarasz, T. J. (2019). Better transfer learning with inferred successor maps. *Advances in Neural Information Processing Systems 2019.*

Mahadevan, S., & Maggioni, M. (2007). Proto-value Functions : A Laplacian Framework for Learning Representation and Control in Markov Decision Processes. *Journal of Machine Learning Research*, *8*, 2169–2231.

Mankin, E. A., Sparks, F. T., Slayyeh, B., Sutherland, R. J., Leutgeb, S., & Leutgeb, J. K. (2012). Neuronal code for extended time in the hippocampus. *Proceedings of the National Academy of Sciences*, *109*(47), 19462–19467.

Markus, E. J., Qin, Y.-L., Leonard, B., Skaggs, W. E., McNaughton, B. L., & Barnes, C. A. (1995). Interactions between location and task affect the spatial and directional firing of hippocampal neurons. *Journal of Neuroscience*, *15*(11), 7079–7094.

Mathys, C., Daunizeau, J., Friston, K., & Stephan, K. (2011). A Bayesian Foundation for Individual Learning Under Uncertainty. *Frontiers in Human Neuroscience*, *5*, 39.

Mattar, M. G., & Daw, N. D. (2017). A rational model of prioritized experience replay. *The 3rd Multidisciplinary Conference on Reinforcement Learning and Decision Making, The University of Michigan*.

Mattar, M. G., & Daw, N. D. (2018). Prioritized memory access explains planning and hippocampal replay. *Nature Neuroscience*, 225664.

McDannald, M. A., Jones, J. L., Takahashi, Y. K., & Schoenbaum, G. (2014). Learning theory: a driving force in understanding orbitofrontal function. *Neurobiology of learning and memory*, *108*, 22–27.

McDannald, M. A., Lucantonio, F., Burke, K. A., Niv, Y., & Schoenbaum, G. (2011). Ventral striatum and orbitofrontal cortex are both required for model-based, but not model-free, reinforcement learning. *Journal of Neuroscience*, *31*(7), 2700–2705.

McDannald, M. A., Takahashi, Y. K., Lopatina, N., Pietras, B. W., Jones, J. L., & Schoenbaum, G. (2012). Model-based learning and the contribution of the orbitofrontal cortex to the model-free world. *European Journal of Neuroscience*, *35*(7), 991–996.

Mcdonald, R. J., & White, N. M. (1994). Parallel Information Processing in the Water Maze : Evidence for Independent Memory Systems Involving Dorsal Striatum and Hippocampus. *Behavioral and Neural Biology*, *270*, 260–270.

Mehta, M. R., Quirk, M. C., & Wilson, M. A. (2000). Experience-Dependent Asymmetric Shape of Hippocampal Receptive Fields. *25*, 707–715.

Miller, K. J., Botvinick, M. M., & Brody, C. D. (2017). Dorsal hippocampus contributes to model-based planning. *Nature Neuroscience*, *20*(9), 1269–1276.

Miyoshi, E., Wietzikoski, E. C., Bortolanza, M., Boschen, S. L., Canteras, N. S., Izquierdo, I., & Da Cunha, C. (2012). Both the dorsal hippocampus and

the dorsolateral striatum are needed for rat navigation in the Morris water maze. *Behavioural Brain Research*, *226*(1), 171–178.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., et al. (2015). Human-level control through deep reinforcement learning. *nature*, *518*(7540), 529–533.

Moita, M. A., Rosis, S., Zhou, Y., LeDoux, J. E., & Blair, H. T. (2004). Putting fear in its place: Remapping of hippocampal place cells during fear conditioning. *Journal of Neuroscience*, *24*(31), 7015–7023.

Momennejad, I., Russek, E. M., Cheong, J. H., Botvinick, M. M., Daw, N. D., & Gershman, S. J. (2017). The Successor Representation in Human Reinforcement Learning. *Nature Human Behaviour*, *1*(9), 680–692.

Montague, P. R., Dayan, P., & Sejnowski, T. J. (1996). A framework for mesencephalic dopamine systems based on predictive Hebbian learning. *Journal of neuroscience*, *16*(5), 1936–1947.

Morris, R. G., Garrud, P., Rawlins, J. N., & O'Keefe, J. (1982). Place navigation impaired in rats with hippocampal lesions. *Nature*, *297*(5868), 681–683.

Mulder, A. B., Tabuchi, E., & Wiener, S. I. (2004). Neurons in hippocampal afferent zones of rat striatum parse routes into multi-pace segments during maze navigation. *European Journal of Neuroscience*, *19*(7), 1923–1932.

Muller, R. U., Bostock, E., Taube, J. S., & Kubie, J. L. (1994). On the directional firing properties of hippocampal place cells. *The Journal of neuroscience : the official journal of the Society for Neuroscience*, *14*(12), 7235–51.

Muller, R. U., & Kubie, J. L. (1987). The effects of changes in the environment on the spatial firing of hippocampal complex-spike cells. *Journal of Neuroscience*, *7*(7), 1951–1968.

Murphy, K. (1998). Switching kalman filters. *Dept. of Computer Science, University of California, . . .*, (August), 1–18.

O'Keefe, J., & Burgess, N. (1996). Geometric determinants of the place fields of hippocampal neurons. *Nature*, *381*(6581), 425–428.

O'Keefe, J., & Conway, D. H. (1978). Hippocampal Place Units in the Freely Moving Rat: Why They Fire Where They Fire. *Experimental Brain Research*, *31*(2), 573–590.

O'Keefe, J., & Dostrovsky, J. (1971). The hippocampus as a spatial map: Preliminary evidence from unit activity in the freely-moving rat. *Brain research*.

Packard, M. G. (1999). Glutamate infused posttraining into the hippocampus or caudate-putamen differentially strengthens place and response learning. *Proceedings of the National Academy of Sciences of the United States of America*, *96*(22), 12881–12886.

Packard, M. G., & McGaugh, J. L. (1996). Inactivation of Hippocampus or Caudate Nucleus with Lidocaine Differentially Affects Expression of Place and Response Learning. *Neurobiology of Learning and Memory*, *72*(0007), 65–72.

Pazzaglia, F., Meneghetti, C., & Ronconi, L. (2018). Tracing a Route and Finding a Shortcut: The Working Memory, Motivational, and Personality Factors Involved.

Pearce, J. M., & Hall, G. (1980). A model for Pavlovian learning: variations in the effectiveness of conditioned but not of unconditioned stimuli. *Psychological review*, *87*(6), 532.

Pearce, J. M., Roberts, A. D. L., & Good, M. (1998). Hippocampal lesions disrupt navigation based on cognitive maps but not heading vectors. *Nature*, *62*(1989), 1997–1999.

Penny, W. D., Zeidman, P., & Burgess, N. (2013). Forward and backward inference in spatial cognition. *PLoS Comput Biol*, *9*(12), e1003383.

Pessiglione, M., Seymour, B., Flandin, G., Dolan, R. J., & Frith, C. D. (2006). Dopamine-dependent prediction errors underpin reward-seeking behaviour in humans. *Nature*, *442*(7106), 1042–1045.

Pfeiffer, B. E., & Foster, D. J. (2013). Hippocampal place-cell sequences depict future paths to remembered goals. *Nature*, *497*(7447), 74–79.

Piray, P., & Daw, N. (2019a). Linear reinforcement learning: Flexible reuse of computation in planning, grid fields, and cognitive control.

Piray, P., & Daw, N. D. (2019b). A transparent model for learning in volatile environments.

Poldrack, R., & Packard, M. (2003). Competition among multiple memory systems: converging evidence from animal and human brain studies. *Neuropsychologia*, *41*, 245–251.

Precup, D., Sutton, R. S., Sutton, R. S., & Singh, S. (2000). Eligibility Traces for Off-Policy Policy Evaluation.

Quirk, G. J., Muller, R. U., & Kubie, J. L. (1990). The firing of hippocampal place cells in the dark depends on the rat's recent experience. *Journal of Neuroscience*, *10*(6), 2008–2017.

Ragozzino, K., Leutgeb, S., & Mizumori, S. (2001). Dorsal striatal head direction and hippocampal place representations during spatial navigation. *Experimental Brain Research*, *139*(3), 372–376.

Rescorla, R. A., & Wagner, A. R. (1972). A theory of Pavlovian conditioning: Variations in the effectiveness of reinforcement and nonreinforcement. *Classical conditioning II: Current research and theory*, *2*, 64–99.

Roesch, M. R., Esber, G. R., Li, J., Daw, N. D., & Schoenbaum, G. (2012). Surprise! Neural correlates of Pearce-Hall and Rescorla-Wagner coexist within the brain. *European Journal of Neuroscience*, *35*(7), 1190–1200.

Russek, E. M., Momennejad, I., Botvinick, M. M., & Gershman, S. J. (2017). Predictive Representations Can Link Model-Based Reinforcement Learning to Model-Free Mechanisms. *PLoS Computational Biology*, 1–42.

