Cognitive graphs: Representational substrates for planning

Jungsun Yoo[a, *], **Elizabeth R. Chrastil**[a, b, c], **Aaron M. Bornstein**[a, b]
[a]Department of Cognitive Sciences, University of California, Irvine, Irvine, CA, USA, 92697
[b]Center for the Neurobiology of Learning and Memory, University of California, Irvine, CA, USA, 92697
[c]Department of Neurobiology & Behavior, University of California, Irvine, Irvine, CA, USA, 92697
[*]To whom correspondence should be addressed: jungsuy@uci.edu

Abstract. Making plans for upcoming actions is a computationally demanding process. To mitigate these demands, agents can build representations – of states, actions, and their sequential relationships – that allow for efficient execution of plans when needed. For example, "bootstrapped" representations can, after sufficient experience, efficiently capture these sequences, allowing them to be used for rapid planning even for novel or changing rewards. However, before such compact representations are available or reliable – for instance, in situations where sequential structure is unknown or changing – individuals may rely on less-compact representations of environmental contingencies. Here, we review recent work on the multitude of representations that can support planning. In particular, we discuss *cognitive graphs*, a framework with roots in both cognitive psychology and computer science. Conceptualizing representations of experience as forms of graphs places them on a continuum where different kinds of structured sequences can be used to support different kinds of planning-related behaviors. We discuss how each of these forms of cognitive graph are created during learning, and used to transfer, and generalize knowledge across environments. We close with a discussion of future directions for theoretical and empirical work.

Keywords: cognitive graph, planning, reinforcement learning.

1 Introduction

Planning is a common, and complex, form of decision-making. It requires both representing actions along with their precedents and consequences, and also sequencing them appropriately. The means by which planning occurs has been a focus of study in cognitive psychology (Solway et al. 2014; Opheusden et al. 2023) and of corresponding algorithmic development in computer science (Sutton and Barto 2018). In cognitive psychology, for example, identifying individual differences in the use of forward planning, rather than more reactive strategies (Chen et al. 2015; Sebold et al. 2017; Brown et al. 2020; Hunter et al. 2022), can be a potential behavioral marker for clinical intervention (C. M. Gillan et al. 2016). In the domain of computer science, one area of focus is the design of efficient autonomous agents that plan routes through complex environments such as warehouses (M. Liu et al. 2019; Li et al. 2021). Although the applications can diverge, planning in the two domains is grounded on the same, *representation-centric*, principle: agents should learn – via experience or instruction – an accurate model of the environment, and use the model strategically to execute actions. This common principle has allowed the two fields to be in fruitful dialogue (Neftci and Averbeck 2019): work in machine learning examines the algorithmic and normative properties of the necessary representations (Jiang et al. 2015), and work in cognitive psychology has examined how people use these representations to navigate different environments (Ho et al. 2022; Correa et al. 2023).

An implication of the representation-centric view of planning is that a key problem for agents to solve is how best to plan when there is little information available to build necessary world model – e.g. in early experience with new environments. One approach to this problem is to *transfer* learning from related environments, which requires first identifying similar situations, and then selecting the relevant aspects of that previously learned structure. When experience in related environments is extensive, allowing the agent to infer generalizable latent structure, one could apply compact, "map-like" representations that allow for efficient planning with minimal error (Geerts et al. 2022; Whittington, T. H. Muller, et al. 2020). However, as the overlap between well-learned settings and the current environment decreases, one must rely on more approximations, such as sampling instances from previous experiences with the current or similar environments (Zhao, Richie, and Bhatia 2022). Internal simulations informed by these sorts of instance samples can be used for iterative, vicarious evaluation of decision problems that not only informs the decision at hand, but allows the agent to accelerate the inference of more general latent structure representations (George et al. 2021). Both of these representations – instance-based samples, generative latent structures, and full-featured maps, as well as many intermediate forms – can be described using graph formalisms, with environmental states represented by nodes, and the transitions between them as various types of edges, depending on the information available (Chrastil and Warren 2014). Formalizing these structures as graphs allows researchers to connect seemingly disparate types of planning, and to reason about their related algorithmic and implementational properties (Zhang, Yang, and Stadie 2021) and how consolidation transfers information from one from one form to another (Feld et al. 2022).

In this review, we discuss the range of graph-like representations that support situations ranging between these two extremes, considering them as endpoints of a continuum of approaches to planning (Chrastil and Warren 2014, Figure 1). Recent work supports the co-existence of representations that support this multitude of approaches, distinguished by the content of the learned representations and also in what this content entails for the dynamics of their learning and use in deliberative decision-making (Doya et al. 2002; Bornstein and Nathaniel D Daw 2012; Bornstein and Nathaniel D Daw 2013; Smith and Graybiel 2013; Tambini et al. 2023). We review findings that suggest that these multiple graphs are learned simultaneously, that they influence behavior in accordance to their suitability to the task at hand, and that the apparent shift in behavioral control from one form to others is characterized by the transformation of information between representational formats, with attendant tradeoffs in function and fidelity.

2 Cognitive graphs

At their extremes, these graphs take forms that range from "uncompressed" to "compressed" (Figure 2). The most uncompressed form, in which states are encoded as experienced sequences with minimal latent structure inference, conceptually aligns with the notion of a "cognitive graph" (R. U.

Muller, Stead, and Pach 1996; Chrastil and Warren 2014; George et al. 2021). A cognitive graph can be characterized as a directed graph (R. U. Muller, Stead, and Pach 1996), with nodes representing states and edges indicating state transitions. These edges may be labeled, augmenting the topology with local metric information (Chrastil and Warren 2014; Warren 2019). They may also be weighted, reflecting the transition probability between states (Natarajan and Kolobov 2022; George et al. 2021; Sutton and Barto 2018). A cognitive graph is formed through learning how different sequences of state transitions connect at intersections, enabling agents to flexibly navigate conceptual and spatial networks by recombining the segments in novel ways (Warren 2019; Peer et al. 2021). Additionally, their abstract nature supports counterfactual simulations and generalizations to novel environments, thereby accelerating the learning process (G. Zhu et al. 2020). Though the entire continuum of representations are graph-structured, we will by default refer to this most uncompressed extreme form as "cognitive graphs", and the most compact representations as "bootstrapped cognitive graphs."

