Computational Brain & Behavior (2020) 3:472–493
https://doi.org/10.1007/s42113-020-00095-7

ORIGINAL PAPER

Computational Neural Mechanisms of Goal-Directed Planning and Problem Solving

Justin M. Fine[1] · Noah Zarr[1] · Joshua W. Brown[1]

Accepted: 2 November 2020 / Published online: 13 November 2020

Abstract

The question of how animals and humans can solve arbitrary goal-driven problems remains open. Reinforcement learning (RL) methods have approached goal-directed control problems through model-based algorithms. However, RL focus on maximizing long-term reward is inconsistent with the psychological notion of planning to satisfy homeostatic drives, which involves setting goals first, then planning actions to achieve them. Optimal control theory suggests a solution: animals can learn a model of the world, learn where goals can be fulfilled, set a goal, and then act to minimize the difference between actual and desired world states. Here, we present a purely localist neural network model that can autonomously learn the structure of an environment and then achieve any arbitrary goal state in a changing environment without relearning reward values. The model, GOLSA, achieves this through a backwards spreading activation that propagates goal-values to an agent. The model elucidates how neural inhibitory mechanisms can support competition between goal representations, serving to push needs-based planning versus exploration. The model performs similar to humans in canonical revaluation tasks used to classify human and rodent behavior as goal-directed. The model revaluates optimal actions when goals, goal-values, world structure, and need to fulfill drive changes. The model also clarifies a number of issues inherent in other RL-based representations such as policy dependence in successor representations, while elucidating biological constraints such as the role of oscillations in gating information flow for learning versus action. Together, our proposed model suggests a biologically grounded framework for multi-step planning behaviors through consideration of how goal representations compete for behavioral expression in planning.

Keywords Neural network · Goal-directed decision making · Planning · Model-based · Cognitive control · Systems neuroscience

Introduction

Actions derived from habits and reflexes offer a surprising amount of power for producing successful behavior, and they require minimal cognitive control for production. This reduced complexity, however, renders them inflexible because they are driven predominantly in a stimulus reactive mode within an environment. Such inflexibility is insufficient for supporting all behaviors, particularly when the cached actions will not lead to successful goal-driven outcomes (Dolan and Dayan 2013). The capacity for goal-oriented and flexible control behavior is abundantly recognized across species and thought to involve learning an internal model of the environment (Dayan and Berridge 2014). Early evidence for model-based behavior was Tolman's (1948) latent map learning and demonstrations showing animals can plan novel actions without significant relearning. Humans and animals can also readily replan actions when reward, optimal policies, and environmental structure change (Alvernhe et al. 2011; Momennejad et al. 2017; Daw et al. 2005). These findings have generally converged on a conception of cognitive planning of novel actions as a goal-driven behavior that uses a mental simulation with an environmental model (Behrens et al., 2018; Daw and Dayan 2014; Ivey et al. 2011). How this occurs has been proposed to involve animals and humans employing representations of goals, actions, and rewards that operate in conjunction with an environmental (map) representation of task space (Behrens et al. 2018; Epstein et al. 2017; Tolman 1948; Wilson et al. 2014). Despite empirical and theoretical implications, there is a dearth of models providing a biological account of how goals guide flexible planning behavior in both spatial and abstract cognitive task spaces.

✉ Joshua W. Brown
jwmbrown@indiana.edu

[1] Dept. of Psychological and Brain Sciences, Indiana University, 1101 E Tenth St, Bloomington, IN 47405, USA

Explicitly, several key questions are (1) what biologically plausible planning mechanisms work in service of goals, (2) how do environmental and reward or goal representations support flexible planning, and (3) how do constraints and the need to fulfill multiple goals compete for control during planning?

Solving this issue requires brain mechanisms that can support planning novel actions towards new goals, away from devalued rewards, or when encountering changing environments. Models of goal-directed control are typically formulated with model-based reinforcement learning (MBRL; Doll et al. 2012; Sutton and Barto 2018). MBRL allows prospective planning through simulating possible future scenarios based on representations of rewards within environmental maps and the model of task space (Daw et al. 2005; Daw and Dayan 2014). An advantage to model-based control is flexibility in replanning when task variables such as reward or environment structure change. Because planning with MBRL problems involves either extensive dynamic programming or exhaustive tree search (Niv et al. 2006) for estimating rewarding states far into the future, this is problematic for resource constrained biological systems (Huys et al. 2012; Simon 1956). These issues leave the biological feasibility underlying MBRL planning uncertain (Daw and Dayan 2014).

Growing evidence indicates some model-based computations are tractable through operations similar to model-free RL, thus reducing computational complexity. The successor representation (SR; Dayan 1993) has been proposed as a viable mechanism that could circumvent these issues with exhaustive planning in full model-based control (Gershman 2018; Momennejad et al. 2017). The SR operates through learning and caching expectations about which states will be visited over a future horizon from any given state. The SR's biological motivation partly rests on findings indicating model-based processes are linked to the Dopaminergic learning circuit originally constrained to explanations of model-free RL (Russek et al. 2017). Experimental and modeling studies have shown that humans and agents using an SR could solve reward re-valuation problems thought originally to require full model-based evaluation (Momennejad et al. 2017; Russek et al. 2017). This instance of goal-directed control emerges because the SR separates expectations of state visitations from rewards, affording it representational flexibility (Dayan 1993).

There are several points to consider with the both the SR and model-based RL that, presently, leave issues open about the computations underlying goal-based (re)planning. The SR alone can only solve replanning problems involving revaluation of already rewarding and visited states (Russek et al. 2017). However, humans are capable of several replanning behaviors not predicted by the SR alone and call on model-based control (Momennejad et al. 2017). The SR restriction occurs because it learns its expectancies on-policy, evaluating expected future states contingent on a fixed policy that was reinforced. This policy dependence makes SRs inflexible if new states not predicted under the SR become required to acquire reward. For example, this can occur in detour problems when the previously available state transitions are blocked. To overcome this limitation, model-based and SR hybrids are necessary to explain other replanning behaviors such as policy change for visiting states that are not expected under the SR or a change in task environment structure (Russek et al. 2017). This indicates human planning involves more than a policy dependent predictive map.

The focus of RL models maximizing an arbitrary future reward from a state also juxtaposes with planning as motivated by time-varying goals and needs, as understood psychologically (de Wit and Dickinson 2009; O'Reilly 2020). Therein, desired outcomes contextualize the expected states of an agent such as sating hunger, and drive the optimal actions needed to reach those states. A goals-first planning system as devised in optimal control or planning algorithms employ backwards induction or search by starting at a goal and finding the optimal state-based controller (Bertsekas, 2010; Todorov 2009). Desired outcomes should influence choice. RL methods using forward solutions to optimize reward expectations are inconsistent with this notion of goal-directed behavior.

Psychological evidence for backward, goal-based search comes from studies of cue-induced physiological states such as drug cravings leading to seeking actions (Dayan 2009), Pavlovian instrumental specific transfer with goals increasing actions that produce it (Dickinson and Balleine 1994), and even reaching towards a spatial goal (Liu and Todorov 2007). Goals can be construed as inducing behaviors that fulfill drives such as hunger or avoiding pain, and represent a form of cognitive or physiological setpoint (Hull 1943; Juechems and Summerfield 2019; O'reilly et al. 2014). Nonetheless, a neuronally plausible model that accounts for flexible, goals-based planning remains to be established.

Here, we present the Goal-Oriented Learning and Sequential Action (GOLSA) model. GOLSA illustrates one way a neural system could plausibly learn to navigate an environment in pursuit of goals in line with optimal control solutions. The network is comprised of continuous-time dynamic model neurons that utilize associative learning to construct a neural representation of an environment's state space. GOLSA's planning mechanism builds on the core premises of optimal control applied to goal-directed control. GOLSA demonstrates how these mechanisms support planning to goal states that fulfill agent needs or drives. The model's notion of goals is motivated by drive-reduction theory applied to motivation (Hull 1943). GOLSA can solve both a plethora of planning and replanning problems that are similarly performable by humans and animals. It can flexibly plan when faced with changes in goal or environmental structure, over spatial and nonspatial problems, and plan with respect to state-dependent

drives or goals. Two core intuitions emerged when elaborating this system. First, solving goals- and drives-based planning akin to optimal control involves a mechanism that spreads backwards as an activity gradient from goals to an agent's adjacent states. The other intuition regards the dual-roles problem wherein the same neural layers are involved in multiple time-dependent task phases as planning and action or learning. This problem is elaborated by findings showing prefrontal cortex (PFC) activation supports goal representations and goal-directed planning, and that its neurons encode multiple task aspects (Badre and Nee 2018; Botvinick and Cohen, 2014). Solving this problem requires a mechanism to control the flow of pre-synaptic information to these areas at the appropriate time. GOLSA solves this problem through an oscillatory gating and inhibition system like cross-frequency coupling found in human and animal electrophysiology (Bonnefond et al. 2017).

Methods

GOLSA Model

GOLSA (Fig. 1) provides a novel account of goal-directed behavior that can handle many of the same tasks as reinforcement learning algorithms, by implementing processes akin to optimal control theory through Hebbian learning, and neural oscillations. GOLSA agents can select any known state as a goal to pursue rather than "reward." A state is defined here as a particular set of conditions of the agent and its environment, which can be operationalized as a unique vector. States are represented by corresponding neurons in GOLSA. Goal states can be imposed externally or selected based on drive-reduction principles (Hull 1943). Once selected, a gradient of activity across the map is established and has a peak at the goal location. The network proceeds by selecting a state adjacent to the current state. The choice of adjacent state is the one that brings the agent closest to the peak of the gradient. An action is then chosen to move the agent to the desired next state. This process is repeated until a goal is reached. All learning in the model is essentially Hebbian, satisfying constraints of localist representation and broader neural plausibility. The model is conceptually similar to the A* (Hart et al. 1968) and Dijkstra (Dijkstra 1959) algorithms commonly used in consumer GPS navigation devices, although the GOLSA model differs in that it respects neurobiological constraints (Supplementary Material).

