Neuroscience and Biobehavioral Reviews 176 (2025) 106282

Contents lists available at ScienceDirect

Neuroscience and Biobehavioral Reviews

journal homepage: www.elsevier.com/locate/neubiorev

Review article

A taxonomy of spatial navigation in mammals: Insights from computational modeling

Sandhiya Vijayabaskaran, Xiangshuai Zeng, Behnam Ghazinouri, Laurenz Wiskott, Sen Cheng *

Institute for Neural Computation, Faculty of Computer Science, Ruhr University Bochum, Bochum, Germany

ARTICLE INFO

Keywords:
Spatial navigation
Spatial behavior
Taxonomy
Computational models
Artificial agents
Spatial representations
Generalization

ABSTRACT

Spatial navigation is a vital cognitive process in nearly all animals, relying on complex neuronal mechanisms to extract, process, and act upon spatial representations. To advance the understanding of spatial navigation and its neural mechanisms, Parra-Barrero et al. (2023) have proposed a taxonomy of spatial navigation processes based on extensive behavioral and neural studies. These processes are hierarchically organized in two levels with navigation strategies at the top and behaviors at the bottom. Building upon this taxonomy, here, we review computational modeling studies on spatial navigation in mammals to provide an overview of the current state of the art and further analyze the navigation processes within the proposed taxonomy. We specifically focus on the representations required by navigation processes, how these representations are extracted, and the computations necessary to execute each strategy and behavior. We propose that the key to understanding what representations and computations are being used by agents lies in testing their ability to generalize to novel situations. We identify three types of generalization relevant for navigation and analyze to what extent current computational models are capable of achieving these types of generalization. Our review shows that while significant progress has been made in modeling navigation, substantial work remains to model and fully understand spatial navigation in mammals.

Contents

---

* Corresponding author.
E-mail address: sen.cheng@rub.de (S. Cheng).

https://doi.org/10.1016/j.neubiorev.2025.106282
Received 10 March 2025; Received in revised form 15 June 2025; Accepted 7 July 2025

1. Introduction

Navigating through space is a task that animals apparently solve quite intuitively in their day-to-day lives and is an essential skill for the success of most species. Systematic laboratory studies of spatial navigation began over a century ago. Early work by John B. Watson investigated how animals, specifically white rats, learn to navigate mazes (Watson, 1907). He emphasized the importance of kinesthetic (movement-related) and organic (internal) sensations in guiding behavior. His work laid the groundwork for behaviorism, a dominant school of thought in psychology during the early 20th century. Although some of his conclusions are no longer considered valid, his methodology — using mazes to study spatial navigation — remains widely used. This approach was later refined by Edward C. Tolman in his revolutionary studies on spatial learning (Tolman and Honzik, 1930; Tolman, 1948). In contrast to the widely held strictly behaviorist views at the time, Tolman demonstrated that navigation involves more than simple stimulus–response associations, and that the observed behaviors must require some underlying internal representations. These foundational studies paved the way for the search for these underlying representations in the brain, leading to the discovery of various spatially tuned neurons, including place cells (O'Keefe and Dostrovsky, 1971), grid cells (Hafting et al., 2005; Fyhn et al., 2007), boundary cells (Hartley et al., 2000), and head direction cells (Taube et al., 1990a,b).

Together, these and many other studies have revealed that spatial navigation requires internal representations of the environment to guide behavior, enabling the determination and maintenance of a course or trajectory from one location to another (Tolman, 1948; O'Keefe and Nadel, 1978; Gallistel, 1990; Franz and Mallot, 2000; Parra-Barrero et al., 2023). On the other hand, the precise nature of the connection between neural representations and the computations and behaviors they support remains elusive, in part due to a disconnect between fields and the somewhat disproportionate focus on the detailed characterization of spatial representations at the expense of understanding what computations and behaviors they support.

Spatial navigation also involves a hierarchical interplay of high-level computations and low-level motor commands. High-level processes include tasks such as localization, path planning, and goal-directed decision making, often relying on abstract representations of the environment. In contrast, low-level motor control translates these plans into precise actions, such as moving forward, adjusting speed, and turning. Thus, given that it engages both higher-level cognitive processes and lower-level automatic mechanisms, along with the wealth of behavioral and neural data available, spatial navigation serves as a unique window into exploring fundamental principles of brain function.

Beyond advancing our understanding of the brain, studying spatial navigation has practical applications in robotics, urban design, and medicine. For example, insights into spatial navigation can inspire the development of autonomous robots (Milford et al., 2004), guide the design of easily navigable urban spaces (Meilinger and Knauff, 2008; Silavi et al., 2017; Bibri and Krogstie, 2019), and can aid in the early diagnosis of diseases such as Alzheimer's (Colmant et al., 2023; Bierbrauer et al., 2020) and Parkinson's disease (Lin et al., 2014; Thurm et al., 2016).

Computational modeling has been instrumental in advancing our understanding of the cognitive processes and neural mechanisms that underlie spatial navigation (Bermudez-Contreras et al., 2020; Ghazinouri et al., 2024). Models have to formalize the object under study, make assumptions explicit, propose specific representations and computations, simulate their interactions to see their outcomes, and make predictions. The results and predictions of the model can be analyzed and compared with experimental observations to draw conclusions about the validity of the features of the model used (Kriegeskorte and Douglas, 2018). Computational modeling can also inspire experimentalists to design and perform novel experiments to test model predictions.

Interest in computational modeling of spatial navigation has increased significantly over the last decade, resulting in an extensive body of literature. This is because, on the one hand, advances in computer science, both in software and hardware, have enabled sophisticated modeling and analysis of spatial navigation (Schwalger et al., 2017; Eliasmith and Trujillo, 2014; Diekmann et al., 2023). On the other hand, there has been progress in autonomous robotics and vehicles, where understanding spatial navigation is essential for developing intelligent systems (Milford and Schulz, 2014; Milford et al., 2016).

In light of these advances, we consider it both appropriate and beneficial to review and synthesize what we have learned so far from computational modeling of spatial navigation. To this end, we adopt the taxonomy of spatial navigation introduced by Parra-Barrero et al. (2023). This hierarchical framework considers spatial navigation processes on two levels: five navigation behaviors at the bottom level controlled by three navigation strategies at the top. In addition, two organization processes govern the selection and integration of the behaviors and strategies available to the agent. In this review article, we identify the representations necessary for each behavior and strategy and the abstract-level computations required to support them. To ensure that agents are truly solving the task using the hypothesized navigation process rather than by some other means, e.g., based on stimulus–response associations, we argue that it is critical to investigate their capacity for generalization (Lee et al., 2020; Zeithamova and Bowman, 2020; Farebrother et al., 2020).

Since it would be virtually impossible to systematically cover all computational models of spatial navigation, we restrict ourselves to models of mammalian spatial navigation (or those closely resembling them) in this review article. In addition, we primarily focus on two classes of computational models. First, we consider models that are biologically plausible and based on neuroanatomy and/or account for neural activity patterns, including spiking neural networks and rate-based models. Spiking neural networks simulate the precise timing of individual spikes in neuronal communication, capturing temporal dynamics critical for understanding real neural processes (Maass, 1997; Taherkhani et al., 2020). Rate-based models, on the other hand, use a more abstract, yet insightful, representation of neural activities as average firing rates over time (Brette, 2015). Second, we consider models based on reinforcement learning (RL) and models from the robotics domain when they offer insights for mammalian navigation. These models use network architectures and learning algorithms that are not subject to biological constraints. RL is a machine learning paradigm that has received significant attention from neuroscientists in recent years. In RL problems, an agent learns by interacting with its environment and optimizing behavior based on rewards or penalties. Unlike supervised learning, RL relies on exploration and self-directed learning, making it well-suited for modeling adaptive, goal-directed behaviors such as spatial navigation (Sutton and Barto, 1998). RL is a broad class of approaches, which can be model-free, where strategies are learned directly from experience, or model-based, where the agent builds an internal model of the environment to plan and predict outcomes (Sutton and Barto, 1998; Dayan and Berridge, 2014). Due to this vast body of literature, what follows is not a systematic review, but rather a critical examination of the literature on computational modeling of navigation through the lens of the taxonomy. As a first step, we searched the literature using keywords corresponding to the navigational processes defined in the taxonomy including related search terms that were used previously by other researchers in the field (e.g. beaconing for aiming, cognitive graph for topological navigation etc.). The full list of related terms corresponded to Table 1 in Parra-Barrero et al. (2023). In the second step, we selected a subset of the papers that are at least applicable to mammalian navigation. For most navigation processes, no other inclusion/exclusion criteria were necessary. For guidance and map navigation, a further refinement was necessary due to the vast over-representation of these two processes in the literature. We selected papers to ensure balanced coverage that encompasses a breadth of ideas. In addition to including models from both biologically plausible and RL-based approaches, we select examples that span a diverse range of representational formats and computations within each category.

This review article is structured as follows: In Section 2, we briefly review the taxonomy of spatial navigation processes by Parra-Barrero et al. (2023) and identify the defining representations and computations for each navigation process, as well as outline our notion of generalization in detail. In Section 3, we review computational models and what they reveal about navigational processes through the lens of the taxonomy. In Section 4 , we discuss the key insights from computational studies of spatial navigation and point out directions for future research. Finally, in Section 5 we conclude our work.

2. Computations and representations underlying spatial navigation processes

2.1. Generalization of navigation processes

In this review, we adopt the taxonomy of spatial navigation introduced by Parra-Barrero et al. (2023) to organize and interpret the computational models discussed and develop it further. Based on decades of behavioral and neural evidence, the taxonomy introduced eight navigation processes and offers insights into how these processes may be organized and interact with one another. After briefly recapitulating the taxonomy in the following, we significantly expand the earlier work by identifying the representations and computations that define each navigation process and outline the process-specific requirements for generalization.

According to the taxonomy, navigation processes are hierarchically structured in two levels, with navigation strategies at the top and navigation behaviors at the bottom (Fig. 1). Navigation strategies take the final goal of navigation and break it down into a series of subgoals, which can be reached by using one or more of the navigation behaviors available at the bottom level. The navigation behaviors are elementary navigation processes that generate abstract movement instructions (move forward, turn right, etc.), not motor commands, that guide the agent towards a subgoal. In addition to the strategies and behaviors themselves, two additional processes organize strategies and behaviors by selecting or integrating strategies or behaviors available to the agent.

Each process is defined by the representations and the computations underlying it, and it is these representations and computations that distinguish one process from another (Tables 1 and 2). We emphasize that both components are important to fully characterize a process, since, for example, the same representations may be used in different computations resulting in different behaviors (Table 1). Thus, focusing solely on representations is not enough to fully characterize the processes.

This means that in order to pinpoint what process an agent, be it artificial or biological, is using to solve a navigation task, the experimenter or modeler must get at both the representations and computations being used by the agent. While it is evident that this can be quite tricky to do experimentally, it may seem, at first glance, quite straightforward for computational models. This is indeed the case when the modeler explicitly builds in representations and computations in the model, i.e., in a more or less hard-coded model of a process. However, since most computational models are self-organized to some degree, and in some cases, such as deep neural network models, the inner workings of the model are black boxes unless explicitly probed, this issue is not trivial even for modeling.

We therefore propose that the key to pinpointing what process is at work lies in testing the agent's ability to generalize to novel situations. Since generalization is a fairly broad term and an agent can generalize over different aspects of the task and the environment, here we clarify "generalization" and which aspects we believe are important to pinpoint the underlying processes. We identify three forms of generalization that are relevant to navigation behaviors and strategies. The generalization principles do not apply to the organizational processes since these operate on the behaviors and strategies and do not make navigational decisions themselves.

The first type of generalization is *sensory generalization*, i.e., the ability to acquire a target under different sensory conditions or over different sensory modalities. For instance, take an agent that is able to navigate only to a very specific type of target, such as a red dot generated by a laser. While this restriction certainly makes the process highly specialized and in some situations less useful in the real world, it does not inherently change the representations and computations being used. Thus, we propose that, while practically useful, sensory generalization is not strictly required to qualify as a navigation process in our taxonomy.

A second, more important type of generalization for navigation is what we term *spatial generalization*. This is the agent's ability to generalize over three parameters (start location, goal location, and environmental features) and is essential to identify which navigation behavior is being used and to rule out alternative explanations for the observed behavior. These explanations include (i) not being a navigation behavior at all, e.g. having memorized a very specific sequence of steps from start to goal (ii) being a different navigation process than the one under consideration, e.g. having learned to navigate only to a single location in space (iii) using some features of the environment to navigate that were deemed irrelevant by the experimenter or modeler. An agent's ability to generalize can be examined through probe trials or a test phase in which the agent has to navigate from different starting and/or to different goal positions or in which the environment has been altered to a reasonable degree. An ecologically relevant example of the latter is the ability to execute the same navigation process in spring and winter, where several properties of the environment may have changed. Similarly, in a laboratory or simulation setting, navigation should generalize over changes to any features of the environment that are not relevant to the task, such as background features and lighting — as long as these changes do not interfere with target acquisition.

A third type of generalization that we consider is the degree to which an agent can find an alternative route to a goal, i.e. *planning generalization*. This type of generalization only applies to the navigation strategies, since only these processes engage in planning to produce a sequence of subgoals. For instance, when navigating to a target along a previously planned trajectory, the agent comes upon an obstacle and, thus, has to plan an alternative trajectory. The alternative trajectory planned by the agent reveals something about the knowledge representation, i.e. which strategy, that the agent uses. An agent that can plan more generally would be capable of solving more complex tasks due to its increased flexibility.

In the following, we discuss the navigation behaviors and their organization, followed by the navigation strategies and their organization. We outline the representations and computations underlying each of the processes (summarized in Tables 1 and 2) and briefly discuss what the types of generalization outlined above might look like for each process.

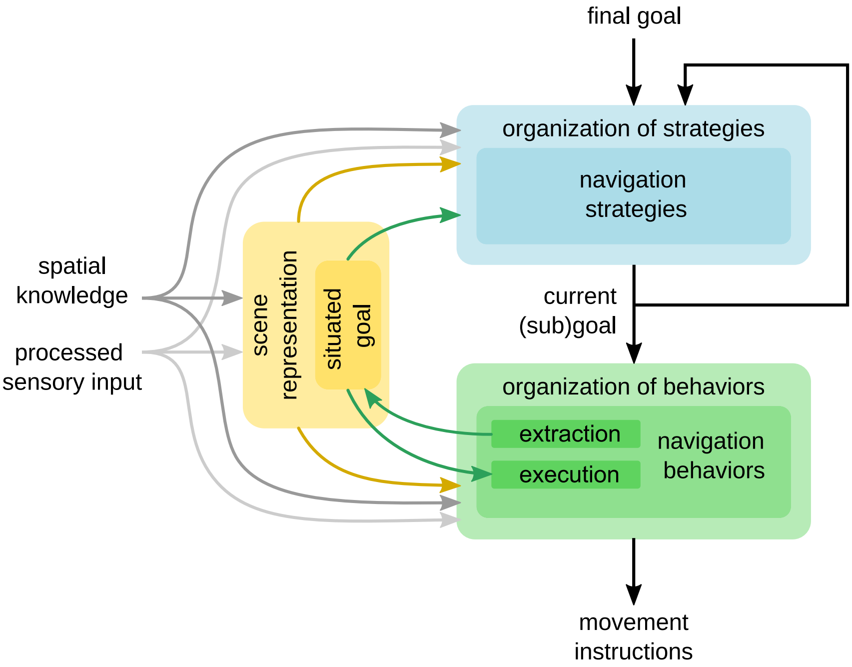

Fig. 1. The organization of navigation processes. Navigation strategies take a final goal and produce a series of subgoals. Each subgoal is then passed on to the organization of behaviors, which chooses one or more navigation behaviors, which generate abstract movement instructions to reach the current subgoal. Each behavior has an extraction and an execution phase, and receives processed sensory input and spatial knowledge in addition to the current subgoal. A scene representation that contains relevant information about the current environment in working memory is also available to the navigation processes.
Source: Reproduced from Parra-Barrero et al. (2023) with permission.

2.2. Navigation behaviors and their organization

The taxonomy of Parra-Barrero et al. (2023) distinguishes five navigation behaviors: aiming, direction field navigation, path following, vector movement, and guidance (Fig. 2). These navigation behaviors translate subgoals computed by the navigation strategies into an operational objective and output movement instructions that allow the agent to reach the goal. These movement instructions are not conceived of as motor commands, but rather as high-level representations such as movement direction and speed. Each behavior is thought to be divided into two phases: an extraction phase in which the goal or movement direction is identified and an execution phase in which movement towards the goal occurs. These two phases may occur sequentially, overlap, or be interleaved (Fig. 2).

Aiming is one of the simplest navigation behaviors, where the goal location is defined by an object. It involves orienting to the target object and moving through space until it is reached (Fig. 2A). Also known as beaconing in other taxonomies (Gallistel, 1990; Toledo et al., 2020), this behavior is most commonly known for visual landmarks, although it also works with other sensory cues, such as sound, that allow the agent to directly perceive the target object. Aiming requires maintaining a representation of the direction and approximate distance to the target (Table 1). This distance representation does not necessarily need to be explicit; for example, consider simple visual aiming which can be accomplished by keeping the object more or less centered in the field of vision and moving towards it. In this simple case, the distance to the goal can be inferred from, for example, semantic knowledge about the usual size of the object or from how fast the size of the goal changes in the visual field as the agent moves towards it. Computationally, aiming requires keeping a perceptible target within the range of the sensory systems while reducing the agent's distance to the target. This distance estimation is useful in order to produce meaningful behavior, such as moving faster when the goal is further away and slowing down when the goal is closer (this feature might also be useful for pursuit). Crucially, the representations are constantly extracted from the perception of the goal, which means that aiming immediately breaks down when the target object is not detected.

An agent might be able to aim for a variety of objects (sensory generalization), but, as discussed above, this is not strictly necessary and might not even be expected in certain cases — including in biological agents. For example, while animals may very quickly aim for food directly, they might be less inclined to do so for random targets, preferring to engage in guidance instead (Devenport and Devenport, 1994). In artificial agents, failure to generalize to new targets only implies that the recognition model being used may not be sophisticated enough and does not imply a deficit in aiming *per se*. On the other hand, identifying whether aiming is actually being used, even if only to a single target, requires checking for spatial generalization. The ability to generalize to novel start and goal locations indicates that the agent is able to extract direction and approximate distance to the target object and rules out navigating to a specific spatial location or memorizing a specific movement sequence. In a similar vein, the ability to generalize over changes to the surrounding environment demonstrates that the agent indeed extracts direction and distance to the target object itself and not from sensory properties of the surrounding environment. Similar arguments regarding generalization apply to the other navigation behaviors as well; thus, we only discuss concrete examples and process-specific nuances for the behaviors that follow.

In **Direction field navigation**, the agent uses a local directional signal that extends over the environment to compute and execute a local movement to reach the goal (Fig. 2B). Common examples of direction fields that could be used for navigation include gradient signals, such as odor or altitude and compass direction inferred from the earth's magnetic field. In direction field navigation, the agent requires the representation of the local signal throughout the navigation to move at a certain angle relative to the signal or the gradient of the signal until the goal is reached (Table 1). The directional signal is local in the sense that it is valid only at a particular location and may be quite different at nearby locations. For successful navigation, this directional signal must be available over an extended area that covers the start and goal locations and some extended region connecting the two. Spatial

Table 1

Representations and computations of the navigation behaviors: The typical representations are defined as follows: d — the approximate distance to the goal from the agent's current location, θ_{goal} — the approximate direction to the goal, θ_{local} — the local movement direction, $\mathbf{x}_{\text{self}}$ — the representation of the agent's location, and $\mathbf{x}_{\text{goal}}$ — the representation of the goal location.