At the other end of the spectrum, bootstrapped representations contain information that is fully predictive of the N-step consequence of taking a given action a in the current state s (Figure 1, right). The transition from the uncompressed graph to the bootstrapped form is implemented by repeatedly sampling and aggregating features from neighboring nodes. This is analogous to message passing in graph convolutional networks (Hamilton, Ying, and Leskovec 2017) (Figure 3). At the beginning of the learning process, the cognitive graph resembles an undiffused graph where a node, or a given state, holds limited information about others, thus requiring the agent in a state to explicitly traverse edges to infer about other states. At the same time, uncompressed graphs provide full representations of the contingency structure between states and actions, which allow for flexibility at the cost of greater computation time and behavioral variability. With more experience, the cognitive graphs undergo a transformation into a bootstrapped representation where information about future states is aggregated into each adjacent state, making explicit edge-based inferences between states less important. Caching these distal outcomes subserves rapid planning, while still retaining sensitivity to changes in reward availability. However, without additional mechanisms, it also confers a relative insensitivity to contingency changes that may be undesirable in novel or volatile environments.

In this paper, we initially delve into the differences between planning predicated on the flexible recombination of action sequences and planning employing compressed representations. Subsequently, we propose cognitive graphs as a potential mechanism supporting decisions based on sampling instance sequences. This is supported by several functions that are offered by cognitive graphs, such as unifying disjoint sequences or distinctively representing seemingly same observations under different contexts. We conclude with a discussion of further research directions that could further explore this concept of a continuous representation, such as identifying a spectrum of control strategies that might emerge to the continuum of cognitive graphs. This, in turn, could

potentially clarify the differential use of types of control in different stages of learning.

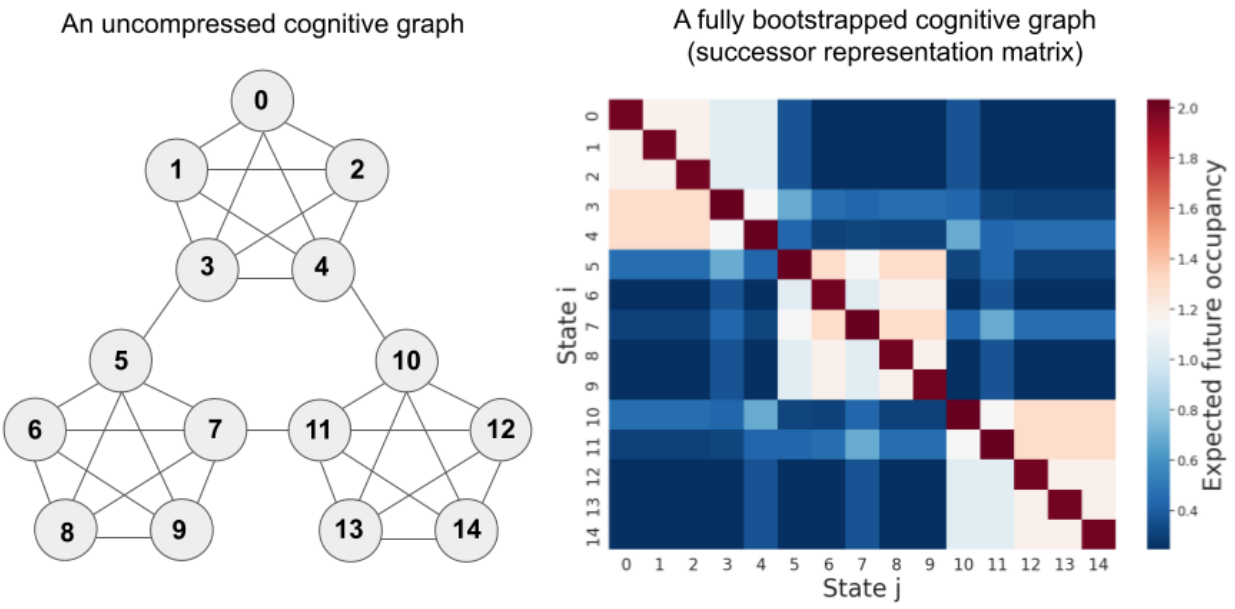

Figure 1: A graphical illustration of the two extremes of representations as a function of bootstrapping: an uncompressed cognitive graph (left) and a fully bootstrapped cognitive graph ("successor representation") (right). **left** Numbers indicate labels of a node, or a discrete state, in a cognitive graph. Edges between two nodes depict the transition between two states. **right** A bootstrapped version of a cognitive graph that fully captures future trajectories from a given node (state). Row identification numbers (ID; i) indicate the current state, and column IDs (j) indicate the successor state. The values in the matrix represent the expected future occupancy of j from i, and are color-coded for visualization.

3 Planning as a function of representational compactness

3.1 Planning in a Markov Decision Process

For simplicity, planning is often conceptualized within the context of a Markov Decision Process (MDP). In a classical MDP, the environment in which an agent plans is characterized as a tuple of $< S, A, T, R, \pi >$ where S is a finite and discrete state space that is comprised of states, and A is a set of actions that can be executed in each state $s \in S$. The models consist of two functions, where $T(s, a, s\prime)$ is the transition function for each $s \in S$ and $a \in A$, and $R(s, a, s\prime)$ is a reward function that provides the immediate reward or value obtained after taking action a in state s and transitioning to state $s\prime$. π refers to the policy that determines which action leads to largest value in a given state. We assume that an agent starts from an initial state s_0 and executes a sequence of possible actions in the successor states ($s\prime$) up to a terminal or goal state $s \in S_G$. The agent's goal in planning is to learn and execute actions based on an optimal solution, or policy, that maximizes the cumulative value from an initial state to a goal state.

