Biological Cybernetics manuscript No.
(will be inserted by the editor)

Modeling awake hippocampal reactivations with model-based bidirectional search

Mehdi Khamassi · Benoît Girard

Received: 30 July 2019 / Accepted: date

Abstract Hippocampal offline reactivations during reward-based learning, usually categorized as replay events, have been found to be important for performance improvement over time and for memory consolidation. Recent computational work has linked these phenomena to the need to transform reward information into state-action values for decision-making and to propagate it to all relevant states of the environment. Nevertheless, it is still unclear whether an integrated reinforcement learning mechanism could account for the variety of awake hippocampal reactivations, including variety in order (forward and reverse reactivated trajectories) and variety in the location where they occur (reward site or decision-point). Here we present a model-based bidirectional search model which accounts for a variety of hippocampal reactivations. The model combines forward trajectory sampling from current position and backward sampling through prioritized sweeping from states associated with large reward prediction errors until the two trajectories connect. This is repeated until stabilization of state-action values (convergence), which could explain why hippocampal reactivations drastically diminish when the animal's performance stabilizes. Simulations in a multiple T-maze task show that forward reactivations are prominently found at decision-points while backward reactivations are exclusively generated at reward sites. Finally, the model can generate imaginary trajectories that are not allowed to the agent during task performance. We raise some experimental predictions and implications for future studies of the role of the hippocampo-prefronto-striatal network in learning.

This work has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No 640891 (DREAM Project), and from the CNRS 80|PRIME research program (RHiPAR project). This work was performed within the Labex SMART (ANR-11-LABX-65) supported by French state funds managed by the ANR within the Investissements d'Avenir programme under reference ANR-11-IDEX-0004-02.

M. Khamassi and B. Girard
Sorbonne Université, CNRS
Institute of Intelligent Systems and Robotics (ISIR)
F-75005 Paris, France
Tel.: +33-6-50764492
Fax: +33-1-44275145
E-mail: firstname.lastname@sorbonne-universite.fr

Keywords Reinforcement Learning · Navigation · Hippocampal Replay · Computational Neuroscience

1 Introduction

The hippocampus contains place cells that encode the position of a rodent during active navigation (O'Keefe and Dostrovsky, 1971), by integrating exteroceptive and interoceptive sensory information. These cells also activate while the animal is passive, this has first been shown during sleep (Wilson and McNaughton, 1994; Lee and Wilson, 2002), and later during awake immobility (Foster and Wilson, 2006). These offline reactivations often appear to lack specific temporal organization, but a non negligible proportion of them correspond to sequences of places that have been previously visited (played in a forward or backward manner). In experimental settings where the animal is allowed to circulate in alleys in one direction only (for example on linear tracks, or in the looped mazed used by (Gupta et al., 2010) and that we simulate in this article, forward reactivations are defined as sequences of place cell activations that follow the same order as what was experienced, while backward reactivations correspond to sequences of place cells that were experienced in the reverse order (Fig. 1) (Foster and Wilson, 2006; Diba and Buzsáki, 2007). Some of these offline reactivations even correspond to so-called *imaginary* sequences (Gupta et al., 2010), e.g. the concatenation of two experienced sequences that share their respective last and first place cell, but that have never been experienced successively. These offline reactivations are compatible with longstanding memory consolidation theories (Buzsáki, 1989), which proposed that labile memories accumulated during daytime would be stabilized by nighttime replays, as has recently been shown in a causal manner (Maingret et al., 2016). It has however also been shown that they probably play a role in reinforcement learning, as disrupting sleep reactivations decreases learning speed (Girardeau et al., 2009) and stimulating the reward system during sleep reactivations creates place preference associations (de Lavilléon et al., 2015). Besides, awake hippocampal reactivations are classically associated to deliberation, possibly reflecting planning mechanisms to guide future behavior (Johnson et al., 2007; Pfeiffer and Foster, 2013). This multiplicity of roles has been reviewed in details in (Ólafsdóttir et al., 2018).

From a computational perspective, the reinforcement learning framework might be particularly suited to account for these hippocampal reactivations (Johnson and Redish, 2005; van der Meer et al., 2012; Khamassi and Humphries, 2012; Pezzulo et al., 2013; Foster, 2017; Cazé et al., 2018; Mattar and Daw, 2018). Various flavors of reinforcement learning algorithms have been successful at modeling animal behavior as well as providing a framework to explain neurophysiological data underlying the interactions between prefrontal cortex, basal ganglia, hippocampus and midbrain dopaminergic nuclei (Barto, 1995; Schultz et al., 1997; Guazzelli et al., 1998; Arleo and Gerstner, 2000; Foster et al., 2000; Daw et al., 2005; Khamassi and Humphries, 2012). In a recent review, we examined which of these algorithms could make use of offline activations of place representations (Cazé et al., 2018), and therefore would be potential candidates to explain the reinforcement learning-related hippocampal place-cell replay events. We stressed on the fact that experimentally observed "replays" may indeed sometimes correspond to the replay of the stored memory of a sequence of episodes, but could also

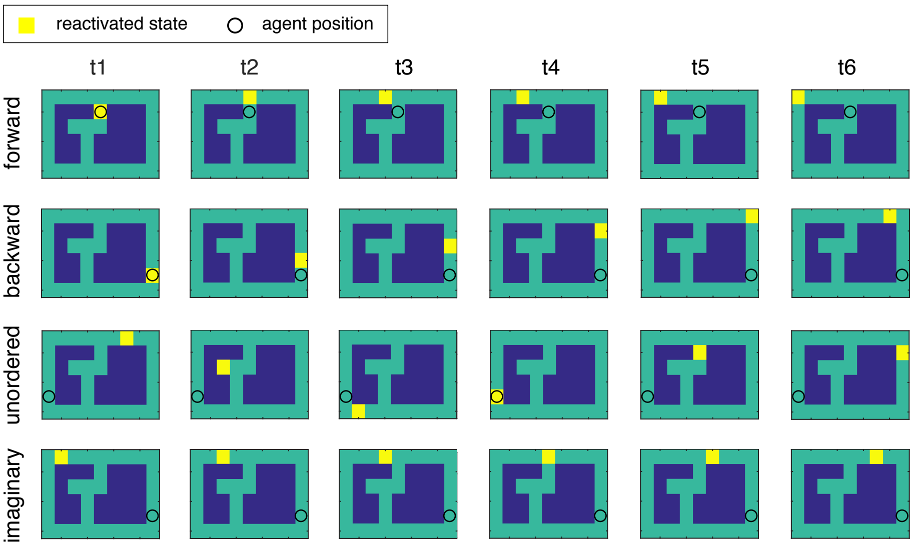

Fig. 1 Taxonomy of different orders of sequences of states (i.e. locations) mentally reactivated during 6 consecutive timesteps while an agent (here artificial) is immobile in a maze. The depicted maze is a simplified version of the multiple T-maze task of Gupta et al. (2010). The reactivated states were here generated through model simulations. Forward sequences correspond to the same order as what the agent usually performs while moving. Backward sequences correspond to the reverse order. Unordered sequences correspond to what a human experimenter could classify as random or noise. Imaginary sequences correspond to the concatenation of two trajectories whose combination has never been performed by the agent during task performance, and which may thus not be considered as a simple replay of past experience (Gupta et al., 2010).

sometimes be generated by the simulation of the effect of a series of actions given a starting point, through a sort of *mental traveling* using an internal model of the world. These two types of computational reactivations, model-free (MF) and model-based (MB), are difficult to separate experimentally, while both make sense from an algorithmic point of view. It has recently been proposed that despite their different mechanisms (i.e., model sampling versus replay of past experience), both types of reactivations may have the common goal to propagate reward information to all relevant states of the environment in order to learn and stabilize optimal state-action values (Pezzulo et al., 2017; Cazé et al., 2018; Mattar and Daw, 2018).

Among the different families of algorithms reviewed by (Cazé et al., 2018), the most promising ones to account for hippocampal awake replays were identified as the MB and the Dyna ones, especially the variants using *trajectory sampling* or *bidirectional search* as heuristics to get the most out of a limited offline activation budget. It has moreover been shown that the *prioritzed sweeping* heuristic alone in a Dyna algorithm (Moore and Atkeson, 1993; Peng and Williams, 1993) can generate awake reactivations that look like hippocampal backward replays, but is nevertheless not sufficient to explain forward and imaginary offline activations (Aubin et al., 2018). Besides, MF algorithms may be more suitable to account for hippocampal sleep replays because they reactivate recent experience in episodic memory rather than generating potential action plans for upcoming behavior. In parallel, Mattar and Daw (2018) proposed a variant to the Dyna algorithm with an additional component akin to *trajectory sampling*, thus extending the explainable phenomena. Nevertheless, while their model offers a normative perspective to explain why hippocampal reactivations are useful in terms of reward maximization, it suffers from some limitations such as unrealistic computation requirements, and agent omniscience about reward availability within the environment.

In the present work, we explore how a novel *bidirectional search* method under budget constraints can be implemented in the inference step of a model-based algorithm, by combining *prioritized sweeping* and *trajectory sampling.* We show that in a navigation context, the combination of these two search heuristics can reduce the computational cost. It moreover appears that it can be advantageous to use different exploration/exploitation trade-off parameter values for the online and the offline action selection steps, depending on whether performance or computational cost is to be optimized. The evolution of the amount of reactivations generated by the model during task learning is compatible with animal experimental data. Moreover, reactivations are generated preferentially at decision points and reward sites, and exhibit forward, backward and imaginary sequences, consistent with neurophysiological recordings.

2 Model description

Our current proposal is set in a standard Markov decision problem setting, where an agent visits discrete states $s \in \mathcal{S}$, using a finite set of discrete actions $a \in \mathcal{A}$. States represent here unique locations in space, equally spaced on a square grid, an information expected to be provided by place cell activity in the hippocampus. The model proposed here can be generalized to more continuous representations of space and actions, but the discrete case was kept for the sake of simplicity.

The family of reinforcement learning algorithms based on an internal model of the world learns a model of the transition probabilities between states, $T(s, a, s')$, and of the rewards $R(s, a)$. They use it to infer the value of the actions in each state by propagating the reward information in the whole transition graph. This propagation can be a computationally costly process, when all (s, a) are to be visited multiple times before accurate value estimation. Heuristics have been proposed to improve the inference cost (Sutton and Barto, 1998): *prioritized sweeping* proposes to visit the states starting from those whose value has changed recently, and then to their predecessors, the predecessors of their predecessors, and so on (Peng and Williams, 1993; Moore and Atkeson, 1993); *trajectory sampling* proposes to generate continuous trajectories from the current position until reward is reached (Barto et al., 1995), with the idea that it will favor the update of relevant states (avoiding wasting resources on states that have a very low probability of being visited). As we highlighted in (Cazé et al., 2018), these two can be associated in a *bidirectional search* process inspired by bidirectional planning approaches (Pohl, 1971). We thus propose here an implementation of a model-based algorithm which alternates *prioritized sweeping* and *trajectory sampling* phases to infer state-action values $Q(s, a)$ (i.e. to perform valuation, in terms borrowed from the neuroscience decision-making field), see Fig. 2.