Behavior	Extraction	Typical representation	Execution
Aiming	Direct sensory perception of goal	θ_{goal}, d	• Rotate to minimize θ_{goal} • Move in direction that reduces d • Adjust velocity and stop based on d
Direction Field Navigation	Infer gradient from sampling intensity at different locations or infer gradient directly (like slope or magnetic field)	θ_{local}	• Move in direction dictated by θ_{local} • Stop when no direction information available
Path Following	Sensory perception of path and initial orientation to follow path	θ_{local}, lateral deviation from path	• Move forward in approximate direction of path given by θ_{local} • Limit lateral path error while maintaining orientation • Stop if path ends or goal is reached
Vector Movement	From path integration on outbound path, existing/externally given spatial knowledge or extraction phase of other behaviors	θ_{goal}, d	• Move in direction of θ_{goal} • Update θ_{goal} and d using path integration • Adjust velocity and stop based on d
Guidance	Configural cues	$\mathbf{x}_{\text{self}}$, $\mathbf{x}_{\text{goal}}$ in term of configural cues	• Move in order to make $\mathbf{x}_{\text{self}}$ align with $\mathbf{x}_{\text{goal}}$

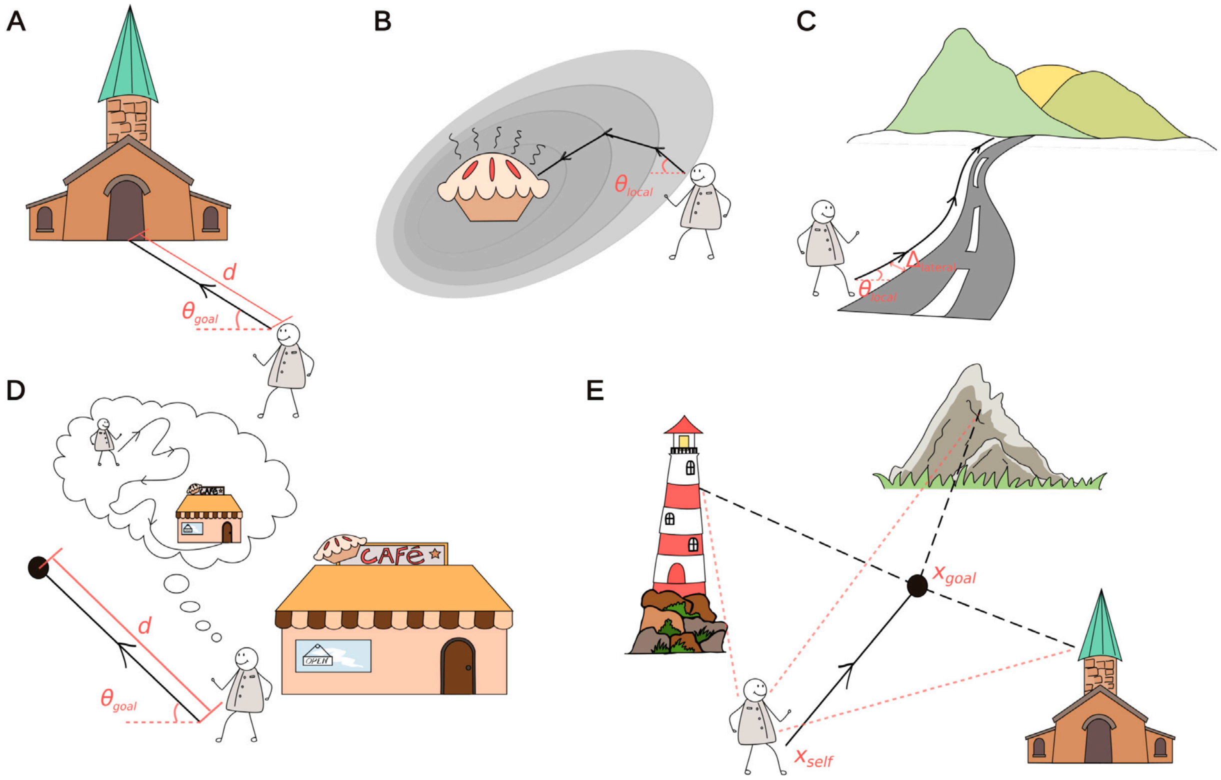

Fig. 2. Navigation behaviors and required representations. Each panel illustrates a navigation behavior and its required representations. Variables are the same as in Table 1. **A**: Aiming, movement is directed towards a perceptible goal. **B**: Direction field navigation, the agent moves based on a local signal towards the goal, an odor source in this case. **C**: Path following, the agent follows a path leading to the goal. **D**: Vector movement, the agent uses a vector pointing towards the goal for navigation. The vector could have been computed, for example, through path integration on the outbound travel. **E**: Guidance, the agent navigates to an unmarked goal defined by the configuration of several landmarks. *Source:* Adapted from Parra-Barrero et al. (2023) with permission.

generalization in direction field navigation implies the ability to start navigation from any location in the direction field, and also the ability to navigate even if source of the field were moved, as long as the local directional signal remains available. In the example of an odor gradient field emanating from a source, this means that the agent can navigate to the source from any point in the gradient field as well as navigate successfully even if the source itself is moved as long as it can still sense the local gradient.

Path following, as the name implies, involves moving along a particular path at which the goal is located until it is finally reached (Fig. 2C). Aside from knowing which path to choose, such as a particular river, the agent must also know in which direction the path must be followed, for instance, do I follow the river upstream or downstream? Computationally, path following involves identifying the path that leads to the goal and constantly adjusting the agent's position in order to stay on the path, e.g. movement of a train on the tracks, or to stay close to the path and maintain a distance from it, e.g. moving along the meanders of a river (Table 1). Thus, path following involves maintaining a representation of the lateral deviation from the path in addition to the local direction in which to follow the path. To test for spatial generalization in path following, the agent should be tested on various path configurations, for example, a path in different directions and shapes (straight vs. curved), as well as novel start and goal locations.

Vector movement involves using the representation of a goal vector that indicates the direction and distance to the goal to navigate. Computationally, this involves continuously updating this vector by path integration to keep track of progress towards the goal (Fig. 2D and Table 1). While the initial source of the goal vector could be path integration from a previous traversal, such as in homing, it could also come from other sources such as a metric map of the environment or from sensory cues in the environment that allow the computation of such a vector. Similarly, path integration could also be supported by self-motion cues from different sensory sources: optic flow, vestibular or proprioceptive inputs. Thus, for vector movement, sensory generalization can occur in two ways — in the acquisition of the initial vector (this is exemplified well by humans, who can even acquire the initial vector from verbal instructions) and in the sensory systems supporting path integration. Testing for spatial generalization involves assessing whether the agent can navigate from arbitrary start to arbitrary goal locations once the goal vector is available. Note that this constitutes a simple form of shortcut-taking, where an agent can move towards the goal along a direct path indicated by the goal vector (see e.g. Ekstrom et al., 2014). However, this is limited to locations for which the goal vector is available, such as the home location in homing.

Guidance is the most complex of the navigation behaviors (Fig. 2E). It is defined as navigating to a target location by using information about its spatial relationship to landmark(s) and distal orienting cue(s). For guidance, the agent maintains a representation of the goal location and its self-location with respect to some configural cues (Table 1). There are multiple possible ways of execution, but the general criterion is that the computation should guide the agent such that the representations of the goal location and self-location become more similar with each step. In addition to testing for spatial generalization with novel start and goal locations in relation to a stationary cue configuration, the entire cue configuration may be rotated and/or translated to ensure that the representations of goal and self are computed with those cues. Note that the ability to perform guidance with different configurations of cues, i.e., when the cues are arranged differently with respect to each other, is a form of sensory generalization and is not strictly required to qualify as guidance.

Organization of behaviors: Real-world navigation is complex, and agents may often find themselves in situations where multiple behaviors can be used to navigate to the subgoal, or conversely, where no single behavior alone is sufficient to successfully navigate to the subgoal. These conditions can change dynamically, and the agent must be able to adapt its behavior accordingly. For example, when navigating in a city, one might initially navigate by following a particular road (path following), but upon encountering a road closure, one might need to use knowledge about the goal's location in relation to landmarks, such as tall buildings, to navigate (guidance). The taxonomy outlines two ways for making use of multiple navigation behaviors, namely *selection* and *integration*. In the former, the agent must select one of available behaviors and only one is active at any given time. Selection can occur in one of two ways: *relay switching*, in which the agent hands over control from one behavior to another at a natural termination point of the first behavior, and *dynamic switching*, where the behaviors are computed in parallel and compete for control of the agent's navigation. We hypothesize that vector movement may play a special role in dynamic switching, acting as a fallback process for the other behaviors when external sensory information is unavailable or unreliable.

Integration, the second possibility for organization, involves multiple behaviors running in parallel, but with shared control over navigation. Integration can take place in different ways, either in the extraction or the execution phase. In the extraction phase, integration involves combining multiple behaviors to successfully extract the representations required for navigation. For example, guidance could be used to constrain the region of visual search for locating the target for aiming. In the execution phase, one possible implementation of integration is when different behaviors output movement proposals in a common format, e.g. movement distance and direction, and these proposals are then combined in a weighted manner to produce a final movement instruction.

2.3. Navigation strategies and their organization

Navigation strategies are complex processes that take the final goal and structure navigation into a sequence of subgoals. Depending on the available representations, each subgoal can be reached successively by one of the navigation behaviors or a combination of them. The taxonomy outlines three navigation strategies: route navigation, topological navigation, and metric navigation (Fig. 3).

Like in the case of the behaviors, testing for generalization can serve as a diagnostic tool for determining whether a particular navigation strategy is being used. However, unlike for the behaviors, where spatial generalization was key, planning generalization takes center stage for delineating strategies. While for the behaviors, the experimenter or modeler would ask — *do we observe spatial generalization corresponding to this behavior?* — for the strategies, the question becomes more nuanced, asking to what extent planning done by the agent generalizes. The degree of generalization, i.e. how flexibly an agent can plan, is then a signature of which strategy is being used and depends both on the spatial knowledge structure (route/graph/metric map) and the specific planning algorithm that operates on that structure. These two aspects, i.e. the spatial knowledge structure and the planning algorithm are closely related because the spatial knowledge structure being used greatly limits the kinds of planning algorithms that can be operated on it. In the following, we describe the three navigation strategies and touch upon these two aspects for each of them.

Route navigation is the simplest navigation strategy where navigation to the next subgoal in a sequence is triggered by reaching the current subgoal (Fig. 3B). The determination of the specific behavior(s) used to navigate to the next subgoal is governed by the organization of behaviors rather than determined by the strategy. While repeated experience can bias agents towards using specific behaviors to reach each subgoal in the sequence, route navigation still allows for flexibility in the choice of behaviors. This flexibility enables agents to adapt to obstacles or changes in the environment that do not interfere with identifying subgoals as needed. However, while using route navigation, agents cannot recover easily from missed cues that define the next subgoal and must either backtrack to a known location along the route or make use of alternative strategies, if available. The spatial knowledge

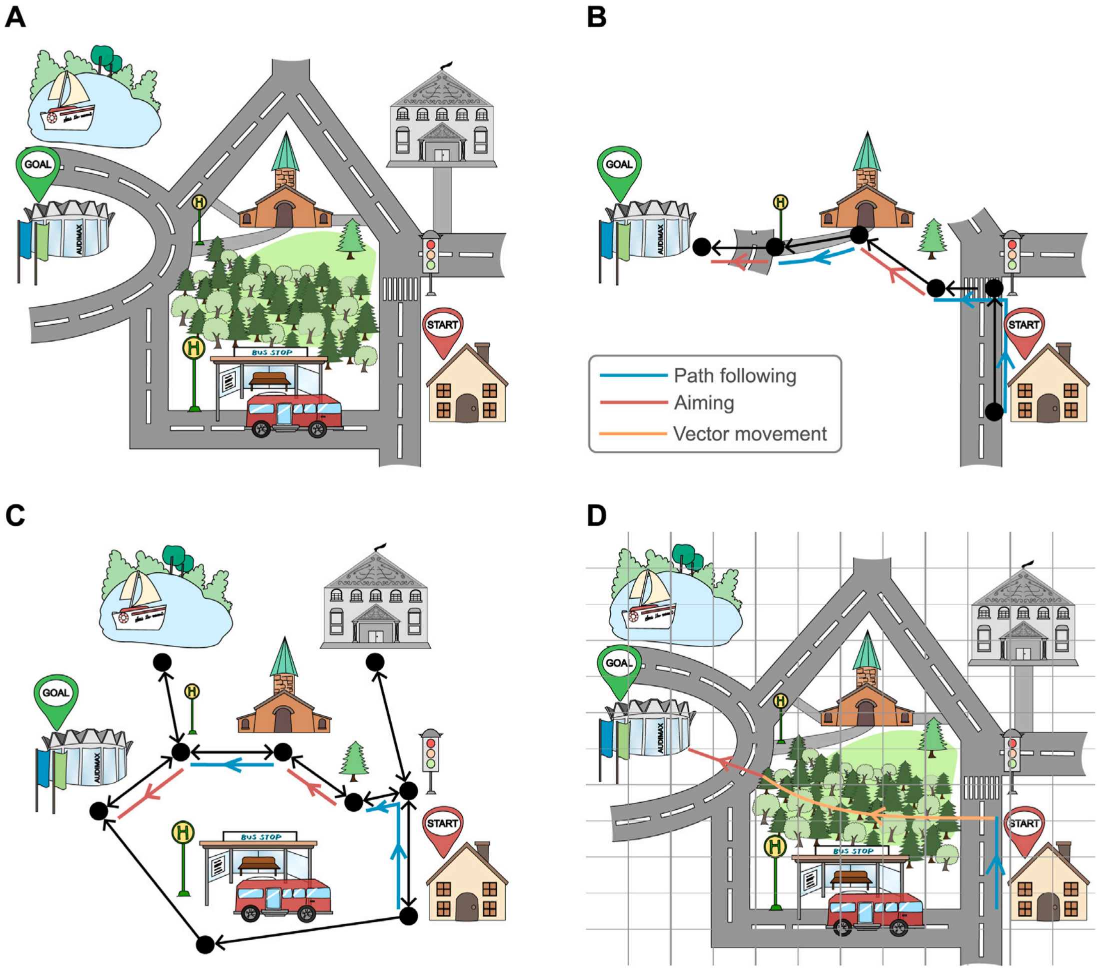

Fig. 3. Illustration of navigation strategies. A: The layout of a city. The agent needs to navigate from home to the university campus. **B**: In route navigation, the agent uses a fixed sequence of subgoals to navigate to the final goal. In the example above, the agent uses path following along the road until the traffic light, then across the pedestrian crossing, then aims towards the church steeple and so on. **C**: In topological navigation, the agent plans the route based on its topological graph representation of the city. The agent is able to generate other routes, e.g. via the bus stop, based on its graph. **D**: In metric navigation, a metric map indicates the relationships between all elements in the city, which allows the agent to discover a direct shortcut through the forest using vector movement, even if the agent has never traveled through the forest before.
Source: Reproduced from Parra-Barrero et al. (2023) with permission.

structure comprises representations for a sequence of subgoals as well as the current subgoal (Table 2). Planning in route navigation is quite limited and in the simplest case only involves picking a route from different known routes at the start of navigation. A more sophisticated version also allows for concatenating routes in order to reach the final goal along a new route stitched together from the individual segments. As additional routes with overlapping subgoals are learned and stitched together, the spatial knowledge structure gradually evolves into a topological map, marking a transition from one structure to another. Interestingly, route navigation shares some similarities with serial recall tasks employed in episodic memory research. Building up route knowledge has parallels to ordinal strategies used in these tasks (Addis and Kahana, 2004; Kahana, 2020), and the Temporal Context Model (Howard and Kahana, 2002), a well-known computational model of list learning, has been extended to spatial domains (Howard et al., 2005). In addition, some experimental evidence suggests that verbal list learning effects may generalize to a route navigation setting (Meilinger et al., 2016; Hilton et al., 2021). These similarities are indicative of a broader, but not complete, overlap between navigation and memory (Ekstrom and Hill, 2023).

Topological navigation is based on a topological graph representation of the environment (Fig. 3C). This graph can be thought of as a simplified representation of the environment, much like a subway map, which focuses on connectivity rather than precise geometric relationships and consists of nodes that correspond to known locations in the environment. The nodes are connected by edges where there is the possibility of navigating using one of the behaviors (Table 2). Although this map lacks metric information, edges can, but do not have to, be associated with some measure of cost (time of travel, distance, or other quantities) that may influence the choice of path. Similarly, nodes on the graph could be associated with position coordinates or angles, which could then guide navigation. There is indeed experimental evidence for such labeling in graphs used for navigation (Warren et al., 2017; Warren, 2019; Chrastil and Warren, 2014, also see Parra-Barrero et al., 2023 for a comprehensive discussion).

The agent needs to localize itself on the graph, so that it knows which node it is at or between which nodes it is located. Paths can be planned on this graph between any two nodes that are connected by a sequence of edges (Table 2), making this strategy quite flexible. For instance, if the preferred path is blocked, the agent is able to plan another path consistent with the graph. However, in a pure, unlabeled topological graph, direct navigation between two unconnected nodes is not possible outside vista space, thus excluding the use of shortcuts that are not connected by edges of the graph. This limitation may, however, be overcome in labeled graphs. Depending on the type and extent of labels, direct navigation between any two nodes may be possible, even if they are unconnected by edges (Warren, 2019).

Table 2
Representations and computations of navigation strategies.

Strategy	Spatial knowledge	Planning
Route Navigation	Series of subgoals leading to the final goal	Pick route at the beginning of navigation if multiple routes are known
Topological Navigation	(Possibly weighted) Topological graph consisting of nodes, representing potential subgoals, connected by edges	Localize oneself on graph and plan over graph to optimize path length or other cost
Metric Navigation	Metric map with position, distance, and angle relationships	Localize oneself on map and plan path based on distance, time, or other criteria while taking into account obstacles or available shortcuts

Metric navigation is the third and most flexible navigation strategy. It involves maintaining an approximately Euclidean map representation of the environment and using it to plan paths between locations on it (Fig. 3D). The extent of the map is influenced by both sensory access to the environment, referred to as the vista space — for example, standing atop a hill provides a broader view of the surroundings — and familiarity with the environment, such as having a well-defined metric map of the interior of our homes. Navigation using a metric map implies being able to localize oneself within the map and then planning a path to the goal via different subgoals, also represented on the map (Table 2). An optimization strategy based on minimizing a cost (e.g., the distance, time, metabolic cost, or a combination thereof) would be used to plan the path. The key advantage of a metric map is that it allows planning shortcuts through novel terrain to optimize navigation and detours through novel areas when known paths are blocked. Direct paths between any two arbitrary points can be planned, as long as they are within the extent of the map. However, direct experimental evidence for this type of navigation is still lacking, as we outline in Parra-Barrero et al. (2023). Additionally, the flexibility offered by a metric map comes at a cost; the construction, storage and use of such a map is computationally expensive, making it less efficient than the two strategies above, especially over larger scales.