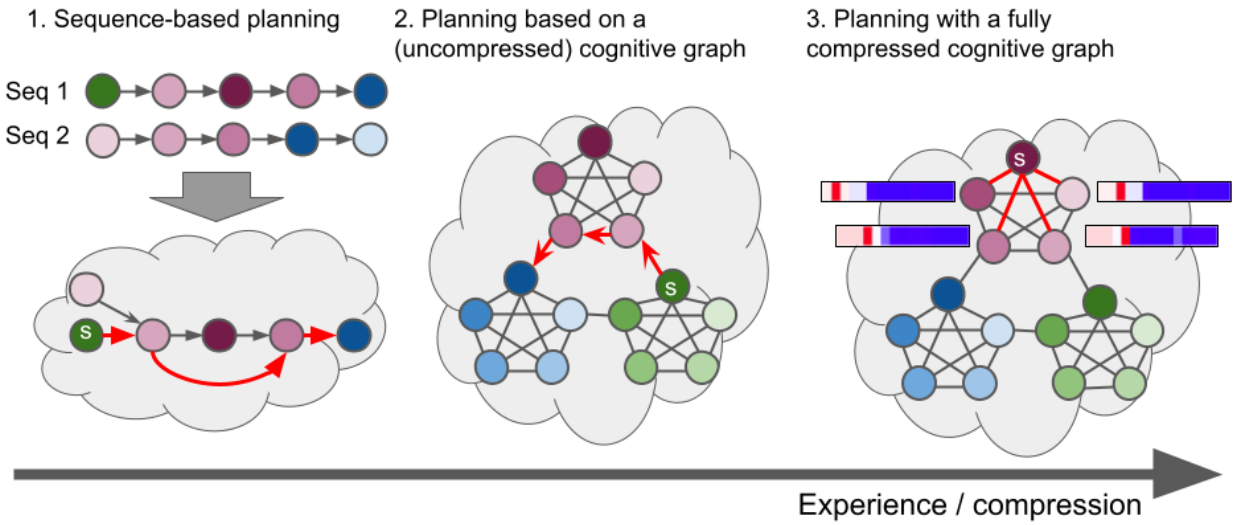

Figure 2: A graphical illustration of the suggested continuum of the representations – specifically, the degree to which they are pre-compiled – as a function of experience. Nodes indicated with 's' represent starting points. Arrows, or directed edges, describe possible plans for the agent in the starting node to execute. The grey clouds represent the form of model the agent is using to plan in a given phase. **1. Sequence-based planning** This phase represents the early learning phase where an agent has not yet constructed a model of the environmental dynamics. Here, agents are assumed to plan based on sampling instances previously experienced trajectories. First, two sample trajectories are shown here, labeled 'seq 1' and 'seq 2.' From these two trajectories, an agent is able to create a combined representation and plan efficiently with it (e.g., taking the shortcut as seen in the red arrows). **2. Planning based on a (uncompressed cognitive graph)** After a few experiences, an agent is able to build cognitive graphs by conjoining past trajectories. Agents are thought to iteratively sample next actions based on the cognitive graph. As agents are gaining more experience, simultaneously, a 'diffusion-like' process is thought to take place such that information about neighboring nodes are integrated into each node. **3. Planning with a fully compressed cognitive graph** After sufficient experience, a compressed trajectory from a given node to each other node is available in a summary format. Agents are able to plan rapidly using these compressed representations.

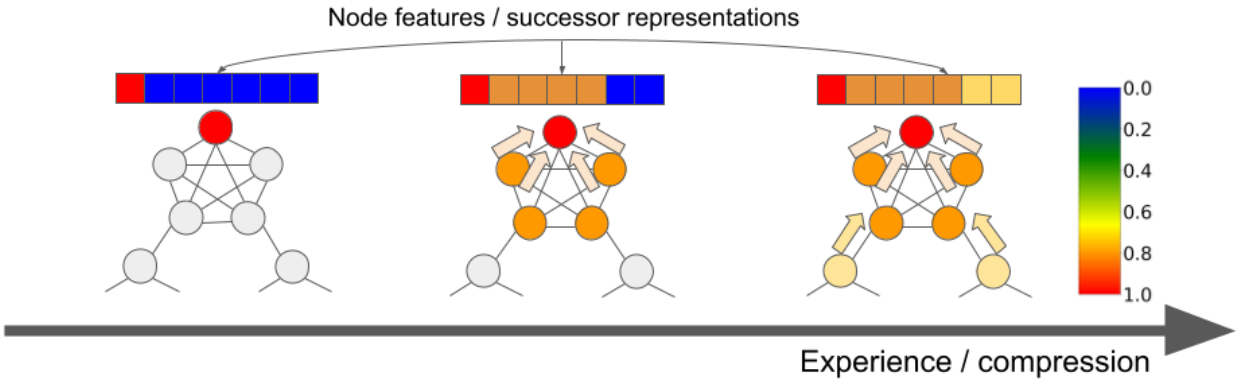

Figure 3: A graphical illustration of the analogy between representation bootstrapping and graph convolutional networks. Both of the algorithms involve propagation of information across neighboring nodes, and update the "features" of the node using the aggregated information. In light of successor representations, the features of the nodes correspond to the expected future occupancy of successor states derived by compressing possible trajectories. **left** At the initial learning phase, no information about the neighboring nodes (or, in terms of successor representations, future states) is reflected in the red node's feature. **middle** With more experience, the trajectories starting from the red is aggregated to the node's feature. Thus, information about future states, or bootstrapped trajectories, is being developed. **right** This describes the phase where extensive experiences have been occurred, so that the learning of the node features has converged. In this phase, the node has full information about possible future trajectories – or, in other words, fully bootstrapped sequences.

One of the most crucial components for successful planning is having an accurate internal model of the environment, because the model is used for simulating or predicting behavior; inaccurate models could entail incorrect predictions and thereby result in a chain of sub-optimal actions (Talvitie 2017). It is also important to adopt the most suitable models for each specific context, given that optimal type of model to use may vary depending on the relationship of the model to the environment – for instance, whether the model is known with certainty to correspond exactly to the environment (Jiang et al. 2015). Below, we delve into the kind of model utilization that may be optimal in circumstances where agents possess sufficient experience for bootstrapping to occur (Section *Planning with bootstrapped state/action sequences*), or in scenarios where agents are still in the preliminary phases of environment interaction (Section *Planning with uncompressed representations: sampling instances*).

3.2 Planning with learned cognitive graphs: uncompressed vs. bootstrapped

Models in planning capture statistical regularities of the environment, and could be either given *a priori* or learned from experience. If an agent has full information about the transition structure of the environment, then the agent is able to plan even without experience. This is conceptually relevant to a classical control problem or search algorithm (Korf 1987): for example, for the game Tic-Tac-Toe, an agent can be endowed with a complete model (or a human can be verbally in-

structed about the rules of the game). Given this starting point, the player can construct a tree-like graph of possible future states and actions, and perform search to find the optimal decision (Sriram et al. 2009). However, in more naturalistic contexts, the dynamics of the environment are unknown to us initially and our internal models develop and change with our experience with the environment (Schrittwieser et al. 2020; Lengyel and Dayan 2007). We confine further discussion to these latter, learned models of the environment.

Cognitive graphs at different levels of compression could serve as models that can support different forms of planning. Raw, uncompressed cognitive graphs support planning via iterative sampling of subsequent states or actions from a given state, or node. Here, individual instances or nodes have minimal information about other nodes, thus making it crucial to traverse graphs based on the relationship between nodes, or edges. Thus, in this form of planning, the sampling algorithm is critical.

At the other extreme, actions and states in a cognitive graph are fully bootstrapped – these are referred to as successor or predecessor representations. In the successor representation, each node-state contains the expected future state occupancy given a current state and according to a given policy (Dayan 1993); these can be thought of as integrated trajectories sampled from the current state. Conversely, predecessor representations can be thought of as fully bootstrapped versions of eligibility traces, a memory-like mechanism that assigns credit to past states and actions from a given state (Bailey and Mattar 2022; Sutton and Barto 2018). Predecessor representations could be seen as a hindsight version of successor representations in that it bootstraps the possible trajectories *that could have lead to* a current state. Whether directed forward or backward, once these bootstrapped representations converge, the expected cumulative reward or credit can be efficiently computed for planning: just taking the product of the representation and a separate reward function. With fully bootstrapped cognitive graphs, the need for edge-based sequences, or sampling next states, becomes negligible.