The online part of the algorithm (Algo. 1) is quite classical: in the current state s, the agent selects the next action from a probability distribution over actions in state s computed from the Q-values with a softmax function:

$$P(a|s) = \frac{e^{\beta Q(s,a)}}{\sum_{i \in \mathcal{A}} e^{\beta Q(s,i)}} \tag{1}$$

where β is called the *inverse temperature* which regulates the exploration/exploitation trade-off by modulating the level of stochasticity of choice: the closer β is to zero, the more the contrast between Q-values will be attenuated, the extreme being for $\beta = 0$ which produces a flat action probability distribution (random exploration); in contrast, the larger the value of β, the more the contrast between Q-values will be enhanced, which makes the probability of the action with the highest Q-value close to 1 when β tends towards ∞ (exploitation).

Then, given the observed reward r (which in the present simulations will be equal to 1 in a single rewarding state of the environment representing the current reward location, and 0 elsewhere) and the new state s' reached by the agent after performing action a, the world model is updated and then the Q-value $Q(s, a)$ is updated with the updated model in a model-based manner. For the update of the world model, we use a very basic statistical approach for the transition function T which consists in simply counting how many times (s, a) was followed by s' normalized by the total number of times (s, a) was encountered. This method is fine when the structure of the environment is stable (which is always the case in the present simulations), and could be extended in cases where the simulations involve the introduction of obstacles or of other changes of maze configuration. Besides, updating the reward function R in the world model is here based on the latest feedback only, which works fine in the tasks involving a deterministic reward studied here. For the Q-value update, we perform one step of the value iteration algorithm (Sutton and Barto, 1998):

$$Q(s, a) \leftarrow R(s, a) + \gamma \sum_{s'} T(s, a, s') max_{k \in \mathcal{A}} Q(s', k) \tag{2}$$

It is worthy of note that this equation works in the general case where transition between states of the task are stochastic (probabilistic). In that case, for a given action a performed in a state s, the transition function T is a probability distribution over all possible states of the task. However, in the simulations presented in this paper, the world is deterministic (i.e. navigation in a maze), so that a given state-action couple always leads to the same resulting state s' with probability 1, while the probability is 0 for all other states. Nevertheless, we keep this formulation for generality purposes.s

After this step, we measure how much the absolute value of the Q-value has been changed according to:

$$\delta = |Q_{t+1}(s, a) - Q_t(s, a)| \tag{3}$$

If δ is large (which means the outcome of action a in state s has been surprising), the agent has good reasons to think that updating the Q-values of neighboring states will also result in important value changes. Thus, following Moore and Atkeson (1993); Peng and Williams (1993), if δ is higher than a certain threshold, all possible predecessors of the current state are added to a priority queue *PQueue* (that will be used during the *prioritized sweeping*), with a priority equal to δ, attenuated by γ (the value cost of one step) and by the probability to effectively reach state s from the predecessor under consideration.

Following classical machine learning work (Lin, 1992; Peng and Williams, 1993), after each action performed by the agent in the environment, an offline reactivation phase starts, which in our algorithm corresponds to a series of iterations of a *bidirectional search* inference process with the world model (Fig. 2). This offline reactivation phase thus corresponds to hippocampal awake replays only, and is described in Algo. 2. Updates of Q-values can in principle be unordered, though it is more efficient to start from rewarding (s, a) combinations, and progressively propagate their value to their predecessors first. This leads to the more general idea of *prioritized sweeping*: update first the observations whose value has changed recently and has been associated with a high level of surprise, and propagate that surprise to their predecessors. This is thus a dynamic programming equivalent of backward search. The level of surprise δ is computed with Eqn. 3. Because the predecessors of a given state s can be difficult to determine in a stochastic world, Moore and Atkeson (1993) propose to consider as predecessors all the states s' which have, at least once in the history of the system, been followed by a one-step transition $s' \rightarrow s$ through an action a'. Thus the priority associated to a predecessor s' shall be weighted by the probability of transition $T(s', a', s)$ between s and s'. As we are propagating the surprise observed in state s to its predecessors, the priority is attenuated by the discount factor γ. To sum up, for each predecessor s', we compute the Q-value variation δ' that would result from updating the $Q(s', a')$ based on the recently changed Q-value of s. As a result, the final priority considered by our model for each predecessor s' is: $\gamma T(s', a', s)\delta'$.

We then use the opposite optimization: rather than trying to update values for all observations, most of which are not going to be visited, concentrate on the current situation by updating the values starting from the current observation (*i.e.,* the current estimated position of the animal within the environment) and considering its successors (a strategy called *trajectory sampling* (Sutton and Barto, 1998)). This process is similar to the computation of the *need* to update each state's value depending on the probability to visit it in the near future, which Mattar and Daw (2018) compute through the use of the *Successor Representation (SR)* (Stachenfeld et al., 2017). While the SR provides less flexibility than pure MB techniques, it can capture additional properties of the hippocampal system and has in common to the *trajectory sampling* method used here to produce a sort of mental traveling through forward sweeps from the current position until possible future states. Nevertheless, one key question with these techniques is when to stop the current sequence of actions that is sampled during this mental traveling. Taking inspiration from the *bidirectional planning* classical technique (Pohl, 1971; Levy, 1996), we sample actions forward until reaching a state that has previously been reached by the backward *prioritized sweeping* process: we stop model-based mental traveling when forward and backward search process have reached a connection state (Fig. 2).

To be more precise, each inference iteration begins with a *prioritized sweeping* phase: the element with the highest priority in $PQueue$ is processed (using the same update rule defined in Eqn. 2), and its predecessors are added to the queue with the same rule as in the online phase, this processing of the queue stops either when a given budget of $nbPSmax$ cycles is consumed, or when the priority of the first element in the queue is below ν. It is followed by a *trajectory sampling* phase: a sequence of actions is simulated using the world model, starting from the current position, using the same update rule, until consumption of the budget (*nbTSmax* cycles) or until the arrival in a state present in *PQueue* (i.e. whose value has already been updated by *prioritized sweeping*). The whole process is repeated until the sum of the modifications of the Q values processed falls under a given threshold ϵ (a budget constraint could be added here, as proposed in Cazé et al. (2018), but in this work we let the system reach this criterion).

Algorithm 1 Online algorithm

```
INPUT: $s_0$, $Q$ // initial state, Q-values
OUTPUT: $Q$ // updated Q-values
$nbActions \leftarrow 0$
$s_t \leftarrow s_0$
PQueue ← {} // PQueue: empty priority queue
T ← 0 // T: transition statistics storage
R ← 0 // R: reward function
repeat
  $a_t \leftarrow draw(softmax_\beta(Q(s_t))$
  $nbActions \leftarrow nbActions + 1$
  Take action $a_t$ receive $s_{t+1}, r_{t+1}$
  $T(s_t, a_t, s_{t+1}) \leftarrow T(s_t, a_t, s_{t+1}) + 1$
  $R(s_t, a_t) \leftarrow r_{t+1}$
  $Q_{old} \leftarrow Q(s_t, a_{t+1})$
  // 1-step MB update:
  $Q(s_t, a_{t+1}) \leftarrow R(s_t, a_{t+1}) + \gamma \sum_{s'} \frac{T(s_t, a_{t+1}, s')}{\sum_i T(s_t, a_{t+1}, i)} max_{k \in \mathcal{A}} Q(s', k)$
  $\delta = |Q(s_{t-1}, a_{t+1}) - Q_{old}|$
  for each $(s, a)$ so that $T(s, a, s_t) \neq 0$ do
    $p \leftarrow \delta \times \gamma \times \frac{T(s,a,s_t)}{\sum_i T(s,a,i)}$
    if $p > \nu$ then
      if $(s, a) \notin$ PQueue then
        Put $(s, a)$ in PQueue with $priority$ $p$
      else
        Update $priority$ of $(s, a)$ in PQueue with $p$
      end if
    end if
  end for
  $s_t \leftarrow s_{t+1}$
  $Q \leftarrow$ bidirectionalInference($Q$,PQueue,$s_t$) // offline phase
until nbActions = nbActionsMax
```

We can anticipate from the defined algorithm that three different types of state offline activations will be produced, depending on the situation:

- If the priority queue contains surprising-enough states (which happens most of the time at reward location when an unexpected reward is obtained, or an expected reward is missed), there is a (short) phase of prioritized sweeping, which results in backward reactivations, followed by a (short) phase of trajectory sampling. This is repeated until convergence of Q-values, before the agent can make a new action and go to a new state.
- If the priority queue does not contain surprising states, there is a minimal duration (i.e., the budgeted $nbTSmax = 10$ iterations) of trajectory sampling performed just to see whether this results in big changes in Q-values, so that

