



Flexible Path Planning in a Spiking Model of Replay and Vicarious Trial and Error

Jeffrey L. Krichmar¹(✉) , Nicholas A. Ketz^{2,3}, Praveen K. Pilly³,
and Andrea Soltoggio⁴

¹ Department of Cognitive Sciences, Department of Computer Science,
University of California, Irvine, Irvine, CA 92697-5100, USA
jkrichma@uci.edu

² Colossal Biosciences, Madison, WI, USA
nick@colossal.com

³ Center for Human-Machine Collaboration, Information and Systems Sciences
Laboratory, HRL Laboratories, Malibu, CA 90265, USA
pkpilly@hrl.com

⁴ Computer Science Department, Loughborough University,
Loughborough LE11 3TU, UK
a.soltoggio@lboro.ac.uk

Abstract. Flexible planning is necessary for reaching goals and adapting when conditions change. We introduce a biologically plausible path planning model that learns its environment, rapidly adapts to change, and plans efficient routes to goals. Our model addresses the decision-making process when faced with uncertainty. We tested the model in simulations of human and rodent navigation in mazes. Like the human and rat, the model was able to generate novel shortcuts, and take detours when familiar routes were blocked. Similar to rodent hippocampus recordings, the neural activity of the model resembles neural correlates of Vicarious Trial and Error (VTE) during early learning or during uncertain conditions and preplay predicting a future path after learning. We suggest that VTE, in addition to weighing possible outcomes, is a way in which an agent may gather information for future use.

Keywords: Cognitive map · Hippocampus · Navigation · Preplay · Spiking neural network

1 Introduction

Flexible planning is an important aspect of cognition that is especially useful when achieving a goal under uncertain conditions. Multi-step planning can be

Supported by the Air Force Office of Scientific Research (AFOSR) Contract No. FA9550-19-1-0306, by the National Science Foundation (NSF-FO award ID 2024633), and by the United States Air Force Research Laboratory (AFRL) and Defense Advanced Research Projects Agency (DARPA) under Contract No. FA8750-18-C-0103.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2022
L. Cañamero et al. (Eds.): SAB 2022, LNAI 13499, pp. 177–189, 2022.
https://doi.org/10.1007/978-3-031-16770-6_15

thought of as a process that uses a cognitive map to guide a sequence of actions towards a goal [12]. In the context of spatial navigation, humans and other animals have the ability to choose alternate routes when necessary. Moreover, they can express spatial knowledge in the form of novel shortcuts over locations not previously explored [3].

Neural correlates of planning have been observed in rodent hippocampus place cell responses. In hippocampal replay, plans are formed by reactivating place cells of previously experienced location sequences [4, 13]. Computational models have shown how hippocampal preplay, which is sometimes called forward replay, can plan paths towards new goals in familiar environments and replan when familiar paths are no longer viable [11, 15]. However, these models of preplay do not address the decision-making process when faced with uncertainty, and they may not explain how knowledge of never experienced locations can be stored and expressed.

In the mid-twentieth century, Edward Tolman described the flexible, intelligent behavior observed in animals as a cognitive map [16]. One aspect of a cognitive map was Vicarious Trial and Error (VTE), which is the ability to weigh one’s options before taking decisive action. Similar to hippocampal preplay, neural correlates of VTE have been observed in hippocampal CA1 where neurons with place-specific firing exhibit “sweeps” of activity representing the locations at which the animal considers its left versus right turn choice [8, 14]. VTE seems to occur when the animal is uncertain about which path to take. After experience, hippocampal preplay occurs for the path the animal intends to take, but not the alternatives.

In this paper, we introduce a computational model of path planning that demonstrates VTE during early learning or when environmental conditions change. We further show that this model can acquire knowledge about never experienced paths, and rapidly adapt to express this knowledge when challenged. We suggest this activity is comparable to VTE observed in animals and that VTE may assist in the acquisition of knowledge that can be later expressed as novel shortcuts or rapid re-routing.