3.2.1 Planning with uncompressed representations: sampling instances

During early stages of learning an environment, planning could be facilitated by instance-based methods instead of relying on an explicit model. Take, for example, the task of choosing a restaurant to dine in an area that one has just moved to and thus has limited experience with. It might be more effective to decide based on a few recent dining experiences rather than attempting to build and decide based on a general summary of what little experience one has. This non-parametric, kernel-based method offers several cognitive advantages that could potentially bolster planning efficiency. For instance, a small group of samples reduces memory load (G. Barron and Erev 2003), simplifies the decision rule (Fiedler 2000; Hertwig and Pleskac 2010), facilitates generalization to unseen observations (Wimmer and Shohamy 2012; H. C. Barron, Dolan, and Timothy EJ Behrens 2013), and reduces time (Hertwig, G. Barron, et al. 2004; Fox and Hadar 2006).

Sampling instances at decision time, rather than maintaining a representation of decision variables, has been shown to explain human decisions (Zhao, Richie, and Bhatia 2022; Bornstein, Khaw, et al. 2017; Wang, Feng, and Bornstein 2022). Specifically, in a repeated decision task, individuals appear to rely on small numbers of samples of recent experiences. When intermittent reminder probes were added to the task, manipulating the recency of past experiences, these probes had a significant effect on subsequent choice (Bornstein, Khaw, et al. 2017). The selection of which instances to sample can depend on the demands of the task: recent items may be more likely to be sampled in an environment that does not have an explicit structure, but different dimensions of environmental structure lead to sampling a sequence of events that are most similar to the decision at hand (Plonsky, Teodorescu, and Erev 2015).

Episodic control, a strategy in reinforcement learning (RL) where an agent decides based on specific past instances ("episodes") rather than building a generalized model of the environment, applies this idea of memory-based RL to artificial agents to facilitate rapid learning and generalization to new observations (Lengyel and Dayan 2007; Blundell et al. 2016). Episodic control enables agents to make an informed guess about the value of unseen states by averaging the values of the stored past instances that are most similar to the current state. This improves sample efficiency since the same amount of observations could be used to inform estimates about a greater number of states, compared to classic RL. Simulation results show that the advantage of episodic control is evident during initial learning stages, as it accelerates the learning process (Lengyel and Dayan 2007; Blundell et al. 2016). Meanwhile, concurrently, a model-based system improves by learning the full structure of the environment, eventually superseding the episodic system.

One drawback of episodic control is that the search process could be inefficient as the number of episodes stored increases. For scalability, neural episodic control (NEC) uses deep learning methods to embed the keys of each state into a fixed-length vector (Pritzel et al. 2017). Embedded inputs are then fed into a differentiable neural dictionary, or a learnable episodic memory system that maps keys to values. The final value of an observation is obtained by the weighted sum of the values in the differentiable neural dictionary, where the weights are computed by the similarity between the current key and the keys of states in the memory system.

In addition to individual samples of instances, leveraging the *sequences* of samples could facilitate flexible planning. In realistically large and highly uncertain environments, creating fully bootstrapped representations will take a long time, and may indeed be infeasible. Silver and Veness (2010) showed that it is possible to achieve asympotically strong performance in this situation using Monte-Carlo trajectory sampling of transitions from a generative model (Silver and Veness 2010). Another advantage of representing the full trajectory, with all its intermediate instances, is that one can perform 'far' jumps across state space with some probability that changes according to the needs of the task or environment (J. Zhu, Sanborn, and Chater 2018); the resulting "Lévy flight" behavior matches observations of biological agents exploring novel environments (Hunt et

al. 2021).

3.2.2 Planning with bootstrapped state/action sequences

Earlier, we introduced the concept of bootstrapped representations as a way of incrementally learning compact summaries of multi-step contingencies. Successor representations have been devised to balance the possible computational intractability of fully model-based methods and inflexibility of computationally cheap model-free methods, providing a robust solution to this problem (Dayan 1993). A successor representation is also an example of boostrapped representations, since it summarizes expected future occupancy of successor states from a current state-action given a policy. Using successor representations compresses the multi-step planning process into a single-step process, since timescales of all possible future trajectories are considered at once (Dayan 1993). This not only reduces computational complexity, but it also facilitates generalization and learning when adapting to variable reward contingencies. Empirical evidence from studies conducted on humans (Momennejad et al. 2017) and artificial agents (Barreto et al. 2017) suggests that using transition dynamics compressed in a successor representation lead to faster adaptation to value-function changes, because only the reward function requires re-learning, thus significantly enhancing learning efficiency.

Another example of compressing sequences of observation, or states, is seen in robust predictable control (RPC; (Eysenbach, Salakhutdinov, and Levine 2021). This algorithm is explicitly encouraged to find a compressed policy by penalizing the complexity of information bits, which is operationally defined as the amount of information needed from observations for a policy to make decisions. The intuition behind this is that agents will rely less on gathering information from observations as they become better at predicting the future accurately. Agents trained on compressed policies are less susceptible to unknown or missing observations (i.e., perturbations), since compressed policies have been trained to use less bits of information per observations. This leads to improved *open-loop control* – producing a plan of action sequences at the beginning and executing it without checking the progress along the way.

In sum, compressed representations lowers the cost of planning by reducing complexity at the representational level. As seen in SR and RPC, this kind of representation also fosters open-loop planning by enabling the execution of actions sequences as a single operation (Eysenbach, Salakhutdinov, and Levine 2021). This could be efficient in environments where transition dynamics are relatively well-known and unchanging. On the other hand, when models of the environment has not been fully developed yet, instance-based control can be useful. In particular, sampling trajectories of instances to preserve the sequential nature of experiences provides a method with less complexity and greater scalability, while still maintaining high performance. In the following section, we discuss how graph-structured representations can improve trajectory-based planning.

4 Ways in which uncompressed cognitive graphs could facilitate planning

Graph-structured models improve planning in software agents. While planning based on sequences of actions has been successful for various domains, it comes at a cost: as the planning horizon of the agents gets longer, the outcomes start to stray away from the original observation. This has been tackled by transforming planning into a graph-search problem (Savinov, Dosovitskiy, and Koltun 2018; K. Liu et al. 2020). One study leveraged graph-based representations to identify landmarks or subgoals in latent graphs, and then performed graph search on the nodes (Zhang, Yang, and Stadie 2021). Here, edges between the nodes are weighted with "reachability" between nodes, making it as a form of a labeled graph. In the domain of spatial navigation, algorithms construct graphs based on subgoals and then plan based on the constructed graphs for efficiency (Bagaria, Senthil, and Konidaris 2021).