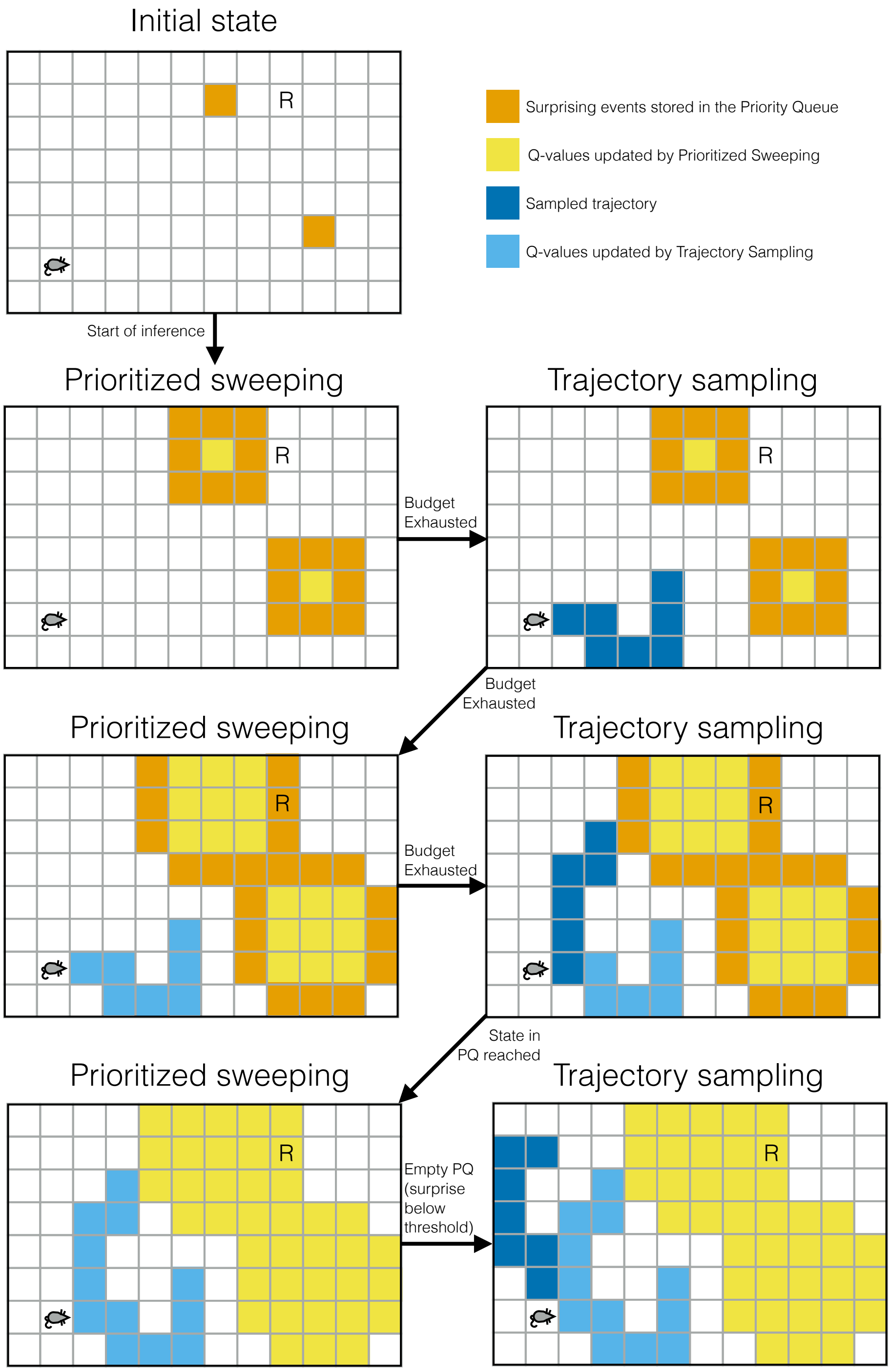

Fig. 2 Schematic representation of the operation of the bidirectional inference algorithm: prioritized sweeping and trajectory sampling phases alternate until an inference stop criterion is reached (Q-value convergence, exhaustion of a general budget, etc.). The mouse represents the agent's position, and R the reward location. We illustrate here the case where the total reactivation budget is not unlimited, hence the initial state can contain surprising events inhertited from the previous set of reactivations, rather than corresponding to the agent's last move only. Each prioritized sweeping phase stops either when budget is exhausted or when no element in the priority queue has a priority above a fixed threshold. Each trajectory sampling phase stops either when budget is exhausted or when the trajectory reaches a state stored in the priority queue. Figure by Girard, B., available at https://doi.org/10.6084/m9.figshare.8306132

Algorithm 2 Bidirectional inference algorithm (offline phase)

```
INPUT: $Q$, PQueue, $s_t$ // Q-values, priority queue, current state in the real world
OUTPUT: $Q$ // updated Q-values
$nbLoops \leftarrow 0$
repeat
  $Sum\delta \leftarrow 0$
  $nbLoops \leftarrow nbLoops + 1$
  // budgeted prioritized sweeping:
  $nbPS \leftarrow 0$
  while $priority(PQueue[0]) > \nu$ and $nbPS < nbPSmax$ do
    $nbPS \leftarrow nbPS + 1$
    $(s, a) \leftarrow$ PQueue[0] // item with the highest priority in the Queue
    $Q_{old} \leftarrow Q(s, a)$
    $Q(s, a) \leftarrow R(s, a) + \gamma \sum_{s'} \frac{T(s,a,s')}{\sum_i T(s,a,i)} max_{k \in \mathcal{A}} Q(s, k)$ // 1-step MB update
    $\delta \leftarrow |Q(s, a) - Q_{old}|$
    $Sum\delta \leftarrow Sum\delta + \delta$
    Update $priority$ of $(s, a)$ in PQueue with $\delta$
    for each $(s', a')$ so that $T(s', a', s) \neq 0$ do
      $p \leftarrow \delta \times \gamma \times \frac{T(s,a,s_t)}{\sum_i T(s,a,i)}$
      if $p > \nu$ then
        if $(s', a') \notin PQueue$ then
          Put $(s', a')$ in PQueue with $priority$ $p$
        else
          Update $priority$ of $(s', a')$ in PQueue with $p$
        end if
      end if
    end for
  end while
  // budgeted trajectory sampling:
  $nbTS \leftarrow 0$
  $s \leftarrow s_t$ // start from the current location in the real world
  // loop until a state of PQueue is reached or until budget expended
  while $s \notin PQueue$ and $nbTS < nbTSMax$ do
    $nbTS \leftarrow nbTS + 1$
    $a \leftarrow draw(softmax_{\beta_R}(Q(s)))$
    $s' \leftarrow draw(probabilityProportionateSelection(T(s, a)))$
    $Q_{old} \leftarrow Q(s, a)$
    $Q(s, a) \leftarrow R(s, a) + \gamma \sum_{s'} \frac{T(s,a,s')}{\sum_i T(s,a,i)} max_{k \in \mathcal{A}} Q(s, k)$ // 1-step MB update
    $Sum\delta \leftarrow Sum\delta + |Q(s, a) - Q_{old}|$
    $s \leftarrow s'$
  end while
until $Sum\delta < \epsilon$ or $nbLoops > nbLoopsMax$ // no significant Q-value update in the last
loop or global budget exhausted
```

the algorithm determines whether a long trajectory sampling is required until the Q-values converge. If Q-values are modified more than a certain threshold, this means that Q-values have not yet converged and a series of (short) phases of trajectory sampling are performed until Q-values converge.

- Finally, in the case where the priority queue does not contain surprising states and the short trajectory sampling performed does not change Q-values more than the ϵ threshold, then the algorithm considers that Q-values have already converged and that inference can be stopped. In other words, this corresponds to most cases after learning, where no surprising event occur, the performance

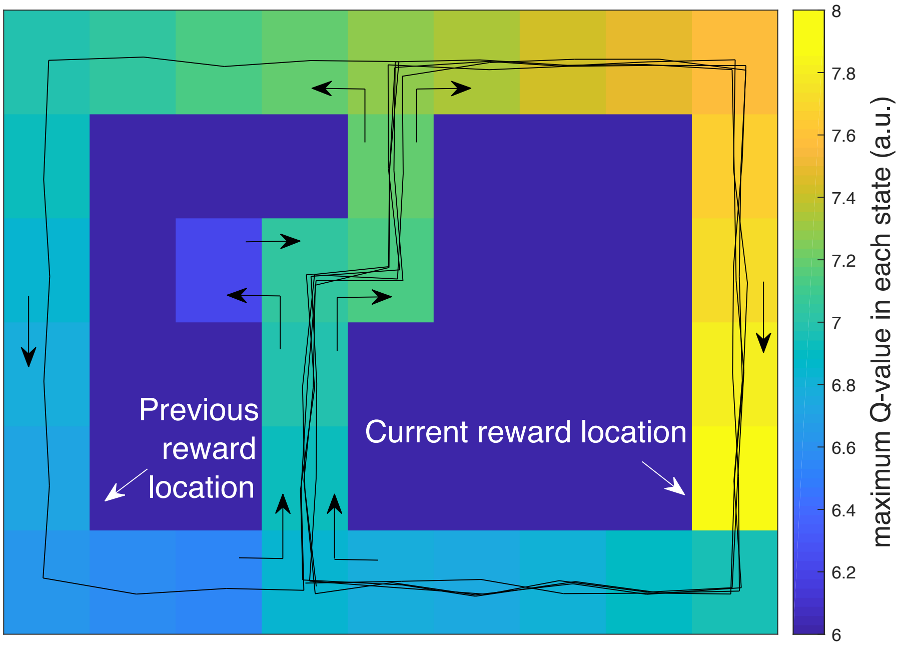

Fig. 3 Example of Q-values learned by the model during simulations of a simplified version of the multiple T-maze task of Gupta et al. (2010). The color code illustrates the gradient of Q-values learned at the end of the experiment, when the reward is located in the right arm. The continuous black line represents the trajectories (noised for illustrative purpose) performed by the agent during the last six trials of the experiment. Black arrows indicate the directions that are allowed to the agent during task performance. The two states in the central arm where the agent has the choice to go either left or right are called the *decision points* throughout the manuscript. The agent is not allowed to immediately get back to the state visited at the previous timestep, except for the state located at the left of the central decision-point, where the only possibility is to move towards the east. The small Q-values in dark blue correspond to areas of the maze unreachable by the agent, due to the corridors' walls. They thus remain unchanged during the experiment.

is high, and the model thus does not consider that it should stop for a long time in order to start a long offline reactivation phase before acting.

3 Results

The model-based bidirectional search algorithm (hereafter noted *MB-RL bidirectional search*) was simulated for a series of numerical experiments in the multiple T-maze task of Gupta et al. (2010) (Fig. 3)[1]. This task involves a central arm where at each trial the agent has to choose to either turn left or right, in hope of getting a reward at the end of the chosen external arm (left or right). Fig. 3 shows an example of 5 simulated trials of the MB-RL bidirectional search algorithm just after a change in the task rule occurred: the reward had been located in the left arm for at least 50 trials (the change occurred after the model had performed 2000

[1] The code is available at https://github.com/MehdiKhamassi/RLwithReplay

Table 1 Parameter values of the model

Parameter	Value	Meaning
$nbActionsMax$	5000	maximal duration of the simulations
β	20	online exploration/exploitation trade-off
	[0,200]	(range used in parameter exploration)
β_R	10	offline exploration/exploitation trade-off
	[0,200]	(range used in parameter exploration)
γ	0.99	discount factor
ν	0.001	threshold defining surprising Q-value updates
$nbPSmax$	10	budget for one *prioritized sweeping* cycle
$nbTSmax$	10	budget for one *trajectory sampling* cycle
ϵ	0.1	threshold on the accumulated Q-value updates, stopping the inference

actions; see all task and model parameters in Table 1) and then was located on the right arm until the end of the simulated experiment (the experiment ended once the model had performed 5000 actions). In this example, the model performed a single perseverative error (illustrated by the single trajectory along the left arm) followed by systematic visits to the right trials (illustrated by the trajectories of the following four trials). This example illustrates how such a model-based learning method with a prioritized offline inference process can quickly adapt to changes in the task rule in simple tasks with a relatively small number of discrete states (see Cazé et al. (2018) for a few comparisons of performance of simpler MB-RL prioritized inference methods with an MB-RL algorithm with unordered inference, or with model-free and Dyna variants of such prioritized methods). Fig. 3 finally illustrates the maximum Q-values (i.e., state-action value function) learned by the algorithm in each state of the maze. The color code indicates that the algorithm successfully learned to assign the largest value to the state where reward can be obtained, and to learn a gradient over states: the further away from reward, the lower the value in a given state.