Organization of strategies: Like the navigation behaviors, the navigation strategies are also orchestrated by an organization process with the two possibilities, selection and integration. Selection implies that agents choose from available strategies, and only one strategy is active at any given moment. The transition between strategies can occur dynamically. For instance, an agent may switch to metric navigation in vista space from one of the other navigation strategies upon encountering a blocked path. The preferred strategy may also switch as the agent gathers more experience. This transition may occur in both directions — either starting with metric navigation in completely novel terrain and transitioning to less computationally intensive route navigation with repeated experience, or when a route is initially available, building it up gradually into graphs and maps as more spatial knowledge is acquired. Integration of strategies could occur at different levels: at the information level (adding metric information from a metric map to the edges of a topological graph), at the subgoal level (different strategies vote for different subgoals, which are then integrated), and at the process level (constraining the search on a topological map based on heuristics from the metric map, like in A* (Hart et al., 1968)).

3. Insights into navigation processes from computational modeling

3.1. Computational models of navigation behaviors

3.1.1. Aiming

Aiming involves moving towards a directly perceptible target at the goal along a relatively straight path.

A well-known model that produces aiming-like behavior is the Braitenberg vehicle (Braitenberg, 1986). One of the simplest versions is a vehicle with a wheel and sensor on the two sides, where the sensors measure the intensity of the target stimulus, such as a bright light or an intense odor. In this simple version, the left and right sensors have an excitatory connection to the opposite wheel. This cross-wiring allows the vehicle to represent the direction towards the goal and, thus, to move towards the stimulus source by continuously adjusting its heading direction. Whenever one sensor detects a stronger signal than the other, the contra-lateral wheel speeds up, causing the vehicle to turn in the direction of the target and approach the goal. Although the Braitenberg vehicle's behavior may resemble aiming, an alternative interpretation is that the simplest version of this model, is more accurately characterized as direction field navigation (discussed in Section 3.1.2), since the model does not represent the distance to the goal.

This limitation can be easily overcome, and models based on the Braitenberg vehicle are a powerful tool for modeling behaviors in neuroscience (Shaikh and Rañó, 2020). For instance, Milde et al. (2017) implemented aiming similar to a Braitenberg vehicle on a neuromorphic device, i.e. an implementation of biologically inspired neural networks on a chip, which can be used for real-time neural processing on robots. Their robot is equipped with a camera providing visual input to the network and is tasked with aiming towards a blinking LED light. The visual field is split in two halves, with the right half projecting to one population and the left half projecting to the other population. The entire spiking neuromorphic network has attractor dynamics where the decision about the movement (left or right) is implemented by a winner-take-all mechanism. Importantly, the speed is adjusted based on the size of the object in the visual field, with the robot slowing down as its distance to the target decreases. Thus, this model has a representation of both direction and approximate distance. The agent is tested on different starting and goal locations, with moving goals, and in different environmental conditions such as poor lighting and demonstrates the capacity for spatial generalization. However, it lacks a sophisticated object detection module, which restricts it to a very specific target, i.e., a blinking LED. Despite this lack of sensory generalization, this implementation qualifies as aiming, as we noted in Section 2.

A weakness of Braitenberg vehicle-like implementations is that the aiming behavior is hard-wired and not learned, which can result in low flexibility. To overcome this limitation, Chavarriaga et al. (2005) and a follow-up study by Dollé et al. (2010) implement aiming using neural networks with brain-inspired coding features. Unlike in Braitenberg vehicles, there is no particular constraint in the connections from sensors (sensory input neurons) to the movement units (action neurons). Instead, the network forms a preference to target cues in the connections via temporal difference learning when the agent finds the goal or is guided to it after a given amount of time. The model uses a population of landmark cells, which encode the angle between the target and the cell's preferred direction in an egocentric reference frame. Additionally, as the agent moves closer to the target, the width of a Gaussian centered on the landmark direction increases, indicating that the target takes up a larger portion of the visual field, thus providing the agent with a distance estimate. Since the model consists of multiple behaviors that interact, among which aiming is only one, it is difficult to evaluate the degree of generalization exhibited by the aiming module alone. However, results from their simulations suggest that under conditions where a landmark indicating the goal moves frequently, the aiming module is favored and can navigate to new goal locations based on the position of the landmark, indicating at least some degree of spatial generalization.

The models of aiming described thus far represent the direction and distance to the goal by the activity of the sensory units, with the aiming behavior being driven by the weights between the sensory and motor units, which can either be hard-wired or learned. While this simplicity makes it relatively easy to analyze the model's behavior and computations, it also has some limitations. For instance, the distance and direction to the goal are given directly to the module in some cases (Chavarriaga et al., 2005; Dollé et al., 2010), which effectively sidesteps the modeling of the extraction phase of the behavior. Additionally, while sensory generalization is not strictly necessary to qualify as a certain navigation behavior, it is still practically useful, but it is unclear how the extraction phase could be added to these models.

Deep RL models could potentially overcome some of these shortcomings, but they also bring their own set of challenges and limitations, as we discuss next. Generally speaking, most state-of-the-art deep RL algorithms are generally capable of learning aiming using visual input (Beattie et al., 2016; Chaplot, 2016; Jaderberg et al., 2016; Kempka et al., 2016; Zhu et al., 2016; Mirowski et al., 2017; Kulhanek et al., 2019). However, even though the behavior of these RL agents might appear to be performing aiming, the agents may not be doing so in the way we envision it in our taxonomy. For instance, an agent could theoretically memorize a very specific sequence of actions that lead to a visual goal or use other features of the environment to guide it to the goal. Nevertheless, in general it seems that these models can generalize quite well to new goal and start locations. For instance, Chaplot (2016) compare the performance of several model-free deep RL agents in a 3D game environment to show that most demonstrate at least some degree of generalization. Importantly, since the underlying representations that guide aiming in these agents are rarely examined, it becomes hard to evaluate what exactly these agents have learned.

One study that examines the representations that underlie aiming in a deep RL agent finds that egocentric vector representations, i.e. a representation of direction and distance to the goal, emerge in the network when aiming (Vijayabaskaran and Cheng, 2022). They also show that the agent can learn aiming-like behavior in non-intuitive ways, for example, by relying on other features of the environment such as distal cues and lighting. In line with our suggestions, they find that testing for spatial generalization using novel start and goal locations and removing environmental features irrelevant to aiming can help identify genuine aiming behavior. However, much like the biologically plausible models we identify above, this model is also unlikely to exhibit sensory generalization, since it is only capable of navigating to the specific sensory target (a red cylinder in the simulation environment) that the agent was trained on.

Like the findings of Vijayabaskaran and Cheng (2022), the studies by Chavarriaga et al. (2005) and Dollé et al. (2010) also suggest that aiming is likely carried out in an egocentric reference frame. An interesting possibility that we suggest should be studied in the future with a more detailed analysis of the deep RL agent's behavior and network connectivity is that Braitenberg vehicle-like dynamics that include distance estimates might emerge to support aiming even in these complex agents. This may emerge, for instance, as a cross-connectivity pattern of the weights to the action units.

Finally, a model that comes closest to exhibiting sensory generalization in addition to spatial generalization is that of Zhu et al. (2016). The agent is trained to navigate in indoor scenes and can approach an object, which is shown to the agent at the start of navigation. It can thus generalize to novel objects and even to novel scenes. However, the addition of the components necessary for sensory generalization render this model far more complex than those that demonstrate a lesser degree of generalization. One potential solution is to use a pretrained object detection module that could potentially be shared with other navigation behaviors and aid sensory generalization in multiple processes.

3.1.2. Direction field navigation

In direction field navigation, the objective is to move at a specific angle relative to a local directional signal, such as a gradient, until the target is reached.

The extraction of a local directional signal from direction field can be done in two ways (Table 1). The first involves using the intensity and/or time of arrival of a signal in order to infer a gradient. This can be done by sampling and comparing the signal intensity at different locations (Brokaw et al., 2021) or by estimating the relative position of the source based on differences in arrival time or intensity when sampling simultaneously with two sensors (Carlini et al., 2024). An agent can then navigate by following the gradient of the signal, either moving towards increasing intensity (gradient ascent) or decreasing intensity (gradient descent) until the goal is reached. In robotics, olfactory navigation is largely inspired by how animals with two nostrils use their sense of smell to find food, detect hazards, and navigate their environment (Catania, 2013; Villarreal et al., 2016; Wang et al., 2023a). A key application of this in technology is finding the origin of gas leaks or other chemical emissions (Martinez et al., 2006; Soegiarto et al., 2022). Martinez et al. (2006) designed an experimental setup where a robot has to move in the direction in which the odor intensity increases. When the odor intensity reaches the maximum value, i.e. the robot cannot find a direction where the odor intensity increases, the robot stops. That point is considered the source of the leakage. Similar to most of the olfactory navigation models, the robot has two sensors, i.e. E-Noses, each containing an array of metal-oxide sensors, which were integrated with a spiking neural network for odor recognition. This model requires two sensors, consistent with findings from a study by Porter et al. (2007) which highlights that two nostrils in humans are more effective than one. The robot navigates towards the source using either binary or analog navigation methods. The binary navigation method makes a simple left-or-right decision based on which side detects a higher odor concentration, without considering the exact difference in concentration. In contrast, the analog navigation method relies on the precise difference in odor concentration between the two sides, enabling more gradual and fine-tuned turns. While the analog method performs well in stable environments with a smooth odor concentration gradient, leading to more controlled trajectories, the binary method proves more efficient in highly turbulent conditions, where the odor plume is fragmented and fluctuates rapidly. In such environments, the binary method allows for quicker and more robust decisions based on simple concentration comparisons. Switching between these two methods allows the agent to cope with different types of environments.

Following a similar concept, Huang et al. (1999) introduced a mobile robot that can navigate towards increasing sound intensity. The sound source following is based on the precedence effect model of the human auditory system (Carlini et al., 2024), allowing the robot to cope with echoes and reverberations. The system uses three microphones arranged in a triangular configuration and calculates inter-aural time differences to find the direction of the received sound. The results show that the robot can find the sounding objects in a reverberant environment and approach the objects without collisions, even when the objects were behind obstacles with background noise. The microphones basically function like the ears of humans and other animals. The major reasons that the model has three ears are the need to enhance directional accuracy, eliminate front–back confusion, and increase robustness to background noise.

The robotics models discussed above are engineered to be able to navigate to novel goal locations from novel start locations and adapt to reasonable environmental changes. That is, they exhibit a high degree of spatial generalization. Looking to these models in order to understand how mammals may achieve such generalization and identifying where any differences lie could provide insights for computational modelers interested in the neural mechanisms of direction field navigation. This is a promising avenue that could be pursued in the future.

The second method of extracting the directional signal involves directly inferring the gradient from environmental properties, such as altitude or magnetic field. Some animals, such as birds (Wiltschko and Wiltschko, 2023), fish (Lohmann et al., 2022), turtles (Lohmann et al., 2022), and mammals like rodents (Burda et al., 1990), bats (Holland et al., 2006), or controversially, humans (Baker, 1987) use the earth's magnetic field for navigation. They possess a sense known as *magnetoreception* (Lohmann et al., 2022). While we found no models of spatial navigation in mammals specifically based on magnetoreception, a few biologically inspired models of spatial navigation have been proposed for birds and fish that rely on the earth's magnetic field. For example, Zein et al. (2021) introduced multiple models of navigation using the magnetic field of the earth. They compare the simulation trajectories with GPS (Global Positioning system) data of the greater white-fronted geese migrating from their Russian Arctic breeding grounds to their European wintering sites. The models include geomagnetic taxis, where birds are guided by moving towards or away from a specific geomagnetic property, such as intensity or inclination. Another model is constant heading navigation (compass navigation), in which birds maintain a fixed direction relative to the earth's magnetic field, much like using a compass. The results indicate that simulations based on underlying gradients that minimize geomagnetic values (taxis), especially during autumn migration, were more likely to produce migratory trajectories closely resembling the actual observed geese migratory paths than other models.

Note that there is a difference in terminology that might cause confusion. In several direction field navigation models, the authors use the terms *localization* and *following*. The first is used to describe the process of finding a goal, such as the source of an odor or smell. In contrast, in this review paper, we define localization as the process by which the agent identifies the target's location from a distance (see Section 3.1.1) or the agent's own location (see Sections 3.1.5 and 3.1.6). The term "following" is used by others to mean moving in the direction where the gradient of that signal points. We use it differently in the context of path following (see Section 3.1.3).

3.1.3. Path following

Path following involves identifying the path leading to the goal and constantly adjusting the agent's position in order to stay on or near the path. The agent might move directly on the path, such as on a trail or street or keep a certain distance from it, such as when following a river. Path following is a fundamental behavior from both biological and computational perspectives. Biologically, a variety of species, including non-mammalian ones, such as snails, exhibit the ability to follow trails (Goss et al., 1990; Roessingh, 1989; Ng et al., 2013), and paths or trails form naturally in open spaces as a result of repeated use by people (Helbing et al., 1997) or animals (Croft, 2019).

Computationally, fairly basic algorithms enable robots to autonomously follow paths. Hung et al. (2023) review several of these algorithms and propose that in principle they all fall into one of two categories — those that minimize path following error either in the agent's reference frame and those that do so with respect to a reference frame attached to a point moving along the path. A notable example of the latter is the pure pursuit algorithm (Amidi and Thorpe, 1991; Coulter, 1992), where the agent constantly pursues a look-ahead point at a set distance ahead on the path. However, these solutions are highly technical, and their implementation details are challenging to relate directly to mammalian navigation. Whether the dichotomy put forth by Hung et al. (2023) also applies to mammalian navigation is an open question that can be addressed by future experimental research and computational modeling. Such a difference may be observable at the behavioral level. An algorithm like pure pursuit, which is able to detect curves and other variations in the path ahead and adjusts its course in advance, might map onto how animals use visual or other look-ahead information to stay on a path. On the other hand, detecting and correcting errors in a strictly body-based reference frame would mean that a deviation from the path is only corrected upon first encountering it, which may be analogous to thigmotaxis used by rodents in burrows and in the Morris Water Maze. The difference between these two computations may be subtle and less apparent while following relatively straight or smooth paths, but are likely to become more evident in paths that have sharp and sudden changes.

Perhaps due to the perceived simplicity of this behavior, in contrast to the relative abundance of path following models found in the robotics literature (Hung et al., 2023), there is a dearth of models that explain how biological agents other than simple organisms (Edelstein-Keshet, 1994; Sharpe and Webb, 1998) might achieve this behavior. This absence of modeling studies also limits our ability to assess generalization. However, the evidence from the robotics literature suggests that achieving spatial generalization in models may not be particularly challenging. We note that this is an important gap that can be addressed by future research.

3.1.4. Vector movement

Vector movement requires representing and updating a goal vector, i.e. the distance and direction to the goal from the agent.

Vijayabaskaran and Cheng (2024) study how the goal vector signal by a deep RL agent is used to navigate. At each time step, the agent receives a pre-computed goal vector and also a visual input. They find that using the goal vector to navigate is necessary for robust navigation in complete darkness or when vision is temporarily lost. This is consistent with our suggestion that vector movement acts as a fallback process that is continually active (Parra-Barrero et al., 2023). However, they show that this robustness comes at the expense of accurate navigation in cases where the goal vector is moderately noisy (for example, due to accumulating path integration errors). In addition, they find that both the goal vector signal and vision are responsible for place cells emerging in the model. The authors do not explicitly test for generalization in this model, however, given previous results that show generalization in a similar task (Vijayabaskaran and Cheng, 2022) and the fact that the model receives pre-computed goal vectors, it is likely that the model would exhibit generalization at least to novel start and goal locations.

While the model above uses pre-computed vectors, other models address the issue of how the brain might compute this goal vector. The prevailing view is that grid cells underlie this computation. For example, Banino et al. (2018) train a deep RL agent on goal-directed navigation tasks, including homing. They find that in order to successfully navigate to the goal, the agent must receive a "goal grid code", which is the activity of grid units in the network when the agent reaches the goal as additional input. The agent's performance suffers significantly in the absence of this extra input. This model thus provides indirect evidence that vector navigation is driven by a goal vector computation that compares the grid code at the current location to the grid code at the goal.

How might this comparison of the current and goal grid codes be executed? Bush et al. (2015) present an algorithmic solution based on the Fourier shift theorem and two general possibilities for neural network implementations (also see Stemmler et al., 2015). The first class of implementations computes the vector by using an additional decoding module to decode the grid cell activity at the current and goal location and then computes the distance between those two locations. The second class involves sampling trajectories between the current and goal location, either sweeping forward from the current location until the goal is encountered or changing head direction. Edvardsen et al.'s (2019) model is a recent example of the first type of implementation. The agent contains a topological map where snapshots of grid cell activities corresponding to each location are attached to the nodes. As the agent moves, the goal vector is computed and updated by a grid cell decoder (Edvardsen, 2018) which explicitly compares the grid cell activity of the agent at the current node to that at the goal node. The goal vector is thus extracted from a topological map of the environment that is enriched with coordinate information.

Erdem and Hasselmo (2012), Erdem et al. (2015) implement a mechanism of the second class in their model using a biologically plausible search algorithm for the rewarded location by sampling linear look-ahead trajectories on a combined grid cell and place cell map. Each virtual trajectory along a certain direction is initiated while the agent remains stationary and triggers a full chain from head direction cells, grid cells, place cells, and up to reward cells whose activities indicate an association with the reward signal. The agent then moves along the direction where a reward cell is found until it arrives at the corresponding location. Similar to Edvardsen et al.'s (2019) model, the vector is extracted from a map of the environment. Vector movement demonstrates a high degree of spatial generalization in these two models, since it can navigate between different start and goal locations based on the grid code.

A vector pointing from the agent to the goal does not necessarily require explicit representations of the two locations. In Goldschmidt et al. (2017), for example, the model directly maintains a homing vector pointing from the home location to the agent's current location by using a layer of memory neurons which integrate the velocity modulated signals of the head direction neurons, i.e. a path integration algorithm. As the agent encounters a rewarded goal, the homing vector at the moment is copied onto another array of vector cells by employing a reward-modulated associative learning rule. Later, the vector from the agent's current location to the goal can be computed by subtracting the current homing vector from the stored vector.

In addition to computing the goal vector, vector movement also requires the agent to update it as it moves towards the goal. For example, if the agent knows that the goal is 20 meters to the north-east at the start of navigation, it must be able to keep track of its motion along this vector in order to know when it has traveled 20 meters and arrived at the goal. This can be accomplished by path integration, which is often modeled using grid cells (see Section 3.1.6).

In general, vector movement is one of the behaviors that has received relatively more attention from the modeling community, largely propelled by interest in uncovering the computational role of grid cells. However, there are still some important questions that remain unanswered. For instance, the disproportionate focus on grid cells means that other potential substrates for vector movement have not been fully explored, such as cells explicitly encoding the vector to the goal which have been discovered in bats (Sarel et al., 2017). Another question is whether the goal vector is directly represented and updated, or if there are separate vectors representing the current and goal locations, which are then subtracted to compute the goal vector. One can also ask, if the vector is represented in a Cartesian or polar coordinate system, which would manifest as differences in the distribution of errors made by the agent.

3.1.5. Guidance

Guidance is defined as navigating to a goal location by using information about its spatial relationship from landmarks and distal orienting cues. When designing a task to study guidance, the most obvious choice is to require the agent to navigate to an unmarked goal location in a maze surrounded by distal cues as, e.g. in the Morris water maze (Morris et al., 1982). As for representations, many computational models of guidance focus on place cells that encode spatial location based on configural cue (Shapiro et al., 1997).