Figure 2 a shows the overall architecture of the model. Each red box in Fig. 2 b represents a layer of rate-coded artificial neurons. Depending on the layer's function, each unit represents either an environmental state, an action the agent can take, or a state transition. The gray ovals represent nodes

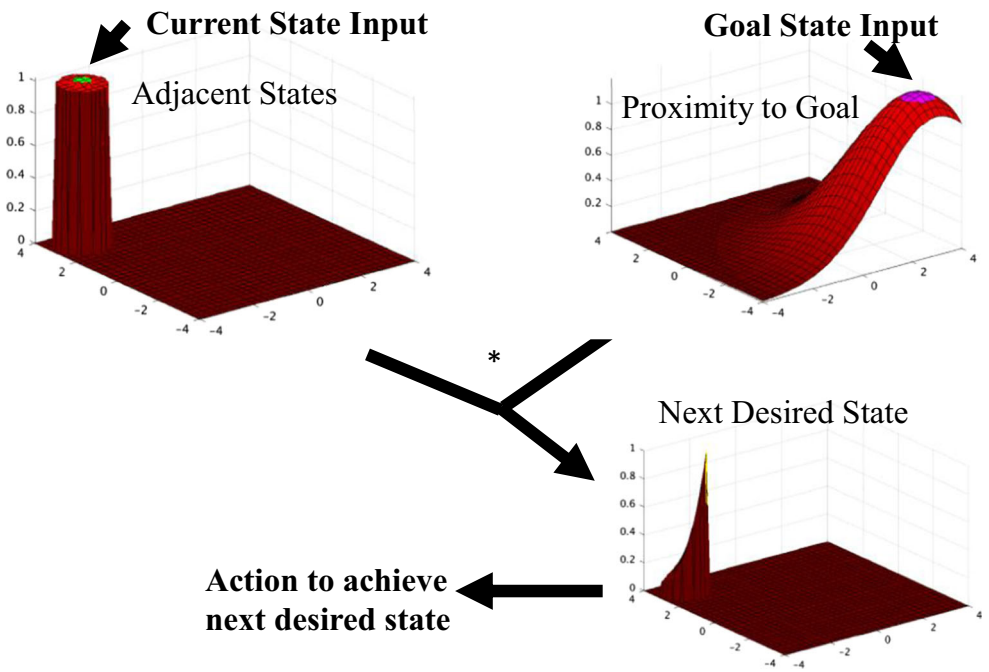

Fig. 1 GOLSA model intuition. The model takes as input the current state and goal states and generates as output an action to move closer to the goal state. The model represents each state with a corresponding neuron in a given layer, and neurons representing nearby states that can be reached with a single action also share a synaptic connection. All state layers have representations of all possible states. The proximity to goal layer diffuses activity backward from the activated goal state. The current state input activates all adjacent states unit representations that are one step away from the current state. The next desired state is simply the unit-wise product of activities of the adjacent states and proximity to goal layers, so that the most active unit represents the state that can be reach in one step and is closest to the goal state. Other model components (not shown) map the next desired state to an action aimed at achieving that state

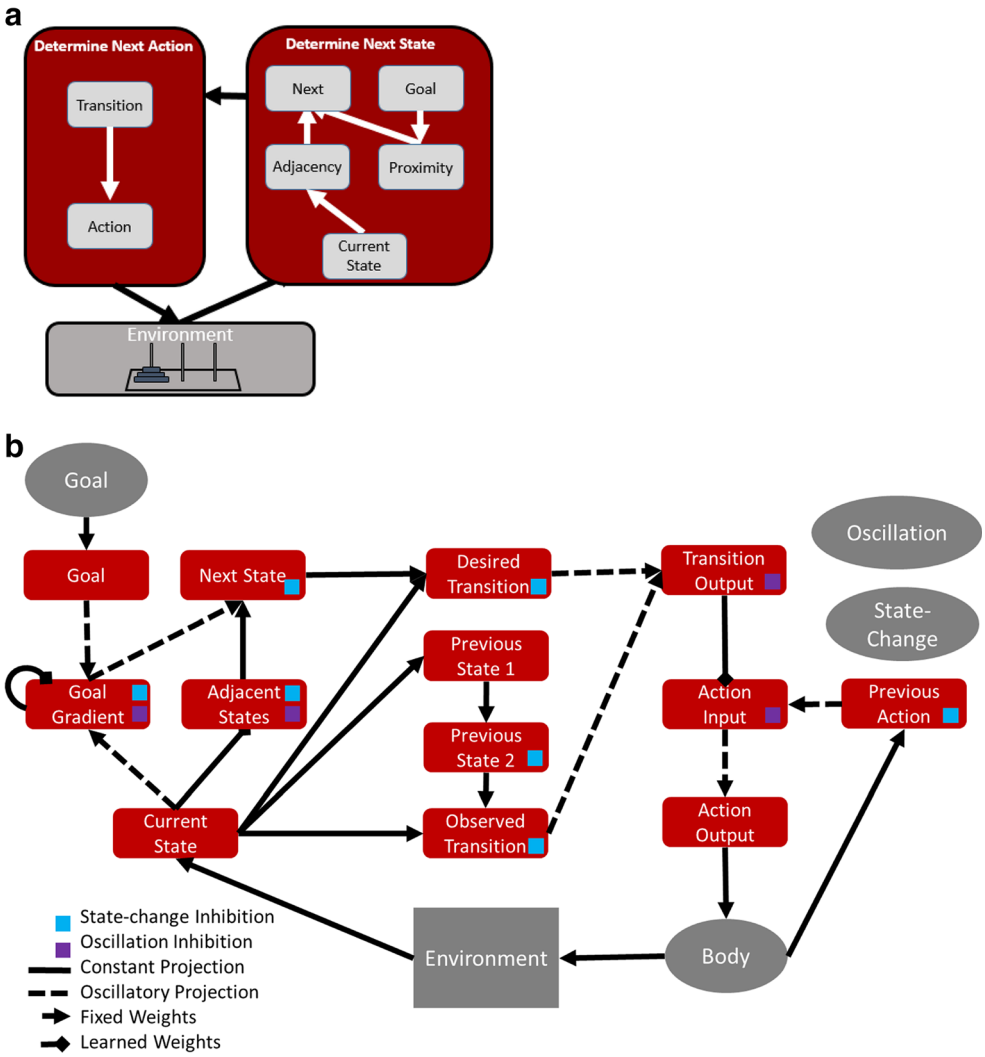

Fig. 2 **a** Schematic overview. The right box encapsulates Fig. 1 above, while the left box depicts the desired transition (conjunction of current state and desired next state), which activates a corresponding action (i.e., an inverse model). This in turn causes changes to the current state of the agent in the environment. **b** Full diagram of core model. Each rectangle represents a layer and each arrow a neural projection. The body is a node, and two additional nodes are not shown which provide inhibition at each state-change and oscillatory control (below and Supplementary Material). The colored squares indicate which layers receive inhibition from these nodes. Some recurrent connections not shown

which provide external input or control signals to the model that are not represented in terms of neural activity. Many of the projections between model layers are hard-coded one-to-one mappings, though several key projections are fully plastic and instantiate the synaptic weight learning processes described in detail below.

GOLSA Algorithm

The GOLSA model's behavior can be described by the following algorithm.

1. Determine which states can be reached from the current state.
2. Identify one of these states that will best bring the agent closest to the current goal.
3. Take the action that will likely implement the transition between the current state and the desired next state.

The two minimum components learned for this algorithm are the adjacency structure, i.e., the topology of the state space and the mapping from transitions to actions. This information is instantiated in the connections of three key projections between layers in the network: from *current state* to *adjacent states*, from *goal gradient* to itself, and from *transition output* to *action input*. Informally, core aspects of GOLSA planning processes can be seen as a neural realization of a particular class of algorithms for optimal control solutions to model-based planning.

Here, we link these core GOLSA operations to previous algorithmic solutions for solving Markov decision processes (MDP). Notably, typical MDP solutions often involve solving the Bellman equation through dynamic programming to derive a value function and a nonlinear action function that maximizes future reward (Sutton and Barto 2018). In contrast, GOLSA uses an interaction of adjacency information and the backward diffusing goal-gradient which closely aligns GOLSA with Todorov's (2009) algorithmic framework for

solving optimal control problems. Todorov (2009) demonstrated that if the value function of Bellman equation is exponentiated, this removes the nonlinear maximization problem. The linearized solution then becomes a soft-maximization process compatible with interactions between biological neurons (Maass 2000). This important insight affords optimal control solutions that are formally equivalent to inferential message passing algorithms for graphical models such as the forward-backward algorithm (Attias 2003; Levine 2018). The overlap of GOLSA and Todorov (2009) like forward-backward solutions to optimal control can be seen at several levels, which we describe qualitatively rather than via formal derivation for expositional simplicity.

Both GOLSA and Todorov (2009) use forward-backward interactions to solve for a next desired state or optimal successor state transition. This occurs through both the types of representations and their interaction. Representationally, both GOLSA and Todorov (2009) separate mappings of transitions between nonterminal (nongoal) states and nonterminal to terminal (goal) states. GOLSA's approximate equivalents are the adjacency and goal-gradient, respectively. GOLSA's desired transition is computed via the desired next state, which emerges through the interaction of both the forward steps by the adjacency and backwards goal distance through the *goal-gradient*. In GOLSA, the adjacency matrix passes a forward message from state s to s' and approximates p(s'|s) as uniform with a matrix inverse equivalent to a random-walk successor representation. The effect of this forward-backward interaction is GOLSA agents ameliorate the policy-dependence issues that limit the SR for sufficiently explaining human planning behavior. Note, in the forward-backward algorithm, the backwards process is like the backward induction process used in standard dynamic programming. Advantageously, and unlike dynamic programming, the GOLSA and Todorov (2009) solutions offer a computationally feasible neural solution. Neither requires exhaustive forward searches for planning found in MBRL or several dynamic programming (value or policy) iterations for online policy control or when a goal changes.

Below, we present a simple example of how these processes drive model activity after successful learning, followed by an explanation of the learning process. For simplicity, the model activity will generally be described in the context of a six-state grid world arranged as shown in Fig. 3, where each state can be reached from the states physically adjacent to it. Although this appears identical to a simple 2D navigation problem, the approach generalizes to high dimensional states representing arbitrarily complex problems. A more complete description of the model is in the Supplementary Material.

Figure 3 illustrates the basic mechanisms of the algorithm with a simple example. Each of the four maps of the environment corresponds to one of the layers in the network, and each of the six boxes in the map represents the corresponding unit in the layer. Each layer represents the same six states. The colors indicate the level of activity expected in each unit for a trial in which the agent starts in state 1 and has the goal of navigating to state 6.

For the representation of the current state, only the unit corresponding to the agent's actual location should be active (Fig. 3). The representation of adjacent states includes all and only the states reachable from the current state, including the current state itself. Within the *adjacent*-states layer, the current state is actually more active than the adjacent states to facilitate learning (Supplementary Materials). The goal gradient has a peak of activity at the goal location (state 6) and then a smoothly decreasing level of activity diffusing outward from the goal to representations of preceding states (this allows that some state transitions may only be possible in one direction and not bidirectional). This representation encodes each state's distance from the goal as a pattern of activity. The goal gradient and adjacent-state representations interact to produce a representation of the next desired state. The pattern of activity in this layer answers the question, "Of all the states adjacent to the current state, which shows the most activity in the goal gradient?" Here, there are two states that are both adjacent to the current state and equally close to the goal state. In the actual network, a winner-take-all (WTA) system breaks the symmetry and selects a single state to move to next. Using the current state and the next desired state, the network identifies the appropriate transition to implement and then takes the appropriate action based on a learned association between actions and transitions. If the goal gradient is thought of as a three-dimensional hill, the model climbs the hill by successively enacting the transitions that move the agent as far up the hill, and towards the goal, as possible.

Learning

The core model, described above, depends upon learning in three projections, namely from *current state* to *adjacent states*, from *goal gradient* to itself, and from *transition output* to *action input* (Fig. 2b). All other connections utilize hard-coded mappings with a one-to-one or simple combinatorial structure.