We then compared the average performance of the MB-RL bidirectional search algorithm over 10 simulation experiments with two standard algorithms (Fig. 4): MB-RL prioritized sweeping and MB-RL trajectory sampling (taken from Cazé et al. (2018)), which are the two main components for offline inference that have been assembled in our model (see the full model description in Section 2), which is similar in spirit but heuristic-based and thus computationnally less costly than the normative proposal of Mattar and Daw (2018). From Fig. 4 we can see that the three algorithms perform similarly in terms of reward rate (ANOVA test, $df = 2$, $F = 0.26$, $p = 0.77$). They all learn to reach a reward rate of about 1 (optimal performance in this task) in less than 10 trials, and then show a drop in reward rate after the change in reward location occurring at trial 50, and then recover a nearly optimal performance again in less than 10 trials. The striking difference in the performance of the three algorithms relies in the amount of time (i.e., Napierian logarithm of the number of iterations) they took on average per trial to perform offline inference. Each of these moments of offline inference correspond to moments where the simulated agent decided to stop in the current state (no matter where it was located in the environment, we'll come back to this issue later) in order to use its internal world model to update Q-values until these Q-values converge and stabilize. These moments are thus supposed to mimic moments where

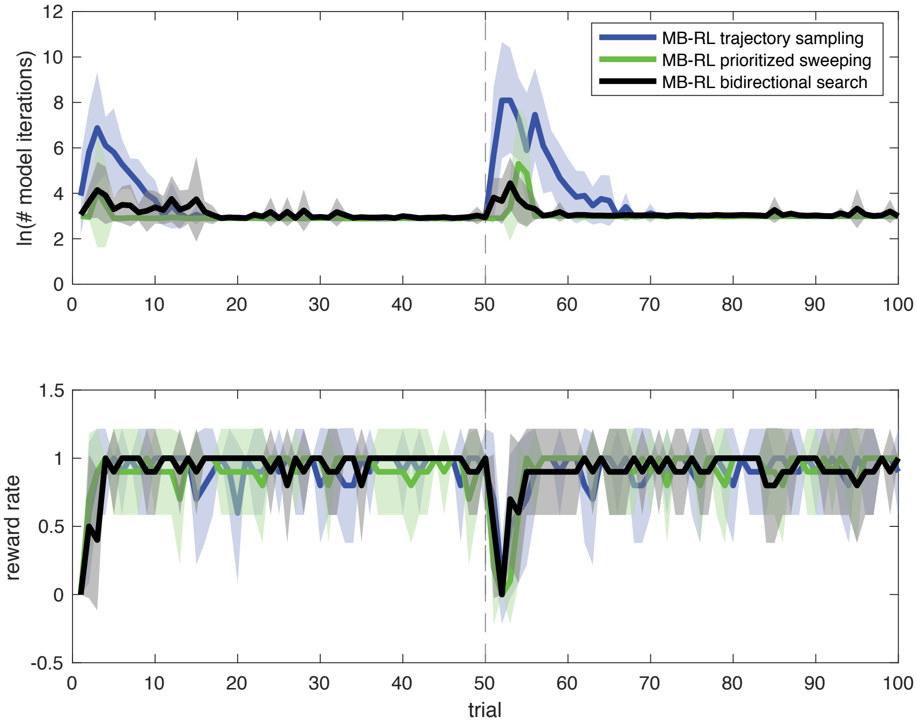

Fig. 4 Performance of the model compared to a pure prioritized sweeping algorithm and a pure trajectory sampling algorithm. Top: Napierian (natural) logarithm of number of model iterations during offline inference phases. Bottom: Performance of the models in terms of reward rate. All algorithms reach similar reward rate and adapt to the change in reward location (from left to right arm of the maze) within less than 10 trials. The pure trajectory sampling method performs the largest number of offline inference iterations after a change in reward location, while the MB-RL bidirectional search performs the smallest number of such iterations. Note that here the parameters of the MB-RL bidirectional search model are chosen to give a good trade-off between performance and offline inference duration, as analyzed later on in the paper.

an animal pauses and where awake hippocampal replays have been extensively observed during neurophysiological experiments in rodents (Foster and Wilson, 2006; Johnson and Redish, 2007; Karlsson and Frank, 2009; Gupta et al., 2010; Diba and Buzsáki, 2007; Pfeiffer and Foster, 2013; Ólafsdóttir et al., 2015; Redish, 2016; Ólafsdóttir et al., 2018).

In order to fairly compare the three algorithms, from the top curves plotted in Fig. 4 we have removed from each algorithm the 10 initial inference iterations which are systematically performed in each state to decide whether further offline inference to stabilize Q-values is required or not. Strikingly, while MB-RL trajectory sampling usually needs a large number of offline inference iterations to converge (here on average 833 iterations), MB-RL prioritized sweeping and MB-RL bidirectional search are much quicker (on average 60 and 36 respectively; Kruskal-Wallis test, $\chi^2 = 24.13$, $df = 2$, $p = 5.75e-6$). Importantly, while the difference in average replay duration is small between MB-RL prioritized sweeping and MB-RL bidirectional search, it is nevertheless significant (Wilcoxon Mann-Whitney test,

$df = 1$, $ranksum = 148$, $zval = 3.21$, $p = 0.0013$). The fact that MB-RL bidirectional search and MB-RL prioritized sweeping perform a much smaller number of offline iterations than trajectory sampling is probably explained by the constrained nature of the maze. Trajectory sampling is expected to be useful so as to avoid updating the value of states one iss not going to visit to solve the task (Sutton and Barto, 1998). In a navigation context, this optimization should be much more important in open environments, rather than in the restrained mazes we used here.

A striking property common to the three algorithms is that they spend a lot of time doing offline inference at the beginning of the task and after a task rule change, but barely perform any offline inference (in other words barely any 'replay' or 'reactivation') the rest of the time when the reward rate is high and stable. This is similar to the alternation between periods of vicarious trial and error and automatized behavior observed in animals (Redish, 2016). This property, highlighted in MB-RL trajectory sampling and MB-RL prioritzed sweeping in Cazé et al. (2018), is here replicated and also obtained with the new MB-RL bidirectional search algorithm. It is important to notice that this is an emerging property of the algorithms. Nothing is built within the models to tell them that it is at these precise moments that offline inference should be invoked. It is because the present approach consists in performing offline inference only when Q-values are not stable, until they converge, that the algorithms start performing long offline inference phases each time there is something new to learn which affects Q-values and make them change from their previous values.

Interestingly, there is a trade-off in the parameters of the MB-RL bidirectional search to produce either the highest possible reward rate or the lowest possible number of offline inference iterations in this task. To study this, we performed 5 simulation experiments of the model for each combination of two key parameters in the model: β which corresponds to the random exploration level during actual task performance (i.e., decision of which action the agent will then actually perform in the maze, according to Eqn. 1); β_R which corresponds to the same decision-making process for virtual actions mentally performed by the trajectory sampling process during offline inference. The results are shown in Fig. 5. If β is low, the agent is very exploratory during actual task performance, and thus does not maximize reward rate. Thus there is a minimal value of β required to maximize reward rate (typically 10 in this task). For β_R, the influence on model performance is different. For high β_R, offline inference phases will consistently exploit and thus focus the action sequences that seem best at a given moment. This will lead to minimizing the duration of these offline reactivation events. Nevertheless, if $\beta_R > \beta$, the model will not be able to maximize reward rate. In contrast, if $\beta_R < \beta$, the model is more exploratory during offline inference, which takes more reactivation time, but also permits to mentally well explore all alternatives and thus to maximize reward rate during task performance. In Cazé et al. (2018) we have argued that this property, in the case of MB-RL trajectory sampling, could be an explanation why hippocampal awake replay events in rodents do not systematically focus only on the trajectory that leads to reward (Johnson and Redish, 2007), and possibly even why hippocampal reactivations during sleep are more noisy and less accurate than during task performance (Roumis and Frank, 2015): possibly because the time pressure (i.e., the *horizon* in machine learning terms) is smaller during sleep than during task performance, and thus it is optimal to perform a higher degree

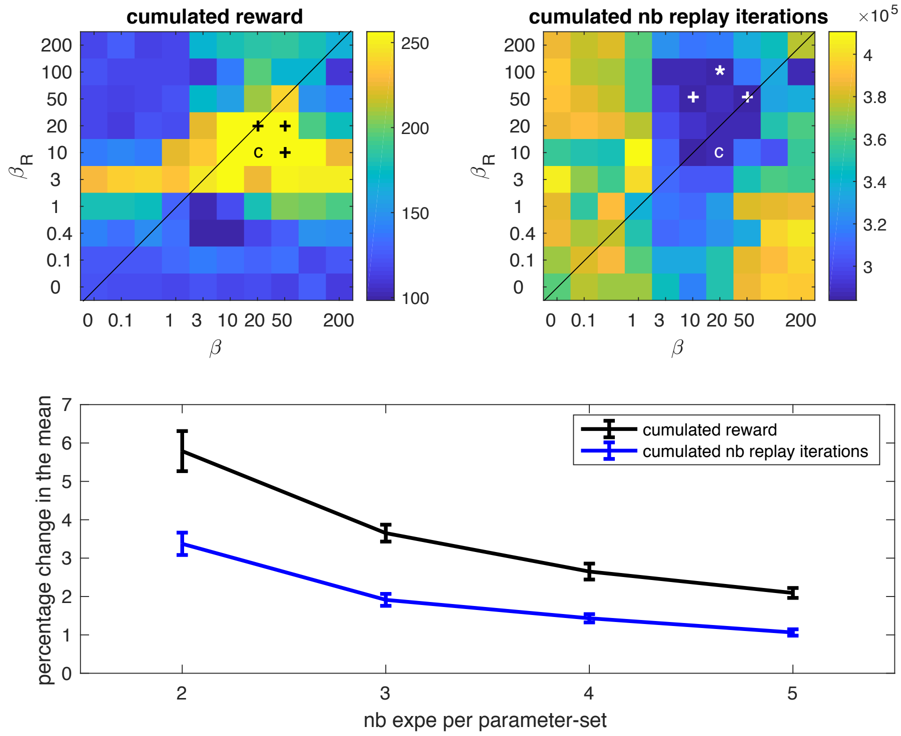

Fig. 5 Optimization of the performance (cumulated number of reward) or of computational cost (cumulated number of reactivation iterations) during a fixed experiment duration of 5000 iterations, as a function of two model parameters: β used for random exploration level during online decision-making, β_R used for random exploration level during offline decision-making during trajectory sampling. For each pair of (β, β_R) values, the figure shows the average performance and offline inference duration over 5 full experiments of 5000 iterations each. Marker c indicates the best parameter-set according to the Chebyshev aggregation function (Viejo et al., 2015), which search for the solution which minimizes the distance with the point corresponding to the highest possible performance and lowest possible offline inference duration. Marker $*$ indicates the best parameter-set for a given measure (either performance or offline inference iterations). Note that for the top-left plot, $*$ is not indicated because it is confounded with c. Markers $+$ indicate parameter-sets that are less than 1% away from the optimum. The bottom figure shows how the number of experiments changed this mean, which illustrates that 5 experiments are enough to have a good estimate of the mean.

of mental exploration. Here we expand this hypothesis by showing that it is also true for the MB-RL bidirectional search model.