Sheynikhovich et al. (2009) presented a computational model of rodent navigation that consists of two pathways to solve the task. The first pathway contains a visual module that processes visual input from the environment and an egocentric action module that generates actions by receiving the processed visual information. The second pathway starts from the self-motion signals, passes through a grid cell module and a place cell module, and eventually to another allocentric action module. Importantly, learning occurs in both pathways but only in the immediate connections to the action modules. In the simulations, the second pathway clearly outperformed the first one, suggesting an advantage of having an explicit encoding of space (place cell module). Several studies (Ghazinouri and Cheng, 2025; Ghazinouri et al., 2024; Scleidorovich et al., 2020; Brzosko et al., 2017) use a two-layer spiking neural network to demonstrate the benefits of having a place cell module in the Morris water maze task. The first (input) layer is a purely spatial representation, where each unit mimics a place cell. As a result, the population activity pattern represents the position of the animal. The second layer is the action selection layer. The policy is learned by adjusting the feed-forward synaptic weights between two layers by using reward-triggered STDP learning rules. The above models have demonstrated that a place cell module together with an action selection model can solve a guidance task.

These models probe the representations and computations underlying guidance. Agents learn the location of the goal and localize themselves (using place cells in the above models) and use this information to move towards the goal. To avoid this explicit computation, the result of that computation is learned and stored for each location, such that the appropriate action is triggered when the location is reached. For example, all the models in the preceding paragraph use an explicit encoding of space using place cells in their first layers and encode a preferred moving direction for each location in the connection weights from place cells to the action neurons. This shortcut has severe consequences. Since there is no explicit representation of the goal location and appropriate computation of the movement direction, the agents can only navigate to one goal location at any given time. Navigation to another goal requires reversal learning (Ghazinouri and Cheng, 2025). Similar representations and computations emerge in deep RL agents that receive only camera images as inputs (Vijayabaskaran and Cheng, 2022). As the spatial location is not supplied as an input, the agent develops a spatial representation internally. Similar to the neural network models, model-free RL models also associate an action with each location. The authors test this model on its ability to navigate from novel start locations, and show that place cell-like representations facilitate the ability of the agent to demonstrate spatial generalization. However, not all aspects of spatial generalization were probed, since the distal cues were not moved, rotated, or otherwise manipulated.

Other models take a view-matching approach to guidance, inspired by insect navigation. For instance, Baddeley et al. (2012) proposed an extreme case of representing the goal with visual information: the goal is stored as a panoramic snapshot of where the agent receives the reward, where the snapshot includes both distal and local cues. During navigation, the agent keeps minimizing the errors between the goal image and rotated images from the nearby locations to go back to the goal location. Due to the over-simplification of the view-matching algorithm, the agent can only reach those locations that are quite close to its current location. To reach a previously visited goal that is far away, the agent maintains a sequence of snapshots connected to each other along the route from the start to the goal location, so that it can move from one "sub-goal" to the next in order to reach the final goal. This means that guidance by view matching only happens between successive subgoals along a route. Thus, this model exhibits a low degree of spatial generalization when the goal is far away, because it can only reach the goal by starting from one of the sub-goals along a stored route. However, whether this is a problem of all view-matching algorithms is not clear. One possibility could be that these algorithms need to be combined with an exploration module to effectively find the goal from different starting locations.

For a navigation behavior to qualify as guidance in our taxonomy, the agent has to be able to generalize its behavior to a different goal locations. This requires the explicit calculation of a movement direction by subtracting the current location from the goal location. For instance, in the deep RL model by Cruse (2003), a recurrent network receives vectors that describe the relationship between the landmarks themselves and between the landmarks and the goal as inputs. Although the network has a relatively simple architecture, it uses built-in redundancies to be able to continue moving towards the goal even when a landmark becomes invisible for a while, which is crucial in real world navigation. Frommberger (2008) used a vector of positions of walls relative to the agent as input representations in a Q-learning agent. These representations speed up learning compared to a similar agent that relied on a metric representation (x-y coordinates and head direction). The policy learned in the landmark-based agent also transferred to novel environments, showing some spatial generalization. Mirowski et al. (2019) showed that a similar approach, using only visual input and goal encoding as a vector of distances to a set of landmarks, can be extended to navigation in large cities such as London, Paris, and New York. These methods can generalize quite well since they rely on explicit representations of the relationships to landmarks. How these representations are extracted, i.e. the extraction phase of the behavior, is however not considered by these models.

There are multiple possible ways of execution, but the general criterion is that the computation should guide the agent such that the representations of the goal location and self-location become more similar with each step.

3.1.6. *Path integration*

Path integration is a process that allows agents to keep track of their position with respect to a reference point. Although it is not a separate navigation process within our taxonomy, path integration may be used by several of the processes to self-localize when needed. Given its role as a fundamental module shared across different processes, we briefly review models of path integration separately here.

One approach to learning path integration is to train a recurrent neural network using supervised learning (Cueva and Wei, 2018; Banino et al., 2018). That is, the network takes a linear velocity signal as input and, given an initial location, uses it to estimate the agent's current location at every time step. Interestingly, units that look like grid cells emerge naturally in these network (but see Kanitscheider and Fiete, 2017). These studies, however, do not explain why the emergent grid pattern is sometimes square and sometimes hexagonal in a square environment (theoretical work suggests that hexagonal grids are optimal for spatial coding in 2D (Wei et al., 2015)). Other observations, such as grid scale grouping into discrete modules in the network (Banino et al., 2018), similar to that in rodents (Stensola et al., 2012), and the need for regularization for grid patterns to arise, have also yet to be analyzed. In key work, Sorscher et al. (2022) show that using simple biologically inspired constraints, such as using place cells with surround inhibition as outputs and non-negative firing of hidden units, causes the grids in path-integration networks to become more hexagonal. They further examine these networks to demonstrate that attractor dynamics underlie path integration in these trained recurrent networks, establishing a clear relationship to previous mechanistic models of path integration (Samsonovich and McNaughton, 1997; Fuhs and Touretzky, 2006; Burak and Fiete, 2009).

Finally, in a setup similar to the recurrent neural network models discussed above but using angular velocity signals as input, Cueva et al. (2020) demonstrate that cells that emerge in the network share properties with head direction cells and shifter cells that jointly code for head direction and angular velocity. This is consistent with experimental evidence that head direction cells play a role in angular path integration (Valerio and Taube, 2012) and may suggest a division of labor between grid cells and head direction cells in integrating the linear and angular components of self motion.

3.2. *Computational models of navigation strategies*

In our taxonomy navigation strategies are processes at the higher level that plan a path to a goal by identifying a sequence of sub-goals. Here we also include models that generate a behavior or even direct action signals as output, if these models provide interesting insights.

3.2.1. *Route navigation*

In route navigation, a sequence subgoals is stored, so that arrival at a subgoal triggers the next subgoal. The dichotomy between route navigation and map navigation is often mapped onto the distinction between model-free and model-based RL (Chavarriaga et al., 2005; Daw et al., 2005; Dollé et al., 2010). However, we believe that model-free RL does not map directly to a particular behavior or strategy within the taxonomy. Depending on the implementation, representation, task, and other factors, model-free RL could be used to model a variety of navigation processes (for discussion of this issue, see Section 4.3). Instead, hierarchical approaches to RL (Dayan, 1993; Wiering and Schmidhuber, 1997; Sutton et al., 1999; Precup, 2000; Stolle and Precup, 2002) are a better match to our account of route navigation, and have also received considerable attention in neuroscience (Botvinick et al., 2009; Ribas-Fernandes et al., 2011; Botvinick, 2012; Diuk et al., 2013). Hierarchical RL is a framework that organizes learning and decision-making processes into a hierarchy of tasks or subtasks. Unlike traditional RL, which focuses on finding a flat policy that maps states to actions, hierarchical RL decomposes the problem into multiple levels of abstraction (Barto and Mahadevan, 2003; Rasmussen et al., 2017). Commonly a two-tiered hierarchy is used, which is analogous to the hierarchical relationship between navigation strategies and behaviors in our taxonomy. In the following, we discuss the parallels between route navigation and two widely studied approaches to hierarchical RL, the *options* framework (Sutton et al., 1999; Precup, 2000; Stolle and Precup, 2002) and goal-conditioned hierarchical RL (Nachum et al., 2018, 2019b).

In the options framework, a higher-level agent makes a discrete choice between a number of available lower-level behaviors, or *options*, which leads to the execution of a high-level action. Viewed through the lens of our taxonomy, these low-level policies correspond to the navigation behaviors, which share a hierarchical relationship to the higher-level agent, which corresponds to the navigation strategy, in this case, route navigation. However, since the higher-level agent directly chooses a behavior, rather than outputting a sub-goal, the options framework more closely mirrors the recognition-triggered response account of Franz and Mallot (2000). Alternatively, the options framework could correspond to the organization of behaviors in our taxonomy. This possibility is explored in detail later in Section 3.3.

The RL framework that most closely resembles our view of route navigation is perhaps goal-conditioned hierarchical RL (Nachum et al., 2018, 2019b; Wang et al., 2023b). Here, the higher-level agent outputs a sub-goal, which must be reached by the lower-level agent. This lower-level agent can be seen as analogous to the organization of behaviors in our taxonomy. Upon arriving at the subgoal, the lower-level agent hands control back to the higher-level agent which then sets the next subgoal. Building on this concept, Nachum et al. (2019a) introduce a novel approach to enhance hierarchical RL by focusing on near-optimal representation learning. The agent learns a transformation from raw observations into a structured target space that effectively capture the environment's structure and that is useful for the specific tasks that the hierarchical RL agents are faced with. The quality of this goal representation directly impacts the efficiency and optimality of the learned hierarchical policies since they determine what the low-level policy tries to achieve. This approach enables the agent to abstract relevant subgoals and optimize their behavior. More recently, Schiewer et al. (2024) introduced a novel hierarchical model-based RL framework that constructs hierarchical world models, simulating environment dynamics at various levels of temporal abstraction. This approach enables a stack of agents to communicate in a top-down manner by proposing goals to their subordinate agents. This approach allows all agents and models in the hierarchy to be trained simultaneously. This also keeps the decision-making process simpler by reducing the number of abstract actions needed. While the hierarchical model-based RL approach did not achieve higher final episode returns than traditional methods, it effectively enabled decision-making across two levels of abstraction by utilizing compact, low-dimensional abstract actions.

Although we have only discussed two-tier hierarchies so far, one could also imagine that three-tier hierarchies could be used to describe navigation, with the top level corresponding to navigation strategy, the middle level corresponding to the organization of navigation behaviors, and the lowest level to the actual behaviors. Although we did not find any 3-tiered hierarchical models that implement this precise setup, some researchers have used 3-tiered hierarchical models to implement navigation in a different form. For example, Levy et al. (2019) use a three-tier agent to navigate in a four-room environment. When this agent is given the task of navigating between two rooms separated by an intermediate room, the top-level agent first assigns a sub-goal to the middle agent in the intermediate room. The middle agent then selects the points that form a trajectory to the sub-goal, and the lowest level calculates the joint torques required to move the agent from one point to the next until the sub-goal is reached. The key difference to the taxonomy is that the middle agent in this case implements a single behavior, and the lowest agent deals with the calculation of detailed movement instructions, which we do not cover in our taxonomy.

A significant advantage of using hierarchical RL is that higher-level actions are semantically meaningful, aiding in the interpretation of otherwise black-box-like agents. For example, in learning to navigate in mazes, Pertsch et al.'s 2020 RL agent learned two useful skills — walking through narrow doorways, and following corridors, which seem to correspond to aiming and path following in our taxonomy. Thus, deconstructing complex navigation tasks using hierarchical RL and looking for parallels between learned policies and elementary navigation behaviors is a promising area for future research.

3.2.2. *Topological and metric navigation*

In this section, we refer to topological and metric navigation collectively as map navigation, following the taxonomy (Parra-Barrero et al., 2023). In topological navigation, the underlying 'map' takes the form of a graph and contains only connectivity information between nodes, much like a subway map. In metric navigation, the map contains a globally consistent metric, like a city map. We believe that topological and metric navigation are two extremes rather than distinct categories, and most models reviewed in this section fall somewhere between these two extremes. We therefore combine the review of topological and metric navigation.

As described in Section 2, there are two aspects that define the navigation strategies — the spatial knowledge structure (in this case the map) and the planning algorithm that operates on it. Hence, we discuss the models in this section mainly based on the implementation of the map in the models, with a focus on the planning mechanisms. Since map navigation is the most extensively modeled among the processes in our taxonomy, we divide this section into two parts: biologically plausible models and RL-based models.

3.2.2.1. Biologically plausible models. Place cells, and more recently, grid cells, are often regarded as the neural basis of the cognitive map. This view, however, is driven primarily by findings in rodents, and the picture in humans is very likely more complex. Overall, this topic remains the subject of scientific debate and ongoing research (see Ekstrom and Hill, 2023, for a review). While grid cells have been proposed to underlie metric navigation, the view on place cells has been more mixed. While many uphold a metric view of them (O'Keefe and Nadel, 1978), others propose that they form a topological map of the environment (Dabaghian et al., 2014; Dabaghian, 2019). In the context of planning, two neural mechanisms have been proposed as possible candidates: theta sequences (Parra-Barrero and Cheng, 2023) and sharp wave ripple activity, such as replay and preplay of sequences (Buhry et al., 2011; Pfeiffer and Foster, 2013). Consequently, biologically plausible models of map navigation have largely focused on how place and grid cells could be used to guide navigation, predominantly using replay.

The question of how place cell maps may be used to plan and navigate has been considered in models quite early on. For instance, Muller et al. (1996) propose that a place cell map takes the form of a directed, weighted topological graph, with the place cells as nodes and the synapses between them as edges. The synaptic weights thus correspond to the edge weights of the graph, and a graph planning algorithm (the authors suggest Dijkstra (1959)) is used to plan a route to the goal. While this model clearly leans towards the topological side of the spectrum, as evidenced by the fact that it would still need to explore a new route when it becomes available before it can plan using it, the model also has a metric element to it. Specifically, the synaptic strength between place cells in the model decays with the Euclidean distance between the field centers, thus incorporating metric distance information.

Most other models based on place cells do not explicitly align with a topological or metric interpretation. For instance, Gao (2023) proposes a continuous attractor network model of place cells a model where the place cell map is not pre-wired, but formed during a period of random exploration. In this model, place cells whose fields are sequentially visited by the agent are also sequentially connected. Additionally, a second layer of striatal neurons encodes the geodesic distance between place cells and the reward location. Thus, the map encodes some metric information in the form of distance to the goal. However, the distance between arbitrary locations is not encoded, since unlike the model of Muller et al. (1996), the synaptic weights between place cells are not correlated with the Euclidean distance between their place field centers. The planning mechanism in this model incorporates both awake and offline replay. During awake replay, the agent samples multiple forward trajectories akin to forward replay that originates from the agent's current location. The agent then selects the trajectory with the highest probability of reaching the reward location, based on the activity of the striatal neurons. In the offline replay phase, external inputs to the network are removed, and the synaptic weights between place cells and striatal neurons are updated based on network activity during this phase. Interestingly, although the model does not include an explicit randomization mechanism, the offline replay trajectories distribute fairly uniformly across the environment.

Despite not explicitly encoding complete metric relationships, Gao's (2023) model demonstrates a substantial degree of planning generalization. The agent exhibits flexibility by adapting to environmental changes such as blocked paths or newly available routes. However, it relies on re-exploration followed by offline replay to do so, suggesting that this model aligns more closely with the topological end of the spectrum.

A closely related model is that of Gönner et al. (2017), which also uses a continuous attractor network model to simulate place cells in the CA3 region of the hippocampus. An additional layer of place cells, representing the dentate gyrus, projects to the CA3 layer. The model also includes some metric information: inter-place-cell synaptic weights are pre-configured based on the Euclidean distance between field centers. In addition, like in the model by Gao (2023), the agent also encodes distances between place cells and the reward location in the synapses between the dentate gyrus and a separate context layer representing the prefrontal cortex (which we do not discuss here). Thus, the model encodes both distances to the goal as well as distances between place fields in its synaptic weights. The planning mechanism in this model differs in that it generates a single place-cell sequence at the start of navigation, occurring on a compressed timescale. The endpoint of this sequence serves as a subgoal for vector movement, and this is carried out until the goal is reached. Notably, the sequences generated by the model are biased by the rewarded location, and can span through previously unvisited trajectories, again suggesting a metric component. However, unlike Gao's (2023) study, the agent was not tested on its ability to take shortcuts or detours, which makes it difficult to gauge the level of planning generalization that this agent supports.

The two models discussed so far build maps of the environment by relying on a place cell module. Other models go a step further by including grid cells, which have been hypothesized to function as a metric system (Dang et al., 2021; Ginosar et al., 2023). For example, the model proposed by Erdem and Hasselmo (2012) consists of a chain of layers of head direction, grid, place cells and prefrontal cortical columns. The place cells encode pure location information, which is enriched with neighborhood connectivity by the prefrontal units, thus encoding the topology of the environment. The map thus consists of metric information from the grid cells and topological information from the place and prefrontal units. The planning mechanism relies on forward linear look-ahead trajectory probes. The head direction activity shifts the spiking phase of grid cells, sequentially reading out potential future trajectories from the agent's current position. In addition, the planning mechanism also relies on a reward signal that originates at the goal and diffuses through the place cell map. The model then selects the look-ahead trajectory that activates the place cells with the highest reward signal. The authors demonstrate that these two mechanisms — linear look-ahead and reward diffusion — enable the agent to navigate along novel routes and take shortcuts, making this an instantiation of metric navigation under our taxonomy. A limitation of this model, however, is that these look-ahead computations occur at fixed intervals, meaning that subgoals are always selected at a uniform distance from the previous location. This rigidity does not fully capture the flexible subgoal selection that we propose.

Similarly, the model of Edvardsen et al. (2019) also integrates both place cell and grid cell modules to build the internal map. The place cell module is implemented by an abstract graph where each node contains a snapshot of the grid cell activity associated with that location, while the firing patterns of the grid cells encode the coordinates of locations within the environment. The connections between the nodes are formed during a phase of random exploration. The place cell map thus takes on the structure of a topological graph that has coordinate information associated with each node, and the grid cell network forms a metric map of the environment. During goal-directed navigation, a place cell (graph node) is associated with a reward signal. Planning in the agent involves both the grid and place cell modules. First, a grid cell decoder is used to compute a vector pointing from the agent to the reward location by using the grid cell patterns associated with the current node and the reward node. Thus, in an open field without obstacles, the agent would plan a direct path to the goal using the grid cell network. The place cell module kicks in only when the computed vector leads the agent to an obstacle that cannot be overcome by simple obstacle avoidance. Planning on the place cell module involves a backward replay mechanism that looks for a subgoal that leads the agent around the obstacle. This subgoal is then again reached by directly planning a path to it using the grid cell module. Thus, in cluttered environments, both the topological and metric components interact to plan a path to the final goal.

One clear takeaway from the discussion above is that modeling efforts have almost exclusively focused on place and grid cells as the *de facto* neural substrates of map navigation. While their regular and readily interpretable firing properties make these cell types attractive candidates for modeling, experimental evidence increasingly points to distributed and mixed representations (Rigotti et al., 2013; Ekstrom et al., 2020; Parra-Barrero et al., 2023; Maisson et al., 2023). Thus, one future research direction for modelers is to incorporate these findings into computational models that better capture the complexity and flexibility of map navigation.