2 Methods

2.1 Spiking Wave Propagation

Spiking wavefront propagation is a neuromorphic navigation algorithm inspired by neuronal dynamics and connectivity of neurons in the brain [7]. The algorithm is loosely based on the responses of place cells in the hippocampus during preplay. The spiking wavefront propagation algorithm learns by adjusting axonal delays. This was inspired by biological evidence suggesting that the myelin sheath, which wraps around and insulates axons, may undergo a form of activity-dependent plasticity [5]. These studies have shown that the myelin sheath becomes thicker with learning motor skills and cognitive tasks. A thicker myelin sheath implies faster conduction velocities and improved synchrony between neurons. In the

present work, adjusting axonal delays fits better with the idea behind wave propagation than the more commonly used synaptic weight updates.

The spiking wavefront propagation algorithm assumes a grid representation of space. Each grid unit corresponds to a discretized area of physical space, and connections between units represent the ability to travel from one grid location to a neighboring location. Each unit in the grid is represented by a single neuron with spiking dynamics, which are captured with a model described by the equations below. Further description of model can be found in [7].

The membrane potential of neuron i at time $t + 1$ is represented by Eq. 1:

$$v_i(t + 1) = u_i(t) + I_i(t + 1) \quad (1)$$

in which $u_i(t)$ is the recovery variable and $I_i(t)$ is the synaptic input at time t , which denotes the advancement of the path planning algorithm and is not related to clock time. In practice, the algorithm is lightweight and t is much shorter than real time.

The recovery variable $u_i(t + 1)$ is described Eq. 2:

$$u_i(t + 1) = \begin{cases} -5 & \text{if } v_i(t) = 1 \\ \min(u_i(t) + 1, 0) & \text{otherwise} \end{cases} \quad (2)$$

such that immediately after a membrane potential spike, the recovery variable starts as a negative value and linearly increases toward a baseline value of 0.

The synaptic input I at time $t + 1$ is given by Eq. 3:

$$I_i(t + 1) = \sum_{j=1}^N \begin{cases} 1 & \text{if } d_{ij}(t) = 1 \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

such that $d_{ij}(t)$ is the delay counter of the signal from neighboring neuron j to neuron i . This delay is given by Eq. 4:

$$d_{ij}(t + 1) = \begin{cases} D_{ij} & \text{if } v_j(t) \geq 1 \\ \max(d_{ij}(t) - 1, 0) & \text{otherwise} \end{cases} \quad (4)$$

which behaves as a timer corresponding to an axonal delay with a starting value of $D_{ij}(t)$, counting down until it reaches 0.

The value of $D_{ij}(t)$ depends on the environmental cost associated with traveling between the locations associated with neurons i and j . Synaptic input comes from neighboring connected neurons. When a neighboring neuron spikes, the synaptic input I_i is increased by 1. This triggers the neuron to spike. After the spike, the recovery variable u_i is set to -5 , then gradually recovers back to 0, modeling the refractory period. Also, immediately after spiking, all delay counters d_{ij} for all neighbor neurons j are set to their current values of D_{ij} . In the present paper, and in contrast to [7], D_{ij} are set to some initial value (10 in the experiments described in Sect. 3.1 and 5 in the experiments described in Sect. 3.2) and then change based on the agent's observations in the environment with the learning rule described in Sect. 2.2.

2.2 E-Prop and Back-Propagation Through Time

The E-Prop algorithm was introduced to learn sequences in spiking recurrent neural networks [2]. Learning was dictated by a loss function related to the desired output. The credit assignment problem was resolved by subjecting each neuron to an eligibility trace based on the neuron’s recent activity. Weights between neurons were updated based on the loss and the value of the eligibility trace. In this way, the E-Prop algorithm resembled Back-Propagation Through Time (BPTT).

In the present model, E-Prop is used to learn sequences of movements through an environment based on the traversal cost. The active neurons during the wave propagation are eligible to be updated. Once eligible, they are subject to an exponential decay due to an eligibility trace. The most eligible neurons are those most recently active relative to when the wave reaches the goal destination. After the path is calculated, E-Prop is applied to weights projecting from neurons along the calculated path. Since these weights are connected to locations adjacent to the path, we assume the agent can observe the features (e.g., traversal cost) at these map locations. In this way, E-Prop solves the credit assignment problem by rewarding paths that lead to goal locations, while also learning about the environmental structure of nearby map locations.