In order to analyze the sensitivity of the MB-RL bidirectional search model to parameters, we performed a series of additional simulations of the multiple T-maze task where we explored different parameter-sets on a grid. For each parameter-set, we performed ten simulations of the experiment and report the average performance and average number of reactivation iterations (Fig. 6). We found that the threshold ϵ for the convergence of Q-values during offline reactivations and the global budget $nbLoopsMax$ imposed on the number of iterations per offline reactivation phases only marginally affect the performance of the model. In contrast,

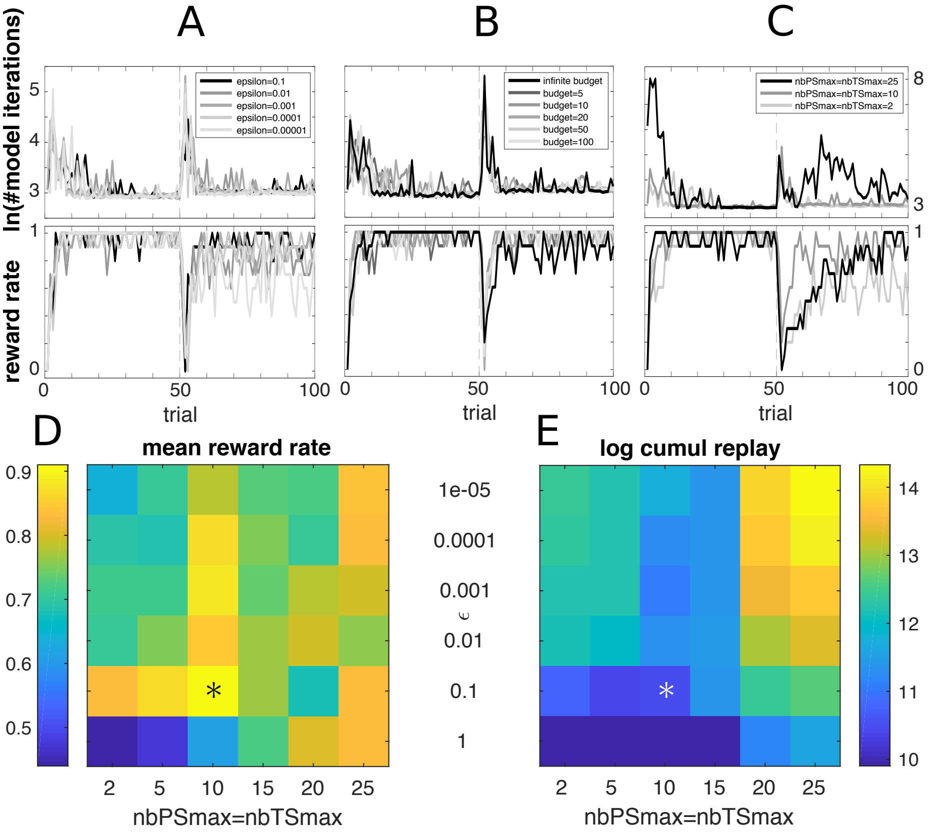

Fig. 6 Sensitivity to model parameters. A) Decreasing the threshold ϵ for the convergence of Q-values increases the number of model iterations during offline reactivations while slightly decreasing the reward rate. B) Varying the global budget imposed on the number of iterations per offline reactivation phases only marginally affects performance. In contrast, releasing this constraint (infinite budget) makes the performance more stable during initial learning but slower to stabilize after the task rule change (after trial 50). C) Alternating between prioritized sweeping (PS) and trajectory sampling (TS) every 10 trials during offline reactivation constitutes an optimum in terms of performance. D-E) The highest reward rate (D) combined with one of the lowest number of reactivation iterations (E) was obtained for $\epsilon = 0.1$ and $nbPSmax = nbTSmax = 10$ (the optimum is indicated by a $*$ on the figure).

we found an optimum frequency of alternation between prioritized sweeping (PS) and trajectory sampling (TS) during offline reactivations. From this analysis, the optimal parameter-set provides on average 91% of the optimal reward rate, and an average of 3.6e-4 offline reactivation iterations per experiment (which is 46 times smaller than what can be obtained with the worst parameter-set tested here).

After this analysis, it is interesting to look at two important properties of the model: where in the maze does it predict that hippocampal reactivations should occur; and in what order (forward, reversed, unordered) does it predict that consecutive states within the environment should be reactivated during offline inference. The first property is illustrated in Fig. 7 which shows that during early learning (i.e., the first 50 trials of the experiment), the simulated agent decides to stop mainly at the reward location (bottom of left arm) and in the central arm in order to start performing offline inference. The color code in the figure also shows that the simulated agent spends a little bit of time performing offline inference in each state of the left arm (where the model goes most of the time, in accordance with the reward rate curves of Fig. 4). This is again due to the simplistic approach adopted here consisting in performing an initial series of 10 offline inference iterations in order to see whether this resulted in changes in the Q-values and thus to decide whether a long offline reactivation phase is needed or not. Nevertheless, this reactivation duration is lower in all states other than reward location and central arm. It is important to note that due to the normative perspective adopted by Mattar and Daw (2018), in their work they have forced the model to perform offline inference only at reward site or at starting point, while here this is another emerging property of the model. The large amount of time spent during offline inference at reward location is due to the large level of surprise associated to the first encountering of reward there. This is produced by the prioritized sweeping component of the model, whose probability of triggering an offline inference phase is proportional to the absolute value of reward prediction errors. Similarly, during the 'post rule change' phase after a change in reward location (from left to right), the model increases the number of offline inference iterations at the previous reward location (left arm) because there are negative surprise signals there, and at the new reward location (although mildly in our simulations). In Cazé et al. (2018) we showed that the classical MB-RL prioritized sweeping predicted offline inference only at reward locations, but not in the central arm, in contrast to MB-RL trajectory sampling. Here we show that the MB-RL bidirectional search model combines these properties and performs a larger number of offline inferences at both reward location and central arm. Furthermore, within the central arm, the model predicts that offline reactivation events should occur mostly at decision points. This is because performing the initial 10 iterations of offline inference on a set of states that include one side of the decision point can lead to large changes in the decision point's Q-values if these values were mostly affected by the values of the other side (e.g., if the agent most of time chooses the other arm of the maze). Thus, similarly to Mattar and Daw (2018), the model can both account for hippocampal reactivations at the reward locations (Gupta et al., 2010; Papale et al., 2016) and at decision-points (Diba and Buzsáki, 2007; Pfeiffer and Foster, 2013).

The second important property of the model is illustrated in Fig. 8. The figure shows the analysis of the simulation data during all offline inference phases regrouped together. The analysis focuses on the sequences of states sampled by the model during reactivation phases, i.e. during moments where the agent is immobile in the maze. It consists in counting the proportion of sequences of 3 (or 5) consecutively sampled states that can be labeled as *forward* because they correspond to the same direction the agent followed during actual task performance, or labeled as *backward* because they occur in the reverse order, or labeled as *unordered* because they correspond to sequences of states that appear unordered or random, even if they have been generated by the same MB-RL bidirectional search algorithm. Finally, within forward and backward cases, we separately count state sequences that correspond to trajectories that have never been performed by the agent during actual performance, hence called *imaginary* sequences (as illustrated in Fig. 1), following the terminology proposed by Gupta et al. (2010). For instance,

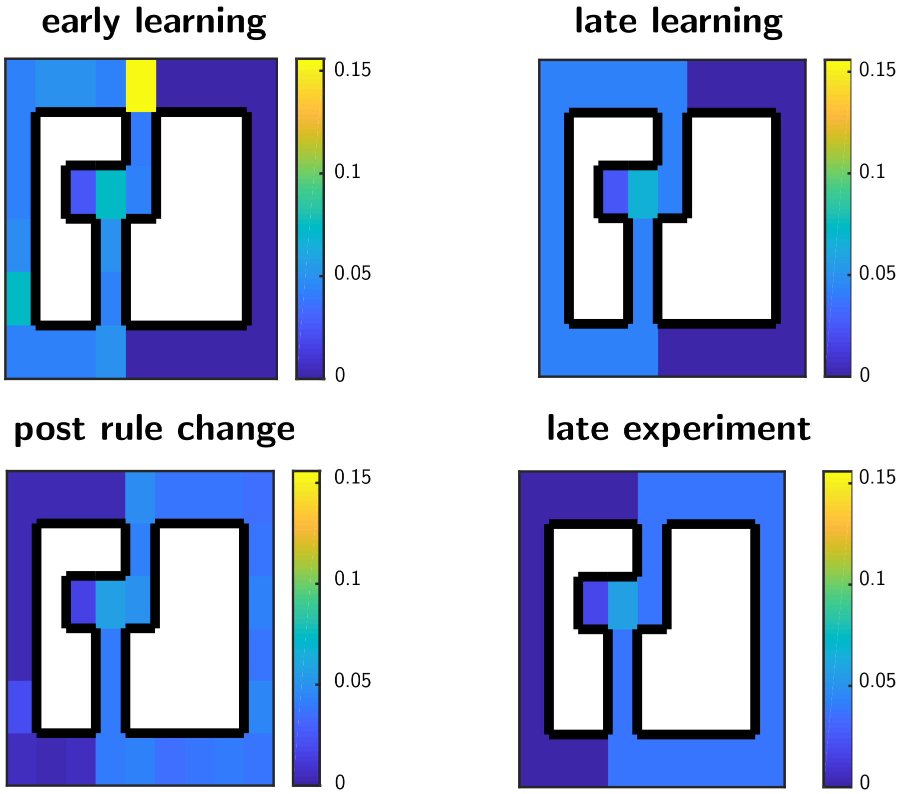

Fig. 7 The experiment was split into four phases of equal duration in terms of number of trials: 50 first trials of the experiment (early learning), 50 trials before change in reward location (late learning), 50 trials after change in reward location (post rule change), 50 last trials of the experiment (late experiment). For each phase of the experiment, the color scale indicates the normalized (thus relative) proportion of time (number of iterations) spent by the agent doing offline reactivations in each state of the maze. The figure shows that the model initially spent most of its time doing offline reactivations at the decision points in the central arm, and at the initial reward location (bottom of left agent). Then reactivations were much reduced during late learning. After a rule change, the agent did some reactivations at previous reward location because of the surprise due to reward omission, some reactivations in the central arm (including decision-point), and a few reactivations around new reward location (bottom of right arm). Finally, offline reactivations were reduced during late experiment.

moving forward from the bottom-left arm and continuing onto the bottom-right arm is something that was not allowed to the agent during the task because the agent is required to get back to the central arm to initiate a new trial (see black arrows in Fig. 3 for trajectories allowed during task performance). This is also true for the same trajectory in reverse order, when the agent is coming back from the bottom-right arm. Similarly, the agent is not allowed to go from the top-left arm onto the top-right arm. This is because during any trial of the task, once the agent has decided to go either left or right from the top decision-point, the animal cannot go back and has to go all through the arm until the bottom of the maze in order to complete the trial. Thus any such sequence of states played by the

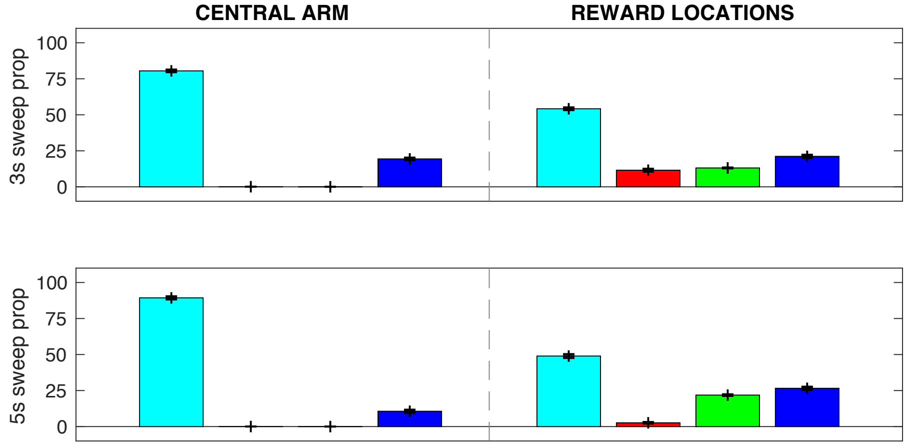

Fig. 8 Proportion of forward (cyan), backward (red), imaginary (green) and unordered (dark blue) state sequence orders simulated by the model during off-line reactivations. The figure shows that backward and imaginary reactivations occurred mostly at reward locations. More forward reactivations were generated by the model in the central arm than at reward locations. These are consistent with experimental results (Gupta et al., 2010).

model during offline inference corresponds to an *imaginary* trajectory of mental traveling.