Sanders, H., Wilson, M. A., & Gershman, S. J. (2020). Hippocampal remapping as hidden state inference. *eLife*, *9*, 1–31.

Sargolini, F., Fyhn, M., Hafting, T., McNaughton, B. L., Witter, M. P., Moser, M.-B., & Moser, E. I. (2006). Conjunctive representation of position, direction, and velocity in entorhinal cortex. *Science*, *312*(5774), 758–762.

Schacter, D. L., Addis, D. R., & Buckner, R. L. (2007). Remembering the past to imagine the future: The prospective brain. *Nature Reviews Neuroscience*, *8*(9), 657–661.

Schapiro, A. C., Turk-Browne, N. B., Norman, K. A., & Botvinick, M. M. (2016). Statistical learning of temporal community structure in the hippocampus. *Hippocampus*, *26*(1), 3–8.

Schmitzer-Torbert, N. C., & Redish, A. D. (2008). Task-dependent encoding of space and events by striatal neurons is dependent on neural subtype. *Neuroscience*, *153*(2), 349–360.

Schoenfeld, F., & Wiskott, L. (2015). Modeling place field activity with hierarchical slow feature analysis. *Frontiers in Computational Neuroscience*, *9*(May), 1–20.

Schuck, N. W., Cai, M. B., Wilson, R. C., & Niv, Y. (2016). Human Orbitofrontal Cortex Represents a Cognitive Map of State Space. *Neuron*, *91*(6), 1402–1412.

Schultz, W., Dayan, P., & Montague, P. R. (1997). A neural substrate of prediction and reward. *Science*, *275*(5306), 1593–1599.

Scoville, W. B., & Milner, B. (1957). Loss of recent memory after bilateral hippocampal lesions. *Journal of neurology, neurosurgery, and psychiatry*, *20*(1), 11.

Sharpe, M. J., Chang, C. Y., Liu, M. A., Batchelor, H. M., Mueller, L. E., Jones, J. L., Niv, Y., & Schoenbaum, G. (2017). Dopamine transients are sufficient and necessary for acquisition of model-based associations. *Nature Neuroscience*, *20*(5), 735–742.

Sheynikhovich, D., Chavarriaga, R., Strösslin, T., Arleo, A., & Gerstner, W. (2009). Is there a geometric module for spatial orientation? Insights from a rodent navigation model. *Psychological review*, *116*(3), 540.

Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., Lanctot, M., Sifre, L., Kumaran, D., Graepel, T., Lillicrap, T., Simonyan, K., & Hassabis, D. (2018). A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. *Science*, *362*(6419), 1140–1144.

Smith, K., Battaglia, P., & Vul, E. (2013). Consistent physics underlying ballistic motion prediction. *Proceedings of the 35th Annual Conference of the Cognitive Science Society*, 3426–3431.

Solstad, T., Boccara, C. N., Kropff, E., Moser, M.-B., & Moser, E. I. (2008). Representation of geometric borders in the entorhinal cortex. *Science*, *322*(5909), 1865–1868.

Squire, L. R., & Zola-Morgan, S. (1991). The medial temporal lobe memory system. *Science*, *253*(5026), 1380–1386.

Srinivas, N., Krause, A., Kakade, S. M., & Seeger, M. (2009). Gaussian process optimization in the bandit setting: No regret and experimental design. *arXiv preprint arXiv:0912.3995*.

Stachenfeld, K. L., Botvinick, M. M., & Gershman, S. J. (2017). The Hippocampus as a Predictive Map. *Nature Neuroscience*, *20*(11), 1643–1653.

Steinberg, E. E., Keiflin, R., Boivin, J. R., Witten, I. B., Deisseroth, K., & Janak, P. H. (2013). A causal link between prediction errors, dopamine neurons and learning. *Nature neuroscience*, *16*(7), 966–973.

Stella, F., Baracskay, P., Neill, J. O., & Csicsvari, J. (2019). Hippocampal Reactivation of Random Trajectories Resembling Brownian Diffusion. *Neuron*, 1–12.

Stensola, H., Stensola, T., Solstad, T., Frøland, K., Moser, M.-B., & Moser, E. I. (2012). The entorhinal grid map is discretized. *Nature*, *492*(7427), 72–78.

Sutton, R. S. (1988). Learning to Predict by the Methods of Temporal Differences. *Machine Learning*, 9–44.

Sutton, R. S. (1991). Dyna, an integrated architecture for learning, planning, and reacting. *ACM SIGART Bulletin*, *2*(4), 160–163.

Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning: An introduction*. MIT press.

Sutton, R., & Barto, A. (1998). *Reinforcement Learning: An Introduction* (Vol. 9). MIT Press.

Taube, J. S., Muller, R. U., & Ranck, J. B. (1990). Head-direction cells recorded from the postsubiculum in freely moving rats. I. Description and quantitative analysis. *Journal of Neuroscience*, *10*(2), 420–435.

Thompson, L. T., & Best, P. J. (1990). Long-term stability of the place-field activity of single units recorded from the dorsal hippocampus of freely behaving rats. *Brain research*, *509*(2), 299–308.

Thompson, W. R. (1933). On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, *25*(3/4), 285–294.

Thorn, C. A., Atallah, H., Howe, M., & Graybiel, A. M. (2010). Differential dynamics of activity changes in dorsolateral and dorsomedial striatal loops during learning. *Neuron*, *66*(5), 781–795.

Todorov, E. (2009). Efficient computation of optimal actions. *Proceedings of the National Academy of Sciences of the United States of America*, *106*(28), 11478–11483.

Tolman, E. C. (1948). Cognitive Maps in Rats and Man. *Psychological Review*, *55*(4), 189–208.

Tomov, M. S., Schulz, E., & Gershman, S. J. (2019). Multi-Task Reinforcement Learning in Humans, 1–16.

Tomov, M. S., Truong, V. Q., Hundia, R. A., & Gershman, S. J. (2020). Dissociable neural correlates of uncertainty underlie different exploration strategies. *Nature communications*, *11*(1), 1–12.

Tricomi, E., Balleine, B. W., & O'Doherty, J. P. (2009). A specific role for posterior dorsolateral striatum in human habit learning. *European Journal of Neuroscience*, *29*(11), 2225–2232.

Tsai, H.-C., Zhang, F., Adamantidis, A., Stuber, G. D., Bonci, A., De Lecea, L., & Deisseroth, K. (2009). Phasic firing in dopaminergic neurons is sufficient for behavioral conditioning. *Science*, *324*(5930), 1080–1084.

Tulving, E. (1972). Episodic and semantic memory. *Organization of memory*, *1*, 381–403.

Valentin, V. V., Dickinson, A., & O'Doherty, J. P. (2007). Determining the neural substrates of goal-directed learning in the human brain. *Journal of Neuroscience*, *27*(15), 4019–4026.

van der Meer, M. A. A., Johnson, A., Schmitzer-Torbert, N. C., & Redish, A. D. (2010). Triple dissociation of information processing in dorsal striatum, ventral striatum, and hippocampus on a learned spatial decision task. *Neuron*, *67*(1), 25–32.

Vargha-Khadem, F., Gadian, D. G., Watkins, K. E., Connelly, A., Van Paesschen, W., & Mishkin, M. (1997). Differential effects of early hippocampal pathology on episodic and semantic memory. *Science*, *277*(5324), 376–380.

Vertes, E., & Sahani, M. (2019). A neurally plausible model learns successor representations in partially observable environments. *arXiv*.

Vikbladh, O. M., Meager, M. R., King, J., Blackmon, K., Devinsky, O., Shohamy, D., Burgess, N., & Daw, N. D. (2019). Hippocampal Contributions to Model-Based Planning and Spatial Memory. *Neuron*, *102*(3), 683–693.

Waelti, P., Dickinson, A., & Schultz, W. (2001). Dopamine responses comply with basic assumptions of formal learning theory. *Nature*, *412*(6842), 43–48.

Wan Lee, S., Shimojo, S., & O'Doherty, J. P. (2014). Supplementary information to Neural Computations Underlying Arbitration between Model-Based and Model-free Learning. *Neuron*, *81*(3), 687–699.

Ward-Robinson, J., Coutureau, E., Good, M., Honey, R. C., Killcross, A. S., & Oswald, C. J. P. (2001). Excitotoxic lesions of the hippocampus leave sensory preconditioning intact: Implications for models of hippocampal functioning. *Behavioral neuroscience*, *115*(6), 1357.

Watkins, C. J. C. H., & Dayan, P. (1992). Q-learning. *Machine Learning*, *8*(3-4), 279–292.

White, N. M. (2004). The role of stimulus ambiguity and movement in spatial navigation: a multiple memory systems analysis of location discrimination. *Neurobiology of learning and memory*, *82*(3), 216–229.