Weights denote the axonal delay in sending an action potential or spike from the pre-synaptic neuron to the post-synaptic neuron. We apply the E-Prop algorithm to these weights so that the spiking neural network learns an axonal delay corresponding to environmental features observed in map_{xy} , which is the spatial location at Cartesian coordinates (x, y) of the map . The values of D_{ij} are updated when the agent reaches the goal destination. The learning rule is described by Eq. 5:

$$D_{ij}(t+1) = D_{ij}(t) + \delta(e_i(t)(map_{xy} - D_{ij}(t))) \quad (5)$$

where δ is the learning rate, set to 0.5, $e_i(t)$ is an eligibility trace for neuron i , and map_{xy} represents the observed cost for traversing the location (x, y) , which corresponds to neuron i . This rule is applied for each of the neighboring neurons, j , of neuron i . By using this method of axonal plasticity, the agent can simultaneously explore and learn, adapting to changes in the environment. The loss in Eq. 5 is $map_{xy} - D_{ij}$.

The eligibility trace for neuron i is given by Eq. 6:

$$e_i(t+1) = \begin{cases} 1 & \text{if } v_j(t) \geq 1 \\ e_i(t) - \frac{e_i(t)}{\tau} & \text{otherwise} \end{cases} \quad (6)$$

where τ is the rate of decay for the eligibility trace. The setting of τ will be explored in Sect. 3.1.

2.3 Extracting a Path from the Spike Wavefront Algorithm

We illustrate the spiking wavefront algorithm with a simple example (Fig. 1). In this example, there is a 4×4 spiking neural network that has converged to a learned

representation of the space (Fig. 1A). The inner section of the environment contains an obstacle that is twice as costly to traverse than the outer section.

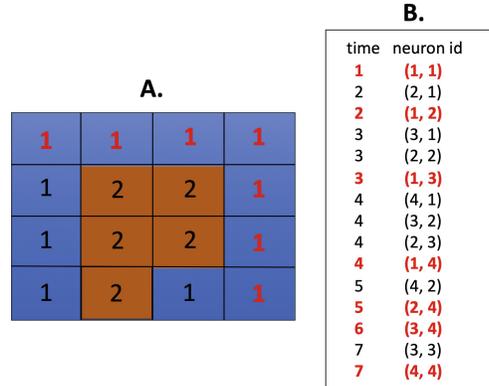


Fig. 1. Simple example with a 4×4 neural network. A neuron is connected to its 4 neighbors (North, South, East, West). The numbers denote the learned axonal delays between each neuron and its neighbors. The task is to find a path from (1,1) to (4,4). The resulting path found by the spike wave propagation algorithm is denoted in red font. A. 4×4 neural network representing a simple environment. The delays, which correspond to a cost map of the environment, are numbered. B. Example of using the spike table for extracting a path from (1,1) to (4,4). The extracted path is shown in red font. (Color figure online)

To extract a path from these learned delay-encoded costs, the neuron corresponding to the current location (1,1) is induced to spike. This causes an input current to be sent to neighboring neurons, starting a traveling wave of activity across the area covered by the grid. As each neuron spikes, the spike index and the time of spike are logged in a table. Figure 1B illustrates how this information is used to trace a path from the start to the goal location (4,4). To extract a path from the spike table, a list of neuron IDs is maintained, starting with the goal neuron. The first spike time of the goal neuron is found (see Fig. 1B). Then, the timestamps are decremented until the spiking of a neuron neighboring the goal neuron is found. The process then continues by finding spikes of neurons neighboring the most recent neuron. The process is repeated until the start neuron is found. The result is an optimal path between the start and goal (see red font in Fig. 1B).