The first important result illustrated by Fig. 8 is that the proportions of different orders of state sequences generated by the model are different between moments when the agent was at reward locations and moments when it was in the central arm. This was confirmed by a Chi-square proportion test applied to the mean proportion of each order per location ($\chi^2 = 30.1$, $df = 3$, $p < 0.001$). When not only looking at mean proportions but instead taking into account the proportions generated by each of the 10 simulations experiments, the previous result was confirmed by a significant interaction ($F = 2.4e^3$, $p < 0.001$) obtained with a two-way analysis of variance (ANOVA) with order x location as factors. The ANOVA also showed no main effect of agent location during the offline inference phase ($F = 7.3e^{-28}$, $p = 1$), and a significant main effect of the sequence orders that were generated during offline inference ($F = 2.4e^4$, $p < 0.001$). Posthoc investigation revealed that the model prominently generates state sequences in forward order ($T > 46$, $p < 0.05$). This is mostly due to the *trajectory sampling* algorithm. Strikingly, the model generates a smaller proportion of forward state sequences at reward locations (54.2%) than in the central arm (80.5%; Kruskal-Wallis test, $\chi^2 = 14.3$, $df = 1$, $p < 0.001$). In contrast, backward state sequences (which mimic hippocampal reverse replay (Foster and Wilson, 2006)) are generated by the model nearly exclusively at reward locations (11.5%) but are nearly absent in the central arm (0.1%). This is because backward state sequences are due to the *prioritized sweeping* algorithm which is mainly triggered when large surprises are encountered by the agent, like an unexpected reward or omission of an expected reward. Once these surprises are detected, the agent stops for some time where it is (i.e., at reward locations) and starts doing a long phase of offline inference. In our simulations, because we allow an infinite budget of offline inference, the model takes as much time as needed to make the Q-values of the whole maze incorporate the new surprising outcome and then converge (i.e., stabilize). During this phase, *prioritized sweeping* tends to propagate these surprising outcomes to neighboring states (Fig. 2). Because these neighbor states are here constrained by a corridor (Fig. 3), this tends to generate clear sequences, including sequences in backward order compared to the trajectory that the agent has just performed. The proportion of backward sequences generated by the present MB-RL bidirectional search model at reward locations (11.5%) is smaller than those generated by the MB-RL prioritized sweeping algorithm in the same experiment (25%; Cazé et al. (2018)) and by a neural network implementation of prioritized sweeping (Aubin et al., 2018). Experimental data typically show varying proportions between individuals, from 6% for some rats up to 72% for others, with a mean of 27% (Foster and Wilson, 2006). Future improvements of our model like removing the systematic short invocation of trajectory sampling (which generates forward sequences) at the beginning of each offline inference phase, and replacing this by a more explicit measure of uncertainty (Pezzulo et al., 2014) to determine whether more offline inferences are needed, could help making these percentages match experimental data.

Importantly, unlike the Mattar and Daw (2018) model, the MB-RL bidirectional search model also generates a substantial proportion (13.1%) of *imaginary* state sequences while the agent was located at reward sites (Fig. 8), which correspond to never-experienced trajectories during task performance. This is due to the surprising outcome triggering long offline inference phases, where prioritized sweeping alternates with trajectory sampling. The latter mentally simulates trajectories with a higher exploration rate than during task performance, as established through the model property analysis (Fig. 5). As a consequence, the model is unlikely to only generate state sequences corresponding to the familiar trajectories performed by the agent during the task. This is a key property of the bidirectional search process which requires the agent to generate a variety of forward trajectories in the hope of making at least one of them connect with the backward trajectories generated from reward location with prioritized sweeping (Fig. 2). This could thus provide a computationally-grounded reason why *imaginary* state sequences can sometimes be observed experimentally when an animal is waiting at a reward site (Gupta et al., 2010).

Another key difference between our model and the one of Mattar and Daw (2018) leads to different experimental predictions. In our model, the sudden absence of an unexpected reward (omission) generates a surprise signal (absolute prediction error) which is sufficient to trigger offline inference. This is a classical property of the prioritized sweeping algorithm which starts anytime absolute prediction errors above a certain threshold are detected (Moore and Atkeson, 1993; Peng and Williams, 1993). As a consequence, our model predicts that in case of extinction, a short (<5 trials) but significant increase in hippocampal reactivation should be observed (Fig. 9). In contrast, the Mattar and Daw (2018) predicts an asymmetric effect of positive and negative prediction errors due to their differential effects on behavior. In particular, their theory specifies that propagating negative prediction errors is unhelpful if no better action is available. Thus in the extinction experiment considered here, which involves negative surprises without alternative best options, their model predicts no increase in hippocampal reactivation. While they showed an example of experimental data showing asymmetric modulation by

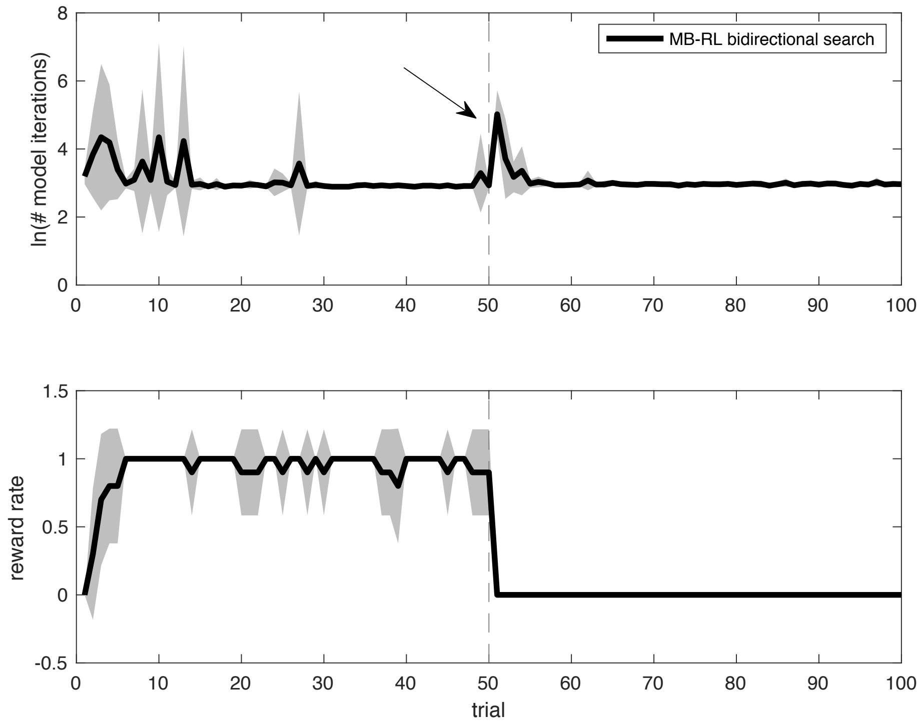

Fig. 9 Experimental prediction in the case of an extinction experiment. After 50 trials of learning where the reward was systematically located on the left arm, reward is no longer of the maze during another 50 trials of simulation. The figure illustrates that a negative surprise such as those occurring during reward omission shall lead to a small transient increase in offline inference iterations. This prediction is in contrast with the normative proposal of Mattar and Daw (2018) which predict that no offline reactivation should occur in the case where no best alternative option exist.

reward, we think this is very unlikely in the general case because it would require rats to be omniscient and systematically knowing whether there exist alternative outcomes. This is a general property of their model: because it is a normative model, which is very important to establish why offline reactivations are useful at all, it implies “the [...] unrealistic [...] calculation of gain [... which] requires that the agent knows the effect of [offline reactivation] on its policy prior to deciding whether to perform it.” (Mattar and Daw, 2018). In the general case, we predict that rats encountering a salient negative surprise in the case of an omission should most of the time stop to mentally process this surprise, which according to our model should be accompanied by offline hippocampal reactivations, as illustrated in Fig. 9.

4 Discussion

In this paper, we have presented a new MB-RL bidirectional search model of hippocampal awake offline reactivations during rodent reward-based navigation.

Similar to Cazé et al. (2018) and Mattar and Daw (2018), the model's central assumption is that the role of offline reactivation processes is to use the world model internally learned by the agent to update and stabilize state-action values (Q-values) and thus subsequently make better decisions during task performance. The novelty here consists of allowing the model to start an offline reactivation phase at any moment in any state of the environment, as long as there is either a surprise signal (absolute reward prediction error above a certain threshold) or Q-value instability detection signal (which implicitly signals choice uncertainty). In either case, the model's offline reactivation phase is organized like a bidirectional search algorithm: alternating between a few iterations of prioritized sweeping that propagate surprise signals from surprising states (i.e., mostly reward sites in a maze) to predecessor states, and a few iterations of trajectory sampling which starts from the agent's current position in the environment and travels forward in the aim of connecting with a state reached by prioritized sweeping. The model considers that there is an infinite offline reactivation budget in the sense that any started offline reactivation phase should last as long as needed to stabilize Q-values (convergence criterion). Nevertheless, we also showed that the same model can work with a finite budget for each reactivation duration, which in terms of neural implementation could represent some modulation of deliberation time as a function of the urge to act in the environment (Cisek et al., 2009).

We presented a series of simulation experiments in a 54-states discretized version of the multiple T-maze task (Gupta et al., 2010). Simulation results showed that the model can adapt as fast as classical prioritized sweeping and trajectory algorithms in response to changes in reward locations, while saving computation cost during offline inference compared to these two methods. Importantly, a first emerging property of the model is to drastically reduce offline reactivation time after learning, due to the absence of surprise and Q-value instability signals, and sharply increase offline reactivation duration in response to task changes. This mimics *vicarious trial-and-error* behaviors (Redish, 2016), where animals increase decision time in response to task changes, as if they were hesitating between alternative options. Strikingly, these hesitations of the animals at decision-point have been experimentally showed to be accompanied by hippocampal reactivations (Johnson and Redish, 2007), where forward trajectories can be decoded from hippocampal activity before the animal initiates movement, consistent with the trajectory sampling part of the present MB-RL bidirectional search model. Consistently, a second emerging property of the model is that forward trajectories generated during offline reactivations were more prominent at the decision-points of the central arm than at reward locations. This is due to (1) higher Q-value instability at decision-points because of the existence of alternative options, and (2) a substantial proportion of offline reactivation time at reward locations dedicated by the model to generate reverse trajectories from rewarding states to preceding states. As a corollary, the model generates backward reactivations only at reward sites, consistent with experimental findings (Foster and Wilson, 2006; Diba and Buzsáki, 2007; Gupta et al., 2010). Notably, the model is also able to generate *imaginary* state sequences during offline reactivations, corresponding to trajectories that were not allowed to the agent during task performance (Gupta et al., 2010). This is due to the higher level of mental exploration during offline trajectory sampling than during actual decision-making during task performance, which was shown here to be appropriate in the model in terms of performance maximization in this task.