3.2.2.2. Robotics and RL models. In this section, we discuss robotics and RL-based models of navigation. We first discuss broadly on how these models approach the representation of the map and planning and then analyze specific computational models in terms of the map structure and planning mechanisms they use.

Most studies of the Simultaneous Localization And Mapping (SLAM) problem are driven by engineering demands and they might, therefore, seem rather remote to this review. However, some approaches to SLAM use more biologically plausible algorithms (Milford et al., 2004; de Souza Muñoz et al., 2022) and some studies even use neural networks to implement similar algorithms (Zhao et al., 2022). The approach of having both topological and metric representations, discussed in the previous section for biologically plausible models, has proven fruitful in the field of robot SLAM, too. There are two primary strategies for building such hybrid maps in SLAM. The first strategy starts with a global metric map, which is later simplified into a topological representation using clustering or graph partitioning (Thrun, 1998; Zivkovic et al., 2006). This results in a map representation analogous to those proposed by some of the biologically plausible models (Erdem and Hasselmo, 2012; Edvardsen et al., 2019). An alternative proposal favored by many is that of a large-scale topological map with embedded local metric maps (Poucet, 1993; Meilinger, 2008; Parra-Barrero et al., 2023) that may be deformed (Parra-Barrero et al., 2023; Lynch, 1964). This approach in robotics is inspired by the spatial semantic hierarchy theory (Kuipers, 2000a), and builds a global topological map first and attaches local metric maps to its nodes. Unlike global metric maps, these local maps do not share a consistent reference frame; instead, an agent is localized relative to its current map.

Planning in SLAM-based models is typically computed using graph search algorithms such as A* or Dijkstra's algorithm (Hart et al., 1968; Dijkstra, 1959), which find an optimal route between two nodes in the topological map. Once the path is planned on the global scale, local movement, is handled using a navigation stack of different algorithms such as wall- or path-following and obstacle avoidance. Since, by definition, these models simultaneously build maps of unexplored territory, they exhibit a high degree of planning generalization.

Although these approaches are primarily technical solutions for map-building, localization and navigation, there are some valuable lessons for those interested in mammalian navigation. For one, they neatly demonstrate the role that spatial scale plays in the type of map that is most effective for navigation. While at smaller scales, metric maps are feasible, large-scale planning and navigation are much more efficient when using topological maps. For instance, Konolige et al. (2011) demonstrate that at short distances, planning times for metric and topological maps are nearly identical. However, as the scale increases, the differences become much more pronounced, with planning on a metric map taking roughly seven times longer than on a topological map at larger scales. Notably, while the planning time varies significantly, the actual length of the planned path remains largely similar between the two map types. In addition, these approaches demonstrate where metric maps become essential — planning and navigating in environments that require flexibility and precise localization, such as cluttered indoor spaces, consistent with Edvardsen et al.'s (2019) findings.

While SLAM methods focus on constructing the maps that enable navigation and accurate localization, RL models address the question of how map navigation could be learned and optimized using interactions with the environment. Model-based RL (Sutton and Barto, 1998) is the most obvious candidate for modeling map navigation, since it constructs an explicit model of the environment, which corresponds to the map. An alternative representation in RL-based navigation is the successor representation, which encodes the expected future occupancy of states (Stachenfeld et al., 2017; Menezes et al., 2025). The predictive horizon of successor representation is controlled by a discount factor, which determines how far into the future state relationships are considered relevant. The structure of successor representation can resemble either a metric or topological map, depending on how the states are represented. If the states are defined on a regular grid by the modeler, the successor representation will reflect metric relationships up to a distance controlled by the discount factor. One can also construct a hierarchical version of successor representation using different predictive horizons to represent different levels of the hierarchy (Stachenfeld et al., 2017), and there is some experimental evidence that this hierarchical version is able to fit human behavior (Wientjes and Holroyd, 2024).

Broadly speaking, planning in RL models can be categorized into online (decision-time) planning and offline (policy-learning) planning. Decision-time planning corresponds to a more traditional view of planning that involves evaluating possible future trajectories based on the map and selecting the optimal one based on some cost function. This approach exhibits a high degree of planning generalization, since it can flexibly re-plan trajectories if, for instance, paths are blocked. However, a major drawback of using decision-time planning on the map is the computational cost, which increases exponentially as the size of the map grows. This can be mitigated by using offline planning, i.e. using the model exclusively to generate simulated experiences, which are used to improve the policy learned by a model-free controller, like the DYNA family of models (Sutton et al., 1999). This also means that, if the environmental structure or goals change, the policy can be adjusted accordingly based on simulated experiences.

Of course, online and offline planning are not mutually exclusive. Indeed, highly successful deep RL algorithms like AlphaZero (Silver et al., 2016, 2018) and MuZero (Schrittwieser et al., 2020) combine them, using decision-time planning via Monte Carlo tree search, as well as offline learning using the results of the Monte Carlo tree search to improve its model. However, these models have not been explicitly tested on navigation tasks to the best of our knowledge, although they should extend quite naturally to them.

Planning in the successor representation framework differs from the notions of planning we discussed above, and consists of two different aspects. First, since the successor representation essentially captures a compressed version of the environment's transition dynamics, long-term values can be computed quickly, even for states that are far away. This mimics the forward simulation aspect of online planning, although alternative trajectories are not explored and no explicit cost computation takes place. Second, since the reward function is decoupled from the predictive map, the algorithm can flexibly adapt to changes in reward contingencies, e.g. if the goal is moved, which resembles re-planning in traditional algorithms. However, changes in the environmental structure take longer to learn and adapt to, and the previous policy can strongly bias the behavior, making it less flexible than traditional online planning algorithms.

These different notions of planning highlight the trade-off between flexibility and computational complexity. While online planning exhibits a high degree of planning generalization, it has a high computational cost. On the other hand, other notions of planning, such as offline planning, which effectively re-frames planning as learning from replay (Mattar and Daw, 2018; Diekmann and Cheng, 2023), and planning in the successor representation (Momennejad et al., 2017), exhibit less planning generalization but are computationally far more efficient. Note that this trade-off may not apply to biological systems in the same way as it does in RL — for instance, one model demonstrates that rapid and parallel path planning can be executed relatively easily in biological networks (Ponulak and Hopfield, 2013).

To round-off this section, we now turn to a discussion of some representative RL models of map navigation. Models that use the map for planning at decision time typically use a topological graph-like representation to plan on. This is because these models typically represent the environment as a set of discrete state transitions or other structured format. For example, Savinov et al. (2018) propose a semi-parametric topological memory architecture that stores a topological graph containing connectivity information from various locations in the environment in its memory. The agent has a retrieval network that takes the current visual observation and the observation at the goal and outputs the corresponding nodes in the topological map, thus establishing a connection between visual observations of the environment and the stored topological map. As mentioned before, planning is done at decision time using Dijkstra's algorithm (Dijkstra, 1959). Given this path, the semi-parametric topological memory agent calculates a waypoint, or subgoal, on the path and outputs the corresponding observation. This observation is then used by a separate locomotion network that generates the actions required to move from one subgoal to the next until the final destination is reached. Thus, the semi-parametric topological memory network, like our conception of map navigation, plans a path to the goal and outputs subgoals, and the attainment of each subgoal is handled by another process — in this case, the locomotion network.

In a somewhat similar approach, Eysenbach et al. (2019) use the replay memory buffer of a deep RL agent to construct a topological graph on which to navigate, known as "Search on the Replay Buffer". The replay memory refers to a memory of all experiences that the agent has had in the environment, including an environmental state (usually the visual observation at each location), the action taken at that state, and the reward and next state resulting from the action. A topological graph is built on top of the states in the replay memory, thus connecting the actual environmental observations and the map. The graph is weighted by using the values of the states to predict the distance between them, and using the predicted distances as weights associated with the edges. Like the semi-parametric topological memory agent (Savinov et al., 2018), a planning algorithm such as Dijkstra's algorithm is used on this graph to plan a path and calculate subgoals.

Finally, a notable example of models that use offline planning is the Dreamer class of models (Hafner et al., 2019, 2020), which use a latent space world model to generate synthetic data, which it uses to improve its policy. An interesting feature of this model is that the simulated experiences are based on the latent space states, unlike other forms of experience replay directly replay the sensory states (Lin, 1993; Mnih et al., 2015; Diekmann and Cheng, 2023). This has parallels to how replay in the brain involves place cells (a latent space representation) rather than direct sensory experiences.

3.3. Computational models of the organization of behaviors and strategies

In most navigation computational models, the strategy and behavior of navigation are predetermined i.e. hard coded. However, in some models, at the start of a navigation task, the agent can choose a strategy and during the task can change the type of navigation behavior. In this section, we review models where the agent dynamically selects a strategy and/or behavior during navigation tasks.

3.3.1. Organization of behaviors

In our taxonomy, an agent can either select a behavior or integrate competing behaviors. Integration can occur either by combining sensory information from the goal and/or environment (i.e., representations) in the extraction stage or by integrating movement signals in the execution stage. We categorize models into three main classes: hard-coded behavior selection, learned behavior selection, and integrated signal models.

In models with hard-coded behavior selection, behaviors are selected based on a fixed algorithm. For instance, the agent controller might operate as a state machine, determining which behavior is active at any given time (Edvardsen et al., 2019). In this model, the agent employs vector navigation to calculate direct paths in open fields, but switches to topological navigation in cluttered environments. In doing so, the model adaptively switches from relying on grid to place cells, demonstrating dynamic interactions between grid and place cell systems in varying environments. The model of rodent navigation by Chavarriaga et al. (2005) incorporates a striatum-based response system for cue-driven, habitual navigation (aiming) and a hippocampus-based system for flexible, exploratory navigation (guidance). Their model replicates observed changes in rodent strategy under changing environmental familiarity and complexity (Goodroe et al., 2018).

To overcome the limitations of hard-coded models, the second class of models learn the behavior selection process. For instance, in the models of Dollé et al. (2010) and Sheynikhovich and Arleo (2010) agents must choose between a taxon and locale strategy, closest to aiming and guidance in our taxonomy. The switching is controlled by gating units that control the probability of either module being selected over the other. The individual modules as well as the gating network are trained via RL. As a result, gating, i.e. behavior selection, becomes dependent on the performance of the modules. Although each module proposes an action at each time step, selection only occurs when the accumulated prediction error of the currently active module crosses a threshold (dynamic switching). Although, at a given time step, an action is chosen only from one module, the other modules participate in learning about the goal position. Another example is the hierarchical RL model (see Section 3.2.1), where behaviors — referred to as options — can be learned through experience, facilitating flexible and hierarchical navigation, particularly in complex or dynamic environments (Wang et al., 2023b).

The third class of models integrate behaviors by combining signals weighted by its reliability in a given environment (Cheng et al., 2007), often using Bayesian integration. Vijayabaskaran and Cheng (2024) exemplify this approach, showing that the integration or selection of processes depends on the robustness of the input signal to noise. When input representations are noisy, the agent favors processes that use more robust signals. At lower to moderate noise levels, behaviors integrate cooperatively rather than compete. At high noise, however, the unreliable behavior is ignored, which amounts to permanent selection of the more reliable signal. The form of integration suggested by Vijayabaskaran and Cheng (2024) applies to the extraction stage, when goal-related information is processed, not the execution stage. We did not find any models that integrate behaviors at the execution stage, i.e. where an agent executes a weighted combination of behaviors simultaneously rather than switching discretely between them.

3.3.2. Organization of strategies

An agent can rely on multiple navigation strategies (see Section 2.3) to determine the final or an intermediate goal during a navigation task. Before discussing the computational models of how the strategies are organized, it is useful to first consider the scenarios in which the agent needs to switch from one strategy to another. In principle, if the agent possesses an accurate, complete metric map of the environment with detailed spatial information of obstacles and landmarks and has sufficient computational resources, it does not need to invoke other navigation strategies other than planning over the metric map. However, in most cases an agent or animal builds the map of the environment by itself, which covers only parts of the environment and might suffer from the noisy measurements of the space due to inaccurate sensors. Hence, switching to topological or even route navigation becomes necessary in certain occasions. Also an agent might have limited computational resources, e.g. due to inherent limitations or time pressure, so that the agent might rely on route navigation when the route is reliable, because the other two strategies require planing over a map and, thus, require more computations.

Franz et al. (2008) developed a robotic system that utilizes all three navigation strategies during the process of building a metric map of the environment. At the behavioral level, the agent stores a snapshot of the inverse distances to the surrounding objects at a particular location (the “disparity signature”) to mark that location. It then moves to another location by comparing the current signature to a stored one which has been memorized within a certain neighborhood. In a new environment, the robot begins by random exploration, marking and connecting a sequence of locations to form a route. The robot is then able to perform route navigation starting from any location in a stored route. Whenever two separate routes are detected to share one or more same locations, they are integrated. After a certain amount of exploration, multiple routes will be connected and form a topological representation of the environment. The agent then utilizes standard graph planning algorithms to find available routes to a given goal location. The topological graph is only used for planning and the robot immediately switches to route navigation once it finds a route to the goal to save computational resources. A general problem in topological navigation, perceptual aliasing, occurs when multiple nodes in the graph have similar stored snapshots and there is ambiguity in self-localization. To solve this problem, Franz et al. (2008) embedded a global metric into the graph and included odometry signals of the robot into the disparity signature, so that nodes become more distinguishable from each other. This metric information also enables the robot to plan a route over unexplored terrain in the environment. Although Franz et al. (2008) did not systematically test the capabilities of the robot in goal-directed navigation, but rather focus on the map of the environment, it presents a system that naturally switches from route navigation, to topological, and then to metric planning as the map develops.

As Franz et al. (2008) and many other studies of robotic navigation already pointed out, a general issue in building a metric map for large environments is that errors accumulate in the distance sensors, such that metric information is inconsistent across the different parts of the environment. Kuipers (2000b) and Beeson et al. (2010) introduced and tested a navigation framework, where they break down the mapping problem in large environments into building metric representations in local, small-scale spaces and topological representation in global, large-scale space. The same arrangement was previously suggested to be used by mammals as well (Parra-Barrero et al., 2023). The local metric maps provide precise spatial information necessary for tasks such as obstacle avoidance and accurate positioning. On the other hand, as the robot navigates through the environment, it identifies significant places (e.g., intersections, doorways) and paths, build an abstract representation of the environment as a topological map. During long-distance planning, the robot uses the topological map to determine the sequence of nodes and edges to reach a goal efficiently (e.g., room A → corridor → elevator → room B). When the robot reaches a node in the topological map, it uses the local metric map to navigate through that specific place accurately. In addition, the robot periodically uses the local metric map to confirm it is on the correct path described by the topological map, adjusting for deviations caused by obstacles or errors. In this way of organization, the robot can navigate through long distances with high accuracy without the need to maintain a global metric representation of the entire environment.

These two examples demonstrate the benefits of switching between different navigation strategies to compensate the errors caused by sensors and the lack of complete maps in large environments, or to save computational resources to make navigation more scalable. There are two interesting observations worth mentioning. First, topological and metric maps of an environment are usually built and maintained in a interleaved way, either in a hierarchy (Kuipers (2000a), global to local) or as overlays (Franz et al., 2008). Second, route navigation naturally follows after the planning stage of the topological navigation to avoid planning multiple times.

4. A synthesis of key insights

In the following sections, we discuss the key insights gained from our review of the literature on computational modeling of spatial navigation in mammals and their broader implications for understanding navigation processes.

4.1. What have we learned about the state of the field?

Reviewing the landscape of computational models of navigation using our taxonomy as a backbone has led to several broader insights about the state of research in this field. First, it became apparent that some navigation processes are less frequently modeled and may warrant more attention. A striking example of this is the seemingly straightforward process of path following. While this process has received a lot of attention from the perspective of technical solutions, evidenced by several hundred robotics papers proposing various methods (Hung et al., 2023), there is a notable lack of models exploring how mammals may accomplish path following. Other understudied processes are route navigation as well as the organization of behaviors and strategies. The flip side of this state of affairs is that other processes are overrepresented in the literature, e.g. map navigation and guidance (Best et al., 2001; Madl et al., 2015). We hypothesize that this broader bias could be driven by several factors, such as leaning towards the human experience (vision-based models are strongly overrepresented, especially in RL, e.g. Kulhanek et al., 2019; Jaderberg et al., 2016; Mirowski et al., 2017), the technical ease of implementing certain models, and, perhaps rightfully so, the prioritization of models based on experimentally known neural substrates, such as models based on place and grid cells (Burgess and O'Keefe, 2011). Beyond these factors, a key reason is conceptual blindness about what navigation process is actually being modeled. This is where the taxonomy and our current review article can be especially useful, as they offer a structured framework and specific generalization criteria for distinguishing between the different processes. As pointed out throughout this review article, this blindness often stems from a lack of clarity about the representations and computations that underlie each process. In experiments, this frequently manifests as either neural representations of space considered separated from the behaviors they support (such as on linear tracks or random foraging in small arenas) or alternative explanations for the observed behaviors being overlooked, as discussed in Parra-Barrero et al. (2023).

This leads us to the second broad insight from our review: computational models, perhaps unexpectedly, often suffer from the same shortcomings as experiments. The underlying issue regarding the representations and computations are surprisingly similar between experiments and models, especially as models become more complex. The most extreme example of this comes from the deep RL literature, where almost all state-of-the-art agents are capable of some form of navigation (Jaderberg et al., 2016; Mirowski et al., 2017; Kulhanek et al., 2019, 2021). However, since these agents are usually trained and tested on video games or very complex navigation tasks, we can only draw broad conclusions about how well these agents can navigate in general and not about how they actually do it, i.e. what representations and computations are being used. The first step in solving this issue is to use constrained and well-defined navigation tasks in lieu of video games to better understand how these agents solve them. While more recently, explicitly navigation-based simulation environments have been developed (Kolve et al., 2017; Wu et al., 2018; Savva et al., 2019), the complexity of these environments might represent a challenge rather than a solution for neuroscience research.

The next step that we propose is a common solution to this deficit for both experiments and computational models: testing for generalization. We specifically identified what aspects of the task and environment an agent must generalize over in order to ensure that the observed behavior is truly an instance of a particular navigation process. For the navigation behaviors, this is spatial generalization, and for the strategies, planning generalization. This is an insight that applies equally to computational modeling and experimental research — placing emphasis on generalization enables experimentalists and modelers alike to be more confident in their assessment of which process is at play and draw more reliable and robust conclusions.

Third, models of navigation in pathological or dysfunctional conditions are scarce in the literature. Several experimental studies have demonstrated that normal aging and clinical conditions such as Alzheimer's disease affect navigation abilities (Laczó et al., 2018; Rodgers et al., 2012; Coughlan et al., 2018) and paint a complex picture of which specific processes are impaired. Although one prevailing view is that the use of specific reference frames is selectively impaired in different conditions (Serino et al., 2014; Colombo et al., 2017; Allison et al., 2016; Ruggiero et al., 2020), other results suggest that what is affected is switching or transformation between the two reference frames (Ruggiero et al., 2018; Schmitt et al., 2021). Further complicating the picture surrounding aging is that aging-related navigation deficits seem to be attenuated when using real-world naturalistic tasks (McAvan et al., 2021; Hill et al., 2024), which may often involve multiple, redundant sources of navigationally relevant information presented in different ways. This is consistent with the hierarchical, dynamic view of navigation proposed in the taxonomy — the impairment of some navigation processes may be compensated for by others in naturalistic settings, which may facilitate the use of multiple processes. Looking at these issues through the lens of our taxonomy enables us to ask more principled questions. For instance, are processes at both levels of the navigation hierarchy affected? Deficits associated with general cognitive decline may affect higher-level processes, such as the organization of behaviors and strategies and the navigation strategies (i.e., route and map navigation), while largely sparing navigation behaviors. Alternatively, specific strategies or behaviors may be impaired selectively. Modeling can help resolve some of these issues by providing a structured framework to test hypotheses and distinguish between different underlying mechanisms.