3 Results

The spiking wave propagation algorithm with E-Prop learning was tested in two different environments: 1) Human navigation in a virtual maze [3], and 2) Rodent navigation in the Tolman detour maze [1]. We simulated their experimental protocols and compared the simulation results with the human and rodent experimental results.

3.1 Simulating Human Navigation and Taking Novel Shortcuts

In [3], human participants navigated through a virtual hedge maze where they first took several laps on a fixed path, which we will refer to as the “learned route” in the remainder of the paper, and then were tested by how well they could navigate between locations (Fig. 2). Along the learned route, there were landmarks, such as a chair, mailbox, potted plant, and picnic table. During the test phase, participants were placed at one landmark and told to navigate to another landmark. Some subjects took learned routes to landmarks, and others took novel shortcuts to landmarks. In another experiment, participants were told to take the shortest path to a landmark. This resulted in more participants taking novel shortcuts, and suggested that participants had survey knowledge, an allocentric mental map of the environment, even if they had previously not chosen to express this knowledge.

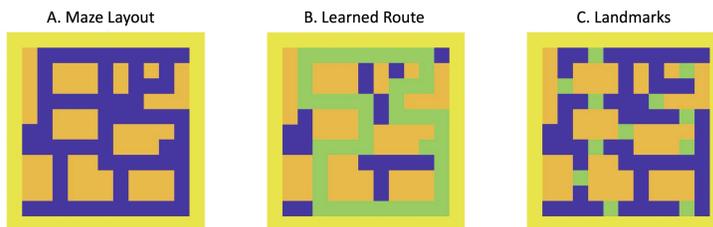


Fig. 2. Maps derived from a virtual environment used in human navigation studies [3]. A. Map is a 13×13 grid. Yellow denotes the border, orange denotes untraversable areas, and blue denotes traversable areas. B. Green denotes the fixed route learned by participants. C. Green denotes landmark locations. (Color figure online)

We converted the virtual maze into a 13×13 grid of Cartesian coordinates, which corresponded to a spiking neural network of the same size. Similar to [3], we forced the model to take a fixed tour of the environment by presenting the path planning algorithm with starting and ending locations that were close by and along a straight line. The weights of the spiking neural network, which correspond to axonal delays, were initially set to 10. Through repeated application of E-Prop learning, the weights began to reflect the costs associated with maze features (i.e., traversable regions, boundaries, and obstacles).

The performance of the spike wave propagation algorithm with E-Prop learning is dependent on the amount of training and how long neurons are eligible for updates. We tested the sensitivity of the algorithm by varying the number of loops on the learned route and the time constant of the eligibility trace (τ in Eq. 6). Similar to the human study, after these training loops, starting and goal locations, which corresponded to landmarks (Fig. 2C), were presented to the algorithm. We chose 24 starting and goal pairs. Navigation errors were defined as the number of times the path calculated by the algorithm attempted to navigate over untraversable regions, such as obstacles or outside the maze borders.

These errors indicated how well the spiking neural network learned the maze layout. With some training and a long enough time constant, the neural network learned the maze well enough that there were little or no navigation errors. Based on these exploratory results, we used 6 loops over the learned route and a time constant τ equal to 25 for the remainder of the simulations.

During experience on the learned route, the activity of the spiking neural network reflected the uncertainty of maze features. This could be observed in the number of neurons eligible for updates. For example on the first loop, more neurons were activated during path planning than on the sixth loop (Fig. 3). The path planning algorithm propagated a wave of neural activity based on the weights corresponding to axonal delays. Therefore, early in the training experience, when features of the maze were unknown, more neurons became active and eligible (see Eq. 6). However, later in the training, these eligible neurons were confined to the learned route, as well as regions near the learned route that could lead to novel shortcuts.