Finally, the MB-RL bidirectional search model predicts that omission of expected certain reward (including during extinction experiments) should generate a sufficient surprise level to trigger a few trials ($<$ 5 trials) of hesitations by the animal, accompanied by a transient increase in hippocampal reactivation. This prediction is in contrast with the model of Mattar and Daw (2018) which, due to its *gain* term, considers the propagation of a reward prediction error through offline reactivation "unhelpful if no better action is available, but advantageous if alternative actions become preferred."

In Cazé et al. (2018) we had performed a systematic simulation and analysis of a series of existing machine learning algorithms for offline reactivations in the same task. We found that different algorithms could generate different types of experimentally observed hippocampal reactivations, namely forward, backward or imaginary. Here we combined different algorithms within the bidirectional search framework, and adopted the same approach consisting in performing offline reactivations only when Q-values are not stable, until they converge. As a result, the model starts performing long offline reactivation phases each time there is something new to learn which affects Q-values and makes them change from their previous values. This property is thus due to the machine learning-based hypothesis that offline reactivation processes using a world model can be a useful mean to learn Q-values, just as direct experience with the world and learning in a model-free manner are. This property originates from the seminal Dyna architecture proposed by Richard Sutton (Sutton, 1990), and has been simultaneously applied to model hippocampal replay by Cazé et al. (2018) and Mattar and Daw (2018). In Cazé et al. (2018) we had compared model-based and Dyna versions of trajectory sampling and prioritized sweeping algorithms, and found that the model-free update rule used in Dyna made learning slower and offline reactivations longer than in model-based methods. As a consequence, the present model employs a model-based learning rule while keeping Dyna's principle to use offline inference as a means to update Q-values, and thus as a way to "propagate reward information over space and time " (Mattar and Daw, 2018).

Mattar and Daw (2018) recently proposed a normative perspective on hippocampal reactivations, considering that they constitute ways to update state-action values during offline inference that are advantageous when they permit to increase the cumulated sum of reward over time. In this sense, their work provides a computational explanation of why offline reactivations are useful at all. Nevertheless, the normative perspective adopted constrained their model to be omniscient about the world in order to decide whether or not to perform offline reactivations. According to the authors, the model thus requires "unrealistic" computations so that the agent knows precisely in which way offline reactivations will change its behavior before even deciding to trigger offline reactivations. Besides, their model predicts an asymmetric effect of positive and negative reward prediction errors due to their differential effects on behavior. Interestingly, while this may be optimal from a normative perspective, it is in apparent contradiction with the observation that human participants have the tendency to privilege confirmatory reward prediction errors over disconfirmatory ones (Palminteri et al., 2017). Another limitation of the Mattar and Daw (2018) model is that offline reactivations were forced to occur either after a feedback at the end of a trial, or at 'decision-point' before the beginning of a trial. Thus the model cannot explain why animals may sometimes stop elsewhere in the maze, and start performing hippocampal reactivations there. In contrast, the present model was allowed to perform offline reactivations in any state of the environment, and chose to perform them mostly at reward sites and decision-points as an emerging property of the characteristics discussed above. Finally, their model cannot generate *imaginary* trajectories during offline reactivations (Gupta et al., 2010), which is the case of the present model.

The present model constitutes a mechanistic and predictive proposal to contribute to a better understanding of animals' *complex spatial navigation* abilities. These can be defined as abilities that go beyond simple stimulus-response behavior, and rather require the integration of past information and future expectations to guide present decisions. One example is the ability of rodents to memorize action sequences from past experience, and replay them mentally. While we adopted in this paper an abstract level of modeling, the question of how a realistic hippocampal-like neural network can encode sequences and replay them, has been addressed in the literature, and is still an active research topic (Levy, 1996; Cutsuridis and Hasselmo, 2011; Saravanan et al., 2015; Jahnke et al., 2015). The reader is referred to (Bhalla, 2019; Rennó-Costa et al., 2019), for recent reviews. However, which cellular or molecular mechanisms could organize the replays so as to implement the strategies used in reinforcement learning, like prioritized sweeping or trajectory sampling, is still to be investigated. Another advantage of computational approaches like the one adopted here is to put an emphasis on the distinction between two types of system reactivations that may contribute to learning: one consists of replaying recent experience from episodic memory, which can be done in a model-free manner; the other consists of generating potential sequence through sampling of an internal model, thus generating model-based mental traveling. While the two may be difficult to disentangle experimentally, the present work highlights some key properties of model-based reactivations, such as the ability to generate *imaginary* sequences, which goes beyond model-free replay of past experience. From a broader perspective, the detailed offline reactivation method proposed here could constitute a refinement of the model-based component of architectures that combine model-based and model-free reinforcement learning (Dollé et al., 2010; Caluwaerts et al., 2012; Renaudo et al., 2014), and which can account for a wider range of animal complex spatial navigation behaviors (Dollé et al., 2008, 2018).

The functions of the present model can also be discussed through the prism of hippocampus-prefrontal cortex and hippocampus-striatum communication during learning (Battaglia et al., 2008). The latter has been shown to be important for learning place-reward associations (Lansink et al., 2009), which constitutes an important mechanisms to learn the reward function of an internal model (Khamassi and Humphries, 2012). The former has been widely studied in terms of transfer of episodic memory-based learning in the hippocampus to long-term memory in the prefrontal cortex (Frankland and Bontempi, 2005). The prefrontal cortex is known for its role in decision-making and cognitive control (Miller and Cohen, 2001) and is thus a good candidate to store in long-term memory the behavioral policy learned by an animal through reinforcement learning (Pasupathy and Miller, 2005; Frank and Claus, 2006; Khamassi et al., 2015). Strikingly, the strength of communication between hippocampus and prefrontal cortex has been found to increase at the decision-point in a Y-maze, specifically during phases of the task where the animal's performance was suddenly rising (in the order of 10 trials) in terms of reward rate (Benchenane et al., 2010). Moreover, it is only after recording sessions where learning occurred (thus sessions including reward rate increases) and not after sessions where the reward rate remained stable that the prefrontal cortex was found to be task-dependently reactivated during subsequent sleep by the animal (Peyrache et al., 2009). These experimental results have been interpreted in terms of tagging relevant information to be subsequently consolidated in memory during sleep (we would say here during offline reactivations in general, including both sleep and quiet wakefulness periods). The present model provides a general principle for learning through offline reactivations that is compatible with these experimental results. The model further predicts that these experimental results are not simply reflecting memory consolidation of things that have been previously well learned, but are more generally related to the learning process itself. Consistently, the hippocampus has been found to be necessary for learning in several reward-based navigation tasks (Girardeau et al., 2009; Jadhav et al., 2012; de Lavilléon et al., 2015) and interactions between the hippocampus and the orbitofrontal cortex have been recently proposed to contribute to exploit the task structure learned by the internal model to infer the value of stimuli within the environment (Jones et al., 2012; Klein-Flügge et al., 2013; Wikenheiser and Schoenbaum, 2016; Zhou et al., 2019; Park et al., 2019).

A number of improvements to the model could be done in the future to overcome its current limitations. First, in terms of the model's mechanisms themselves, we have simplistically considered that a first round of 10 simulated offline reactivation iterations were required to further determine whether an offline reactivation phase should be triggered. This was a means to measure Q-values instability, especially in cases where a recent reward change information had been propagated to one of the maze arm and required more offline reactivations at the decision-point to be further propagated to the rest of the maze. However, less costly means to estimate the need to perform further offline reactivations could be used in future versions of the model. For instance, we could systematically track and memorize recent Q-values instability so that the decision to initiate offline reactivations does not require re-estimating this instability. Alternatively, we could use more explicit measures of decision-making uncertainty (Pezzulo et al., 2013; Viejo et al., 2015) to decide when to initiate offline reactivations. A second important limitation of the present work is that the model has so far only been tested in a single simple discrete navigation task. The model should be further simulated in a number of navigation tasks, including open environments so as to allow trajectory sampling to fully express its potential, and compared to more experimental data. In particular, we have only simulated cases of deterministic rewards: the reward was delivered with probability 1 if the agent had chosen the right arm, and probability 0 otherwise. It would be interesting to simulate the model in a task involving stochastic rewards and analyze how this changes the learning and offline reactivation dynamics. It would also be important to analyze the sensitivity of the model's performance in various tasks to different tested parameter-sets. This would for instance permit to assess whether the need to be more exploratory during offline inference than during task performance is a general property or not. Finally, it is also important to keep in mind that simulating reinforcement learning models in discrete tasks (i.e., grid-worlds) is good to learn about their properties, but makes them far from generalizable to real-world situations. In Aubin et al. (2018) we have simulated a neural network version of a dyna-Q algorithm with prioritized sweeping, so as to be able to manipulate more realistic state descriptions (population coding of spatial position). We found that some model properties were preserved, such as the percentage of generated backward state sequences. But importantly, we were confronted with the necessity of having two types of offline reactivations: those used to estimate the Q-values, which have a similar role to those presented in the present paper, and another set of unrelated unordered offline reactivations necessary to properly learn the world-model when represented with a neural network. Thus, further systematic comparisons of the same models tested in discrete and continuous environments are needed to better learn how robust and generalizable these models are.