It has also been found that people spontaneously construct graph-like representations when observing a sequence of events, where these latent graphs could be either correlational (Rmus et al. 2022; Solomon et al. 2019, undirected graphs) or causal (Gopnik and Schulz 2004; Gopnik, Glymour, et al. 2004; Sommerville and Woodward 2005a; Sommerville and Woodward 2005b, directed graphs). Intriguingly, the general tendency to use plans over model-free approaches appears to be correlated with the ability to infer latent graph-based structure from jumbled sequences of experiences (Rmus et al. 2022), potentially underscoring the utility of graphs in planning.

4.1 Mechanisms by which cognitive graphs could facilitate planning

States and observations or instances may not be mapped onto each other in a one-to-one fashion. This phenomenon, referred to as perceptual aliasing, could potentially destabilize control in RL (Whitehead and Ballard 1991). To overcome this, agents must employ an accurate and parsimonious representation of experience that is able to split identical observations into different underlying states or merge seemingly different observations into a single state for generalization, depending on the context (Niv 2019).

The hippocampus provides empirical solutions to these two challenges: attractor dynamics (Knierim and Neunuebel 2016) and pattern separation (Yassa and Stark 2011). Given the hippocampus's role in representing cognitive graphs (R. U. Muller, Stead, and Pach 1996; K. L. Stachenfeld, Botvinick, and Gershman 2017), it implies that graphs could serve as a beneficial tool. Not surprisingly, a graph structure allows individual states to be identified more exactly by their multiple connected nodes, rather than considering observations independently (Whittington, McCaffary, et al. 2022). Also, in a variant of the cognitive graph – clone-structured cognitive graphs (CSCG) – that utilizes Bayesian inference to derive latent states from a sequence of observations, 'clones' of observations are created when a new context or latent state is encountered (George et al. 2021). This provides a solution for both merging and splitting aliased observations.

Below, we describe how cognitive graphs support both merging and splitting, and what specific mechanisms an unfolded graph could provide to facilitate early stage learning.

4.1.1 Merging: fast generalization by extrapolating trajectories

Associative memory could be seen as the building block of cognitive graphs. One such instantiation is transitive inference, which is an example of leveraging relational information of instances for faster generalization, observed in humans and animals (Bryant and Trabasso 1971; D. J. Gillan 1981; Davis 1992). When an agent experiences $A > B$ and $B > C$, the unobserved relationship between $A > C$ can be inferred without direct experience (Eichenbaum, Dudchenko, et al. 1999). This can be achieved through forming supraordinate representations, comparable to cognitive graphs, such that $A > B > C$, which has been found to be supported by the hippocampus (Greene et al. 2006; Dusek and Eichenbaum 1997; Zalesak and Heckers 2009). Similarly, disparate fragments of event trajectories can be fused together, creating graph-like formations by leveraging the intersections of these trajectories (Eichenbaum and Cohen 2014; Rmus et al. 2022). From these graphs, inferences can be made between instances that were not directly experienced together, supporting flexible recombination and fast generalization (Eichenbaum 2004).

Implementing this associative-memory-based cognitive graphs leads to efficient planning algorithms. For example, an episodic reinforcement learning algorithm called Episodic Reinforcement Learning with Associative Memory (ERLAM) augmented with associative-memory showed increased sample efficiency compared to benchmarks (G. Zhu et al. 2020). Here, trajectories of episodic memory are reorganized into graphs, which speeds the propagation of value learned from one instance to other related instances, thereby enhancing sample efficiency. In addition, CSCG introduced earlier have been shown to be capable of performing transitive inferences (George et al. 2021). Here, two separate rooms are stitched together to form an overlapping regions. Agents navigate each rooms separately and are tested whether they can travel from an non-overlapping region of one room to the region exclusive to other room. Results show that agents are able to construct a latent map by stitching sequential observations from two disjoint episodes; overlapping observations from different trajectories are correctly assigned to the same hidden state.

In addition to conjoining separate sequences, associative memory binds seemingly independent choice options together into a temporal context, so that learning the value of a chosen option also influences the value of unchosen options (Biderman and Shohamy 2021). This is referred to as counterfactual reasoning, another example of associative memory accelerating learning since information about an instance can be propagated to related experiences. Counterfactual reasoning is observed in RL: humans not only deploy "factual" information through direct trial-and-error, but also incorporate counterfactual learning (Boorman, Timothy E Behrens, and Rushworth 2011; Fischer and Ullsperger 2013), where greater learning rates for positive-factual and negative-counterfactual options are consistent with well-established confirmation bias in humans (Palminteri et al. 2017).

Associative memory strength between options in RL is correlated with how much learning about one option influences other unchosen options (Biderman, Gershman, and Shohamy 2023), implying that counterfactual learning in the RL context could also be based on cognitive graphs.

ERLAM provides an example of leveraging counterfactual combinatorial trajectories to facilitate learning (G. Zhu et al. 2020). Trajectories are reorganized into graphs by merging the common elements of the two trajectories, and it intuitively follows that when experiencing two intersecting trajectories that each lead to reward and no-reward, an agent that only relies on episodic RL would take longer than the agent that uses counterfactual thinking based on the graphs. Recently, *expected eligibility traces* have been introduced as a form of leveraging counterfactual trajectories to accelerate learning (Hasselt et al. 2021). Eligibility trace is a mechanism in RL that provides a hindsight credit assignment with regard to the current state by keeping a trace of past experiences weighted by their recency (Singh and Sutton 1996; Sutton and Barto 2018). *Expected* eligibility traces improves the limitation of eligibility traces – that only one directly experienced trace is updated each time – by considering multiple counterfactual sequences that could have preceded a current state.

4.1.2 Splitting: Recovering latent structure from aliased sequences

It is possible that two different states are mapped onto a same observation, which is referred to as aliased states. In this situation, as opposed to the example above that agents should be able to create a graph that *merges* two sequences – $A > D > C$ and $B > D > E$ – , an agent should be able to *split* D into two different nodes according to their contexts. CSCG are able to accurately reconstruct correct latent graphs from sequences of aliased sensory observations by making clones of observations (George et al. 2021). Impressively, CSCG is able to both split aliased observation into latent states, but is able to merge the reconstructed graphs as in transitive inference.