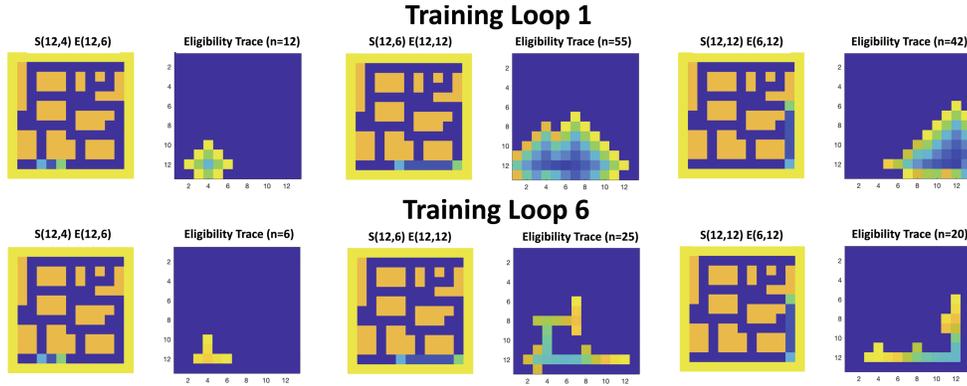


Fig. 3. Left panels. Path calculated by the spike wave algorithm for segments of the learned route. The text S(row,col) and E(row,col) above each figure denotes the starting and ending locations, respectively. Right panels. Active neurons during the path planning. The pixels denote neurons with hotter colors corresponding to more recently active neurons according to the eligibility trace. Dark blue pixels denote inactive neurons. The number of eligible neurons is denoted by n in the text above the figure. (Color figure online)

As learning progressed, the total loss ($map_{xy} - D_{ij}$ in Eq. 5) decreased, which meant the network was learning the maze features, and the number of eligible neurons decreased (Fig. 4). We suggest that the eligibility trace is qualitatively similar to VTE and hippocampal replay. When the agent was uncertain about the path, there were more neurons active, reflecting a number of alternative routes. This is somewhat like the neural correlates of VTE that have been observed in rodent hippocampus during early exposure to an environment [8, 14]. After exposure, the neural activity was mostly confined to the route taken.

This is somewhat like preplay activity observed in rodent hippocampus when the animal is familiar with an environment [13, 14].

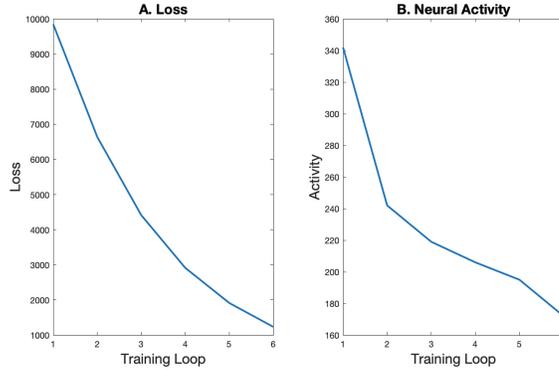


Fig. 4. A. Loss per training loop. B. Size of the eligibility trace.

The neural correlates of VTE may facilitate the construction of cognitive maps and the ability to take novel shortcuts. During learned route training, the eligible neurons spilled over into regions of the environment that were not on the learned route. This may have led to the calculation of novel routes between landmarks (Fig. 5). After learned route training, the spiking wavefront propagation algorithm calculated novel shortcuts on 12 out of 24 trials. Adding blockades to the middle section of the learned route (i.e., coordinates (2,7), (7,7) and (12,7)) led to more shortcuts (i.e., 15 out of 24). Similar to [3], this suggested that there was additional knowledge contained in the neural network, which did not express itself until challenged to do so.

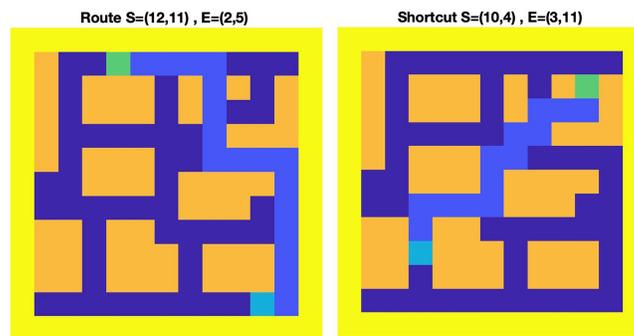


Fig. 5. Paths calculated between landmarks. Left. Path along the learned route. Right. Novel shortcut.