During early learning, the agent lacks knowledge about the structure of the state space and will therefore be unable to determine an appropriate next desired state or the action required to take an appropriate transition. Without determining the correct transition, the agent will obviously fail to take the appropriate actions to reach the goal. In our implementation of the model, after 300 time steps without taking an action, the agent "explores" by taking the available transition that has been implemented least often. Figure 4 shows model activity over the course of the first transition in a trial prior to any learning with the same starting and goal states as in Fig. 3, namely state 1 and state 6.

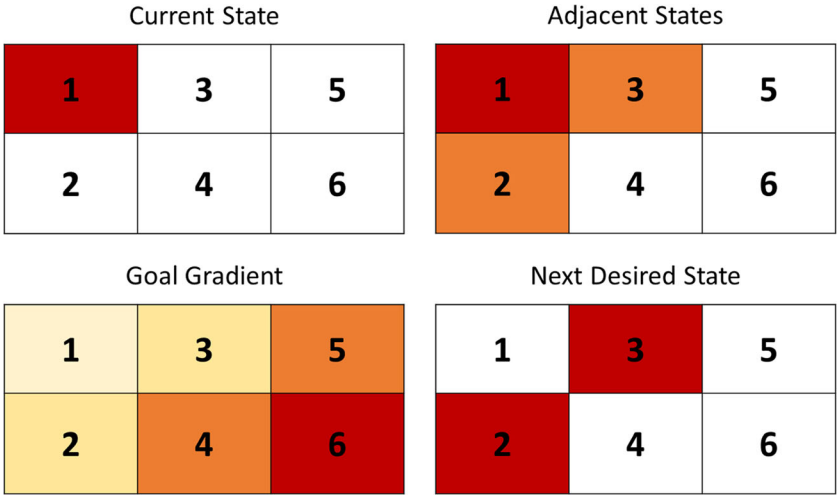

Fig. 3 GOLSA model example and illustration of important model layer activities for selecting the desired next state. Each box corresponds to a unit in the appropriate layer and more red units are more active. In the current-state layer, only the unit corresponding to the agent's state is active. In the adjacent-states layer, units representing states one step away are also active, but less so. Each unit in goal-gradient in rough proportion to the proximity to the goal state. When these two maps are integrated in next desired state, the state(s) which bring the agent one step closer to the goal are activated

As shown in Fig. 4 (time steps 0–600), the *adjacent-states* layer initially mirrors the *current-state* layer, while the *goal-gradient* alternates between a simple representation of the current state and the current goal. Without the necessary information to determine the appropriate next state, the agent explores, moving down into state 2. This results in learning in both the projection from *current state* (the State 1 unit) to *adjacent states* (State 2 unit) and the recurrent projection in

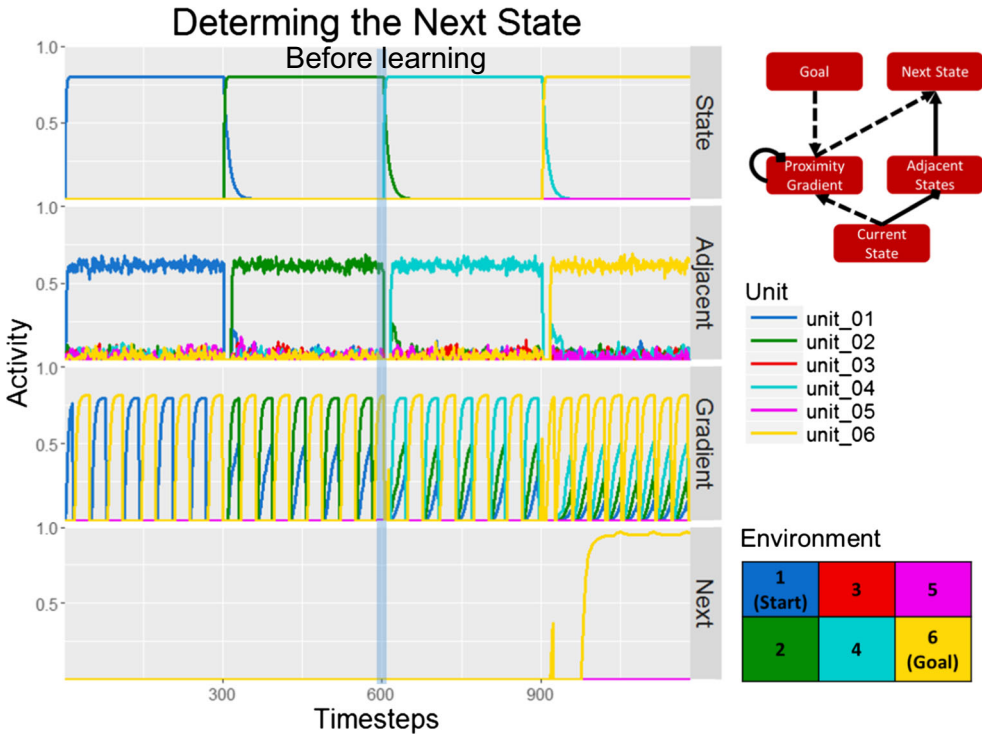

Fig. 4 Time Steps 0–600: Layer activities related to determining the desired next state during the first step of the first trial during learning. As described in Fig. 3, the model is starting at state 1 with the goal of state 6. No states are known to be adjacent so only current state activity is present there in the adjacent layer. After the first transition, the goal gradient learns as an oscillatory signal (describe below) provides input from the current State, which is now state 2. The model thus learns (in connections within the goal-gradient layer) that state 1 can precede state 2. All time steps 0–1200: Network activity related to determining the next state during the entire first trial, before learning has occurred. The trial continues until the agent reaches the goal state via random exploration. Note that when the agent reaches state 6, the goal, the entire trajectory is encoded in the goal-gradient layer. All previously visited states are active in order of their proximity to state 6

goal gradient (from State 2 unit to State 1 unit). Only the latter has a visible effect on the activities here because the *goal gradient* represents backward transitions while the *adjacent-states* layer looks forward into the future. Because the model does not assume that all transitions are bidirectional, it does not yet know that it is possible to move from state 2 to state 1. The *goal gradient*, on the other hand, now contains unit 2 activity during the learning phase, indicating that the model (specifically the *goal-gradient* layer) now knows that if it needs to get to state 2, it can do so via state 1.

The activity across the full first trial, shown in Fig. 4, demonstrates how the gradient continues to expand with experience. Because the agent happened upon the goal state, the trial ended. Note that until the agent actually reaches the goal state, no unit is active in *next desired state* and therefore no *desired-transition* is specified. This is because during most of the trial, no units were active in the *goal-gradient* during the action-phase of the oscillation at the same time that the corresponding unit was active in *adjacent-states*. Once the agent reaches state 6, this is no longer true and state 6 becomes the desired state and the 6-to-6 transition unit is activated.

For most of this trial, no desired next state was specified. Therefore, the network could not select a desired transition to implement with an action. This is shown in Fig. 5 with the lack of activity in *desired transition* and *action output* for most of the trial. Recall that transition layers dedicate one unit to each combination of starting and next states, such that there are 36 units in each transition layer for this simulation. Unit 1 represents the transition from state 1 to state 1 while unit 36 represents the transition from state 6 to state 6. Unit 36 becomes active at the end of the trial, when a unit finally becomes active in *next desired state* and the agent both currently inhabits state 6 and desires to be in state 6 at the next step.

On the next trial (Figs. 6 and 7), the agent is able to use the updated weights to walk the same path to the goal more quickly and without using forced exploration.

On this trial, the *goal-gradient* and *adjacent-states* layers contain enough information to activate the correct unit in *next desired state* and therefore in *next desired transition*. The agent's exploration in the previous trial taught it which actions implement those transitions, so activity is present in *action input* during the acting phase (the smaller peaks) in addition to the previous action information during the learning phase (the larger peaks). This activity accumulates in *action output*, which eventually triggers the *body* node to effect an action. The following section addresses the learning laws which allow the agent to move from blindly exploring in first trial to effectively navigating to the goal in the second trial.

Oscillations

The constraint of biological plausibility leads to a problem with learning the mapping from desired state transitions to actions. Specifically, the *dual-roles problem* requires the same projection, e.g., the mapping from desired state transitions to actions, to be presented with different information at different times, within a single event, for learning vs. performance. Consider that when a desired transition node is active, indicating a desired transition from, say, state 1 to state 3, the action (A) that is generated may instead cause a transition from state

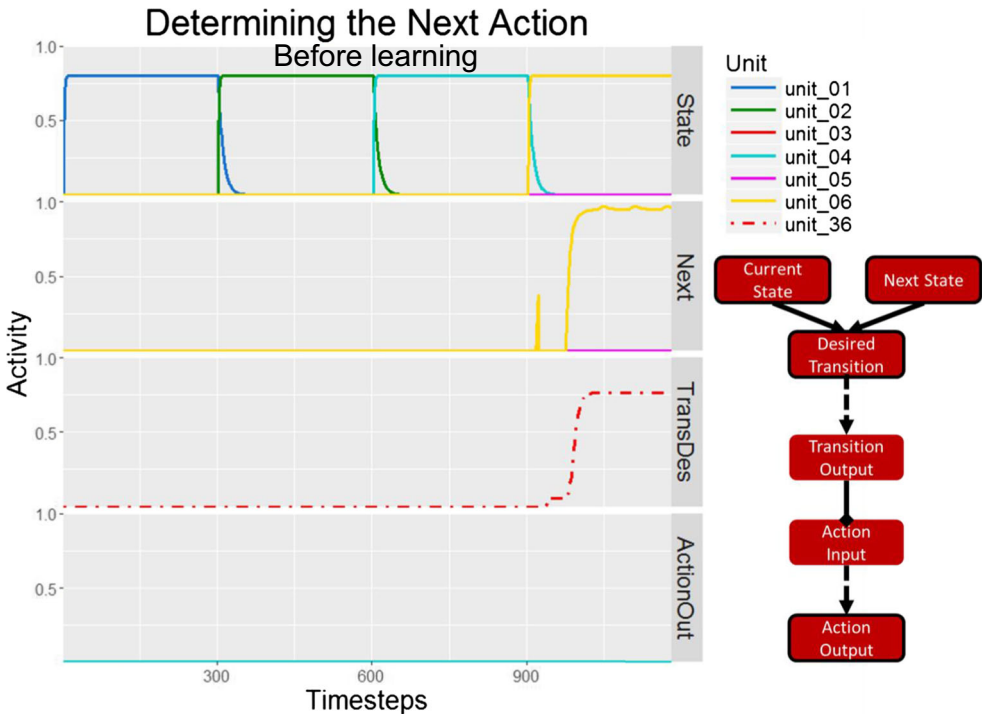

Fig. 5 Network activity in layers related to action selection during the first trial during learning, before learning has occurred. Note that transitions are numbered such that transition 1 is from state 1 to state 1, transition 6 is from state 1 to state 6, transition 7 is from state 2 to state 1, etc. Because no next state is specified during most of the trial, no transition is specified and no action is activated