4.2. What have we learned about the navigation processes, their representations and computations?

We have also gained perspective on issues relating to specific representations and computations underlying the navigation processes. Among these, the question of which processes and computations involve spatial representations, such as place and grid cells, is perhaps the most well-studied. Generally speaking, AI-based and biologically plausible models tend to approach this question from different perspectives.

On the one hand, AI-based models offer insights into which navigation processes might lead to the emergence of distinct spatial representations. Studies in artificial agents have shown that under certain conditions, grid cells emerge to support path integration (Banino et al., 2018; Cueva and Wei, 2018; Sorscher et al., 2022), head-direction cells for angular path integration (Cueva et al., 2020), place cells for guidance (Vijayabaskaran and Cheng, 2022), and egocentric cue direction cells for aiming (Vijayabaskaran and Cheng, 2022). On the other hand, biologically plausible models provide insight into how these representations might be implemented in the brain. These models consider how networks composed of spatially selective neurons might be used in navigation, taking into account biological constraints and mechanisms such as connectivity motifs and plasticity rules (Brzosko et al., 2017; Ghazinouri and Cheng, 2025).

Despite significant insights from both approaches, our understanding of the precise functional role(s) of place, grid, and other cell types is still incomplete. Experimental evidence supports the involvement of place cells in guidance (Gothard et al., 1996; Hales et al., 2014), topological (Dabaghian et al., 2014; Dabaghian, 2019), and metric navigation (Wilson and McNaughton, 1993). Correspondingly, models of place cells exist for these three processes, i.e., guidance (Vijayabaskaran and Cheng, 2022; Ghazinouri et al., 2024), topological (Edvardsen et al., 2019) and metric navigation (Gao, 2023; Gönner et al., 2017). However, there are still asymmetries and gaps in the modeling literature, as we discuss below.

First, in biologically plausible models, place cells have been used to model both guidance and map navigation. However, these models often do not explicitly consider if the map is metric or topological, as we outline in Section 3.2.2. Moreover, many models using place cells for guidance focus on navigation to a single goal location (e.g. Ghazinouri et al., 2024), making it difficult to assess if the model is capable of generalizing to new goal locations (e.g. Ghazinouri and Cheng, 2025). From the RL perspective, one study demonstrated that place cells facilitate generalization to new start locations (Vijayabaskaran and Cheng, 2022), although generalization to novel goals and sensory properties of the environment was not explicitly tested in that study. The role of place cells in map navigation has also received relatively little attention in the RL literature. However, the study by Banino et al. (2018) found that an agent using place cells to navigate was less effective at finding shortcuts compared to one endowed with grid cells.

Second, models of grid cells propose a functional role for them in path integration (McNaughton et al., 2006) and vector movement (Bush et al., 2015), as well as metric navigation (Ginosar et al., 2023). However, while these models illustrate how the grid code could be used to compute a direct vector to the goal, it remains unclear how they might be used to flexibly plan detours around obstacles. Edvardsen et al.'s (2019) model proposes an interplay between place and grid cells in this scenario, where place cells are responsible for re-planning subgoals upon encountering obstacles, while grid cells compute direct paths to these newly identified subgoals.

A perspective on place and grid cells and their role in the cognitive map that has gained considerable attention in recent years is the successor representation theory (Section 3.2.2.2), which proposes that place cells encode predictive relationships between states in the environment (Stachenfeld et al., 2017; Momennejad et al., 2017; Gershman, 2018). However, as we will argue in the next section, while we believe that the successor representation has many potential applications, it is a general representation learning technique rather than a particular type of map, which makes it difficult to relate it to a specific navigation process in the taxonomy.

Third, as we note above, and in Section 3.2.2, place and grid cells have received the bulk of attention in the modeling literature. Thus, a crucial gap in the literature concerns incorporating cells with mixed selectivity (Rigotti et al., 2013; Maisson et al., 2023) as well as considering the contribution of cells that encode other spatial features of the environment (with the notable exception of border cells (Dabaghian, 2023; Santos-Pata et al., 2017)). Cells such as object vector cells (Høydal et al., 2019) and landmark vector cells (Deshmukh and Knierim, 2013) found in the entorhinal cortex, could potentially be integrated into the metric map to indicate the choice of sub-goals for navigation strategies. Understanding how these, and other, cell types contribute to the navigation strategies remains an important open question. As others have argued, rather than studying specific cell types in isolation, it is worth considering them in their larger context as part of a dynamical system, where neural representations flexibly adapt to environmental and task demands to support behavior (Ekstrom et al., 2020).

Finally, the discussion above surrounding place and grid cells also highlights a notable asymmetry in the attention given to the two aspects of the navigation strategies, i.e. the spatial knowledge structure and the planning process. While place and grid cells may form the neural basis of the spatial knowledge structure, the equally critical component of planning has received comparatively less attention in the context of navigation. This imbalance is particularly significant because, as we emphasize at various points, it is planning generalization that determines which strategy is currently at play. Although some models of navigation have considered the role of replay and preplay of neural sequences (Section 3.2.2) in learning (Pezzulo et al., 2019), a similar phenomenon that might be involved in planning — theta sequences (Parra-Barrero et al., 2021; Parra-Barrero and Cheng, 2023) — has been relatively understudied by modelers. Given that theta sequences have been shown to represent meaningful segments of spatial experience (such as sections of a maze) (Gupta et al., 2012) and reflect an agent's current goals (Wikenheiser and Redish, 2015), they present an exciting avenue for computational modeling.

4.3. What have we learned about current computational tools?

Our review raises several important open questions about RL-based models. One key question concerns what model-free RL actually captures when used to model navigation. The simplest perspective suggests that, because each state is associated with an optimal action, model-free RL primarily represents a sequence of learned state–action associations, aligning more closely with habitual, sensorimotor associations rather than 'genuine' navigational processes. This interpretation is reflected in some models that use a model-free controller for precisely this purpose, i.e. to model stimulus–response associations (Chavarriaga et al., 2005; Sheynikhovich et al., 2009; Dollé et al., 2010). We provide a counterpoint to this perspective, and propose that in itself, model-free RL is not limited to a single interpretation, and, rather is a computational tool that can be potentially used to model different processes. For instance, some have used model-free RL to model complex navigation processes (Banino et al., 2018; Vijayabaskaran and Cheng, 2022). How successfully model-free RL can capture a given behavior or strategy should again be evaluated through the lens of generalization, and we can already make some educated guesses about the direction this line of investigation may take. For example, it is unlikely that tabular model-free RL would pass the threshold of spatial generalization, since it cannot easily adapt to new goal (i.e., reward) locations without extensive re-learning. This problem also affects deep RL models, but depending on the network and learning algorithm the agent might be able to extract useful representations for navigation, which could aid generalization (Devo et al., 2020). It is important to continue to test these hypotheses empirically, especially since agents that successfully learn navigation develop representations that resemble place and grid cells (Vijayabaskaran and Cheng, 2022; Kappel and Cheng, 2025; Banino et al., 2018; Sorscher et al., 2022). There are several questions that these models can potentially answer — how, and to what extent, are internal representations and computations of the agent shaped by different factors? What types of representations other than those known in biological agents can aid successful navigation and generalization?

We also view the successor representation (Section 3.2.2.2) through a similar lens, i.e. as a tool that could potentially be used to model different processes. At first glance, key properties of the successor representation give it the appearance of being metric in nature — since the successor representation matrix represents future expected occupancy of states, in a 2D open field, this reflects metric relationships between the states (Stachenfeld et al., 2017). However, this is only true given that the states are defined based on a regular grid and exploration is uniform. If one constructed a successor representation of a subway map, for instance, where connectivity matters more than metric properties, the predictive relationship would more closely reflect topological than metric properties. Another key property of the successor representation that factors into our view of it as a computational tool rather than an instantiation of a particular process is its dependence on the policy, i.e., it reflects the behavior within an environment rather than purely spatial relationships. For example, the successor representation will skew to reflect routes that are more frequently taken.

Our review also highlights a general mismatch between the two broad modeling approaches. On the one hand, methods based on AI and deep RL in particular are highly powerful and are therefore used on fairly complex tasks, where navigation is often not even explicitly the main task (e.g. while playing the video game Doom (Kempka et al., 2016)). In these cases, it is very clear that the agents *can* navigate, but not clear at all how they do it. This type of agent is not so fruitful for neuroscience, as it involves the highly complex, and sometimes intractable, problem of deciphering a black box. On the other hand, biologically plausible models use simplified inputs and simple navigation tasks and often cannot even cope with a changing goal location (e.g. Ghazinouri and Cheng, 2025). In general, our impression is that current models based on synaptic plasticity lack flexibility, which manifests as an inability to switch between different behaviors and strategies. Thus, while it is clear *how* they navigate to the goal, i.e., what mechanism drives the behavior, it is not so clear if navigation is flexible or general enough to qualify as a navigation process in our taxonomy. A consequence of this limited flexibility in biologically plausible models is the scarcity of models that explicitly study the organization of behaviors and strategies (Sections 3.3.1 and 3.3.2). While this gap is also apparent in RL-based models, RL at least theoretically offers a framework for addressing this issue through hierarchical RL (Botvinick, 2012). However, how such flexibility could be achieved in biologically plausible models remains an open question. One potential approach might be the use of neuromodulators to transition the network between different states (Marder, 2012).

4.4. What have we learned about the taxonomy?

The taxonomy serves as a much-needed scaffold for structuring our understanding of spatial navigation and facilitating interdisciplinary communication. However, it must be noted that the boundaries between different processes are defined at a computational level (Marr, 1982), allowing for various implementations and algorithms to achieve the same process. Thus, while the boundaries outlined in the taxonomy are both useful and meaningful, they remain a conceptual framework rather than absolute divisions. This implies that, depending on specific algorithmic choices, some ambiguity may arise in classifying a model as belonging to one process or another. A notable example of this ambiguity is the classic Braitenberg vehicle (Braitenberg, 1986), which illustrates the fuzziness at the boundaries of the behaviors — in this case between aiming and direction field navigation. Behaviorally, a Braitenberg vehicle with crossed sensor–motor connections navigating towards a light source may appear to be engaging in aiming. However, in its simplest form, it does not slow down as it approaches the source and may not even stop upon arrival. Since its movement is guided solely by local differences in sensor measurements and lacks an estimate of distance, it could be interpreted as an instance of direction field navigation within our taxonomy, as discussed in Sections 3.1.1 and 3.1.2. The fact that it moves in a relatively straight line towards the goal is a consequence of the properties of the sensory signal — light propagates in straight lines — rather than the ability to localize the source directly. A modification that enables the Braitenberg vehicle to estimate the distance to the goal, for example, by using intensity readings to adjust its behavior accordingly, could however lead to its reinterpretation as a genuine model of aiming. This example illustrates the subtleties involved in interpreting the processes implemented by a model. Similar issues in interpretation may occur at various boundaries between different processes in the taxonomy, particularly in real-life navigation or in complex settings involving artificial agents. This is because there are many possible ways in which the same real-life navigation task could be solved, and it is very difficult to distinguish which specific process(es) an agent is using. This is further exacerbated by the sensory richness of real-world environments, which often contain redundant information that enables the use of multiple navigation processes, and the dynamic interaction between navigation processes, where agents may integrate different processes or switch between them rapidly. In some cases, it may even be unclear if the observed process is a behavior or strategy. Take the example of navigating in a street network within a city — it can be difficult to distinguish if, while constrained to walk along the streets, an agent is engaged in path following, route navigation, or both. Thus, it is important to exercise caution and keep in mind potential ambiguities while applying the taxonomy.

Despite these ambiguities, we strongly believe that establishing clear terminology is essential for discussing and analyzing these subtleties in a structured and principled manner, which the taxonomy provides. Without such a framework, these nuances might be overlooked or lost entirely.

Finally, the taxonomy remains agnostic about two key issues related to spatial navigation. The first concerns the choice of reference frame to which spatial representations are anchored, namely, whether certain navigation processes inherently rely on an egocentric or allocentric reference frame. Experimental evidence suggests that both types of representations coexist in parallel, with factors such as environmental structure, familiarity with the environment, and task complexity influencing which reference frame is utilized (Burgess, 2006). Relatively few studies have explored this question through modeling. Vijayabaskaran and Cheng (2022) suggest that guidance based on allocentric representations and aiming using egocentric representations result in better generalization performance than using the other representation, respectively. Based on the available evidence and theoretical considerations, we propose that different navigation processes might be biased towards specific reference frames, however, whether this represents a strict dichotomy remains unclear at this moment. For the navigation behaviors, this bias may depend on the nature of the cues required to extract relevant representations, such as whether the cues are fixed, local, or distal. For the navigation strategies, the bias may be shaped by how the spatial knowledge structure,i.e. route, topological or metric map, is constructed, a process which likely unfolds gradually (Golledge et al., 1985). While purely egocentric or allocentric representations may exist as endpoints on a spectrum, it is likely that in most cases, the navigation strategies involve mixtures and distortions influenced by factors such as memory, attention, and experience.

Another, somewhat similar, issue concerns the relationship of sensory modalities to the navigation behaviors, specifically, whether certain navigation processes are inherently biased towards particular sensory modalities. As with reference frames, modeling studies on this topic remain scarce, with a predominant focus on vision-based models, as noted above. We do not assume *a priori* that any navigation behavior is restricted to a specific sensory modality. However, the nature of the sensory signal may strongly bias certain behaviors towards particular modalities. For instance, aiming may be particularly well-suited to vision, given that light travels in straight lines, whereas other sensory signals, such as sound, which is prone to reflection in environments with walls, may be less reliable for precise localization. In addition, the suitability of a modality for navigation depends on the type of sensor that is available to a species and its sensitivity, which varies widely from species to species.

An important open question for future modeling and experimental work is how an agent executing the same behavior using different sensory modalities integrates multiple signals. One possibility is that each modality undergoes a distinct extraction stage to generate the necessary representation before a common process operates on it. Alternatively, parallel processes may extract and process information separately, later interacting through integration or selection mechanisms. This question has been extensively explored in robotics, where sensor fusion is a key technical challenge (Sasiadek, 2002). For the navigation strategies, since the spatial knowledge structure involves abstract goal representations, it is likely that agents incorporate information from multiple sensory sources. Some experimental evidence supporting this idea comes from studies demonstrating that place cell maps can be formed based on landmarks detected across different modalities, such as vision (Poucet et al., 2000), taste (Herzog et al., 2019), and odor (Zhang and Manahan-Vaughan, 2013).

5. Conclusions

In this review, we have significantly expanded on the hierarchical taxonomy proposed by Parra-Barrero et al. (2023), refining and extending it to identify the representations and computations required for each navigation process. A central argument we put forth is that the key to distinguishing between different navigation processes lies in assessing an agent's ability to generalize. We examined both biologically plausible and RL-based computational models of navigation through the lens of our taxonomy. While many computational studies have been performed and much has been learned from them, we have identified several shortcomings in the state of the field, our understanding of the computational tools we use to model navigation, and our understanding of navigation and its neural basis. We have made several suggestions how these shortcomings could be addressed and how this process could benefit from applying our taxonomy of spatial navigation.

Funding support

Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation), Germany – project number 316803389 – through SFB 1280, projects A14 and F01 (S.C.).

Declaration of competing interest

None.

References

Addis, K.M., Kahana, M.J., 2004. Decomposing serial learning: What is missing from the learning curve? Psychon. Bull. Rev. 11 (1), 118–124.

Allison, S.L., Fagan, A.M., Morris, J.C., Head, D., 2016. Spatial navigation in preclinical alzheimer's disease. J. Alzheimer' s Dis.: JAD 52 (1), 77–90.

Amidi, O., Thorpe, C.E., 1991. Integrated mobile robot control. In: Mobile Robots V, vol. 1388, International Society for Optics and Photonics, pp. 504–523.

Baddeley, B., Graham, P., Husbands, P., Philippides, A., 2012. A model of ant route navigation driven by scene familiarity. PLoS Comput. Biol. 8 (1), e1002336.

Baker, R.R., 1987. Human navigation and magnetoreception: The Manchester experiments do replicate. Anim. Behav. 35 (3), 691–704.

Banino, A., Barry, C., Uria, B., Blundell, C., Lillicrap, T., Mirowski, P., Pritzel, A., Chadwick, M.J., Degris, T., Modayil, J., Wayne, G., Soyer, H., Viola, F., Zhang, B., Goroshin, R., Rabinowitz, N., Pascanu, R., Beattie, C., Petersen, S., Sadik, A., Gaffney, S., King, H., Kavukcuoglu, K., Hassabis, D., Hadsell, R., Kumaran, D., 2018. Vector-based navigation using grid-like representations in artificial agents. Nature 1.

Barto, A.G., Mahadevan, S., 2003. Recent advances in hierarchical reinforcement learning. Discrete Event Dyn. Syst. 13 (4), 341–379.

Beattie, C., Leibo, J.Z., Teplyashin, D., Ward, T., Wainwright, M., Küttler, H., Lefrancq, A., Green, S., Valdés, V., Sadik, A., Schrittwieser, J., Anderson, K., York, S., Cant, M., Cain, A., Bolton, A., Gaffney, S., King, H., Hassabis, D., Legg, S., Petersen, S., 2016. DeepMind lab. p. 1612.03801, ArXiv.

Beeson, P., Modayil, J., Kuipers, B., 2010. Factoring the mapping problem: mobile robot map-building in the hybrid spatial semantic hierarchy. Int. J. Robot. Res. 29 (4), 428–459.

Bermudez-Contreras, E., Clark, B.J., Wilber, A., 2020. The neuroscience of spatial navigation and the relationship to artificial intelligence. Front. Comput. Neurosci. 14.

Best, P.J., White, A.M., Minai, A., 2001. SPATIAL processing IN THE BRAIN: the activity of hippocampal place cells. Annu. Rev. Neurosci. 24 (1), 459–486.

Bibri, S.E., Krogstie, J., 2019. Generating a vision for smart sustainable cities of the future: A scholarly backcasting approach. Eur. J. Futur. Res. 7 (1), 5.

Bierbrauer, A., Kunz, L., Gomes, C.A., Luhmann, M., Deuker, L., Getzmann, S., Wascher, E., Gajewski, P.D., Hengstler, J.G., Fernandez-Alvarez, M., Atienza, M., Cammisuli, D.M., Bonatti, F., Pruneti, C., Percesepe, A., Bellaali, Y., Hanseeuw, B., Strange, B.A., Cantero, J.L., Axmacher, N., 2020. Unmasking selective path integration deficits in alzheimer's disease risk carriers. Sci. Adv. 6 (35), eaba1394.

Botvinick, M., 2012. Hierarchical reinforcement learning and decision making. Curr. Opin. Neurobiol..

Botvinick, M., Niv, Y., Barto, A.C., 2009. Hierarchically organized behavior and its neural foundations: A reinforcement learning perspective. Cognition.

Braitenberg, V., 1986. Vehicles: Experiments in Synthetic Psychology. MIT Press.