Whittington, J. C. R., Muller, T. H., Mark, S., Chen, G., Barry, C., Burgess, N., & Behrens, T. E. J. (2020). The Tolman-Eichenbaum machine: Unifying space and relational memory through generalization in the hippocampal formation. *Cell*, *183*(5), 1249–1263.

Wilson, R. C., Takahashi, Y. K., Schoenbaum, G., & Niv, Y. (2014). Orbitofrontal cortex as a cognitive map of task space. *Neuron*, *81*(2), 267–278.

Winocur, G., Frankland, P. W., Sekeres, M., Fogel, S., & Moscovitch, M. (2009). Changes in context-specificity during memory reconsolidation: Selective effects of hippocampal lesions. *Learning & Memory*, *16*(11), 722–729.

Wunderlich, K., Dayan, P., & Dolan, R. J. (2012). Mapping value based planning and extensively trained choice in the human brain. *Nature neuroscience*, *15*(5), 786–791.

Yin, H. H., & Knowlton, B. J. (2004). Contributions of striatal subregions to place and response learning. *Learning and Memory*, *11*(4), 459–463.

Yin, H. H., & Knowlton, B. J. (2006). The role of the basal ganglia in habit formation. *Nature Reviews Neuroscience*, *7*(6), 464–476.

Yin, H. H., Ostlund, S. B., & Balleine, B. W. (2008). Reward-guided learning beyond dopamine in the nucleus accumbens: the integrative functions of cortico-basal ganglia networks. *European Journal of Neuroscience*, *28*(8), 1437–1448.

Yin, H. H., Ostlund, S. B., Knowlton, B. J., & Balleine, B. W. (2005). The role of the dorsomedial striatum in instrumental conditioning. *European Journal of Neuroscience*, *22*(2), 513–523.

Yu, C., Behrens, T. E. J., & Burgess, N. (2020). Prediction with directed transitions: complex eigenstructure, grid cells and phase coding. *arXiv preprint arXiv:2006.03355*.

Ziv, Y., Burns, L. D., Cocker, E. D., Hamel, E. O., Ghosh, K. K., Kitch, L. J., El Gamal, A., & Schnitzer, M. J. (2013). Long-term dynamics of CA1 hippocampal place codes. *Nature neuroscience*, *16*(3), 264.

Appendix A

Supplementary information to Chapter 2

I describe arbitration in our model in more detail. I then describe task-specific adaptations that were made to the model, and some additional experiments.

A.1 Arbitration between hippocampal and striatal systems

I implemented arbitration between the hippocampal and dorsal striatal systems in our model using a rule introduced by Wan Lee et al. (2014). These authors suggested that arbitration between model-based and model-free systems was done based on a *reliability* signal. They also used fMRI to show that inferior lateral prefontal and frontopolar cortex encode such reliability signals, as well as the output of a comparison between these signals. Furthermore, they showed evidence that the connectivity between these regions and model-free value areas is negatively modulated by the degree of model-based control.

Here, I applied their method to arbitration between a hippocampal system based on the Successor Representation and a striatal system based on model-free learning. The idea is that the reliability of both systems is tracked by computing the recent average of prediction errors of both systems. The Pearce-Hall update rule for tracking average prediction error is:

$$\Delta\Omega = \eta(|\delta| - \Omega) \tag{A.1}$$

where $|\delta|$ is the absolute RPE and η is a learning rate. The reliability is defined as:

$$\chi = (\delta_{MAX} - \Omega)/\delta_{MAX} \tag{A.2}$$

with δ_{MAX} being the upper bound of the prediction error, which was set to 1. After each episode, the reliability of each system was updated using the following rule:

$$\Delta\chi = \eta \left[\left(1 - \frac{|\delta|}{\delta_{MAX}} \right) - \chi \right] \tag{A.3}$$

This measure goes to zero as the average prediction error increases ($\Omega \to \delta_{MAX}$), and goes to one as the average prediction error decreases ($\Omega \to 0$).

Following Wan Lee et al. (2014), I use the reliability measure for arbitration. These authors computed transition rates α and β for transitioning from MF to MB states and vice versa as follows. Here I use the same terms but for transitions between MF and SR. These transition rates are functions of the reliability of the respective systems:

$$\alpha(\chi_{MF}) = \frac{A_\alpha}{1 + \exp(B_\alpha \chi_{MF})} \tag{A.4}$$

$$\beta(\chi_{SR}) = \frac{A_\beta}{1 + \exp(B_\beta \chi_{SR})} \tag{A.5}$$

where the A and B parameters in both equations determine transition rate and the steepness of these curves, respectively. These parameters were fitted to behavioural data by Wan Lee et al. (2014) and I matched their parameter values (see Table A.1).

At each time step, the rate of changes of the probability of choosing the SR system P_{SR} was computed using the following differential equation:

$$\frac{dP_{SR}}{dt} = \alpha(\chi_{MF})(1 - P_{SR}) - \beta(\chi_{SR})P_{SR} \tag{A.6}$$

Although not explored here (but see Wan Lee et al., 2014), this means that there is a certain "stickiness" to the model: if the model is currently choosing MF actions, it will take some time to move weight to the MB system.

Following Wan Lee et al. (2014), state action value estimates were given by a weighted average of the two model components:

$$Q(s,a) = P_{SR} Q_{HPC}(s,a) + (1 - P_{SR}) Q_{DLS}(s,a) \tag{A.7}$$

Thus, the degree to which a system contributes to the value estimate is influenced by its reliability. Given these full-model state-action values, the agent chose actions following a softmax policy:

$$\pi(a|s) = \frac{e^{\tau^{-1}Q(s,a)}}{\sum_{a'} e^{\tau^{-1}Q(s,a')}} \tag{A.8}$$

where τ^{-1} is an inverse temperature parameter which sets the balance between exploration and exploitation. The higher the inverse temperature, the more the agent chooses higher-valued actions.

A.2 Task-specific adaptations

Although the general model architecture remained the same throughout all simulations, different adaptations were made to the model described above such that it could be used in the differen state spaces defined by the tasks.

Plus maze

For the Plus Maze task described in Figure 3, landmark cells were tuned to the ends of the maze. I assumed that the landmark cells could not distinguish between the two ends of the maze such that, from the point of view of the striatal system, probe trials and training trials looked the same.

Blocking

For the blocking simulations (Figure 4), I adapted the hippocampal controller (that worked with a tabular state representation as input) to incorporate the effects of boundaries on place cell firing. To that end, I defined the hippocampal SR system using linear function approximation. The agent observes states through a vector of features $\mathbf{f}(s)$ which, if chosen rightly, will be of much smaller dimension than the number of states, allowing the agent to generalise to states that are nearby in feature space. The feature-based SR (Barreto et al., 2016) encodes the expected discounted future activity of each feature:

$$\boldsymbol{\psi}^{\pi}(s) = \mathbb{E}_{\pi}\left[\sum_{t=0}^{\infty} \gamma^t \mathbf{f}(s_t)|s_0 = s\right] \tag{A.9}$$

As in the tabular case, the feature-based SR can be used to compute value when multiplied with a vector of reward expectations per feature, $\mathbf{u}$: $V^{\pi}(s) = \psi^{\pi}(s)^{T}\mathbf{u}$. In the case of linear function approximation, these Successor Features - in Equation A.9 are approximated by a linear function of the features $\mathbf{f}$:

$$\hat{\boldsymbol{\psi}}(s) = W^{T}\mathbf{f}(s), \tag{A.10}$$

where W is a weight matrix which parameterises the approximation.

In the context of hippocampus, the feature-based SR allows us to represent states as population vectors of place cells with overlapping firing fields (the features), rather than having a one-to-one correspondence between place cells and states. Then we are free to model the dependence of the place cell firing on specific environmental features (boundaries). This dependence has been extensively characterised by computational models of boundary vector cells (BVCs) (Barry et al., 2006; Burgess et al., 2000; de Cothi & Barry, 2020; Grieves et al., 2018; Hartley et al., 2000), which were shown to exist in the subiculum (Lever et al., 2009). Accordingly, I modelled a set of hippocampal place cells whose activity $\mathbf{f}_i(s_t)$ was the thresholded sum of a set of BVC inputs (see Barry et al., 2006, for details on how BVC and place cell maps were calculated).

Thus, at every state s (corresponding to a location) in the environment, the agent observed a population vector $\mathbf{f}(s)$ of BVC-driven place cells (see Figure A.1 for an example). It then computed its estimated Successor Features - using its current estimate of weights W and Equation A.10, which encode the discounted sum of future population firing rate vectors $\mathbf{f}$ of the input place cells. In terms of circuitry, W might correspond to the Schaffer collaterals projecting from CA3 to CA1 neurons, corresponding to $\mathbf{f}$ and -, respectively.