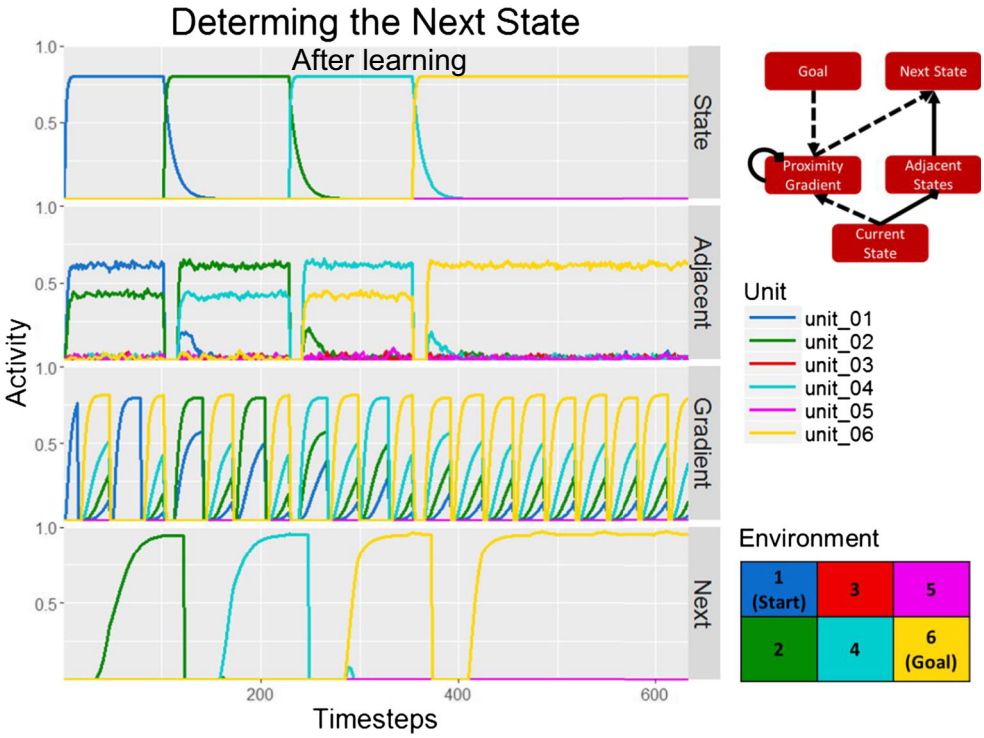

Fig. 6 Network activity in the layers responsible for goal selection during the second trial after learning. The agent utilizes the past experience in the environment to navigate back to the goal path. In each state, the appropriate next state is determined based on the activities in the adjacent-states layer and the goal-gradient layer. Gradient activity is only used during an oscillatory phase in which the goal rather than the current state is an input to the gradient

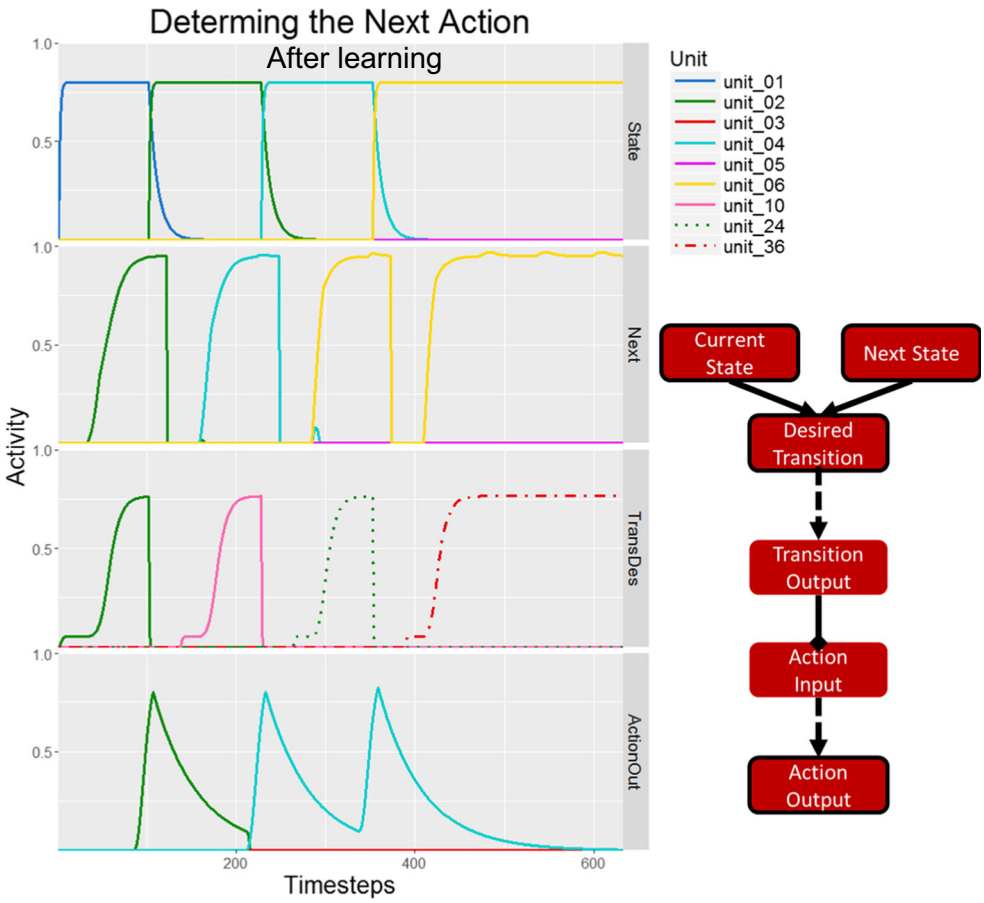

Fig. 7 Network activity in layers related to action selection during the first trial during learning. In the first trial, shown in Fig. 5, a desired transition was not specified for most of the trial because a next desired state was not selected. In this trial, the specification of a next desired state (Fig. 6), results in the specification of a transition in desired-transition (TransDes), which drives the appropriate actions compared to pre-learning where no actions were selected (Fig. 5)

1 to state 2. In that case, it would be inappropriate to strengthen the active synaptic weight from a node representing the desired transition (1➔3) to the action A, based purely on a Hebbian principle of coactivation. Instead, we would like to activate a node representing the actual transition (1➔2), and then strengthen its projection to the actual action A. This leads to another problem, which is that if we activate a representation of the desired transition (1➔ 2), what is to prevent that from activating some other action inappropriately? All of this suggests that the network must operate in two distinct modes, i.e., learning vs. acting. During the acting mode, the desired state transition must activate an action, which will then be generated. During the learning mode however, two changes must be made. First, the activity pattern of the desired transition layer must be changed to represent the *actual* transition that just occurred, instead of the *desired* transition. Second, the action generating layer must have its output disconnected, so that it cannot generate an inappropriate action during learning. To fulfill this requirement, we posit an oscillatory mechanism that rapidly switches the network between the learning and acting modes described above, with cell activities reliably linked to an oscillatory cycle (Klausberger and Somogyi 2008). The details of this mechanism, and its application to the three learned connections in the model, are described in the Supplementary materials.

Results

Small Environment

Over the course of multiple trials, the GOLSA model learns how to accurately navigate its environment to reach any goal state from an arbitrary starting state. Notably, it can generalize, learning to reach goal states without having been on those states as goals. With a constant goal and starting state across trials however, the model would simply repeat the first path it found via random exploration. To counter this tendency, the model was run with alternating starting and goal states. On each trial, it attempted to get from one corner of the small 6 state environment to the other, forcing the agent to take different paths to reach its goals. The agent also had a 20% chance to randomly take a relatively unexplored transition when attempting to take a known state transition. A trial was ended shortly after the goal was reached, or after 4000 time steps, with each time step equal to 50 msec (t = 200 s total) if it was not. Figure 8 shows how the average time to complete a trial and the overall error in the learned projections decreased over the course of 18 trials. The entire sequence of trials was repeated 50 times.

As expected, the time taken in each trial diminished rapidly to a minimum of about 30 time steps, though random exploration sometimes moved the agent off course making each trial take longer. The total sum squared error in the learned weight projections also diminished rapidly over the course of the trials, though failed to completely reach zero because some transitions were never explored. If the model were run for an extensive number of trials, it would eventually explore and learn all possible state transitions due to 3 properties. First, there is a random exploration term that will eventually cause a change in trial course. Second, there is noise in the choice probability as well that will eventually make the agent become adjacent to an unexplored transition. Third, the agent is empowered to explore unknown transitions. These 3 factors in combination ensure that with infinite time, the agent would explore the whole state space with a nonzero probability.

Tower of Hanoi

The GOLSA model can also learn to navigate larger, real-world problems. To demonstrate this, we taught the model to complete the Tower of Hanoi puzzle. The Tower of Hanoi consists of three differently sized discs on three pegs. The typical starting configuration is all three discs on the left-most peg. For a position to be legal, smaller discs cannot be below bigger discs. On each step, participants move the top disc from one peg to another, with the typical goal state being all three discs stacked on the rightmost peg. There are 27 legal states in the three-disc Tower of Hanoi. These states can be arranged into a graph where each node constitutes a legal state and each link represents a valid move.

To put the problem into a format that the GOLSA model can complete, each possible legal configuration was considered a state. There were 6 possible actions, each defined as attempting to move the top disc from one peg to another. The only time this "action" would fail is if there was in fact no disc on the "from" peg, in which case the action could still be taken but would not result in a state transition.

The architecture used was very similar to that used in the small six-state grid world, but with larger layers containing units to represent each of the 27 states or 729 transitions. The decay parameter on the *goal gradient* was also changed from 0 to 1 (effectively turning off the decay), allowing goal-related activity to diffuse more broadly through the network. Without this change, the decay would overwhelm the relatively small effect of goal activity passing backward through many states in a trajectory. While the gradient could still be successfully established after exploration-based learning, a single learned trajectory could not be reliably followed even if repeated many times. After removing the decay term (and adjusting the learning thresholds accordingly), however, the gradient is much more sensitive.