Brette, R., 2015. Philosophy of the spike: rate-based vs. spike-based theories of the brain. Front. Syst. Neurosci. 9.

Brokaw, A.F., Davis, E., Page, R.A., Smotherman, M., 2021. Flying bats use serial sampling to locate odour sources. Biology Lett. 17 (10), 20210430.

Brzosko, Z., Zannone, S., Schultz, W., Clopath, C., Paulsen, O., 2017. Sequential neuromodulation of Hebbian plasticity offers mechanism for effective reward-based navigation. ELife 6, e27756.

Buhry, L., Azizi, A.H., Cheng, S., 2011. Reactivation, replay, and preplay: how it might all fit together. Neural Plast. 2011, 1–11.

Burak, Y., Fiete, I.R., 2009. Accurate path integration in continuous attractor network models of grid cells.. PLoS Comput. Biol. 5 (2), e1000291+.

Burda, H., Marhold, S., Westenberger, T., Wiltschko, R., Wiltschko, W., 1990. Magnetic compass orientation in the subterranean rodentcryptomys Hottentotus (Bathyergidae). Experientia 46 (5), 528–530.

Burgess, N., 2006. Spatial memory: How egocentric and allocentric combine. Trends Cogn. Sci. 10 (12), 551–557.

Burgess, N., O'Keefe, J., 2011. Models of place and grid cell firing and theta rhythmicity.. Curr. Opin. Neurobiol. 21 (5), 734–744.

Bush, D., Barry, C., Manson, D., Burgess, N., 2015. Using grid cells for navigation. Neuron 87 (3), 507–520.

Carlini, A., Bordeau, C., Ambard, M., 2024. Auditory localization: A comprehensive practical review. Front. Psychol. 15.

Catania, K.C., 2013. Stereo and serial sniffing guide navigation to an odour source in a mammal. Nat. Commun. 4 (1), 1441.

Chaplot, D.S., 2016. Transfer deep reinforcement learning in 3 d environments : an empirical study. In: NIPS Deep Reinforcemente Leaning Workshop. Barcelona, Spain.

Chavarriaga, R., Strösslin, T., Sheynikhovich, D., Gerstner, W., 2005. A computational model of parallel navigation systems in rodents. Neuroinformatics 3 (3), 223–241.

Cheng, K., Shettleworth, S.J., Huttenlocher, J., Rieser, J.J., 2007. Bayesian integration of spatial information.. Psychol. Bull. 133 (4), 625–637.

Chrastil, E.R., Warren, W.H., 2014. From cognitive maps to cognitive graphs. PLoS ONE 9 (11), e112544+.

Colmant, L., Bierbrauer, A., Bellaali, Y., Kunz, L., Van Dongen, J., Sleegers, K., Axmacher, N., Lefèvre, P., Hanseeuw, B., 2023. Dissociating effects of aging and genetic risk of sporadic alzheimer's disease on path integration. Neurobiol. Aging 131, 170–181.

Colombo, D., Serino, S., Tuena, C., Pedroli, E., Dakanalis, A., Cipresso, P., Riva, G., 2017. Egocentric and allocentric spatial reference frames in aging: A systematic review. Neurosci. Biobehav. Rev. 80, 605–621.

Coughlan, G., Laczó, J., Hort, J., Minihane, A.-M., Hornberger, M., 2018. Spatial navigation deficits — Overlooked cognitive marker for preclinical alzheimer disease? Nat. Rev. Neurol. 14 (8), 496–506.

Coulter, R.C., 1992. Implementation of the Pure Pursuit Path Tracking Algorithm. Technical report, Carnegie-Mellon Univ Pittsburgh PA Robotics Inst.

Croft, D.B., 2019. Walking in each other's footsteps: do animal trail makers confer resilience against trampling tourists? Environments 6 (7), 83.

Cruse, H., 2003. A recurrent network for landmark-based navigation. Biol. Cybernet. 88 (6), 425–437.

Cueva, C.J., Wang, P.Y., Chin, M., Wei, X.-X., 2020. Emergence of functional and structural properties of the head direction system by optimization of recurrent neural networks.

Cueva, C.J., Wei, X.-X., 2018. Emergence of grid-like representations by training recurrent neural networks to perform spatial localization. p. 1803.07770. http://dx.doi.org/10.48550/arXiv.1803.07770, ArXiv.

Dabaghian, Y., 2019. Through synapses to spatial memory maps via a topological model. Sci. Rep. 9 (1), 1–13.

Dabaghian, Y., 2023. Grid cells, border cells, and discrete complex analysis. Front. Comput. Neurosci. 17, 1242300.

Dabaghian, Y., Brandt, V.L., Frank, L.M., 2014. Reconceiving the hippocampal map as a topological template.. ELife 3.

Dang, S., Wu, Y., Yan, R., Tang, H., 2021. Why grid cells function as a metric for space. Neural Netw.: Off. J. Int. Neural Netw. Soc. 142, 128–137.

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

Dayan, P., 1993. Improving generalization for temporal difference learning: the successor representation. Neural Comput. 5 (4), 613–624.

Dayan, P., Berridge, K.C., 2014. Model-based and model-free pavlovian reward learning: revaluation, revision and revelation. Cogn. Affect. Behav. Neurosci. 14 (2), 473–492.

de Souza Muñoz, M.E., Chaves Menezes, M., Pignaton de Freitas, E., Cheng, S., de Almeida Ribeiro, P.R., de Almeida Neto, A., Muniz de Oliveira, A.C., 2022. xRatSLAM: an extensible RatSLAM computational framework. Sensors 22 (21), 8305.

Deshmukh, S.S., Knierim, J.J., 2013. Influence of local objects on hippocampal representations: landmark vectors and memory. Hippocampus 23 (4), 253–267.

Devenport, J.A., Devenport, L.D., 1994. Spatial navigation in natural habitats by ground-dwelling sciurids. Anim. Behav. 47 (3), 727–729.

Devo, A., Mezzetti, G., Costante, G., Fravolini, M.L., Valigi, P., 2020. Towards generalization in target-driven visual navigation by using deep reinforcement learning. IEEE Trans. Robot. 36 (5), 1546–1561.

Diekmann, N., Cheng, S., 2023. A model of hippocampal replay driven by experience and environmental structure facilitates spatial learning. ELife 12:e82301.

Diekmann, N., Vijayabaskaran, S., Zeng, X., Kappel, D., Menezes, M.C., Cheng, S., 2023. CoBeL-RL: A neuroscience-oriented simulation framework for complex behavior and learning. Front. Neuroinform. 17, 1134405.

Dijkstra, E.W., 1959. A note on two problems in connexion with graphs. Numer. Math. 1 (1), 269–271.

Diuk, C., Tsai, K., Wallis, J., Botvinick, M., Niv, Y., 2013. Hierarchical learning induces two simultaneous, but separable, prediction errors in human basal ganglia. J. Neurosci..

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. Biol. Cybernet. 103 (4), 299–317.

Edelstein-Keshet, L., 1994. Simple models for trail-following behaviour; trunk trails versus individual foragers. J. Math. Biol. 32 (4), 303–328.

Edvardsen, V., 2018. Navigating with distorted grid cells. In: The 2018 Conference on Artificial Life: A Hybrid of the European Conference on Artificial Life (ECAL) and the International Conference on the Synthesis and Simulation of Living Systems (ALIFE). pp. 260–267.

Edvardsen, V., Bicanski, A., Burgess, N., 2019. Navigating with grid and place cells in cluttered environments. Hippocampus.

Ekstrom, A.D., Arnold, A.E.G.F., Iaria, G., 2014. A critical review of the allocentric spatial representation and its neural underpinnings: toward a network-based perspective. Front. Hum. Neurosci. 8.

Ekstrom, A.D., Harootonian, S.K., Huffman, D.J., 2020. Grid coding, spatial representation, and navigation: should we assume an Isomorphism? Hippocampus 30 (4), 422–432.

Ekstrom, A.D., Hill, P.F., 2023. Spatial navigation and memory: A review of the similarities and differences relevant to brain models and age. Neuron 111 (7), 1037–1049.

Eliasmith, C., Trujillo, O., 2014. The use and abuse of large-scale brain models. Curr. Opin. Neurobiol. 25, 1–6.

Erdem, U.M., Hasselmo, M., 2012. A goal-directed spatial navigation model using forward trajectory planning based on grid cells: forward linear look-ahead trajectory model. Eur. J. Neurosci. 35 (6), 916–931.

Erdem, U.M., Milford, M.J., Hasselmo, M.E., 2015. A hierarchical model of goal directed navigation selects trajectories in a visual environment. Neurobiol. Learn. Mem. 117, 109–121.

Eysenbach, B., Salakhutdinov, R., Levine, S., 2019. Search on the replay buffer: bridging planning and reinforcement learning. arXiv:1906.05253. [Cs].

Farebrother, J., Machado, M.C., Bowling, M., 2020. Generalization and regularization in DQN. arXiv:1810.00123. [Cs, Stat].

Franz, M.O., Mallot, H.A., 2000. Biomimetic robot navigation. Robot. Auton. Syst. 30 (1–2), 133–153.

Franz, M.O., Stürzl, W., Hübner, W., Mallot, H.A., 2008. A robot system for biomimetic navigation – from snapshots to metric embeddings of view graphs. In: Jefferies, M.E., Yeap, W.-K. (Eds.), Robotics and Cognitive Approaches To Spatial Mapping. Springer, Berlin, Heidelberg, pp. 297–314.

Frommberger, L., 2008. Learning to behave in space: A qualitative spatial representation for robot navigation with reinforcement learning. Int. J. Artif. Intell. Tools 17 (03), 465–482.

Fuhs, M.C., Touretzky, D.S., 2006. A spin glass model of path integration in rat medial entorhinal cortex. J. Neurosci. 26 (16), 4266–4276.

Fyhn, M., Hafting, T., Treves, A., Moser, M.-B., Moser, E.I., 2007. Hippocampal remapping and grid realignment in entorhinal cortex. Nature 446 (7132), 190–194.

Gallistel, C.R., 1990. The organization of learning. In: The Organization of Learning, The MIT Press, Cambridge, MA, US, viii, 648.

Gao, Y., 2023. A computational model of learning flexible navigation in a maze by layout-conforming replay of place cells. Front. Comput. Neurosci. 17.

Gershman, S.J., 2018. The successor representation: its computational logic and neural substrates. J. Neurosci. 38 (33), 7193–7200.

Ghazinouri, B., Cheng, S., 2025. The cost of behavioral flexibility: reversal learning driven by a spiking neural network. In: Brock, O., Krichmar, J. (Eds.), From Animals To Animats 17. Springer Nature Switzerland, Cham, pp. 39–50.

Ghazinouri, B., Nejad, M.M., Cheng, S., 2024. Navigation and the efficiency of spatial coding: Insights from closed-loop simulations. Brain Struct. Funct. 229 (3), 577–592.

Ginosar, G., Aljadeff, J., Las, L., Derdikman, D., Ulanovsky, N., 2023. Are grid cells used for navigation? on local metrics, subjective spaces, and black holes. Neuron 111 (12), 1858–1875.

Goldschmidt, D., Manoonpong, P., Dasgupta, S., 2017. A neurocomputational model of goal-directed navigation in insect-inspired artificial agents. Front. Neurorobotics 11.

Golledge, R.G., Smith, T.R., Pellegrino, J.W., Doherty, S., Marshall, S.P., 1985. A conceptual model and empirical analysis of children's acquisition of spatial knowledge. J. Environ. Psychol. 5 (2), 125–152.

Gönner, L., Vitay, J., Hamker, F.H., 2017. Predictive place-cell sequences for goal-finding emerge from goal memory and the cognitive map: a computational model. Front. Comput. Neurosci. 11.

Goodroe, S.C., Starnes, J., Brown, T.I., 2018. The complex nature of hippocampal-striatal interactions in spatial navigation. Front. Hum. Neurosci. 12.

Goss, S., Beckers, R., Deneubourg, J.L., Aron, S., Pasteels, J.M., 1990. How trail laying and trail following can solve foraging problems for ant colonies. In: Hughes, R.N. (Ed.), Behavioural Mechanisms of Food Selection. Springer, Berlin, Heidelberg, pp. 661–678.

Gothard, K.M., Skaggs, W.E., Moore, K.M., McNaughton, B.L., 1996. Binding of hippocampal CA1 neural activity to multiple reference frames in a landmark-based navigation task. J. Neurosci. 16 (2), 823–835.

Gupta, A.S., van der Meer, M.A., Touretzky, D.S., Redish, A.D., 2012. Segmentation of spatial experience by hippocampal theta sequences.. Nature Neurosci. 15 (7), 1032–1039.

Hafner, D., Lillicrap, T., Ba, J., Norouzi, M., 2019. Dream to control: learning behaviors by latent imagination.

Hafner, D., Lillicrap, T., Norouzi, M., Ba, J., 2020. Mastering atari with discrete world models.

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

Hales, J.B., Schlesiger, M.I., Leutgeb, J.K., Squire, L.R., Leutgeb, S., Clark, R.E., 2014. Medial entorhinal cortex lesions only partially disrupt hippocampal place cells and hippocampus-dependent place memory. Cell Rep. 9 (3), 893–901.

Hart, P.E., Nilsson, N.J., Raphael, B., 1968. A formal basis for the heuristic determination of minimum cost paths. IEEE Trans. Syst. Sci. Cybern. 4 (2), 100–107.

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.

Helbing, D., Keltsch, J., Molnár, P., 1997. Modelling the evolution of human trail systems. Nature 388 (6637), 47–50.

Herzog, L.E., Pascual, L.M., Scott, S.J., Mathieson, E.R., Katz, D.B., Jadhav, S.P., 2019. Interaction of taste and place coding in the hippocampus. J. Neurosci. 39 (16), 3057–3069.

Hill, P.F., Bermudez, S., McAvan, A.S., Garren, J.D., Grilli, M.D., Barnes, C.A., Ekstrom, A.D., 2024. Age differences in spatial memory are mitigated during naturalistic navigation. Aging Neuropsychol. Cogn. 31 (6), 1106–1130.

Hilton, C., Wiener, J., Johnson, A., 2021. Serial memory for landmarks encountered during route navigation. Q. J. Exp. Psychol. 74 (12), 2137–2153.

Holland, R.A., Thorup, K., Vonhof, M.J., Cochran, W.W., Wikelski, M., 2006. Bat orientation using earth's magnetic field. Nature 444 (7120), 702–702.

Howard, M.W., Fotedar, M.S., Datey, A.V., Hasselmo, M.E., 2005. The temporal context model in spatial navigation and relational learning: Toward a common explanation of medial temporal lobe function across domains.. Psychol Rev 112 (1), 75–116.

Howard, M.W., Kahana, M.J., 2002. A distributed representation of temporal context. J. Math. Psych. 46 (3), 269–299.

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 1.

Huang, J., Supaongprapa, T., Terakura, I., Wang, F., Ohnishi, N., Sugie, N., 1999. A model-based sound localization system and its application to robot navigation. Robot. Auton. Syst. 27 (4), 199–209.

Hung, N., Rego, F., Quintas, J., Cruz, J., Jacinto, M., Souto, D., Potes, A., Sebastiao, L., Pascoal, A., 2023. A review of path following control strategies for autonomous robotic vehicles: theory, simulations, and experiments. J. Field Robot. 40 (3), 747–779.

Jaderberg, M., Mnih, V., Czarnecki, W.M., Schaul, T., Leibo, J.Z., Silver, D., Kavukcuoglu, K., 2016. Reinforcement learning with unsupervised auxiliary tasks. arXiv:1611.05397. [Cs].

Kahana, M.J., 2020. Computational models of memory search. Annu. Rev. Psychol. 71 (1), 107–138.

Kanitscheider, I., Fiete, I., 2017. Training recurrent networks to generate hypotheses about how the brain solves hard navigation problems. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (Eds.), Advances in Neural Information Processing Systems 30. Curran Associates, Inc., pp. 4529–4538.

Kappel, D., Cheng, S., 2025. Global remapping emerges as the mechanism for renewal of context-dependent behavior in a reinforcement learning model. Front. Comput. Neurosci. 18, 1462110.

Kempka, M., Wydmuch, M., Runc, G., Toczek, J., Jaśkowski, W., 2016. ViZDoom: a doom-based AI research platform for visual reinforcement learning. arXiv:1605.02097. [Cs].

Kolve, E., Mottaghi, R., Han, W., VanderBilt, E., Weihs, L., Herrasti, A., Deitke, M., Ehsani, K., Gordon, D., Zhu, Y., Kembhavi, A., Gupta, A., Farhadi, A., 2017. AI2-THOR: an interactive 3D environment for visual AI.

Konolige, K., Marder-Eppstein, E., Marthi, B., 2011. Navigation in hybrid metric-topological maps. In: 2011 IEEE International Conference on Robotics and Automation. pp. 3041–3047.

Kriegeskorte, N., Douglas, P.K., 2018. Cognitive computational neuroscience. Nature Neurosci. 21 (9), 1148–1160.

Kuipers, B., 2000a. The spatial semantic hierarchy. Artificial Intelligence 119 (April 1999), 191–233.

Kuipers, B., 2000b. The spatial semantic hierarchy. Artificial Intelligence 119 (1), 191–233.

Kulhanek, J., Derner, E., Babuska, R., 2021. Visual navigation in real-world indoor environments using end-to-end deep reinforcement learning. IEEE Robot. Autom. Lett. 6 (3), 4345–4352.

Kulhanek, J., Derner, E., de Bruin, T., Babuska, R., 2019. Vision-based navigation using deep reinforcement learning. In: 2019 European Conference on Mobile Robots (ECMR). IEEE, Prague, Czech Republic, pp. 1–8.

Laczó, J., Parizkova, M., Moffat, S.D., 2018. Spatial navigation, aging and alzheimer's disease. Aging 10 (11), 3050–3051.

Lee, K., Seo, Y., Lee, S., Lee, H., Shin, J., 2020. Context-aware dynamics model for generalization in model-based reinforcement learning. In: Proceedings of the 37th International Conference on Machine Learning. PMLR, pp. 5757–5766.

Levy, A., Konidaris, G., Platt, R., Saenko, K., 2019. Learning multi-level hierarchies with hindsight. arXiv:1712.00948. [Cs].

Lin, L.-J., 1993. Reinforcement Learning for Robots Using Neural Networks (Ph.D. thesis). Carnegie Mellon University, Pittsburgh, PA.

Lin, C.-C., Wagenaar, R., Young, D., Saltzman, E., Ren, X., Neargarder, S., Cronin-Golomb, A., 2014. Effects of parkinson's disease on optic flow perception for heading direction during navigation. Exp. Brain Res. 232, 1343–1355.

Lohmann, K.J., Goforth, K.M., Mackiewicz, A.G., Lim, D.S., Lohmann, C.M.F., 2022. Magnetic maps in animal navigation. J. Comp. Physiol. A 208 (1), 41–67.

Lynch, K., 1964. The Image of the City. MIT Press.