Indeed, as implied above, these are exactly the sorts of environments in which CSCG have an advantage over bootstrapped representations. Specifically, when presenting a CSCG agent with a sequence of aliased observations from a graph with community structure (e.g. Figure 1left), the agent is able to recover the modular structure. However, an agent that used a successor representation was not able to recover this structure (George et al. 2021). This suggests that environments in which modular structure is important to the task at hand benefit from having available less-compressed representations of experience. This idea aligns with the finding that sequences of observations generated by a modular vs. lattice graph lead to more robust latent representations (Kahn et al. 2023).

5 Discussion

Multi-step planning is a critical ability for autonomous organisms. Extensive research has identified multiple routes for performing multi-step planning. These distinct approaches rely on different

representational substrates, and have different algorithmic commitments, leading to the inference that they are supported by wholly distinct biological implementations. Here, we suggest that there is a common basis for these multiple planning systems, which can be expressed using graph formalisms that have been fleshed out over multiple empirical and computational investigations.

We showed that the initial phase of representation learning puts substantial importance on sequences of instances. This is because representations of latent states are uniquely characterized by the systematic ordering of observations (Whittington, McCaffary, et al. 2022), and leveraging the sequential structures of experiences yields faster learning and improved performance (Hansen et al. 2018). Adding branching structure to these sequences, as in an uncompressed cognitive graph, allows for sampling trajectories during planning in a manner that reflects the statistics of experience, while also permitting flexibility critical for exploration (J. Zhu, Sanborn, and Chater 2018). This form of representation also allows for significant alterations based on experience, which can dramatically alter subsequent decisions (G. Zhu et al. 2020), and at the same time can allow generalizing from repeated experiences that share latent features (George et al. 2021). Together, these features dramatically reduce the concern about sample efficiency raised about earlier implementations of instance-based reasoning approaches.

At the other end of the continuum, a compressed graph such as the bootstrapped successor and predecessor representation allow for significantly faster planning in environments where the transition structure is well-learned and unlikely to change, while still remaining sensitive to changes in value functions (Figure 2). We proposed that this continuum is mediated by experience and implemented as compression. This proposal reflects both the empirical observation that humans and animals tend to prioritize precision at the expense of computational resources, at first, but then favor computational efficiency with greater experience and stability (Nathaniel D Daw, Niv, and Dayan 2005; Smith and Graybiel 2013; Lengyel and Dayan 2007).

One interesting direction to expand this concept of representational continuum is testing whether different modes of control arise as a function of the degree of compression (Moskovitz et al. 2022). Recall that agents using compressed policies are adept at open-loop control – they cast action sequences into a single operation. This eliminates the need for intermittent re-planning during action execution, due to correctly learned compressed future states (Eysenbach, Salakhutdinov, and Levine 2021). However, if agents plan by sequentially sampling next actions using uncompressed cognitive graphs, taking small steps of edges would be more efficient than open-loop control, since the model has not been compiled yet to provide reliable future trajectories from a state. This edge-based planning is conceptually more similar to closed-loop planning, where an agent stops at each transition to re-plan. This leads us to the overarching question of whether the utility of using closed-loop vs. open-loop planning aligns with the degree of compression in cognitive graphs. This alignment would be similar to the evolution of an episodic control system to model-based, and then finally to model-free systems (Lengyel and Dayan 2007). Based on an interpretation that

the seemingly model-free behaviors could actually be action sequences (Dezfouli and Balleine 2012; Dezfouli and Balleine 2013), we could hypothesize that the model-free system at the end of the spectrum could be in fact representing action sequences formed by open loop control, likely a result of using highly compressed models. Empirical support for this hypothesis would extend the idea of continuum of representations to encompass the continuum of control methods.

6 Conclusions

To conclude, we highlight the kind of representations that could be used to support instance-based planning at early stages of learning – cognitive graphs – and suggest that they could be in a continuum, rather than discrete concepts, with bootstrapped representations at the other end. Further research may link this representational-centric continuum to a continuum of closed versus open-loop control.

7 Acknowledgements

The authors would like to thank Dr. Bruce McNaughton and Dr. Jeffrey L. Krichmar for insightful discussions. Funding was provided by NIA R21AG072673 (to AMB), and NINDS R01NS119468 (to ERC).

8 Statements and Declarations

The authors declare no competing interest.

References

Bagaria, Akhil, Jason K Senthil, and George Konidaris (2021). "Skill discovery for exploration and planning using deep skill graphs". In: *International Conference on Machine Learning*. PMLR, pp. 521–531.

Bailey, Duncan and Marcelo Mattar (2022). "Predecessor Features". In: *arXiv preprint arXiv:2206.00303*.

Barreto, André et al. (2017). "Successor features for transfer in reinforcement learning". In: *Advances in neural information processing systems* 30.

Barron, Greg and Ido Erev (2003). "Small feedback-based decisions and their limited correspondence to description-based decisions". In: *Journal of behavioral decision making* 16.3, pp. 215–233.

Barron, Helen C, Raymond J Dolan, and Timothy EJ Behrens (2013). "Online evaluation of novel choices by simultaneous representation of multiple memories". In: *Nature neuroscience* 16.10, pp. 1492–1498.

Biderman, Natalie, Samuel J Gershman, and Daphna Shohamy (2023). "The role of memory in counterfactual valuation." In: *Journal of Experimental Psychology: General* 152.6, p. 1754.

Biderman, Natalie and Daphna Shohamy (2021). "Memory and decision making interact to shape the value of unchosen options". In: *Nature communications* 12.1, p. 4648.

Blundell, Charles et al. (2016). "Model-free episodic control". In: *arXiv preprint arXiv:1606.04460*.

Boorman, Erie D, Timothy E Behrens, and Matthew F Rushworth (2011). "Counterfactual choice and learning in a neural network centered on human lateral frontopolar cortex". In: *PLoS biology* 9.6, e1001093.

Bornstein, Aaron M and Nathaniel D Daw (2012). "Dissociating hippocampal and striatal contributions to sequential prediction learning". In: *European Journal of Neuroscience* 35.7, pp. 1011–1023.

— (2013). "Cortical and hippocampal correlates of deliberation during model-based decisions for rewards in humans". In: *PLoS computational biology* 9.12, e1003387.

Bornstein, Aaron M, Mel W Khaw, et al. (2017). "Reminders of past choices bias decisions for reward in humans". In: *Nature Communications* 8.1, pp. 1–9.

Brown, Vanessa M et al. (2020). "Improving the reliability of computational analyses: Model-based planning and its relationship with compulsivity". In: *Biological Psychiatry: Cognitive Neuroscience and Neuroimaging* 5.6, pp. 601–609.

Bryant, Peter E and Thomas Trabasso (1971). "Transitive inferences and memory in young children". In: *Nature* 232, pp. 456–458.