References

Arleo A, Gerstner W (2000) Spatial cognition and neuro-mimetic navigation: a model of hippocampal place cell activity. Biological cybernetics 83(3):287–299

Aubin L, Khamassi M, Girard B (2018) Prioritized sweeping neural DynaQ with multiple predecessors, and hippocampal replays. In: Conference on Biomimetic and Biohybrid Systems, Springer, pp 16–27

Barto AG (1995) Adaptive critics and the basal ganglia. In: Houk JC, Davis JL, Beiser DG (eds) Models of Information Processing in the Basal Ganglia, The MIT Press, Cambridge, MA, pp 215–232

Barto AG, Bradtke SJ, Singh SP (1995) Learning to act using real-time dynamic programming. Artificial intelligence 72(1-2):81–138

Battaglia FP, Peyrache A, Khamassi M, Wiener SI, et al. (2008) Spatial decisions and neuronal activity in hippocampal projection zones in prefrontal cortex and striatum. Hippocampal Place Fields: Relevance to Learning and Memory pp 289–311

Benchenane K, Peyrache A, Khamassi M, Tierney PL, Gioanni Y, Battaglia FP, Wiener SI (2010) Coherent theta oscillations and reorganization of spike timing in the hippocampal-prefrontal network upon learning. Neuron 66(6):921–936

Bhalla US (2019) Dendrites, deep learning, and sequences in the hippocampus. Hippocampus 29(3):239–251

Buzsáki G (1989) Two-stage model of memory trace formation: A role for ”noisy” brain states. Neuroscience 31(3):551–570

Caluwaerts K, Staffa M, N’Guyen S, Grand C, Dollé L, Favre-Félix A, Girard B, Khamassi M (2012) A biologically inspired meta-control navigation system for the psikharpax rat robot. Bioinspiration & biomimetics 7(2):025009

Cazé R, Khamassi M, Aubin L, Girard B (2018) Hippocampal replays under the scrutiny of reinforcement learning models. Journal of neurophysiology 120(6):2877–2896

Cisek P, Puskas GA, El-Murr S (2009) Decisions in changing conditions: the urgency-gating model. Journal of Neuroscience 29(37):11560–11571

Cutsuridis V, Hasselmo M (2011) Spatial memory sequence encoding and replay during modeled theta and ripple oscillations. Cognitive Computation 3(4):554–574

Daw ND, Niv Y, Dayan P (2005) Uncertainty-based competition between prefrontal and dorsolateral striatal systems for behavioral control. Nature neuroscience 8(12):1704

Diba K, Buzsáki G (2007) Forward and reverse hippocampal place-cell sequences during ripples. Nature neuroscience 10(10):1241

Dollé L, Khamassi M, Girard B, Guillot A, Chavarriaga R (2008) Analyzing interactions between navigation strategies using a computational model of action selection. In: International Conference on Spatial Cognition, Springer, pp 71–86

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

Dollé L, Chavarriaga R, Guillot A, Khamassi M (2018) Interactions of spatial strategies producing generalization gradient and blocking: A computational approach. PLoS computational biology 14(4):e1006092

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

Foster DJ (2017) Replay comes of age. Annual review of neuroscience 40:581–602

Foster DJ, Wilson Ma (2006) Reverse replay of behavioural sequences in hippocampal place cells during the awake state. Nature 440(7084):680–683

Frank MJ, Claus ED (2006) Anatomy of a decision: striato-orbitofrontal interactions in reinforcement learning, decision making, and reversal. Psychological review 113(2):300

Frankland PW, Bontempi B (2005) The organization of recent and remote memories. Nature reviews Neuroscience 6(2):119–130

Girardeau G, Benchenane K, Wiener SI, Buzsáki G, Zugaro MB (2009) Selective suppression of hippocampal ripples impairs spatial memory. Nature neuroscience 12(10):1222–1223

Guazzelli A, Bota M, Corbacho FJ, Arbib MA (1998) Affordances. motivations, and the world graph theory. Adaptive Behavior 6(3-4):435–471

Gupta AS, van der Meer MAA, Touretzky DS, Redish AD (2010) Hippocampal Replay Is Not a Simple Function of Experience. Neuron 65(5):695–705

Jadhav SP, Kemere C, German PW, Frank LM (2012) Awake hippocampal sharp-wave ripples support spatial memory. Science 336(6087):1454–1458

Jahnke S, Timme M, Memmesheimer RM (2015) A unified dynamic model for learning, replay, and sharp-wave/ripples. Journal of Neuroscience 35(49):16236–16258

Johnson A, Redish AD (2005) Hippocampal replay contributes to within session learning in a temporal difference reinforcement learning model. Neural Networks 18(9):1163–1171, DOI 10.1016/j.neunet.2005.08.009

Johnson A, Redish AD (2007) Neural ensembles in CA3 transiently encode paths forward of the animal at a decision point. Journal of Neuroscience 27(45):12176–12189

Johnson A, van der Meer MA, Redish AD (2007) Integrating hippocampus and striatum in decision-making. Current opinion in neurobiology 17(6):692–697

Jones JL, Esber GR, McDannald MA, Gruber AJ, Hernandez A, Mirenzi A, Schoenbaum G (2012) Orbitofrontal cortex supports behavior and learning using inferred but not cached values. Science 338(6109):953–956

Karlsson MP, Frank LM (2009) Awake replay of remote experiences in the hippocampus. Nature neuroscience 12(7):913

Khamassi M, Humphries MD (2012) Integrating cortico-limbic-basal ganglia architectures for learning model-based and model-free navigation strategies. Frontiers in Behavioral Neuroscience 6:79

Khamassi M, Quilodran R, Enel P, Dominey P, Procyk E (2015) Behavioral regulation and the modulation of information coding in the lateral prefrontal and cingulate cortex. Cerebral Cortex 25(9):3197–3218

Klein-Flügge MC, Barron HC, Brodersen KH, Dolan RJ, Behrens TEJ (2013) Segregated encoding of reward–identity and stimulus–reward associations in human orbitofrontal cortex. Journal of Neuroscience 33(7):3202–3211

Lansink CS, Goltstein PM, Lankelma JV, McNaughton BL, Pennartz CMA (2009) Hippocampus leads ventral striatum in replay of place-reward information. PLoS Biology 7(8), DOI 10.1371/journal.pbio.1000173

de Lavilléon G, Lacroix MM, Rondi-Reig L, Benchenane K (2015) Explicit memory creation during sleep demonstrates a causal role of place cells in navigation. Nature neuroscience 18(4):493–495

Lee AK, Wilson MA (2002) Memory of sequential experience in the hippocampus during slow wave sleep. Neuron 36(6):1183–1194

Levy WB (1996) A sequence predicting ca3 is a flexible associator that learns and uses context to solve hippocampal-like tasks. Hippocampus 6(6):579–590

Lin LJ (1992) Self-improving reactive agents based on reinforcement learning, planning and teaching. Machine learning 8(3/4):69–97

Maingret N, Girardeau G, Todorova R, Goutierre M, Zugaro M (2016) Hippocampo-cortical coupling mediates memory consolidation during sleep. Nature neuroscience 19(7):959–964

Mattar MG, Daw ND (2018) Prioritized memory access explains planning and hippocampal replay. Nature Neuroscience 21(11):1609

van der Meer M, Kurth-Nelson Z, Redish AD (2012) Information processing in decision-making systems. The Neuroscientist 18(4):342–359

Miller EK, Cohen JD (2001) An integrative theory of prefrontal cortex function. Annual review of neuroscience 24(1):167–202

Moore AW, Atkeson CG (1993) Prioritized sweeping: Reinforcement learning with less data and less time. Machine learning 13(1):103–130

O'Keefe J, Dostrovsky J (1971) The hippocampus as a spatial map: Preliminary evidence from unit activity in the freely-moving rat. Brain research 34(1):171–175

Ólafsdóttir HF, Barry C, Saleem AB, Hassabis D, Spiers HJ (2015) Hippocampal place cells construct reward related sequences through unexplored space. eLife 4(JUNE):e06063

Ólafsdóttir HF, Bush D, Barry C (2018) The role of hippocampal replay in memory and planning. Current Biology 28(1):R37–R50

Palminteri S, Lefebvre G, Kilford EJ, Blakemore SJ (2017) Confirmation bias in human reinforcement learning: Evidence from counterfactual feedback processing. PLoS computational biology 13(8):e1005684

Papale AE, Zielinski MC, Frank LM, Jadhav SP, Redish AD (2016) Interplay between Hippocampal Sharp-Wave-Ripple Events and Vicarious Trial and Error Behaviors in Decision Making. Neuron 92(5):1–8

Park SA, Miller DS, Nili H, Ranganath C, Boorman ED (2019) Map making: Constructing, combining, and navigating abstract cognitive maps. BioRxiv p 810051

Pasupathy A, Miller EK (2005) Different time courses of learning-related activity in the prefrontal cortex and striatum. Nature 433(7028):873

Peng J, Williams RJ (1993) Efficient learning and planning within the Dyna framework. Adaptive Behavior 1(4):437–454

Peyrache A, Khamassi M, Benchenane K, Wiener SI, Battaglia FP (2009) Replay of rule-learning related neural patterns in the prefrontal cortex during sleep. Nature Neuroscience 12(7):919–926

Pezzulo G, Rigoli F, Chersi F (2013) The mixed instrumental controller: using value of information to combine habitual choice and mental simulation. Frontiers in psychology 4

Pezzulo G, van der Meer MAA, Lansink CS, Pennartz CMA (2014) Internally generated sequences in learning and executing goal-directed behavior. Trends in Cognitive Sciences 18(12):647–657

Pezzulo G, Kemere C, Van Der Meer MA (2017) Internally generated hippocampal sequences as a vantage point to probe future-oriented cognition. Annals of the New York Academy of Sciences 1396(1):144–165

Pfeiffer BE, Foster DJ (2013) Hippocampal place-cell sequences depict future paths to remembered goals. Nature 497(7447):74

Pohl I (1971) Bi-directional search. Machine intelligence 6(127-140):10

Redish AD (2016) Vicarious trial and error. Nature Reviews Neuroscience 17(3):147–159

Renaudo E, Girard B, Chatila R, Khamassi M (2014) Design of a control architecture for habit learning in robots. In: Conference on Biomimetic and Biohybrid Systems, Springer, pp 249–260

Rennó-Costa C, da Silva ACC, Blanco W, Ribeiro S (2019) Computational models of memory consolidation and long-term synaptic plasticity during sleep. Neurobiology of learning and memory 160:32–47

Roumis DK, Frank LM (2015) Hippocampal sharp-wave ripples in waking and sleeping states. Current opinion in neurobiology 35:6–12

Saravanan V, Arabali D, Jochems A, Cui AX, Gootjes-Dreesbach L, Cutsuridis V, Yoshida M (2015) Transition between encoding and consolidation/replay dynamics via cholinergic modulation of can current: a modeling study. Hippocampus 25(9):1052–1070

Schultz W, Dayan P, Montague PR (1997) A neural substrate of prediction and reward. Science 275:1593–1599

Stachenfeld KL, Botvinick MM, Gershman SJ (2017) The hippocampus as a predictive map. Nature neuroscience 20(11):1643

Sutton RS (1990) Integrated architectures for learning, planning, and reacting based on approximating dynamic programming. In: Proceedings of the seventh international conference on machine learning, pp 216–224

Sutton RS, Barto AG (1998) Reinforcement Learning: An Introduction. Cambridge, MA: MIT Press

Viejo G, Khamassi M, Brovelli A, Girard B (2015) Modeling choice and reaction time during arbitrary visuomotor learning through the coordination of adaptive working memory and reinforcement learning. Frontiers in behavioral neuroscience 9

Wikenheiser AM, Schoenbaum G (2016) Over the river, through the woods: cognitive maps in the hippocampus and orbitofrontal cortex. Nature Reviews Neuroscience 17(8):513–523

Wilson MA, McNaughton BL (1994) Reactivation of hippocampal ensemble memories during sleep. Science (New York, NY) 265(5172):676–679

Zhou J, Montesinos-Cartagena M, Wikenheiser AM, Gardner MP, Niv Y, Schoenbaum G (2019) Complementary task structure representations in hippocampus and orbitofrontal cortex during an odor sequence task. Current Biology 29(20):3402–3409