As in the tabular case, temporal difference learning can be used to update the SR weights:

$$\Delta W = \alpha\left[\mathbf{f}(s_t) + \gamma\boldsymbol{\psi}(s_{t+1}) - \boldsymbol{\psi}(s_t)\right]\mathbf{f}(s_t)^{T} \tag{A.11}$$

Note that the algorithm has not changed with respect to the one-hot state encoding mentioned earlier – it is easy to see that the function approximation version reduces to the tabular case when $\mathbf{f}$ is a one-hot vector. The reward expectation vector $\mathbf{u}$ was updated using a simple delta rule:

$$\Delta\hat{\mathbf{u}} = \alpha\left(r_t - \hat{\mathbf{u}}^{T}\mathbf{f}\left(s_t\right)\right)\mathbf{f}\left(s_t\right) \tag{A.12}$$

Two-step tasks

For the non-spatial two-step tasks (Figure 2.7 and Figure A.3), the DLS cells were assumed to provide a one-hot representation of the task states. While this is significantly different from the landmark cell representation used in the spatial navigation studies, this representation reflected the fact that states were uniquely identifiable as different images. Furthermore, this is consistent with experimental evidence showing that dorsal striatum represents reward-predictive cues (van der Meer et al., 2010).

Hippocampal damage in the two-step and spatial tasks

In order to mimic the individual differences between participants found by Vikbladh et al. (2019), I sampled 20 different agents with varying values for the parameters governing the transition from SR to MF and vice versa (see Equations A.4 and A.5). Specifically, I sampled A_α values (steepness of the transition from MF to SR) uniformly between .5 and 5, and A_β (steepness of the transition from SR to MF) values uniformly between 2 and .5. In addition to the 20 "full agents", I sampled 20 agents for which the hippocampal component was partially inactivated by setting a maximum to the P_{SR}. To mimic variability in the size of the lesion that was present in the dataset of Vikbladh et al. (2019), I sampled max P_{SR} values from a uniform distribution between 0 and 0.35.

A.3 Quantification and statistical analysis

To investigate the relationship between the agents' spatial navigation and non-spatial decision making strategies, I quantified the agents' degree of MB planning, as well as their degree of using an allocentric strategy, and computed their correlation.

For quantifying MB planning, I followed earlier studies (Daw et al., 2011; Vikbladh et al., 2019) and analysed the agents' choices using a mixed-effects logistic regression (estimated using the *statsmodels* Python package, **Seabold2010Statsmodels:Pyt**
For each trial, the dependent variable (stay with the same first-level action or switch) was explained in terms of whether there was a reward on the previous trial, whether the previous transition was of the rare or common type, and the interaction between these factors. The logic of the two-step task is

that an MB learner will stay with the same action if it was rewarded after a common transition, but will be more likely to switch if it gets rewarded after a rare transition. Thus, the degree of MB planning can be quantified as the interaction between previous reward and trial type.

For quantifying the degree of allocentric place memory, I computed the average distance between the previous platform location and the location of the maximum of the agent's value function at the start of the next session. This is akin to the boundary distande error employed by Vikbladh et al. (2019).

After computing the correlation between allocentric place memory and MB planning for both the "healthy" and "lesioned" groups of agents, I asked whether the two correlation coefficients were significantly different from each other by applying the Fisher z-transform (**Fisher1915FrequencyPopulation**) to the coefficients, and testing whether the difference between the transformed coefficients was significantly different from zero.

For the cued water maze task described in FigureA.2, the differences among the groups in relation to the number of agents that chose a place or a cue strategy were analysed by the Fisher exact test as implemented in R (**Team2013R:Computing**).

A.4 Additional tasks

Cue versus place Water Maze

In addition to the hippocampal lesion described in Figure 2, I simulated a DLS lesion in the task used by Pearce et al. (1998). Figure A.2A shows the simulation results: there is little to no learning across sessions for the first trials of each session, indicating impaired acquisition of the landmark-platform association. Fourth-trial performance is not significantly worse than control performance, which is a sign of intact place learning as agents still learn during a session in which the platform has a fixed location. This is consistent with a previous finding showing that dopamine depletion in the DLS impairs egocentric but not allocentric Water Maze navigation (Braun et al., 2015). Figure A.2B shows results from a study by Miyoshi et al. (2012) that investigated the effects of bilateral lesions of the hippocampus, DLS or both in a cue on a probe test in the Water Maze. Animals were trained to swim to a given location in the Water Maze, that was indicated by the presence of a landmark. Then, during a probe trial, the landmark was placed elsewhere in the maze,

and the animals' behaviour wass classified as cue-guided if the animal swam directly to the cued platform, as place-guided if it swam directly to the place the hidden platform was the day before, or as thigmotaxic if the animals swam around the edge of the pool. This dual-solution probe trial is akin to the first trial of each session in Pearce et al. (1998). Figure A.2C shows that our simulations accurately capture these results, where I classified behaviour as "cue" or "place" guided if the agent reached the platform as indicated by the landmark or previous location within a given number of time teps (60), and as "neither" otherwise.

Deterministic two-step task

In the experiment designed by Doll et al. (2015), human participants where shown a pair of two pictures from one of two categories (faces or tools) and were asked to choose one. This was defined as the start state. The participants' initial choice determined which of two second-stage states they would transition to. These second stage states corresponded to a choice from a pair of pictures from one of two new categories (scenes or body parts; see Figure A.3A). Each second-stage option (the 'outcome') was either rewarded with money or not rewarded. The reward probability for each outcome drifted slowly and randomly such that participants continuously learned by trial and error which second-stage choices were most likely to be rewarded. The total expected value of both scene and body part states was made equal to avoid inducing a bias. The first-stage choices deterministically led to different outcomes: selecting one of the tools or one of the faces always led to the scenes, while the other tool or face always led to the body parts.

This task structure dissociates behaviour consistent with MB and MF learning. A model-based learner represents transition probabilities, and uses this transition model to compute the best action. Thus, when a model-based learner encounters a reward, this should affect its behaviour in the next trial regardless of whether it starts in the same state as the previous trial (for example, faces followed by faces) or in a different one (for example, faces followed by tools). In contrast, a MF learner evaluates options in terms of the outcomes they have previously produced. Therefore, a model-free learner, upon receiving a reward, will only increase the probability of taking the same action in the next trial if that next trial starts in the same state as the previous one. Consistent with humans making use of both strategies, Doll and

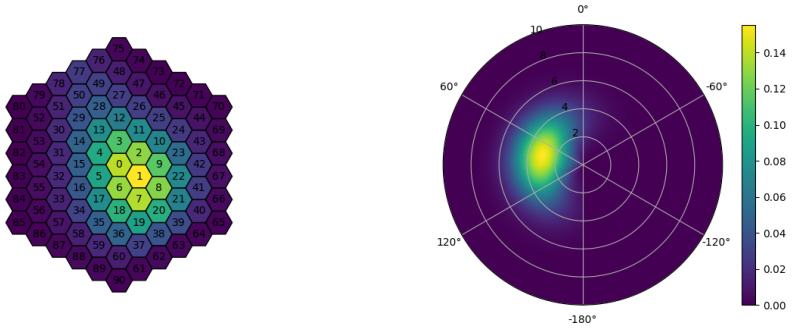


FIGURE A.1: Example receptive fields. Left panel: Example SR place cell map in a discretised maze. Right panel: Example landmark cell receptive field plotted in polar coordinates.

colleagues showed that human performance on this task lies somewhere in between these strategies (Figure A.3B).

Our model recapitulates the main effects found by Doll and colleagues. The SR model mimics model-based behaviour by separating reward information from information about the transition structure. When the goal is reached, value is generalised to states that predict the goal states. Thus, following reward, the hippocampal model will learn to take actions to end up in the same second stage state in the next trial, regardless of whether it has the same or different starting state (Figure A.3C). In contrast, the striatal learner learns separate action values for each state. Therefore, rewards obtained following one start state will not affect action values in the other start state (Figure A.3C). Combining these two models gives a pattern of behaviour in between model-based and model-free, akin to human performance. However, in contrast to our model, human participants showed a higher stay probability for the "same starting state" condition than for the "different starting state" condition. This propensity to stay with the same action does not follow directly from a MF/MB trade-off.