3.2 Simulating Rodent Navigation in Tolman Detour Task

In the Tolman detour task, rats were required to choose detours when well-known paths were blocked. Initially, rats were trained to run along a straight corridor from a start location to a goal location (from A to B in Fig. 6A). After training, rats needed to plan a detour path when barriers were placed along the original path. If the barrier was placed at P1, the rat could only use the long detour to get from A to B. But if the barrier was placed at P2, the rat might choose either the long detour on the right or the short detour on the left.

We constructed an environment to simulate the Tolman maze. The corridors had a cost of 1 (blue regions in Fig. 6A), the maze borders had a cost of 120 (yellow regions in Fig. 6A). During the detour test, the barriers, P1 or P2, were placed at (5,7) and (7,7), respectively and the cost was set to 120. The weights in the neural network were initialized to 5.

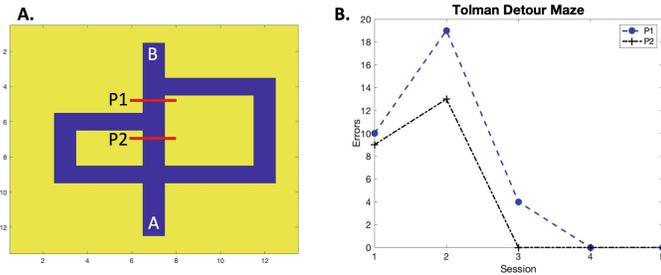


Fig. 6. Tolman detour maze task. A. The maze used in the rodent and modeling experiments. The agent is trained to go from A to B. After training, a barrier is placed at P1 or P2. B. Errors during navigation with barriers at P1 or P2. Each session denotes 4 trials from A to B.

During training, the spiking wave propagation algorithm with E-Prop learned a path from A to B over 20 trials. After training, a barrier was placed at P1 or P2. Then the spike wave propagation algorithm underwent 20 additional trials to plan paths from A to B.

The spiking wave propagation algorithm with E-Prop quickly learned the best detour given the barrier (Fig. 6B). Each session contained 4 trials where the spike wave propagation planned a path from A to B. Errors were whenever the planned path went outside the corridor or attempted to traverse through a barrier. Backtracking and re-routing after an error occurred was not simulated. After the first 2 sessions, the agent planned mostly error-free paths.

Similar to rodent experiments [1], when the barrier was placed at P1, the agent took the longer detour (left panels of Fig. 7). Like the rat, the agent attempted to travel straight from A to B when the barrier was introduced. By the 10th trial, the agent took the longer detour to reach its goal. Note how the number of timesteps (TS), the path length (PL), and the loss (see Eq. 5) decreased with the number of trials.

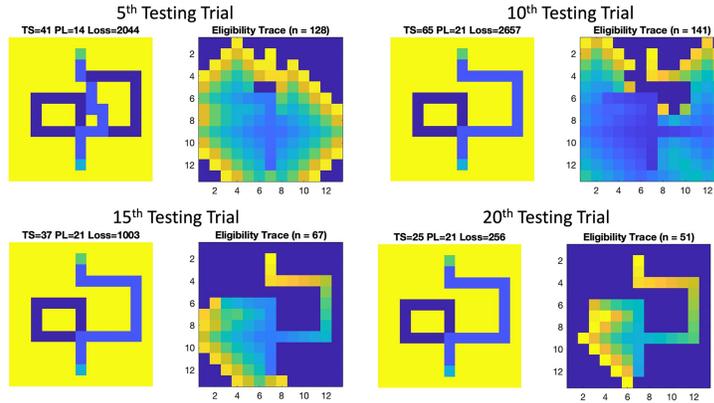


Fig. 7. Tolman detour maze task with barrier placed at P1. Left panels. The path planned by the agent. The cyan pixel denotes the start location, and the green pixel denotes the end location. The light blue pixels denote the planned path. Title text contains timesteps needed to calculate the path (TS), path length (PL), and loss when comparing the neural network weights to the maze values. Right panels. The eligible neurons are shown, with the hotter colors signifying more recent activity in the path planning. (Color figure online)