Chen, Chong et al. (2015). "Reinforcement learning in depression: A review of computational research". In: *Neuroscience & Biobehavioral Reviews* 55, pp. 247–267.

Chrastil, Elizabeth R and William H Warren (2014). "From cognitive maps to cognitive graphs". In: *PloS one* 9.11, e112544.

Correa, Carlos G et al. (2023). "Humans decompose tasks by trading off utility and computational cost". In: *PLOS Computational Biology* 19.6, e1011087.

Davis, Hank (1992). "Transitive inference in rats (Rattus norvegicus)." In: *Journal of Comparative Psychology* 106.4, p. 342.

Daw, Nathaniel D, Yael Niv, and Peter Dayan (2005). "Uncertainty-based competition between prefrontal and dorsolateral striatal systems for behavioral control". In: *Nature neuroscience* 8.12, pp. 1704–1711.

Dayan, Peter (1993). "Improving generalization for temporal difference learning: The successor representation". In: *Neural computation* 5.4, pp. 613–624.

Dezfouli, Amir and Bernard W Balleine (2012). "Habits, action sequences and reinforcement learning". In: *European Journal of Neuroscience* 35.7, pp. 1036–1051.

— (2013). "Actions, action sequences and habits: evidence that goal-directed and habitual action control are hierarchically organized". In: *PLoS computational biology* 9.12, e1003364.

Doya, Kenji et al. (2002). "Multiple model-based reinforcement learning". In: *Neural computation* 14.6, pp. 1347–1369.

Dusek, Jeffery A and Howard Eichenbaum (1997). "The hippocampus and memory for orderly stimulus relations". In: *Proceedings of the National Academy of Sciences* 94.13, pp. 7109–7114.

Eichenbaum, Howard (2004). "Hippocampus: cognitive processes and neural representations that underlie declarative memory". In: *Neuron* 44.1, pp. 109–120.

Eichenbaum, Howard and Neal J Cohen (2014). "Can we reconcile the declarative memory and spatial navigation views on hippocampal function?" In: *Neuron* 83.4, pp. 764–770.

Eichenbaum, Howard, Paul Dudchenko, et al. (1999). "The hippocampus, memory, and place cells: is it spatial memory or a memory space?" In: *Neuron* 23.2, pp. 209–226.

Eysenbach, Ben, Russ R Salakhutdinov, and Sergey Levine (2021). "Robust predictable control". In: *Advances in Neural Information Processing Systems* 34, pp. 27813–27825.

Feld, GB et al. (2022). "Sleep targets highly connected global and local nodes to aid consolidation of learned graph networks". In: *Scientific Reports* 12.1, p. 15086.

Fiedler, Klaus (2000). "Beware of samples! A cognitive-ecological sampling approach to judgment biases." In: *Psychological review* 107.4, p. 659.

Fischer, Adrian G and Markus Ullsperger (2013). "Real and fictive outcomes are processed differently but converge on a common adaptive mechanism". In: *Neuron* 79.6, pp. 1243–1255.

Fox, Craig R and Liat Hadar (2006). ""Decisions from experience"= sampling error+ prospect theory: Reconsidering Hertwig, Barron, Weber & Erev (2004)". In: *Judgment and Decision Making* 1.2, pp. 159–161.

Geerts, Jesse P et al. (2022). "A probabilistic successor representation for context-dependent prediction". In: *bioRxiv*.

George, Dileep et al. (2021). "Clone-structured graph representations enable flexible learning and vicarious evaluation of cognitive maps". In: *Nature communications* 12.1, p. 2392.

Gillan, Claire M et al. (2016). "Characterizing a psychiatric symptom dimension related to deficits in goal-directed control". In: *elife* 5, e11305.

Gillan, Douglas J (1981). "Reasoning in the chimpanzee: II. Transitive inference." In: *Journal of Experimental Psychology: Animal Behavior Processes* 7.2, p. 150.

Gopnik, Alison, Clark Glymour, et al. (2004). "A theory of causal learning in children: causal maps and Bayes nets." In: *Psychological review* 111.1, p. 3.

Gopnik, Alison and Laura Schulz (2004). "Mechanisms of theory formation in young children". In: *Trends in cognitive sciences* 8.8, pp. 371–377.

Greene, Anthony J et al. (2006). "An FMRI analysis of the human hippocampus: inference, context, and task awareness". In: *Journal of cognitive neuroscience* 18.7, pp. 1156–1173.

Hamilton, Will, Zhitao Ying, and Jure Leskovec (2017). "Inductive representation learning on large graphs". In: *Advances in neural information processing systems* 30.

Hansen, Steven et al. (2018). "Fast deep reinforcement learning using online adjustments from the past". In: *Advances in Neural Information Processing Systems* 31.

Hasselt, Hado van et al. (2021). "Expected eligibility traces". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 35. 11, pp. 9997–10005.

Hertwig, Ralph, Greg Barron, et al. (2004). "Decisions from experience and the effect of rare events in risky choice". In: *Psychological science* 15.8, pp. 534–539.

Hertwig, Ralph and Timothy J Pleskac (2010). "Decisions from experience: Why small samples?" In: *Cognition* 115.2, pp. 225–237.

Ho, Mark K et al. (2022). "People construct simplified mental representations to plan". In: *Nature* 606.7912, pp. 129–136.

Hunt, LT et al. (2021). "Formalizing planning and information search in naturalistic decision-making". In: *Nature neuroscience* 24.8, pp. 1051–1064.

Hunter, Lindsay E et al. (2022). "Increased and biased deliberation in social anxiety". In: *Nature Human Behaviour* 6.1, pp. 146–154.

Jiang, Nan et al. (2015). "The dependence of effective planning horizon on model accuracy". In: *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*, pp. 1181–1189.

Kahn, Ari E et al. (2023). "Network structure influences the strength of learned neural representations". In: *bioRxiv*, pp. 2023–01.

Knierim, James J and Joshua P Neunuebel (2016). "Tracking the flow of hippocampal computation: Pattern separation, pattern completion, and attractor dynamics". In: *Neurobiology of learning and memory* 129, pp. 38–49.

Korf, Richard E (1987). "Planning as search: A quantitative approach". In: *Artificial intelligence* 33.1, pp. 65–88.

Lengyel, Máté and Peter Dayan (2007). "Hippocampal contributions to control: the third way". In: *Advances in neural information processing systems* 20, pp. 889–896.

Li, Jiaoyang et al. (2021). "Lifelong multi-agent path finding in large-scale warehouses". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 35. 13, pp. 11272–11281.

Liu, Kara et al. (2020). "Hallucinative topological memory for zero-shot visual planning". In: *International Conference on Machine Learning*. PMLR, pp. 6259–6270.