Learning proceeded in a similar fashion to that used in the small gridworld. While the three discs on the right peg constitute the typical goal state for the Tower of Hanoi, a major strength of the GOLSA model is its ability to navigate to

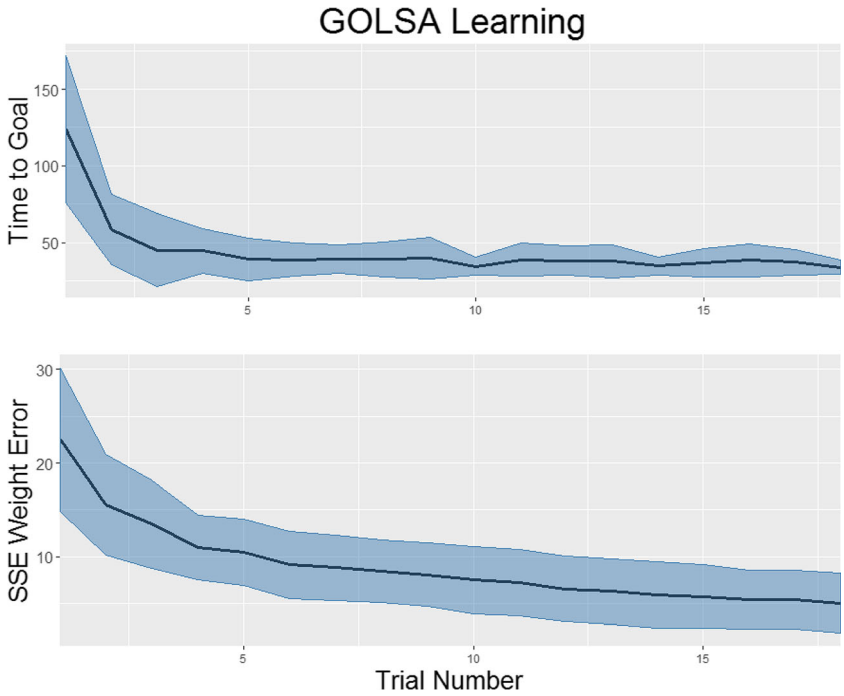

Fig. 8 Top: Average time taken to the goal as a function of the number of trials since learning began, in the six-state gridworld problem. The learning process was repeated 50 times and the shaded area represents the standard deviation across those 50 trial sequences. Bottom: Total sum squared error of the learned weights (actual – ideal) in the three key projections described above

any desired state. To facilitate learning the entire state space, we rotated starting and goal states across three configurations. During these different configurations, the agent attempted to move from one corner of the state space shown in Fig. 9 a clockwise to the next corner. Max trial length was 200 time units (4000 timesteps at dt = .05), while the agent could remain stuck for a maximum of 15 time units before exploring. Though the environment was significantly larger than the six-state grid-world, the model quickly learned the state space and required actions. This is illustrated in Fig. 9 b. These results show GOLSA agents could readily learn the tower of Hanoi task, and do so using state space knowledge rather than storing an explicit policy.

There is one major way in which GOLSA diverges from human performance on this task. Humans performing this task will often make moves that create a configuration that is

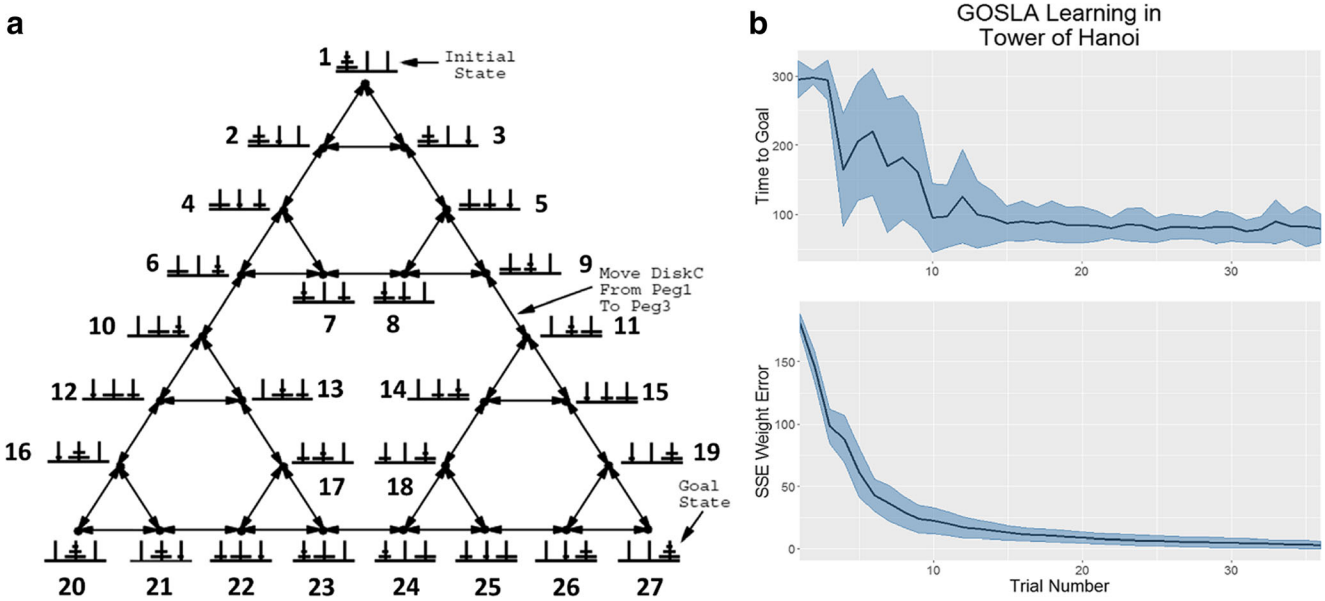

Fig. 9 **a** Graph representation of the Tower of Hanoi state-space. States are numbered according to their representation in the model (e.g., state 22 is represented by the 22nd unit in *current state*, *adjacent states*, etc. Figure adapted from Knoblock (1990). **b** Top: Average time taken to the goal as a function of the number of trials since learning began. The learning process was repeated 50 times and the shaded area represents the standard deviation across those 50 trial sequences. Bottom: Total sum squared error of the learned weights (actual – ideal) in the three key projections described above

visually like the goal but is actually many steps away from the goal in state space topology (Welsh et al. 1995). This does not emerge in GOLSA because the agent only learns the topology and its viable transitions. Humans likely represent the problem space in multiple formats, including a visual representation of the goal in addition to a cognitive map representation like GOLSA (Goel & Grafman 1995). The true and visual representations clearly imply two different distances from a state to a goal in the Hanoi task. The visual representation causes humans to make sub-optimal moves towards the goal (Welsh et al. 1995). GOLSA's focus on planning could be brought into closer alignment with human sub-optimal behavior by the addition of a visual layer representation that competes with the internal state space map during planning.

Latent Learning, Revaluation and Replanning Tasks

The GOLSA model is also able to produce key aspects of novel action or replanning behavior in humans or animals that has been used as evidence for planning using both cognitive maps and model-based behavior. We simulated GOLSA on a suite of tasks and environment that were inspired by Tolman's original studies and others (Russek et al. 2017; Momennejad et al. 2017) examining the capacity of model-based RL or the successor representation (SR) to support re/planning. The map used in these simulations is shown in Fig. 10. It contained 22 viable states. The network architecture and settings were identical to that for the Tower of Hanoi task.

In the simulations below, we focus on model behavior preceding and following different revaluation scenarios. Learning was performed akin to *latent learning* in typical cognitive map experiments (Russek et al. 2017). There was no goal or "reward" during this time, just exploration for structure learning. The agent explored the state space for 30 trials, with a max time of 300 timesteps per trial. The model took approximately the same number of trials here to learn the

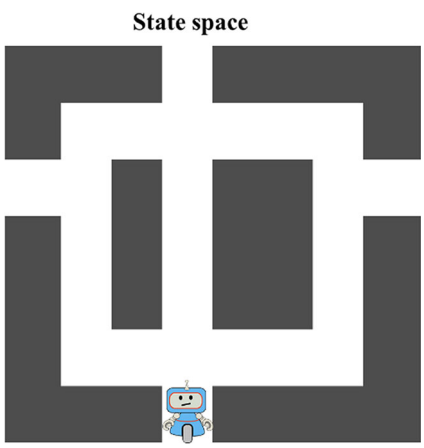

Fig. 10 Graph representation of the maze (top), which the agent explores to learn the structure independent of a goal

requisite layers, e.g., *adjacency,* as it did in the Tower of Hanoi.

Goal (Reward) Revaluation

The first task variant we considered using this maze was a form of reward or goal revaluation (Fig. 11). In this task, the agent first learns the map, then is notified a goal exists in a certain state. The agent readily approaches the goal (Fig. 11, left) on this first trial because the goal activates the gradient layer right away. After several trials, the goal location is switched (Fig. 11, right). Because GOLSA's planning is not linked to a particular action policy, the agent directly navigates to the new location and solves the replanning problem on the first trial following a goal switch. This is because replanning involves a change in goal, but no change in the environment. The model results recapitulate human's typical successes at solving this type of replanning task (Momennejad et al. 2017).

Transition Revaluation: Barrier Task

Another revaluation task variant that is useful for considering replanning mechanisms is when the structure of the environment changes. This is often done by impeding the typical travel path from start to reward. The agent first learns and then solves going to the goal position. It always chooses the shortest path, through the middle hall, in this scenario. By blocking the path on the 31st trial, the agent must learn the transition is no longer viable and must replan on the next trial. The model exhibits the human capacity to exhibit zero-shot revaluation in these tasks. GOLSA can solve the replanning problem immediately. This was achieved by nulling the weights of the blocked state in the adjacency and gradient layers to its connecting states. This is a simplification of one-shot learning (i.e., long-term depression at a synapse) that a particular state transition is no longer possible. Changing this mapping leads the agent to readily solve the replanning problem by taking a different path to the goal (Fig. 12, right side). While GOLSA can solve this like humans, RL formulations require a model-based agent to recompute an SR online (for example, see Russek et al. 2017). Our finding suggests that one manner humans may solve these tasks is by representing the whole state space, rather than just those states linked to a particular action policy.

Policy/Goal-Competition Revaluation

A challenge for several models of planning and learning, particularly in using the SR, is the learning is on-policy. This means the map of expected occupancies through the SR is biased by the previous action policy developed during learning and does not reflect the "optimal" actions. This issue stymies their ability to solve a type of replanning task dubbed

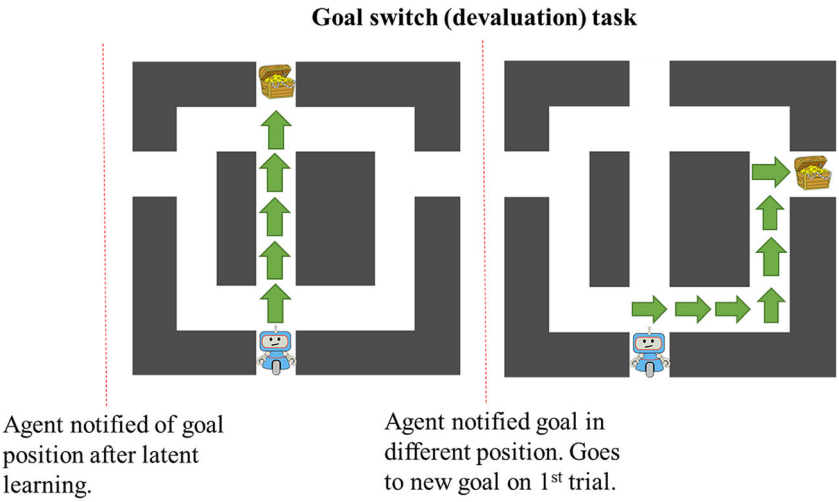

Fig. 11 Goal revaluation (switching) task. In both left and right maps, the green arrows depicting policy taken by agent in > 95% of trials. In the left map, agent approaches goal placement. The right map depicts the agent action on the first trial after the goal switches

"policy revaluation" (Russek et al. 2017). This task involves learning original goals or rewards are in a given state (Fig. 13 left), and later a new goal or reward is introduced at a new state and has a higher value. The expectation is if the agent represents the whole state space and acts optimally within it, they should pursue the more rewarding goal. Humans are often capable of solving these tasks and often make choices to progress them towards the new, more rewarding goal state (Momennejad et al. 2017). Here, we refer to it as goal competition because GOLSA does not learn specific or cached policies. GOLSA can solve these problems readily because rewards or goals are separated from the state-space map and are divorced from a specific policy. This engenders GOLSA agents to traverse the space using the goal-gradient rather than following an on-policy state expectation such as the successor. This result is shown in Fig. 13, indicating the agent navigated towards the higher value goal in 70% of revaluation trials. Thirty percent of the time the agent still went towards the lower value goal. This result repeats the fact that enacting choices towards a goal is driven both about value and proximity to that goal. Mechanically, this happens through competition and WTA dynamics in the goal layer. The "larger valued" goal at certain input ratios effectively inhibits the "smaller valued" goal, leading the agent to enact a goal-focused plan. The effect of this competition on goal-layer representations can be seen in Fig. 13 (top activity trace). This indicates the model solves the problem not solely in a greedy manner. Agent actions and pursued goals are determined by both relative weight and distance of goals during this revaluation. These results recapitulate the important point that humans do not always pursue the most rewarding state.