When the barrier was placed at P2, the model agent took the shorter detour (left panels of Fig. 8). This differed slightly from the rat experiments. In the rat experiments, the rat occasionally took the long detour, but did favor the short detour. This could be simulated by including prior preferences or adding noise to the model. By trial 10, the spiking wave propagation algorithm exclusively planned paths along the shorter detour. Similar to when the barrier was placed at P1, the timesteps (TS), path length (PL), and loss decreased as trials progressed.

Introducing a barrier forced the agent to learn alternative routes. This uncertainty was reflected in the eligibility trace of the neural network model (right panels of Fig. 7 and Fig. 8). In the first few trials, the model was exploring possible alternatives. We suggest this neural activity is comparable to VTE in the hippocampus. By trial 15 and later, activity was mainly confined to the path to be taken by the agent. We suggest that this is more akin to preplay in the hippocampus. Taken together, these results show how the spiking wave propagation algorithm with E-Prop can rapidly adapt and re-route when changes occur. Furthermore, the uncertainty and consideration of alternative routes can be observed in the neural network activity.

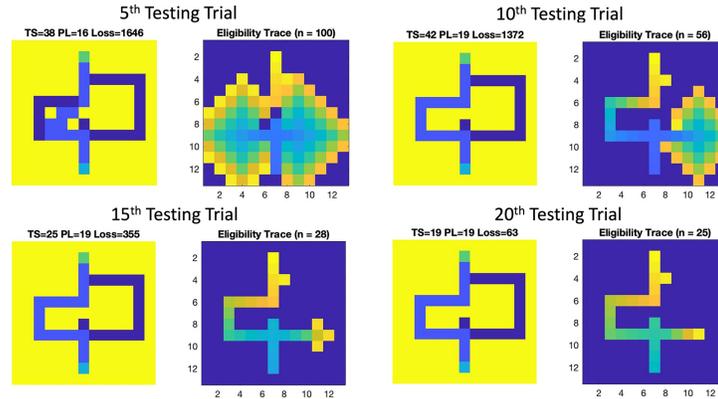


Fig. 8. Tolman detour maze with barrier placed at P2. Notations same as in Fig. 7.

4 Discussion

A hallmark of cognitive behavior is flexible planning [12]. In the present study, we show that a path planning algorithm from robotics, coupled with a biologically plausible learning rule, could demonstrate flexible path planning, and generate neural correlates of assessing alternatives when faced with uncertainty. The spiking wavefront propagation path planner presented here has been used for robot navigation [7], and to simulate human navigation [10]. Unlike other path planners, the spiking wavefront algorithm is a network of spiking neurons that is compatible with power efficient neuromorphic hardware, as was shown its implementation on the IBM TrueNorth NS1e [6]. Although evidence suggests that these activations occur sequentially [14], wavefront propagation operates in parallel, which can speed up processing. This has practical applications for autonomous systems operating at the edge and may be of interest to follow up experimentally now that more neurons can be recorded simultaneously.

The present work explores how a biologically plausible learning rule [2], specifically designed for spiking neural networks, could extend the spiking wavefront propagation algorithm. Rather than having learning be related to changes in synaptic efficacy, the present algorithm changed the delays between pre- and post-synaptic neurons. This was inspired by evidence suggesting that the myelin sheath, which wraps around and insulates axons, undergoes a form of activity-dependent plasticity [5]. Although, we are not suggesting that hippocampal learning is solely due to changes in conductance velocity, it is an intriguing form of plasticity rarely applied to neural networks that also provides a simple and effective method of explainability and uncertainty quantification.

The spiking wavefront propagation path planner presented here has activity similar to neural correlates of VTE: 1) It is more prominent during early learning and when faced with environmental challenges or uncertainty. 2) The wave of activity propagates to alternative choices, and 3) After experience, the wave of activity more resembles preplay in that it becomes confined to the future choice.