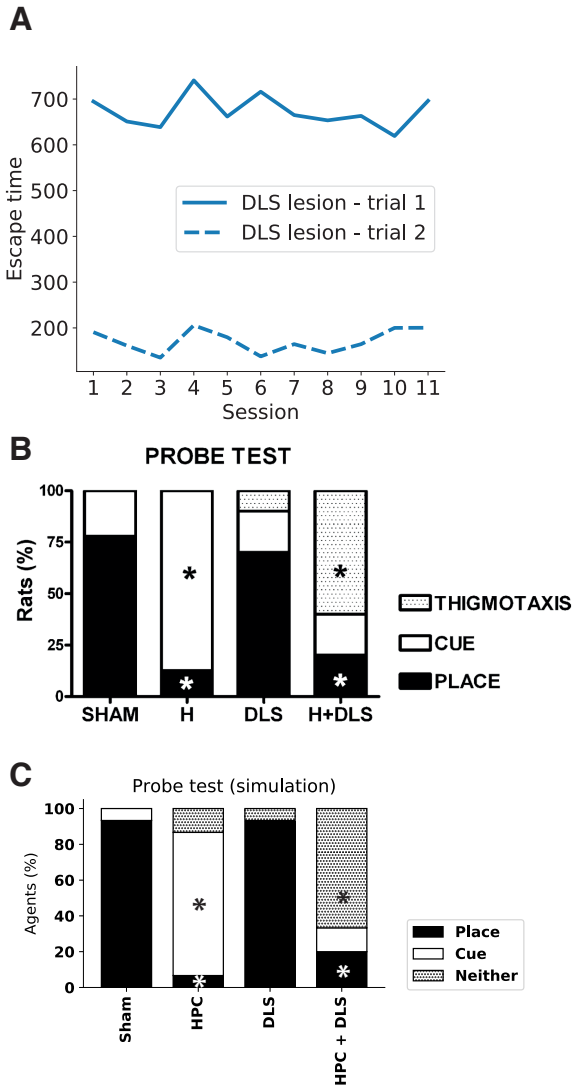


FIGURE A.2: **(A)** Simulation result of a DLS lesion in the Pearce et al. Pearce et al., 1998 study, showing escape time on the first and fourth trial of each session. Landmark and platform were moved together after every session. **(B)** Data from Myoshi et al. Miyoshi et al., 2012 showing the effects of bilateral lesions of the dorsal hippocampus (H) and/or the dorsolateral striatum (DLS) on a probe test carried out after 5 days of training on the Morris Water Maze. Data express the proportion of rats that (i) swam directly to the cued platform, (ii) to the place the hidden platform was the day before, or (iii) exhibited thigmotaxic swimming behaviour (swimming around the edges of the pool) in the first trial in the cued version. *P < 0.05 compared to SHAM animals; Fisher test. **(C)** Simulation results showing the effects of ablating the HPC and/or DLS model components on the task described in (B). * correspond to P < 0.05 in a Fisher test compared to SHAM animals/agents in both (B) and (C).

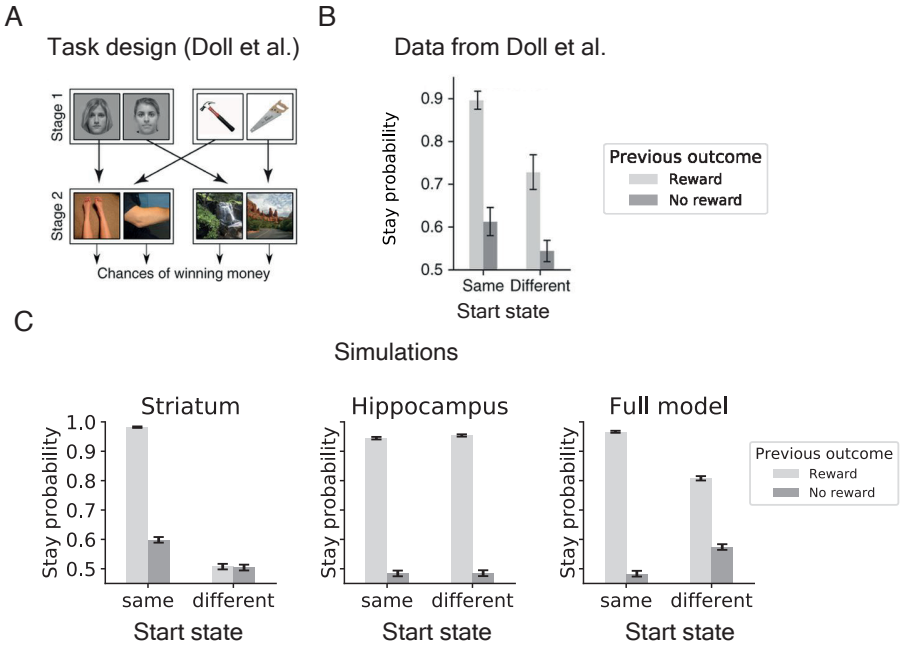


FIGURE A.3: **(A)** Task stucture employed by Doll et al. (2015). **(B)** The probability that human participants chose the same first-stage action as on the previous trial binned by whether the previous choice was rewarded, and whether they started in the same state. **(C)** Simulation results. The hippocampal model mimics the true MB agent presented in the original paper. The striatal model shows MF behaviour. Combining the two models results in a behavioural pattern that shows both effects. As in Doll et al. (2015), MB behaviour was quantified as the main effect of previous reward on choice behaviour (estimate=.96, $Z = 4.3$, $P = 1.66 \times 10^{-5}$). This effect is greater when the current state is the same as the previous one (estimate=2.37, Z=6.64, $P = 3.18 \times 10^{-11}$), indicating the presence of MF behaviour.

Name	Symbol	Value
SR learning rate	α_M	0.07
Q learning rate	α_Q	0.07
Softmax inverse temperature (exploration parameter)	τ^{-1}	5
Discount parameter	γ	0.95
Reliability learning rate	η	0.03
Maximum prediction error	δ_{MAX}	1
Steepness of transition curve MF to SR	A_α	3.2
Steepness of transition curve SR to MF	A_β	1.1

TABLE A.1: Parameter settings

Appendix B

Addressing the bias in Kalman TD

The basic version of the Kalman Temporal Differences (KTD) algorithm introduced in Chapter 1 of this thesis was derived based on the simplifying assumption that the observation noise is white (independent per time step). In reality, this is only the case when the transitions are deterministic. In most cases, the successive uncertainty terms cannot be treated as independent because they are related by the way in which the agent moves through the world. In that case the white noise assumption leads to biased estimates.

Here, I first show how KTD is derived in the deterministic case (B.1), and how a biases arises when this is directly applied to stochastic domains (B.2). Then, I show in B.3 how the problem can be alleviated by using a coloured noise model introduced by Geist and Pietquin (2010b) (and previously by Engel et al., 2005), which leads to extended (X)KTD. Finally, I show empirically that this bias indeed arises on a simple stochastic MDP, and I compare KTD and XKTD's performance on this MDP (B.5). I discuss the value learning case throughout for simplicity but these results apply equally to learning successor features.

B.1 Kalman TD, the deterministic case

In order to see why KTD's cost function is biased when transitions are not deterministic, it is necessary to first show how the original KTD cost function assumes deterministic transitions. When transitions are deterministic, as assumed in this basic version of KTD, the Bellman equations for V and Q simplify to:

$$V^{\pi}(s) = R(s, \pi(s), s') + \gamma V^{\pi}(s'), \forall s \tag{B.1}$$

$$Q^{\pi}(s, a) = R(s, a, s') + \gamma Q^{\pi}(s', \pi(s')), \forall s, a \tag{B.2}$$

where the expectations have disappeared because there is no uncertainty in the transitions. With $\hat{V}_{\mathbf{w}_t}$ as the current estimate of V given the current parameter estimates $\hat{\mathbf{w}}$, we denote

$$g_t(\mathbf{w}_t) = \begin{cases} \hat{V}_{\mathbf{w}_t}(s_t) - \gamma \hat{V}_{\mathbf{w}_t}(s_{t+1}) & \text{for evaluation of } V \\ \hat{Q}_{\mathbf{w}_t}(s_t, a_t) - \gamma \hat{Q}_{\mathbf{w}_t}(s_{t+1}, a_{t+1}) & \text{for evaluation of } Q \end{cases} \tag{B.3}$$

Then, the TD error can be written generically as:

$$\delta_t = r_t - g_t(\mathbf{w}_t), \tag{B.4}$$

and the observation equation (relating the observed rewards to the hidden parameters $\mathbf{w}$) can be written as:

$$r_t = g_t(\mathbf{w}_t) + \boldsymbol{\nu}_t \,. \tag{B.5}$$

The observation noise $\boldsymbol{\nu}$ is assumed white, independent and of variance $\mathrm{P}_{\boldsymbol{\nu}}$. Notice that this assumption does not hold for stochastic MDPs, which is discussed in the next section.