GOLSA Engages State-Specific Drives Through Goal Competition

The demonstrated model capacity, thus far, show GOLSA's flexibility to plan and solve replanning problems in a way like humans. These components scaffold a model framework to examine how planning might occur with multiple, time-dependent, and competing drives that propagate down to goal-state selection. With multiple competing drives that are satisfied in different states, agents must deal with an

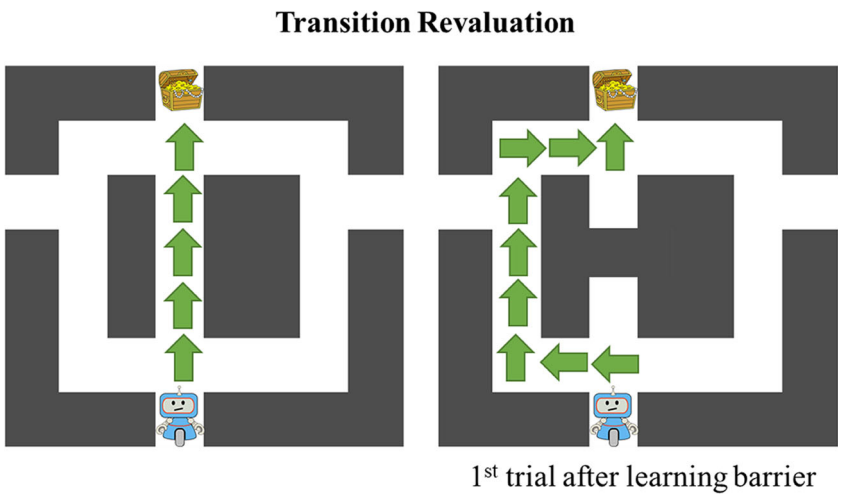

Fig. 12 Transition revaluation task. In both left and right maps, the green arrows depicting policy taken by agent in > 95% of trials. In the left map, agent approaches goal placement. The right map depicts the agent action on the first trial after they learn the state space changes with inclusion of a barrier

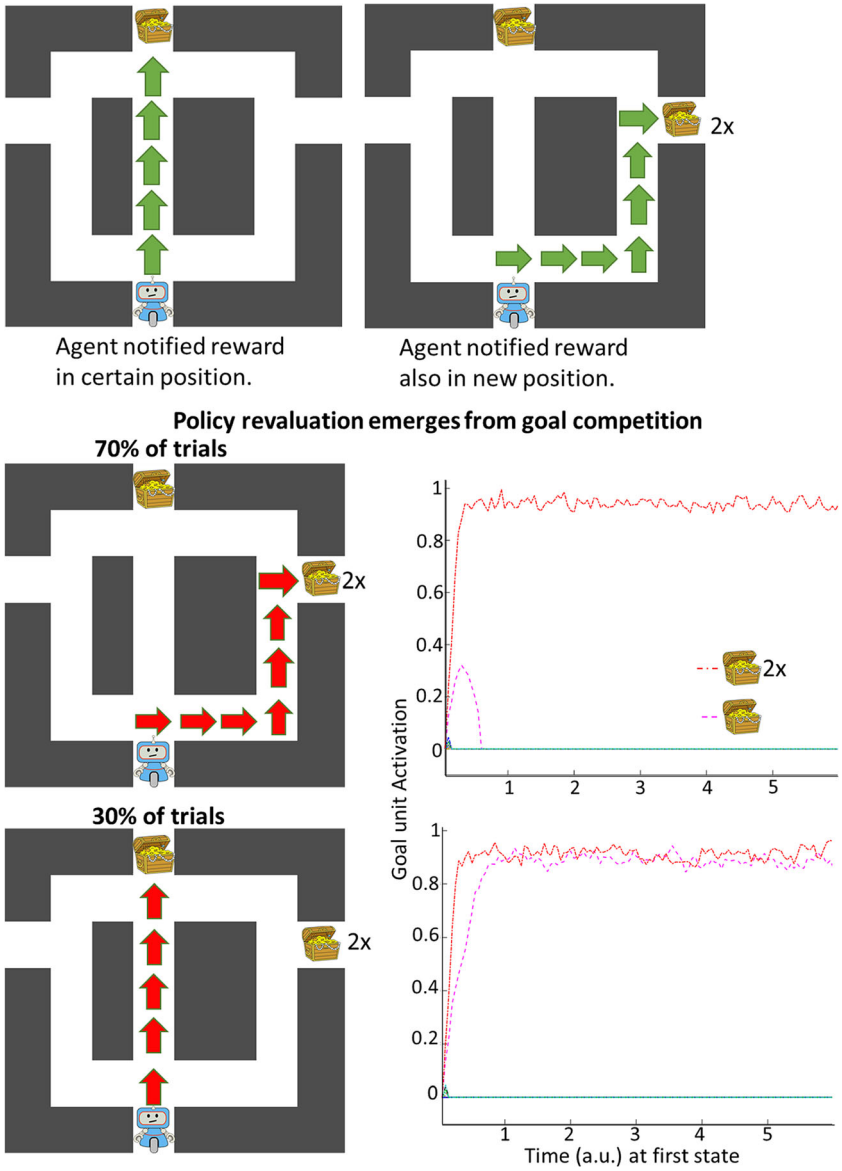

Fig. 13 Policy revaluation task. In both left and right upper maps, the green arrows depicting policy taken by agent in > 95% of trials. In the left map, the agent approaches the original goal. The right map depicts the agent action on the first trial after they learn a larger reward is in a new location. The bottom two maps indicate the policy taken by the agent 70% (top) and 30% (bottom) of the time when both rewards are offered together. The right traces show the representative activation plots of the goal layer units on trials corresponding to 70% actions towards larger reward (top) and 30% actions towards smaller but closer reward (bottom)

interdependent planning problem. For example, an agent with a pressing drive (e.g., hunger) may need to navigate towards a goal state that puts them more distant from other goal-fulfilling states. Thus, proper planning requires agents to not act greedily or merely approach the closest goal-state.

The GOLSA framework achieves a mapping of drives to goals to affect planning through an additional *drives* layer. The agent learns that different drives map on to different *goal-states* the same as other simulations (Fig. 14). In effect, certain drives can only be satisfied by visiting those states (Fig. 14). The *drives* layer follows a function that accumulates activity to a threshold, which is suppressed once that goal state is reached (Hull 1943). The *drives* layer does not have competition. The motivation for this, by example, is that active drives such as hunger and desire for entertainment can co-exist without necessarily suppressing one another. Planning decisions about which drives to pursue reside in the interaction of *drives* to *goal state* and *goal gradient*. Specifically, the *drives* specify the need and gradient specifies the 'distance' to the goal state. GOLSA uses this interaction to select the order of which drives to fulfill. Therefore, GOLSA does not act greedily regarding distance to goals.

We demonstrate GOLSA's ability to exhibit drive to goal-based planning through 4 simulations. In both sets, the state-space was the same as the above replanning tasks. In these simulations, we consider both the relative starting level of different drives and the agent starting position. This combination was done to demonstrate how GOLSA weights the drive, goal state, and gradient interaction to pursue different goals. The simulations included 3 states in the *drives* layer. The drive layer had as many units as there were drives. These were initially connected all-to-all to goal layers for learning (see

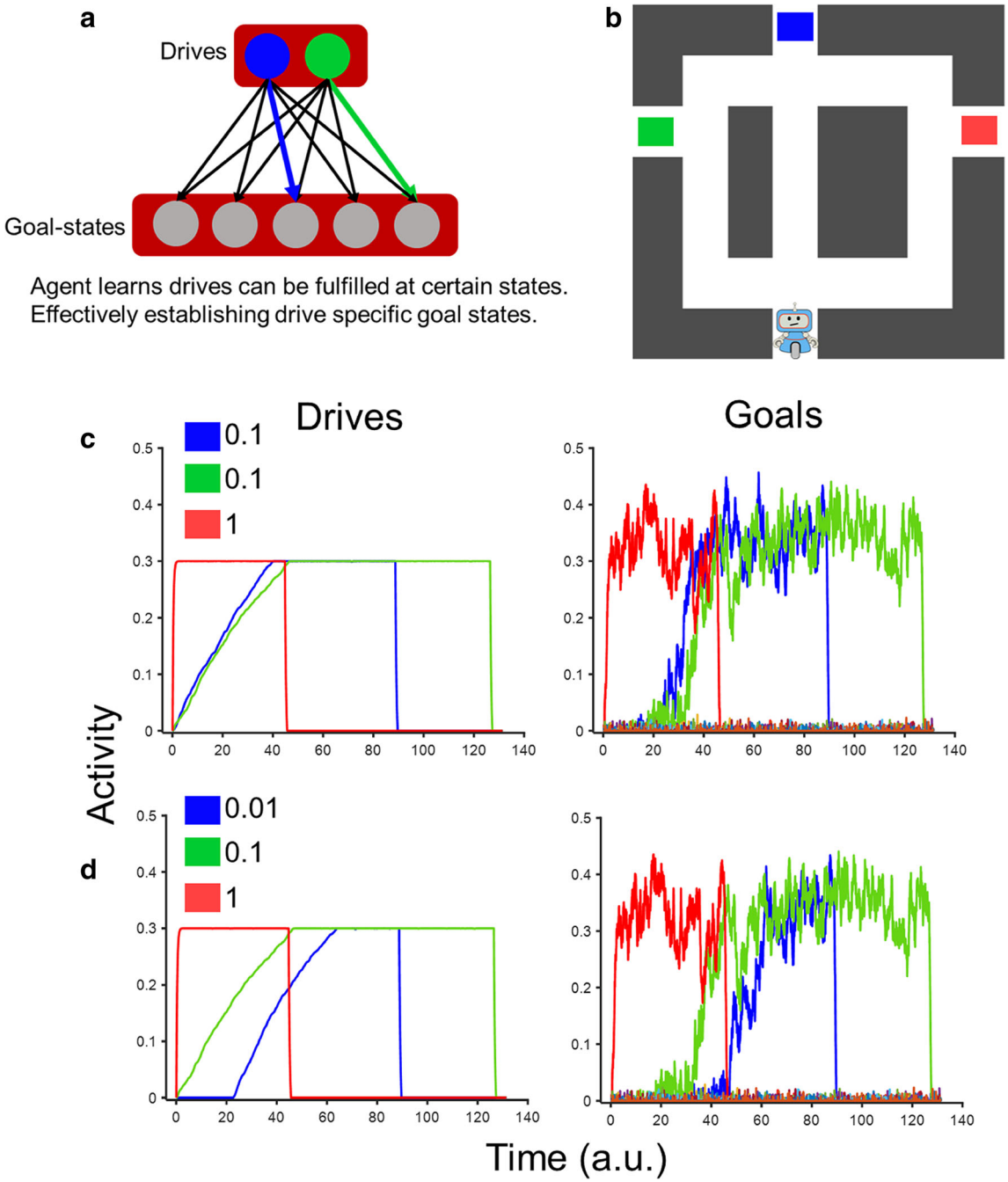

Fig. 14 Overview of drives linked to goal states and how it affects planning. **a** Example network connectivity between *drives* and *goal-state* layers. **b** Layout of environment and drive to goal states agent had to learn through exploration and reinforcement. **c** Representative activity trace in *drives* (left) and *goal-state* (right) layers when drives two drives started off at same level and the other larger. **d** Representative activity trace in *drives* (left) and *goal-state* (right) layers when all drives started off at different levels. **c** and **d** show how the relative drive level, agent position, and distance to those drives changes planning order. In activity traces, when a goal was reached, the activity in the *drives* layer was suppressed that inevitably removed excitation to the *goal-states* layer