We suggest that VTE, in addition to weighing possible outcomes is a way in which an agent may gather information for future use. Such an algorithm could be applied to mobile robots that take cost, which is based on the application goals, into account during path planning rather than deep learning approaches that require off-line training [9].

References

1. Alvernhe, A., Save, E., Poucet, B.: Local remapping of place cell firing in the Tolman detour task. *Eur. J. Neurosci.* **33**(9), 1696–705 (2011). <https://doi.org/10.1111/j.1460-9568.2011.07653.x>
2. Bellec, G., et al.: A solution to the learning dilemma for recurrent networks of spiking neurons. *Nat. Commun.* **11**(1), 3625 (2020). <https://doi.org/10.1038/s41467-020-17236-y>
3. Boone, A.P., Maghen, B., Hegarty, M.: Instructions matter: individual differences in navigation strategy and ability. *Mem. Cogn.* **47**(7), 1401–1414 (2019). <https://doi.org/10.3758/s13421-019-00941-5>
4. Dragoi, G., Tonegawa, S.: Preplay of future place cell sequences by hippocampal cellular assemblies. *Nature* **469**(7330), 397–401 (2011). <https://doi.org/10.1038/nature09633>
5. Fields, R.D.: A new mechanism of nervous system plasticity: activity-dependent myelination. *Nat. Rev. Neurosci.* **16**(12), 756–67 (2015). <https://doi.org/10.1038/nrn4023>
6. Fischl, K.D., Fair, K., Tsai, W., Sampson, J., Andreou, A.: Path planning on the trueneurosynaptic system. In: 2017 IEEE International Symposium on Circuits and Systems (ISCAS), pp. 1–4 (2017). <https://doi.org/10.1109/ISCAS.2017.8050932>
7. Hwu, T., Wang, A.Y., Oros, N., Krichmar, J.L.: Adaptive robot path planning using a spiking neuron algorithm with axonal delays. *IEEE Trans. Cogn. Develop. Syst.* **10**(2), 126–137 (2018). <https://doi.org/10.1109/Tcds.2017.2655539>
8. Johnson, A., Redish, A.D.: Neural ensembles in CA3 transiently encode paths forward of the animal at a decision point. *J. Neurosci.* **27**(45), 12176–89 (2007). <https://doi.org/10.1523/JNEUROSCI.3761-07.2007>
9. Kahn, G., Abbeel, P., Levine, S.: BADGR: an autonomous self-supervised learning-based navigation system. [arXiv:2002.05700](https://arxiv.org/abs/2002.05700) (2020)
10. Krichmar, J.L., He, C.: Importance of path planning variability: a simulation study. *Top. Cogn. Sci.* (2021). <https://doi.org/10.1111/tops.12568>
11. Mattar, M.G., Daw, N.D.: Prioritized memory access explains planning and hippocampal replay. *Nat. Neurosci.* **21**(11), 1609–1617 (2018). <https://doi.org/10.1038/s41593-018-0232-z>
12. Miller, K.J., Venditto, S.J.C.: Multi-step planning in the brain. *Curr. Opin. Behav. Sci.* **38**, 29–39 (2021). <https://doi.org/10.1016/j.cobeha.2020.07.003>
13. Pfeiffer, B.E., Foster, D.J.: Hippocampal place-cell sequences depict future paths to remembered goals. *Nature* **497**(7447), 74–9 (2013). <https://doi.org/10.1038/nature12112>
14. Redish, A.D.: Vicarious trial and error. *Nat. Rev. Neurosci.* **17**(3), 147–59 (2016). <https://doi.org/10.1038/nrn.2015.30>

15. Stachenfeld, K.L., Botvinick, M.M., Gershman, S.J.: The hippocampus as a predictive map. *Nat. Neurosci.* **20**(11), 1643–1653 (2017). <https://doi.org/10.1038/nn.4650>
16. Tolman, E.C.: Cognitive maps in rats and men. *Psychol. Rev.* **55**(4), 189–208 (1948). <https://doi.org/10.1037/H0061626>