The KTD cost function (equation) can be written as the trace of the parameter covariance matrix:

$$\begin{aligned} J_t(\mathbf{w}) &= \mathbb{E}\left[|| \, \mathbf{w}_t - \hat{\mathbf{w}}_{t|t} ||^2 | r_{1:t} \right] && \text{(B.6)} \\ &= \mathbb{E}\left[(\mathbf{w}_t - \hat{\mathbf{w}}_{t|t})^T (\mathbf{w}_t - \hat{\mathbf{w}}_{t|t})) | r_{1:t} \right] && \text{(B.7)} \\ &= \mathrm{Tr}\left(\mathbb{E}\left[(\mathbf{w}_t - \hat{\mathbf{w}}_{t|t})(\mathbf{w}_t - \hat{\mathbf{w}}_{t|t})^T | r_{1:t} \right] \right) && \text{(B.8)} \\ &= \mathrm{Tr}\left(\Sigma_{t|t} \right) && \text{(B.9)} \end{aligned}$$

The central idea of Kalman TD is that the optimal Kalman gain $\boldsymbol{\kappa}_t$ can be computed by finding the derivative of this cost function with respect to the gain. The first step to doing so is to express the covariance as a function of the gain. First, a few definitions. Following Geist and Pietquin, I will here use the tilde notation to denote errors in estimates of the different quantities. For example, the innovation is the expectation (conditioned on past observed data) of the TD prediction error:

$$\tilde{r}_t = r_t - \hat{r}_{t|t-1} = \mathbb{E}\left[\delta_t | r_{1:t} \right] \tag{B.10}$$

Likewise,

$$\begin{aligned}
\tilde{\mathbf{w}}_{t|t} &= \mathbf{w}_t - \hat{\mathbf{w}}_{t|t} \\
\tilde{\mathbf{w}}_{t|t-1} &= \mathbf{w}_t - \hat{\mathbf{w}}_{t|t-1} \\
\Sigma_{t|t} &= cov\left(\tilde{\mathbf{w}}_{t|t} | r_{1:t}\right) \\
\Sigma_{t|t-1} &= cov\left(\tilde{\mathbf{w}}_{t|t-1} | r_{1:t-1}\right) \\
L_t &= cov(\tilde{r}_t | r_{1:t-1}) \\
\Sigma_{\mathbf{w}\, r_t} &= \mathbb{E}\left[\tilde{\mathbf{w}}_{t|t-1} \tilde{r}_t | r_{1:t-1}\right]
\end{aligned} \tag{B.11}$$

The covariance can then be expanded as follows:

$$\Sigma_{t|t} = cov\left(\mathbf{w}_t - \hat{\mathbf{w}}_{t|t} | r_{1:t}\right) \tag{B.12}$$

$$= cov\left(\mathbf{w}_t - \left(\hat{\mathbf{w}}_{t|t-1} + \boldsymbol{\kappa}_t\, \tilde{r}_t | r_{1:t}\right)\right) \tag{B.13}$$

$$= cov\left(\tilde{\mathbf{w}}_{t|t-1} - \boldsymbol{\kappa}_t\, \tilde{r}_t | r_{1:t}\right) \tag{B.14}$$

$$\Sigma_{t|t} = \Sigma_{t|t-1} - \Sigma_{\mathbf{w}\, r_t}\, \boldsymbol{\kappa}_t^T - \boldsymbol{\kappa}_t\, \Sigma_{\mathbf{w}\, r_t}^T + \boldsymbol{\kappa}_t\, L_t\, \boldsymbol{\kappa}_t^T \tag{B.15}$$

Zeroing the gradient of the trace of this matrix with respect to $\boldsymbol{\kappa}_t$ gives the optimal gain: $\boldsymbol{\kappa}_t = \Sigma_{\mathbf{w}\, r_t} L_t^{-1}$, which is used for the basic KTD algorithm used in the thesis.

B.2 Stochastic transitions and bias

The KTD framework outlined in the previous section assumed deterministic transitions. When transitions are stochastic, the assumption that the observation noise $\boldsymbol{\nu}_t$ is white leads to a bias in the cost function. Here, I will show how this bias in the cost function arises when KTD (assuming equation B.5 as observation equation) is applied.

The observation equation for a stochastic MDP is:

$$r_t = \mathbb{E}_{s'|s_t,a_t}\left[g_t(\mathbf{w}_t) + \boldsymbol{\nu}_t\right]. \tag{B.16}$$

Notice that, because transitions are stochastic, the expectation over successor states given the current state and action is added compared to the deterministic case (equation B.5). The difference is that transitions are now not just sampled but averaged. In order to see that the cost function is biased now

that transitions are stochastic, we will compute the expectation of the terms in the cost function under the next-state distribution. Recall that the cost function is:

$$\mathcal{J}_t(\mathbf{w}) = \text{Tr}\left(\boldsymbol{\Sigma}_{t|t}\right) = \Sigma_{t|t-1} - \Sigma_{\mathbf{w}\,r_t}\,\boldsymbol{\kappa}_t^T - \boldsymbol{\kappa}_t\,\Sigma_{\mathbf{w}\,r_t}^T + \boldsymbol{\kappa}_t\,\mathcal{L}_t\,\boldsymbol{\kappa}_t^T \tag{B.17}$$

where the calligraphic symbols denote terms that are defined in the same manner as in the notations above (equation B.11). For example, the covariance between the parameters and the innovation is now:

$$\Sigma_{\mathbf{w}\,r_t} = \mathbb{E}\left[\tilde{\mathbf{w}}_{t|t-1}\tilde{\mathfrak{r}}_t|r_{1:t-1}\right] \text{ with } \tilde{\mathfrak{r}}_t = r_t - \hat{\mathfrak{r}}_{t|t-1} = r_t - \mathbb{E}[\mathbb{E}_{s'|s_t,a_t}[g_t(\mathbf{w}_t)]|r_{1:t-1}] \tag{B.18}$$

Notice, again, the expectation with respect to the destination state s'. The prediction of the reward is unbiased, and so is the innovation:

$$\mathbb{E}_{s'|s_t,a_t}[\hat{r}_{t|t-1}] = \hat{\mathfrak{r}}_{t|t-1} \tag{B.19}$$

$$\mathbb{E}_{s'|s_t,a_t}[\tilde{r}_{t|t-1}] = \tilde{\mathfrak{r}}_{t|t-1} \tag{B.20}$$

The predicted covariance $\Sigma_{t|t-1}$ does not depend on the destination state s', so it is unbiased:

$$\mathbb{E}_{s'|s_t,a_t}[\Sigma_{t|t-1}] = \Sigma_{t|t-1} \tag{B.21}$$

The covariance between the of the parameters and the innovation is linear in the innovation, so it is also unbiased:

$$\mathbb{E}_{s'|s_t,a_t}[\Sigma_{\mathbf{w}\,r_t}] = \Sigma_{\mathbf{w}\,r_t} \tag{B.22}$$

The variance of the innovation, however, has a squared dependence on the innovation, hence it is biased:

$$\mathbb{E}_{s'|s_t,a_t}\left[L_t\right] = \mathbb{E}_{s'|s_t,a_t}\left[\mathbb{E}\left[\tilde{r}_t^2|r_{1:t-1}\right]\right] \tag{B.23}$$

$$= \mathbb{E}\left[\mathbb{E}_{s'|s_t,a_t}\left[\tilde{r}_t^2\right]|r_{1:t-1}\right] \tag{B.24}$$

$$= \mathbb{E}\left[\mathbb{E}_{s'|s_t,a_t}[\tilde{r}_t]^2|r_{1:t-1}\right] + \mathbb{E}\left[\mathbb{E}_{s'|s_t,a_t}[\tilde{r}_t^2] - \left(\mathbb{E}_{s'|s_t,a_t}[\tilde{r}_t]\right)^2\right] \tag{B.25}$$

$$= \mathbb{E}\left[\mathfrak{r}_t^2|r_{1:t-1}\right] + \mathbb{E}\left[\mathbb{E}_{s'|s_t,a_t}[\tilde{r}_t^2] - \left(\mathbb{E}_{s'|s_t,a_t}[\tilde{r}_t]\right)^2\right] \tag{B.26}$$

$$= \mathcal{L}_t + \mathbb{E}[cov_{s'|s_t,a_t}(\tilde{r}_t)|r_{1:t-1}] \tag{B.27}$$

In this derivation, the fact that the expectation of the square of a random variable is the square of the expectation plus the difference between both (i.e. $\mathbb{E}[X^2] = \mathbb{E}[X]^2 + (\mathbb{E}[X^2] - \mathbb{E}[X]^2)$ was used, which follows from the definition of variance ($Var(X) = \mathbb{E}[X^2] - E[X]^2$).