supplementary materials). The agent learned the mapping of *drives* to *goal states* through a latent learning process similar to the above "Tolman" map replanning tasks (Fig. 14).

The first 2 simulations placed the agent at the same starting position as previous replanning simulations (see Fig. 14). In the first simulation, the closest (blue) and 2nd closest (green) drives were set to the same lower starting value (0.1). The furthest drive state (red) was set to the highest. The order of pursued goals is seen in Fig. 14 activity traces. When a goal was achieved, a drive layer activity became suppressed and the agent went on to other goal-fulfilling drive. Essentially, the agent went to the most distant but pressing drive goal-state

first (red). The agent then traversed to the blue and green coded states in that order. The agent behavior with all drives starting off at different levels is seen in the 2nd simulation (Fig. 14d), with the closest (blue) state drive starting at the lowest level (Fig. 14). Thus, given the agent starting distance to goals and their drive states, it planned towards the furthest goal (red) first. These simulations show how high drives move the GOLSA model to flexibly plan first towards distant states in lieu of close goal-states that will not fulfill pressing drives.

A second set of simulations (Fig. 15) was done to further demonstrate how the interaction of drive and goal distance influence planning behavior by changing the agent start state (Fig. 15a). The agent started in a middle state. The first simulation involved the blue and green drives starting at the same value. Unlike the first simulations, the agent distance to goal states and drive values caused the agent to move towards the blue and green states first (Fig. 15b, c). By manipulating the relative green and blue starting drive states, we more clearly demonstrate the agent will pursue a path of goal states that is driven by the interaction of drive need and distance. Even though the red drive was highest in both these simulations, the agent holds off the highest drive till the end (Fig. 15b, c). Together, these *drive*-based simulations show how the GOLSA model flexibly plans in a goal or drive directed manner that respects the topology of the state space and placement of drive fulfilling states. GOLSA inherently solves a cost driven planning problem through interaction of drive level and goal distance.

Discussion

Here, we present a biologically plausible solution to goal-directed planning through a localist neural network named GOLSA. The model uses biologically realistic rate equations, with online plasticity driven solely by local information as opposed to dedicated learning periods (Guerguiev et al. 2017). Learning occurs through Hebbian principles, modified by eligibility traces and oscillatory dynamics. GOLSA acquires an environmental map to plan towards goals through a backwards spreading activation mechanism. GOLSA can navigate simple environments, abstract environments like Tower of Hanoi, while producing flexible planning and replanning towards any goal. GOLSA can also solve a host of other revaluation tasks similar to human's capacity to replan in drive- or reward-dependent manner. GOLSA delivers this flexible control through two key components. First is learning state space topology separate from rewards, goals, or actions. Separating representations enables GOLSA to transfer the representation of the state-space to solve novel

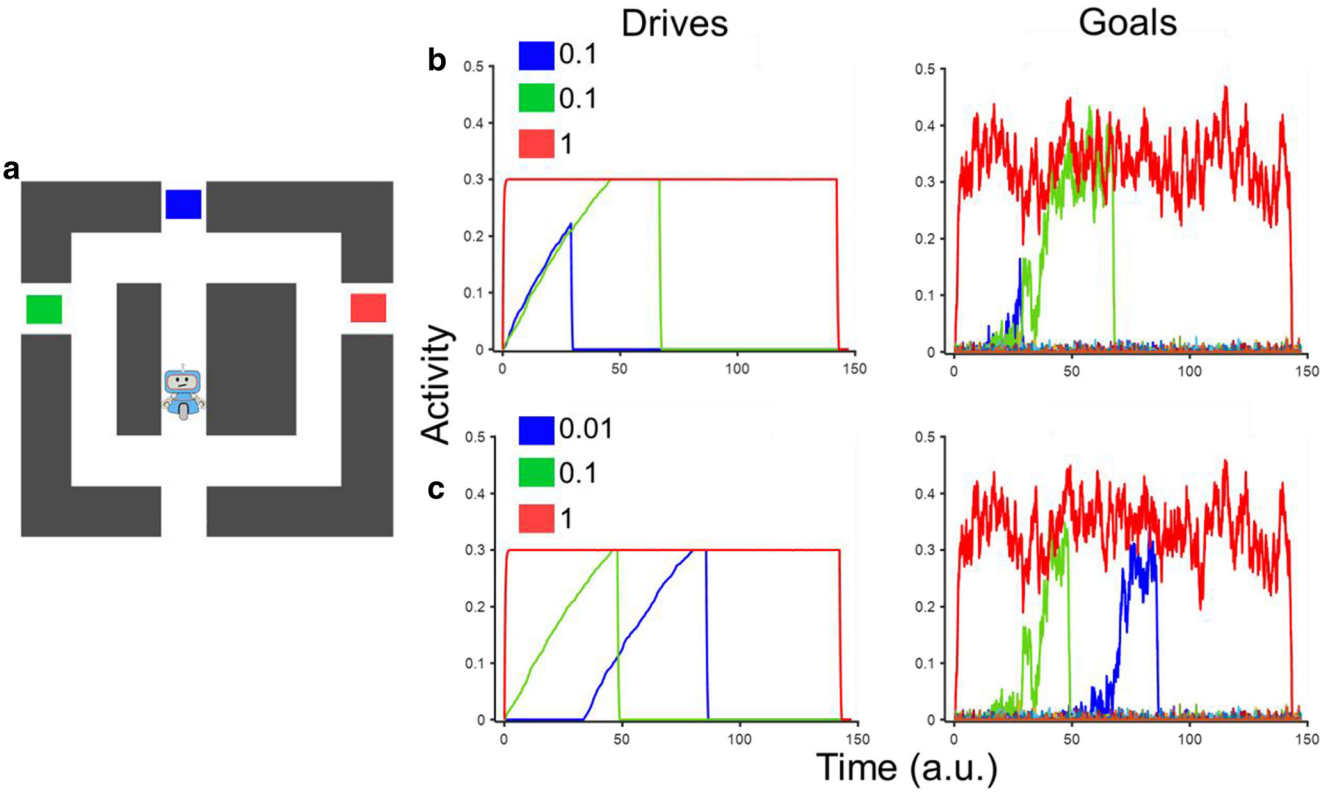

Fig. 15 **a** Layout of environment with agent position changed compared to first drive to goal simulation. **b** Representative activity trace in *drives* (left) and *goal-state* (right) layers when drives two drives started off at same level and the other larger. **c** Representative activity trace in *drives* (left) and *goal-state* (right) layers when all drives started off at different levels

problems. The second component, supporting planning, is the hill-climbing approach in which an agent ascends a gradient distributed across state-space with a peak at the goal.

Goals and Drives

A core GOLSA component was addressing how a motivated, goals-first neural system could operate. We propose that the model's design of desired goal states exhibiting a backwards driving effect on behavior accords with how goal-directed planning is conceptualized psychologically (de Wit and Dickinson 2009), with internal drives consistent with homeostatic set-point regulation (Hull 1943; Toates 1986) and drive to states associates formed through reinforcement. The brain must first assess motivations or drives from essential homeostatic needs. These drives are then prioritized to then set goals. Importantly, this contrasts with the RL model focus of planning based on maximum long-term reward from a given state (Sutton and Barto 2018). Effectively, GOLSA decides between which goals to pursue and plans using competition between goal states. The competition process is influenced by interactions of drive setpoints and goal distance. Combining this interaction with lateral inhibition to implement winner-take-all competition in the drives layer allows GOLSA to pursue goal states more out of the way if the drive for that state is higher. This captures another aspect of behavior not captured by RL models. Animals or humans will pursue states that fulfill high drives, while avoiding other states with higher but unneeded reward offers (Balleine & O'Doherty 2010). Interactions of goal distance and drives also reproduce the observed behavior that animals will also plan a "route" of actions based on the combined distance and pressing need of a drive (Salamone and Correa 2012). This emergent planning essentially captures anticipatory behavior exhibited by living agents that trade-off drive need and goal proximity. GOLSA does not act greedily by merely approaching the largest offer but is motivated to pursue the most pressing drive given its proximity to other goal states.

The issue of multi-dimensional motivated drives is a notable problem for RL models, as they have a singular objective focus on maximizing long-term reward regardless of the agent's internal state. Therefore, planning within RL frameworks do not sufficiently explain goal-directed human and animal tendencies to ignore high-offers in favor of lower-offer but needed states. GOLSA captures this effect through interaction of drive set-point, goal distance, and state space topology. RL frameworks were recently extended to capture motivational and homeostatic drive effects (Keramati and Gutkin 2014). This model proposes RL reward mechanisms are equated with physiological distance to satiating an agent's drive. This effect of reward as homeostatic drive setpoint distance is like our drives layer. Unlike GOLSA, though, a biological implementation of the RL model (Keramati and Gutkin 2014) remains undetermined. Additionally, the RL model relies on several post hoc parameterizations that are biologically undetermined (e.g., distance weighting) and undermine its putative normative claims.