Liu, Minghua et al. (2019). "Task and path planning for multi-agent pickup and delivery". In: *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*.

Momennejad, Ida et al. (2017). "The successor representation in human reinforcement learning". In: *Nature human behaviour* 1.9, pp. 680–692.

Moskovitz, Ted et al. (2022). "A unified theory of dual-process control". In: *arXiv preprint arXiv:2211.07036*.

Muller, Robert U, Matt Stead, and Janos Pach (1996). "The hippocampus as a cognitive graph." In: *The Journal of general physiology* 107.6, pp. 663–694.

Natarajan, Mausam and Andrey Kolobov (2022). *Planning with Markov decision processes: An AI perspective*. Springer Nature.

Neftci, Emre O and Bruno B Averbeck (2019). "Reinforcement learning in artificial and biological systems". In: *Nature Machine Intelligence* 1.3, pp. 133–143.

Niv, Yael (2019). "Learning task-state representations". In: *Nature neuroscience* 22.10, pp. 1544–1553.

Opheusden, Bas van et al. (2023). "Expertise increases planning depth in human gameplay". In: *Nature*, pp. 1–6.

Palminteri, Stefano et al. (2017). "Confirmation bias in human reinforcement learning: Evidence from counterfactual feedback processing". In: *PLoS computational biology* 13.8, e1005684.

Peer, Michael et al. (2021). "Structuring knowledge with cognitive maps and cognitive graphs". In: *Trends in cognitive sciences* 25.1, pp. 37–54.

Plonsky, Ori, Kinneret Teodorescu, and Ido Erev (2015). "Reliance on small samples, the wavy recency effect, and similarity-based learning." In: *Psychological review* 122.4, p. 621.

Pritzel, Alexander et al. (2017). "Neural episodic control". In: *International Conference on Machine Learning*. PMLR, pp. 2827–2836.

Rmus, Milena et al. (2022). "Humans can navigate complex graph structures acquired during latent learning". In: *Cognition* 225, p. 105103.

Savinov, Nikolay, Alexey Dosovitskiy, and Vladlen Koltun (2018). "Semi-parametric topological memory for navigation". In: *arXiv preprint arXiv:1803.00653*.

Schrittwieser, Julian et al. (2020). "Mastering atari, go, chess and shogi by planning with a learned model". In: *Nature* 588.7839, pp. 604–609.

Sebold, Miriam et al. (2017). "When habits are dangerous: alcohol expectancies and habitual decision making predict relapse in alcohol dependence". In: *Biological psychiatry* 82.11, pp. 847–856.

Silver, David and Joel Veness (2010). "Monte-Carlo planning in large POMDPs". In: *Advances in neural information processing systems* 23.

Singh, Satinder P and Richard S Sutton (1996). "Reinforcement learning with replacing eligibility traces". In: *Machine learning* 22.1-3, pp. 123–158.

Smith, Kyle S and Ann M Graybiel (2013). "A dual operator view of habitual behavior reflecting cortical and striatal dynamics". In: *Neuron* 79.2, pp. 361–374.

Solomon, Ethan A et al. (2019). "Hippocampal theta codes for distances in semantic and temporal spaces". In: *Proceedings of the National Academy of Sciences* 116.48, pp. 24343–24352.

Solway, Alec et al. (2014). "Optimal behavioral hierarchy". In: *PLoS computational biology* 10.8, e1003779.

Sommerville, Jessica A and Amanda L Woodward (2005a). "Infants' sensitivity to the causal features of means-end support sequences in action and perception". In: *Infancy* 8.2, pp. 119–145.

— (2005b). "Pulling out the intentional structure of action: the relation between action processing and action production in infancy". In: *Cognition* 95.1, pp. 1–30.

Sriram, Sivaraman et al. (2009). "Implementing a no-loss state in the game of tic-tac-toe using a customized decision tree algorithm". In: *2009 International Conference on Information and Automation*. IEEE, pp. 1211–1216.

Stachenfeld, Kimberly L, Matthew M Botvinick, and Samuel J Gershman (2017). "The hippocampus as a predictive map". In: *Nature neuroscience* 20.11, pp. 1643–1653.

Sutton, Richard S and Andrew G Barto (2018). *Reinforcement learning: An introduction*. MIT press.

Talvitie, Erik (2017). "Self-correcting models for model-based reinforcement learning". In: *Thirty-First AAAI Conference on Artificial Intelligence*.

Tambini, Arielle et al. (2023). "Structured memory representations develop at multiple time scales in hippocampal-cortical networks". In: *bioRxiv*, pp. 2023–04.

Wang, Shaoming, Samuel F Feng, and Aaron M Bornstein (2022). "Mixing memory and desire: How memory reactivation supports deliberative decision-making". In: *Wiley Interdisciplinary Reviews: Cognitive Science* 13.2, e1581.

Warren, William H (2019). "Non-euclidean navigation". In: *Journal of Experimental Biology* 222.Suppl_1, jeb187971.

Whitehead, Steven D and Dana H Ballard (1991). "Learning to perceive and act by trial and error". In: *Machine Learning* 7, pp. 45–83.

Whittington, James CR, David McCaffary, et al. (2022). "How to build a cognitive map: insights from models of the hippocampal formation". In: *arXiv preprint arXiv:2202.01682*.

Whittington, James CR, Timothy H Muller, et al. (2020). "The Tolman-Eichenbaum machine: unifying space and relational memory through generalization in the hippocampal formation". In: *Cell* 183.5, pp. 1249–1263.

Wimmer, G Elliott and Daphna Shohamy (2012). "Preference by association: how memory mechanisms in the hippocampus bias decisions". In: *Science* 338.6104, pp. 270–273.

Yassa, Michael A and Craig EL Stark (2011). "Pattern separation in the hippocampus". In: *Trends in neurosciences* 34.10, pp. 515–525.

Zalesak, Martin and Stephan Heckers (2009). "The role of the hippocampus in transitive inference". In: *Psychiatry Research: Neuroimaging* 172.1, pp. 24–30.

Zhang, Lunjun, Ge Yang, and Bradly C Stadie (2021). "World model as a graph: Learning latent landmarks for planning". In: *International Conference on Machine Learning*. PMLR, pp. 12611–12620.

Zhao, Wenjia Joyce, Russell Richie, and Sudeep Bhatia (2022). "Process and content in decisions from memory." In: *Psychological Review* 129.1, p. 73.

Zhu, Guangxiang et al. (2020). "Episodic reinforcement learning with associative memory". In.

Zhu, Jianqiao, Adam Sanborn, and Nick Chater (2018). "Mental sampling in multimodal representations". In: *Advances in neural information processing systems* 31.