Given the above bias in the variance of the innovation, we can compute the bias in the cost function ($\mathbb{E}_{s'|s_t,a_t}[J_t(\mathbf{w})] - \mathcal{J}_t(\mathbf{w})$):

$$\mathbb{E}_{s'|s_t,a_t}[J_t(\mathbf{w})] - \mathcal{J}_t(\mathbf{w}) = \mathbb{E}_{s'|s_t,a_t}\left[\mathrm{Tr}(\boldsymbol{\kappa}_t(L_t - \mathcal{L}_t)\,\boldsymbol{\kappa}_t^T)\right] \tag{B.28}$$

$$= \mathrm{Tr}(\boldsymbol{\kappa}_t\,\boldsymbol{\kappa}_t^T)\,\mathbb{E}_{s'|s_t,a_t}[L_t - \mathcal{L}_t] \tag{B.29}$$

$$= \boldsymbol{\kappa}_t^T\,\boldsymbol{\kappa}_t(\mathbb{E}_{s'|s_t,a_t}[L_t] - \mathcal{L}_t) \tag{B.30}$$

$$= ||\,\boldsymbol{\kappa}_t\,||^2\,\mathbb{E}\left[cov_{s'|s_t,a_t}(r_t - g_t(\mathbf{w}))|r_{1:t-1}\right] \tag{B.31}$$

Of course, this bias gets larger the more stochastic the environment is. As noted by Geist and Pietquin, this bias is similar to the bias arising from the minimisation of a square Bellman residual, such as in the residual algorithms of Baird (1995) (see also Sutton & Barto, 2018, chapter 11).

B.3 Coloured noise model

To alleviate the issue of bias in KTD, Geist and Pietquin (2010b) introduced a coloured noise model that was first introduced by (Engel et al., 2005) in Gaussian Process TD. The key idea is to replace the white observation noise in the generative model by a "coloured" observation noise, i.e. a noise that is not independent per time step.

Assuming the policy is fixed during evaluation, the MDP is simply a Markov chain with probability transitions $p^\pi(.|s) = p(.|\pi(s))$ and rewards $R^\pi(s,s') = R(s,\pi(s),s')$. We can define the value function as the expectation (over all possible trajectories through the state space) of the following random process describing the discounted return $D^\pi(s)$:

$$D^\pi(s) = \sum_{t=0}^{\infty} \gamma^t R^\pi(s_t, s_{t+1})|s_0 = s, \text{with } s_{t+1} \sim p^\pi(.|s_t) \tag{B.32}$$

Note that the randomness in $D(s_0)$ for any state s_0 is coming from both the stochasticity in the sequence of states that follow s_0 and to the randomness in the rewards. Following Engel, these are referred to as *intrinsic* sources of

randomness. The random process follows a Bellman like recurrence:

$$D^\pi(s) = R^\pi(s,s') + \gamma D^\pi(s'), \text{with } s' \sim p^\pi(.|s) \tag{B.33}$$

where the equality sign marks the equality in the distributions on either side of the equation. By definition, the *value function* is simply the expected value of this random variable, i.e. $V^\pi(s) = \mathbb{E}[D^\pi(s)]$. Thus, we can decompose the discounted return D into its mean and a random, zero-mean residual ΔV:

$$D^\pi(s) = \mathbb{E}[D^\pi(s)] + (D^\pi(s) - \mathbb{E}[D^\pi(s)]) = V^\pi(s) + \Delta V^\pi(s) \tag{B.34}$$

Combining equations B.33 and B.34, we can express the reward as a function of the value plus a noise:

$$R^\pi(s,s') = V^\pi(s) - \gamma V^\pi(s') + N(s,s') \tag{B.35}$$

where the noise is defined as the sum of the residuals of the current and destination state:

$$N(s,s') = \Delta V^\pi(s) - \gamma \Delta V^\pi(s') \tag{B.36}$$

Both Engel et al. (2005) and Geist and Pietquin (2010b) assumed that each of the residuals $\Delta V^\pi(s_t)$ is generated independently of all the others. This is of course a very strong assumption, as transitions between different states are likely to render the residuals dependent, but it allowed the development of a "coloured" noise model for the extended Kalman TD model (XKTD). In XKTD, the noise term n_t in the observation equation ($r_t = g_t(\mathbf{w}_t) + n_t$), rather than a white noise, is a moving average (MA) noise, defined as the sum of two white noises:

$$n_t = \gamma u_t + u_{t-1}, u_t(0, \sigma_t^2) \tag{B.37}$$

B.4 Extending Kalman TD with coloured noise

In order to rederive Kalman TD with a MA noise, the scalar MA noise n_t is expressed as a vectorial auto-regressive (AR) noise. The KTD state-space model is extended to include this vectorial AR noise and, as it turns out, the general KTD algorithm applies well to this new state-space model, as described below.

Let ω_t be an auxiliary random variable. The scalar MA noise (B.37) is equivalent to the vectorial AR noise:

$$\begin{bmatrix} \omega_t \\ n_t \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \omega_{t-1} \\ n_{t-1} \end{bmatrix} + \begin{bmatrix} 1 \\ -\gamma \end{bmatrix} u_t \tag{B.38}$$

From this vectorial AR noise, it can be seen that $n_t = \omega_{t-1} - \gamma u_t$ and $\omega_t = u_t$, so $n_t = -\gamma u_t + u_t - 1$, which is the correct MA model. The noise:

$$u'_t \begin{bmatrix} u_t \\ -\gamma u_t \end{bmatrix} \tag{B.39}$$

is also centered with covariance matrix:

$$\Sigma_{u'_t} = \sigma_t^2 \begin{bmatrix} 1 & -\gamma \\ -\gamma & \gamma^2 \end{bmatrix} \tag{B.40}$$

With this new noise formulation, the state-space formulation of KTD can be extended to:

$$\begin{cases} \boldsymbol{\theta}_t = F\boldsymbol{\theta}_{t-1} + \boldsymbol{\xi}'_t \\ r_t = g_t(\boldsymbol{\theta}_t) \end{cases} \tag{B.41}$$

where the parameter vector is now extended with the vectorial AR noise:

$$\boldsymbol{\theta}_t = \begin{bmatrix} \mathbf{w}_t \\ \omega_t \\ n_t \end{bmatrix} \tag{B.42}$$

Crucially, the observation noise n_t is now a part of the extended parameter vector, such that it will be estimated in the inference process. As for the evolution matrix F, this was an identity matrix in the original KTD formulation. Here, it will take into account the structure of the MA observation noise (B.38). Let p be the number of parameters and I_p the identity matrix of size p. The evolution matrix is given by ($\mathbf{0}$ denotes a $p \times 1$ vector of zeros):

$$F = \begin{bmatrix} I_p & \mathbf{0} & \mathbf{0} \\ \mathbf{0}^T & 0 & 0 \\ \mathbf{0}^T & 1 & 0 \end{bmatrix} \tag{B.43}$$

The evolution noise $\boldsymbol{\xi}$ is also extended to take into account the MA observation noise. It is still of mean zero, but the covariance matrix is extended using (B.40):

$$\mathrm{P}_{\boldsymbol{\xi}}{}' = \begin{bmatrix} \mathrm{P}_{\boldsymbol{\xi}} & \mathbf{0} & \mathbf{0} \\ \mathbf{0}^T & \sigma_t^2 & -\gamma\sigma_t^2 \\ \mathbf{0}^T & -\gamma\sigma_t^2 & \gamma^2\sigma_t^2 \end{bmatrix} \tag{B.44}$$

Finally, the observation matrix remains the same:

$$r_t = g_t(\boldsymbol{\theta}_t) = g_t(\boldsymbol{\theta}_t) + n_t \tag{B.45}$$

However, the observation noise is now part of the evolution equation, and has to be inferred from the data.

The updated state space formulation above has motivated the XKTD algorithm, shown below. It is very similar to the original KTD, except that now we are predicting the mean and covariance of the extended parameter vector $\boldsymbol{\theta}$, using the evolution matrix F (which, in the case of KTD, was the identity matrix). The computational complexity is the same for both algorithms, as the parameter vector was only extended with two scalars. However, because of the memory effects induced by the coloured noise estimation, XKTD cannot be applied to off-policy evaluation.

B.5 Empirical evaluation

In order to empirically asses how damaging the white noise assumption is, I now compare KTD's value estimates to the true (unbiased) value, approximated by Monte Carlo (MC) sampling. In MC sampling, value is estimated by simply averaging sample returns across episodes (Sutton and Barto). For completeness, I also compare these to XKTD estimates.