Maass, W., 1997. Networks of spiking neurons: the third generation of neural network models. Neural Netw. 10 (9), 1659–1671.
Madl, T., Chen, K., Montaldi, D., Trappl, R., 2015. Computational cognitive models of spatial memory in navigation space: A review. Neural Netw. 65, 18–43.
Maisson, D.J.-N., Cervera, R.L., Voloh, B., Conover, I., Zambre, M., Zimmermann, J., Hayden, B.Y., 2023. Widespread coding of navigational variables in prefrontal cortex. Curr. Biol. 33 (16), 3478–3488.e3.
Marder, E., 2012. Neuromodulation of neuronal circuits: back to the future. Neuron 76 (1), 1–11.
Marr, D., 1982. Vision : A Computational Investigation into the Human Representation and Processing of Visual Information. W. H. Freeman & Co..
Martinez, D., Rochel, O., Hugues, E., 2006. A biomimetic robot for tracking specific odors in turbulent plumes. Auton. Robots 20 (3), 185–195.
Mattar, M.G., Daw, N.D., 2018. Prioritized memory access explains planning and hippocampal replay. Nature Neurosci. 21 (11), 1609–1617.
McAvan, A.S., Du, Y.K., Oyao, A., Doner, S., Grilli, M.D., Ekstrom, A., 2021. Older adults show reduced spatial precision but preserved strategy-use during spatial navigation involving body-based cues. Front. Aging Neurosci. 13.
McNaughton, B.L., Battaglia, F.P., Jensen, O., Moser, E.I., Moser, M.-B., 2006. Path integration and the neural basis of the 'cognitive map'. Nature Rev. Neurosci. 7 (8), 663–678.
Meilinger, T., 2008. The network of reference frames theory: a synthesis of graphs and cognitive maps. In: Spatial Cognition VI. Learning, Reasoning, and Talking About Space. In: Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, pp. 344–360.
Meilinger, T., Knauff, M., 2008. Ask for directions or use a map: A field experiment on spatial orientation and wayfinding in an urban environment. J. Spat. Sci. 53 (2), 13–23.
Meilinger, T., Schulte-Pelkum, J., Frankenstein, J., Hardiess, G., Laharnar, N., Mallot, H.A., Bülthoff, H.H., 2016. How to best name a place? Facilitation and inhibition of route learning due to descriptive and arbitrary location labels. Front. Psychol. 7.
Menezes, M., Zeng, X., Cheng, S., 2025. Revealing the mechanisms underlying latent learning with successor representations. BioRxiv 2025.03.01.640768.
Milde, M.B., Blum, H., Dietmüller, A., Sumislawska, D., Conradt, J., Indiveri, G., Sandamirskaya, Y., 2017. Obstacle avoidance and target acquisition for robot navigation using a mixed signal analog/digital neuromorphic processing system. Front. Neurorobotics 11.
Milford, M., Jacobson, A., Chen, Z., Wyeth, G., 2016. Ratslam: using models of rodent hippocampus for robot navigation and beyond. In: Inaba, M., Corke, P. (Eds.), Robotics Research: the 16th International Symposium ISRR. In: Springer Tracts in Advanced Robotics, Springer International Publishing, Cham, pp. 467–485.
Milford, M., Schulz, R., 2014. Principles of goal-directed spatial robot navigation in biomimetic models. Phil. Trans. R. Soc. B 369 (1655), 20130484+.
Milford, M., Wyeth, G., Prasser, D., 2004. RatSLAM: A hippocampal model for simultaneous localization and mapping. In: IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004. IEEE, pp. 403–408.
Mirowski, P., Grimes, M.K., Malinowski, M., Hermann, K.M., Anderson, K., Teplyashin, D., Simonyan, K., Kavukcuoglu, K., Zisserman, A., Hadsell, R., 2019. Learning to navigate in cities without a map. ArXiv:1804.00168.
Mirowski, P., Pascanu, R., Viola, F., Soyer, H., Ballard, A.J., Banino, A., Denil, M., Goroshin, R., Sifre, L., Kavukcuoglu, K., Kumaran, D., Hadsell, R., 2017. Learning to navigate in complex environments. ArXiv:1611.03673.
Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A.A., Veness, J., Bellemare, M.G., Graves, A., Riedmiller, M., Fidjeland, A.K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., Hassabis, D., 2015. Human-level control through deep reinforcement learning. Nature 518 (7540), 529–533.
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. Nat. Hum. Behav. 1 (9), 680–692.
Morris, R.G.M., Garrud, P., Rawlins, J.N.P., O'Keefe, J., 1982. Place navigation impaired in rats with hippocampal lesions. Nature 297 (5868), 681–683.
Muller, R.U., Stead, M., Pach, J., 1996. The hippocampus as a cognitive graph.. J. Gen. Physiol. 107 (6), 663–694.
Nachum, O., Gu, S., Lee, H., Levine, S., 2018. Data-efficient hierarchical reinforcement learning. arXiv:1805.08296. [Cs, Stat].
Nachum, O., Gu, S., Lee, H., Levine, S., 2019a. Near-optimal representation learning for hierarchical reinforcement learning. (arXiv:1810.01257).
Nachum, O., Tang, H., Lu, X., Gu, S., Lee, H., Levine, S., 2019b. Why does hierarchy (sometimes) work so well in reinforcement learning?. arXiv:1909.10618. [Cs, Stat].
Ng, T.P.T., Saltin, S.H., Davies, M.S., Johannesson, K., Stafford, R., Williams, G.A., 2013. Snails and their trails: The multiple functions of trail-following in gastropods. Biological Rev. 88 (3), 683–700.
O'Keefe, J., Dostrovsky, J., 1971. The hippocampus as a spatial map. preliminary evidence from unit activity in the freely-moving rat. Brain Res. 34 (1), 171–175.
O'Keefe, J., Nadel, L., 1978. The Hippocampus as a Cognitive Map. Clarendon Press, Oxford.
Parra-Barrero, E., Cheng, S., 2023. Learning to predict future locations with internally generated theta sequences. PLoS Comput. Biol. 19 (5), e1011101.
Parra-Barrero, E., Diba, K., Cheng, S., 2021. Neuronal sequences during theta rely on behavior-dependent spatial maps. ELife 10, e70296.
Parra-Barrero, E., Vijayabaskaran, S., Seabrook, E., Wiskott, L., Cheng, S., 2023. A map of spatial navigation for neuroscience. Neurosci. Biobehav. Rev. 152, 105200.
Pertsch, K., Lee, Y., Lim, J.J., 2020. Accelerating reinforcement learning with learned skill priors. arXiv:2010.11944. [Cs].
Pezzulo, G., Donnarumma, F., Maisto, D., Stoianov, I., 2019. Planning at decision time and in the background during spatial navigation. Curr. Opin. Behav. Sci. 29, 69–76.
Pfeiffer, B.E., Foster, D.J., 2013. Hippocampal place-cell sequences depict future paths to remembered goals. Nature 497 (7447), 74–79.
Ponulak, F., Hopfield, J.J., 2013. Rapid, parallel path planning by propagating wavefronts of spiking neural activity. Front. Comput. Neurosci. 7.
Porter, J., Craven, B., Khan, R.M., Chang, S.-J., Kang, I., Judkewitz, B., Volpe, J., Settles, G., Sobel, N., 2007. Mechanisms of scent-tracking in humans. Nature Neurosci. 10 (1), 27–29.
Poucet, B., 1993. Spatial cognitive maps in animals: new hypotheses on their structure and neural mechanisms. Psychol Rev 100 (2), 163–182.
Poucet, B., Save, E., Lenck-Santini, P., 2000. Sensory and memory properties of hippocampal place cells.. Rev. Neurosci. 11 (2–3), 95–111.
Precup, D., 2000. Temporal abstraction in reinforcement learning.
Rasmussen, D., Voelker, A., Eliasmith, C., 2017. A neural model of hierarchical reinforcement learning. PLOS ONE 12 (7), e0180234.
Ribas-Fernandes, J., Solway, A., Diuk, C., McGuire, J.T., Barto, A., Niv, Y., Botvinick, M., 2011. A neural signature of hierarchical reinforcement learning. Neuron.
Rigotti, M., Barak, O., Warden, M.R., Wang, X.-J., Daw, N.D., Miller, E.K., Fusi, S., 2013. The importance of mixed selectivity in complex cognitive tasks. Nature 497 (7451), 585–590.
Rodgers, M.K., Sindone, J.A., Moffat, S.D., 2012. Effects of age on navigation strategy. Neurobiol. Aging 33 (1), 202.e15–202.e22.
Roessingh, P., 1989. The trail following behaviour of yponomeuta cagnagellus. Èntomol. Exp. Appl. 51 (1), 49–57.
Ruggiero, G., Iavarone, A., Iachini, T., 2018. Allocentric to egocentric spatial switching: impairment in aMCI and alzheimer's disease patients? Curr. Alzheimer Res. 15 (3), 229–236.
Ruggiero, G., Ruotolo, F., Iavarone, A., Iachini, T., 2020. Allocentric coordinate spatial representations are impaired in aMCI and alzheimer's disease patients. Behav. Brain Res. 393, 112793.
Samsonovich, A., McNaughton, B.L., 1997. Path integration and cognitive mapping in a continuous attractor neural network model.. J. Neurosci. 17 (15), 5900–5920.
Santos-Pata, D., Zucca, R., Low, S.C., Verschure, P.F.M.J., 2017. Size matters: how scaling affects the interaction between grid and border cells. Front. Comput. Neurosci. 11, 65.
Sarel, A., Finkelstein, A., Las, L., Ulanovsky, N., 2017. Vectorial representation of spatial goals in the hippocampus of bats. Science 355 (6321), 176–180.
Sasiadek, J., 2002. Sensor fusion. Annu. Rev. Control. 26 (2), 203–228.
Savinov, N., Dosovitskiy, A., Koltun, V., 2018. Semi-parametric topological memory for navigation. arXiv:1803.00653. [Cs].
Savva, M., Kadian, A., Maksymets, O., Zhao, Y., Wijmans, E., Jain, B., Straub, J., Liu, J., Koltun, V., Malik, J., Parikh, D., Batra, D., 2019. Habitat: a platform for embodied AI research.
Schiewer, R., Subramoney, A., Wiskott, L., 2024. Exploring the limits of hierarchical world models in reinforcement learning. Sci. Rep. 14 (1), 26856.
Schmitt, J., Paradis, A.-L., Boucher, M., Andrieu, L., Barnéoud, P., Rondi-Reig, L., 2021. Flexibility as a marker of early cognitive decline in humanized apolipoprotein e $E4$ (ApoE4) mice. Neurobiol. Aging 102, 129–138.
Schrittwieser, J., Antonoglou, I., Hubert, T., Simonyan, K., Sifre, L., Schmitt, S., Guez, A., Lockhart, E., Hassabis, D., Graepel, T., Lillicrap, T., Silver, D., 2020. Mastering atari, go, chess and shogi by planning with a learned model. Nature 588 (7839), 604–609.
Schwalger, T., Deger, M., Gerstner, W., 2017. Towards a theory of cortical columns: from spiking neurons to interacting neural populations of finite size. PLoS Comput. Biol. 13 (4), e1005507.
Scleidorovich, P., Llofriu, M., Fellous, J.-M., Weitzenfeld, A., 2020. A computational model for spatial cognition combining dorsal and ventral hippocampal place field maps: Multiscale navigation. Biol. Cybernet. 114 (2), 187–207.
Serino, S., Cipresso, P., Morganti, F., Riva, G., 2014. The role of egocentric and allocentric abilities in alzheimer's disease: A systematic review. Ageing Res. Rev. 16, 32–44.
Shaikh, D., Rañó, I., 2020. Braitenberg vehicles as computational tools for research in neuroscience. Front. Bioeng. Biotechnol. 8.
Shapiro, M.L., Tanila, H., Eichenbaum, H., 1997. Cues that hippocampal place cells encode: Dynamic and hierarchical representation of local and distal stimuli.. Hippocampus 7 (6), 624–642.
Sharpe, T., Webb, B., 1998. Simulated and situated models of chemical trail following in ants. In: Pfeifer, R., Blumberg, B., Meyer, J.-A., Wilson, S.W. (Eds.), From Animals To Animats 5: Proceedings of the Fifth International Conference on Simulation of Adaptive Behavior. The MIT Press.

Sheynikhovich, D., Arleo, A., 2010. A reinforcement learning approach to model interactions between landmarks and geometric cues during spatial learning. Computational Cognitive Neuroscience III: Selected Presentations from CCNC-09, Brain Res. Computational Cognitive Neuroscience III: Selected Presentations from CCNC-09, 1365.35–47,

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.. Psychol Rev 116 (3), 540–566.

Silavi, T., Hakimpour, F., Claramunt, C., Nourian, F., 2017. The legibility and permeability of cities: Examining the role of spatial data and metrics. ISPRS Int. J. Geo-Inf. 6 (4), 101.

Silver, D., Huang, A., Maddison, C.J., Guez, A., Sifre, L., van den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., Dieleman, S., Grewe, D., Nham, J., Kalchbrenner, N., Sutskever, I., Lillicrap, T., Leach, M., Kavukcuoglu, K., Graepel, T., Hassabis, D., 2016. Mastering the game of go with deep neural networks and tree search. Nature 529 (7587), 484–489.

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.

Soegiarto, D., Trilaksono, B.R., Adiprawita, W., Idris Hidayat, E.M., Prabowo, Y.A., 2022. Combining SLAM, GDM, and anemotaxis for gas source localization in unknown and GPS-denied environments. | ebscohost. 14, (3), (ISSN: 2085-6830) p. 514.

Sorscher, B., Mel, G.C., Ocko, S.A., Giocomo, L.M., Ganguli, S., 2022. A unified theory for the computational and mechanistic origins of grid cells. Neuron.

Stachenfeld, K.L., Botvinick, M.M., Gershman, S.J., 2017. The hippocampus as a predictive map. Nature Neurosci. 20 (11), 1643–1653.

Stemmler, M., Mathis, A., Herz, A.V.M., 2015. Connecting multiple spatial scales to decode the population activity of grid cells. Sci. Adv. 1 (11).

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.

Stolle, M., Precup, D., 2002. Learning options in reinforcement learning. In: Koenig, S., Holte, R.C. (Eds.), Abstraction, Reformulation, and Approximation. In: Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, pp. 212–223.

Sutton, R.S., Barto, A.G., 1998. Reinforcement Learning: An Introduction.. MIT Press.

Sutton, R.S., Precup, D., Singh, S., 1999. Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning. Artificial Intelligence 112 (1), 181–211.

Taherkhani, A., Belatreche, A., Li, Y., Cosma, G., Maguire, L.P., McGinnity, T.M., 2020. A review of learning in biologically plausible spiking neural networks. Neural Netw. 122, 253–272.

Taube, J.S., Muller, R.U., Ranck, J.B., 1990a. Head-direction cells recorded from the postsubiculum in freely moving rats. i. description and quantitative analysis. J. Neurosci. 10 (2), 420–435.

Taube, J.S., Muller, R.U., Ranck, J.B., 1990b. Head-direction cells recorded from the postsubiculum in freely moving rats. II. effects of environmental manipulations. J. Neurosci. 10 (2), 436–447.

Thrun, S., 1998. Learning metric-topological maps for indoor mobile robot navigation. Artificial Intelligence 99 (1), 21–71.

Thurm, F., Schuck, N.W., Fauser, M., Doeller, C.F., Stankevich, Y., Evens, R., Riedel, O., Storch, A., Lueken, U., Li, S.-C., 2016. Dopamine modulation of spatial navigation memory in parkinson's disease. Neurobiol. Aging 38, 93–103.

Toledo, S., Shohami, D., Schiffner, I., Lourie, E., Orchan, Y., Bartan, Y., Nathan, R., 2020. Cognitive map–based navigation in wild bats revealed by a new high-throughput tracking system. Science 369 (6500), 188–193.

Tolman, E.C., 1948. Cognitive maps in rats and men.. Psychol Rev 55 (4), 189–208.

Tolman, E.C., Honzik, C.H., 1930. Introduction and Removal of Reward, and Maze Performance in Rats, vol. 4, University of California Publications in Psychology, pp. 257–275.

Valerio, S., Taube, J.S., 2012. Path integration: How the head direction signal maintains and corrects spatial orientation. Nature Neurosci. 15 (10), 1445–1453.

Vijayabaskaran, S., Cheng, S., 2022. Navigation task and action space drive the emergence of egocentric and allocentric spatial representations. PLoS Comput. Biol. 18 (10), e1010320.

Vijayabaskaran, S., Cheng, S., 2024. Competition and integration of visual and goal vector signals for spatial navigation. BioRxiv.

Villarreal, B.L., Olague, G., Gordillo, J.L., 2016. Synthesis of odor tracking algorithms with genetic programming. Neurocomputing 175, 1019–1032.

Wang, J., Lin, S., Liu, A., 2023a. Bioinspired perception and navigation of service robots in indoor environments: a review. Biomimetics 8 (4), 350.

Wang, S., Wu, Z., Hu, X., Lin, Y., Lv, K., 2023b. Skill-based hierarchical reinforcement learning for target visual navigation. IEEE Trans. Multimed. 25, 8920–8932.

Warren, W.H., 2019. Non-euclidean navigation. J. Exp. Biol. 222, (Suppl. 1).

Warren, W.H., Rothman, D.B., Schnapp, B.H., Ericson, J.D., 2017. Wormholes in virtual space: from cognitive maps to cognitive graphs. Cognition 166, 152–163.

Watson, J.B., 1907. Kinæ sthetic and organic sensations: their role in the reactions of the white rat to the maze. Psychol. Rev.: Monogr. Suppl. 8 (2), i–101.

Wei, X.-X., Prentice, J., Balasubramanian, V., 2015. A principle of economy predicts the functional architecture of grid cells. ELife 4, e08362.

Wientjes, S., Holroyd, C.B., 2024. The successor representation subserves hierarchical abstraction for goal-directed behavior. In: Graham, L.J. (Ed.), PLoS Comput. Biol. 20 (2), e1011312.

Wiering, M., Schmidhuber, J., 1997. HQ-learning. Adapt. Behav. 6 (2), 219–246.

Wikenheiser, A.M., Redish, A.D., 2015. Hippocampal theta sequences reflect current goals.. Nature Neurosci. 18 (2), 289–294.

Wilson, M.A., McNaughton, B.L., 1993. Dynamics of the hippocampal ensemble code for space. Science 261 (5124), 1055–1058.

Wiltschko, R., Wiltschko, W., 2023. Animal navigation: How animals use environmental factors to find their way. Eur. Phys. J. Spec. Top. 232 (2), 237–252.

Wu, Y., Wu, Y., Gkioxari, G., Tian, Y., 2018. Building generalizable agents with a realistic and rich 3D environment.

Zein, B., Long, J.A., Safi, K., Kölzsch, A., Wikelski, M., Kruckenberg, H., Demšar, U., 2021. Simulation experiment to test strategies of geomagnetic navigation during long-distance bird migration. Mov. Ecol. 9 (1), 46.

Zeithamova, D., Bowman, C.R., 2020. Generalization and the hippocampus: more than one story? Neurobiol. Learn. Mem. 175, 107317.

Zhang, S., Manahan-Vaughan, D., 2013. Spatial olfactory learning contributes to place field formation in the hippocampus.. Cerebral cortex (New York, N.Y. : 1991).

Zhao, D., Zhang, Z., Lu, H., Cheng, S., Si, B., Feng, X., 2022. Learning cognitive map representations for navigation by sensory–motor integration. IEEE Trans. Cybern. 52 (1), 508–521.

Zhu, Y., Mottaghi, R., Kolve, E., Lim, J.J., Gupta, A., Fei-Fei, L., Farhadi, A., 2016. Target-driven visual navigation in indoor scenes using deep reinforcement learning. arXiv:1609.05143. [Cs].

Zivkovic, Z., Bakker, B., Krose, B., 2006. Hierarchical map building and planning based on graph partitioning. In: Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006.. ICRA 2006, pp. 803–809.