I evaluated both algorithms on a simple chain MDP (adapted from Brockman et al., 2016). The MDP has seven non-absorbing states, arranged linearly from state 0 to 6. Making a right move in the final state leads to an absorbing state. The agent can move to the left or right and receives a reward of -0.2 for every step except in the absorbing state, where it receives a reward of 10. The stochasticity in the state transitions will come from the policy, which can

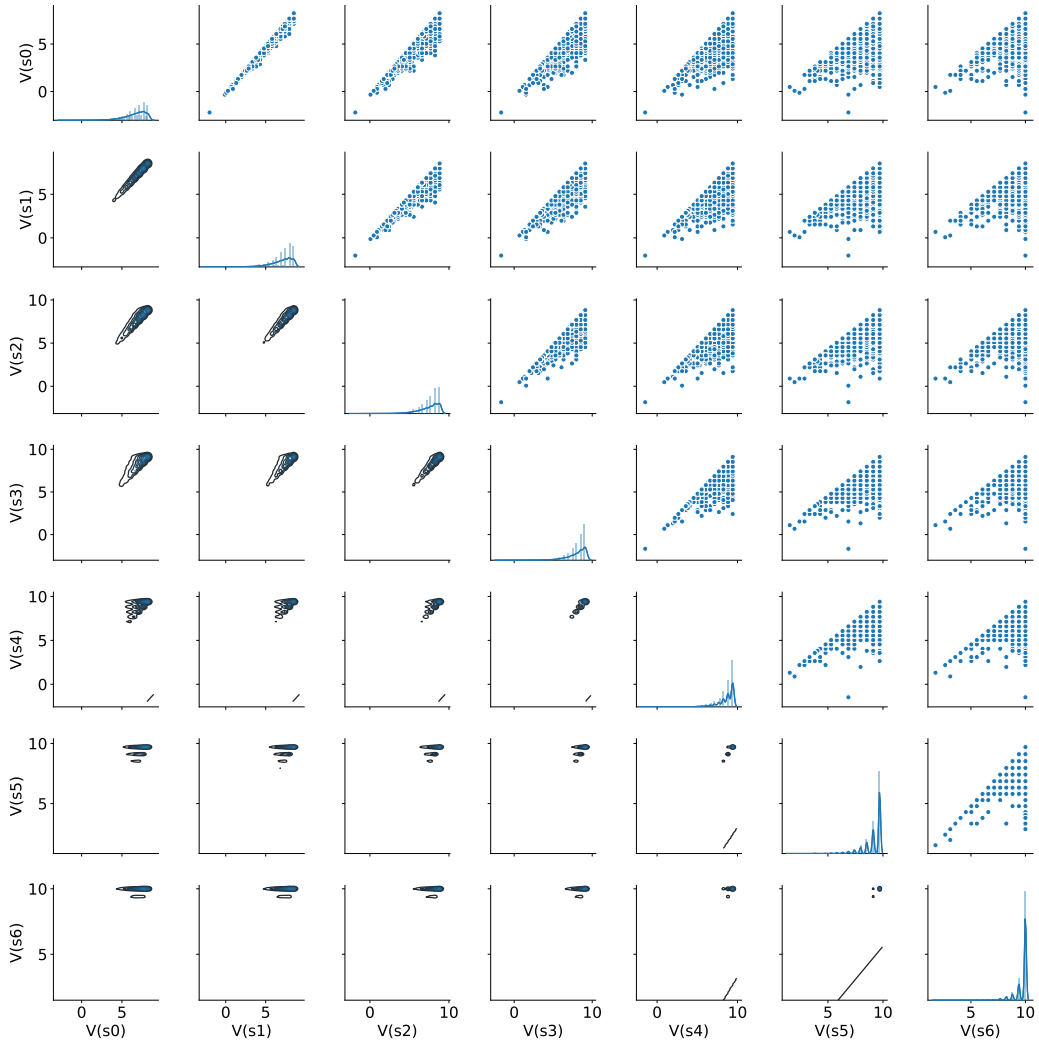


FIGURE B.1: Monte Carlo samples of return for all states on the linear track environment ($P(R) = 0.75$).

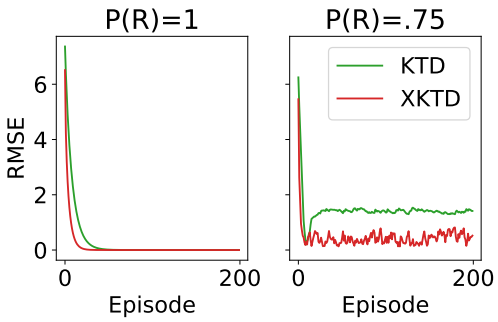


FIGURE B.2: Root mean square error after each episode for an example run of KTD and XKTD in a deterministic (left) and stochastic (right) MDP.

be defined by a single parameter $P(R)$, for the probability of making a step to the right ($P(L) = 1 - P(R)$).

Figure B.1 shows the histograms and kernel density estimates of the return for each state, sampled by Monte Carlo, when running this domain for 5000 episodes with $P(R) = 0.75$. As the number of state visits goes to infinity, the mean of these distributions corresponds to the true value for each state.

I then ran KTD and XKTD on this domain for 200 episodes, with a deterministic optimal policy ($P(R) = 1$) and with a stochastic policy ($P(R) = 0.75$), computing after each episode the root mean square error (RMSE) between the true value function (as estimated by MC) and the algorithm's value estimate (Figure B.2). With deterministic transitions, both algorithms converge to the same low error (left panel), but with stochastic transitions, KTD converges to a wrong value, maintaining higher error, consistent with the aforementioned bias described in B.2. Figure B.3 shows the posterior distribution over value after the example run of 200 episodes for both KTD and XKTD, overlaid with the actual, sampled returns. Indeed, the mean of the posterior for KTD is consistently off, while the XKTD posterior is closer to the true mean.

To quantify how damaging the deviations are as a function of stochasticity of the environment, I then varied $P(R)$ from 1 to 0.5 (completely random transitions), running KTD and XKTD for 200 episodes, repeated this 10 times for each value of $P(R)$ and computed the RMSE, which is shown in Figure B.4. As could be seen theoretically in section B.2, the bias grows as the environment is more stochastic. In addition, the bias is significantly reduced for XKTD, although even for the latter algorithm the bias grows with higher stochasticity.

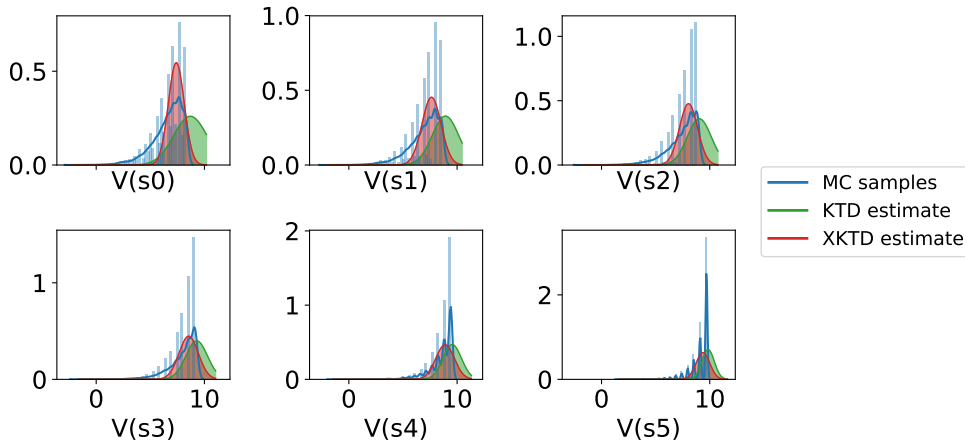


FIGURE B.3: Posterior distributions over value after an example run of 200 episodes for KTD and XKTD, overlaid with the true, sampled distribution of returns.

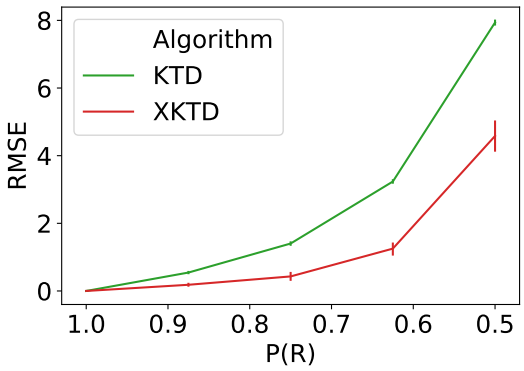


FIGURE B.4: Comparing KTD and XTKD on the linear track environment. RMSE after 200 episodes is plotted as a function of stochasticity of the environment ($P(R) = 0.5$ corresponds to maximum entropy / randomness). Error bars show 95% confidende intervals.

B.6 Conclusion

The Kalman TD algorithm incorrectly treats successive observation noise terms as "white" (independent from each other), while they are related because of the way the agent moves through the world. Any stochasticity in the transitions will therefore bring about a bias in the KTD estimates, as was shown here theoretically and empirically. This problem can be alleviated using XKTD, in which the hidden parameter vector is extended such that the observation noise, which is now assumed to be coloured, can be estimated online. However, as shown in Figure B.4, even XKTD leads to biased estimates under high stochasticity. This is because, while the assumptions are less strong than for KTD, XKTD still incorrectly assumes that the successive residuals are independent from each other (see section B.3).

In conclusion, KTD's uncertainty estimation is incorrect for many realistic MDPs, but the damage this does can be partially remedied by extending KTD with coloured noise estimation. Both these algorithms will only store parameters for a Gaussian posterior distribution, which will only be a rough approximation in most cases.