Algorithmic and Biological Differentiation

Algorithmically, GOLSA is most similar to solving optimal control through forward-backward algorithms underlying linear Markov decision problems for model-based control (Todorov. 2009). Essentially, GOLSA *adjacent-state* activity constitutes a one-state forward sweep from the current state. This interacts with the backward process that is analogous to a parallel sweep propagating information from all goals across all paths. Biologically, this mechanism recapitulates findings of parallel activations in prefrontal cortex (PFC) encoding sequential action plans (Averbeck et al. 2002; Mushiake et al. 2006) and future goal-states in primate PFC (Genovesio et al., 2012a; Yamagata et al. 2012). GOLSA's *desired next state* layer approximates the soft-max solution to these forward-backward interactions. Lateral inhibition strength in this layer implements a control gain like a soft-max temperature, with gain controlling trade-offs between forward information and the states linked to goals. Therefore, unlike typical RL applications, GOLSA is softly biased towards the random action policy implied by the adjacency. The goal-gradient interaction works against this bias, with the efficacy determined by the strength of the gradient at adjacent states and degree of lateral inhibition. This type of trade-off regularization is reminiscent of the notion of cognitive control driving actions in the face of automatic biases (Botvinick and Cohen 2014). An important future question is how this meta-control problem of setting this gain trade-off parameter is driven at a normative and neuromodulator level. The multiplicative nature of control gain suggests a neuromodulatory, possibly dopaminergic, mechanism (Wunderlich et al. 2012). Speculatively, if dopamine modulates this trade-off, this predicts dopaminergic encoding of variables governing control motivation (Berke 2018) distinct from the classic reward prediction error (Roesch et al. 2007; Schultz et al. 1997); this is consistent with the idea that findings that dopaminergic manipulations typically alter task performance variables such as cognitive effort (Westbrook et al. 2020), increases in model-based planning (Wunderlich et al. 2012) and action vigor (Beierholm et al. 2013). In GOLSA, this explanation emerges because goal "value" is already conveyed by the backward gradient, This component of GOLSA delivers a connection to human behavior not explicated by standard MBRL solutions: a biological control mechanism that allows task- and individual-level tuning of the trade-off between prior state expectations and goal-driven behavior.

At an algorithmic level, the Todorov (2009) formulation was also recently adopted to explain several findings about

human behavior in revaluation tasks and cognitive control tasks (Piray and Daw 2020); the model was referred to as linear RL to note the change in parameterization of the Bellman equation introduced by Todorov (2009). Piray and Daw (2020) show the algorithm indeed solves the problems like GOLSA. However, they use a representation of state visitation expectation based on a prior or default policy—similar to a successor—which they call the default representation. GOLSA instead relies on an adjacency rather than state expectation, and GOLSA offers an answer to biological implementation regarding planning through backwards goal propagation. This is an imperative component of planning not addressed in Piray and Daw (2020) or the original work (Todorov 2009). An important issue is future empirical validation of differences in proposed representational formats of states.

GOLSA along with some RL models can complete the revaluation tasks considered here. These tasks are useful because they provide a benchmark comparison between models and human or rodent behavior and have been used to elucidate whether hybrid SR and other RL mechanisms can achieve them. For example, Russek et al. (2017) used the revaluation tasks to test standard SR models, SR-MB, and even SR-DYNA variants that use "offline" model-based replay to plan. In Table 1, we provide a checklist comparing human capacity for solving these tasks with GOLSA and RL models employing an SR alone to SR hybrid variants, and full standard model-based RL with dynamic programming.

The table points out that humans and animals can solve these revaluation tasks, as can GOLSA, SR-DYNA, and full MBRL. Despite using a transition model, the SR alone or SR-MB combinations cannot solve all of these because its expected future states are policy dependent. However, policy revaluation can be accomplished by using a weighted combination of pure MBRL and an SR (Momennejad et al. 2017). This combination was shown to reproduce the fact that in experimental settings, humans only successful revaluated choices under policy changes approximately 50% of the time. Pure MBRL would predict close to perfect revaluation, unlike humans. Even though GOLSA has no SR and is more akin to model-based, it also exhibited a human like trade-off at 70/30 using current parameters (Fig. 13). Notably, this happens without an explicit SR and predicts that this trade-off could emerge from inhibitory goal-competition rather than differing MBRL and SR representations. This does not rule out the utility or viability of an SR. An SR could provide agents with a computationally inexpensive prediction of how to act in each state without planning in a full world model. Additionally, an SR could be added into GOLSA by amending the adjacency weight learning with local rewards and a change of learning law. Changing the adjacency representation to an SR with a biased policy suggests GOLSA could also capture reward-dependent planning biases (Momennejad et al. 2017). SR-Dyna achieves flexibility akin with MBRL and GOLSA by performing offline replay of previous trajectories to update an SR using both states and actions. An important caveat is that SR-DYNA must perform substantial replay events between each decision to produce these flexible evaluations. It remains an open experimental question as to what extent and when online planning in GOLSA (or MBRL) versus offline replay updates in DYNA underlie human and rodent brain planning.

A clear commonality among RL models successful across all these revaluation tasks is they are off-policy or approximate the optimal policy. GOLSA is also "off-policy" and approximates optimal, but not in the traditional RL sense of learning a policy through online or offline optimizing of Q-values. Because GOLSA uses lateral inhibition and self-feedback inhibition in the desired next state dynamics, this approximates a soft-(max)optimality decision function. The soft-optimality of GOLSA is a major way it differs from

Table 1 Comparisons of GOLSA with SR model variants, model-based RL, and human performance on canonical learning and choice tasks, whether they work with multiple goal decisions, and their on- or off-policy learning

Task	GOLSA	SR-MB	SR-TD	SR-DYNA	Model-Based	Humans
Latent Learning	✓	✓	✓	✓	✓	✓
Reward Revaluation	✓	✓	✓	✓	✓	✓
Transition Revaluation	✓	✓	X	✓	✓	✓
Policy Revaluation	✓	X	X	✓	✓	✓
Multi-Goal/Drive	✓	X	X	X	X	✓
On-Policy Off-Policy	Off	On	On	Off	Off	

MBRL and SR-DYNA as well, where the latter two use hard maximization to choose a Q-value (MBRL) during value iteration or offline model-based simulations (SR-DYNA; Russek et al. 2017; Sutton and Barto 2018). This soft-optimality may offer a biological, architectural explanation for why humans and animals may not always take the maximizing state-action predicted by hard-maximizing MBRL or DYNA methods. Soft-optimality balances the desirability of a next state over the influence of all states in a way that encourages the agent to explore when the overall desirability is not high, but exploit when it is. This is clearly seen when multiple drives are considered. To see this, note that a low drive and high drive could render similar differences in activation between desired next states but their absolute within unit activation will differ and so will the effect of stochasticity and competition in choice.

To clarify GOLSA differences from other models, in Table 1, we note that GOLSA is differentiated from other RL formulations in terms of how it handles multiple goals or drives. For clarity, the RL models compared to GOLSA, as well as Keramati and Gutkin (2014), do directly handle multi-drive competition by treating them as homologous with the reward function. However, GOLSA differs in that any state can be considered a goal, regardless of reward, neutral, or aversive outcome. GOLSA's variant of reward-independent goal-state setting also accords well with common understandings of ecologically grounded notions of reward: it is sparse, and goals are often set without any reward per se.

These algorithmic connections clarify that GOLSA has overlap with MBRL, while differing in how optimality emerges. Relevant differences for understanding neural representations supporting planning also exist between how GOLSA, Todorov (2009), and standard MBRL solutions solve for actions with respect to state transitions. MBRL uses symbolic actions (e.g., left or right) through the transition probability (p(s'|s, a)), comparing all actions in a given state Q(s,a). Todorov's (2009) solution represents the optimal transition as p*(s'|s). This approach equates optimal transition with optimal action by removing symbolic action. However, this equivalence means planning actions with stochastic transitions are not viable without amendments. In contrast, GOLSA's approach rests on first solving for the desired next state (s'). This layer's mutual inhibition and WTA dynamics approximate a soft-max state selection similar to p*(s'|s). GOLSA diverges from Todorov (2009) by determining which explicit symbolic action will likely elicit that desired transition (p(a|s',s)). This allows extensions of GOLSA to map actions probabilistically to stochastic transitions. GOLSA could be adapted to plan with stochastic transitions by changing learning laws to allow bidirectional weights changes in the adjacency, gradient, or desired transition layer weights with respect to action weights. This issue is not critical for many tasks like navigation where the relation between control and transition is deterministic.

Comparisons between predicted representations in GOLSA and MBRL also speak directly to biological representation of goal-directed control and planning. These models predict different information carried by neural units and synaptic weights state transition and linked action representations (Behrens et al. 2018). MBRL predicts cells representing (s,a) conjunctions with transition probabilities in synaptic weights, with the conjunction indexing the joint conditioned probability in the transition matrix p(s'|s, a). GOLSA units represent desired state-transition conjunctions (ss') and action selection as p(a|s,s'). Indeed, application of Bayes rule shows the MBRL representation is not required to solves these problems as the transition matrix partitions into the GOLSA representations of p(a|s,s') and p(s'|s) normalized by p(a|s). The question is whether there is stronger biological evidence in for either type of representation that could further validate either model. Several studies have established strong links between hippocampal 'map' representations and prospective planning activity supported by rodent and human place cells (O'Keefe and Nadel 1978; O'Keefe et al. 1998; Ekstrom et al. 2003). GOLSA representations of desired transition conjunctions accord with a hippocampal prospective representations and longstanding propositions that place cell connections represent state transitions (p(s'|s); Poucet et al. 2004) or the cells could themselves encode conjunctive transitions (ss'; Gaussier et al. 2002). The imperative is these prospective representations are independent of action encoding. Growing evidence provides support for the latter in hippocampal place cells, indicating that units encode expected state transitions like an SR or ss' conjunctions (Stachenfeld et al. 2017). We take this as evidence that GOLSA's representation for solving goal-directed state transitions more closely aligns with hippocampal place cell patterns than MBRL.

Turning state transition activity into action selection could occur through hippocampal projections to ventral striatum (shell) via transmission of state transition information. We suspect that synaptic weights bridging hippocampal-striatal interactions could support GOLSA's proposed action selection (p(a, s', s)) process. The evidence for this is based on fMRI showing striatum encodes model-based predictions (Daw et al. 2011), and other fMRI (Guitart-Masip et al. 2012), electrophysiology (Lau and Glimcher 2008), and causal manipulation of striatum (Tai et al. 2012) indicating striatal neurons encode action values and learning.

Dual-roles, Oscillations and Informational Gating

Developing biological models that capture how both learning and action are represented are faced with a *dual-roles* problem. For the same layers to participate in both planning and acting or learning online, these neurons need a mechanism to gate the projection of different input at different times. In GOLSA, for example, updating recurrent connection weights

in the *goal-gradient* uses goal layer input during planning and current state input during learning. Our solution to the dual-roles problem was motivated by studies showing cognitive processes such as working memory, attention, sensorimotor integration, memory retrieval versus encoding, and behavioral timing (Berger 1929; Caplan et al. 2003; Cavanagh and Frank 2014; Coon et al. 2016; Herrmann et al. 2004; Palva et al. 2010; Roux and Uhlhaas 2014) are putatively linked to oscillatory gating and inhibition mechanisms.

GOLSA uses an oscillatory gating mechanism as a solution to open and close pre- to post-synaptic projections based on oscillatory phase. Effectively, this controls the temporal window for updating new information versus maintenance of information necessary for actions. This phase-based oscillatory coupling mechanism is indeed found in humans and animals as cross-frequency coupling (Canolty and Knight 2010; Roux and Uhlhaas 2014). Furthermore, our utilization of oscillatory gating supports current theorizing that suggests cross-frequency coupling underwrites neuronal communication via phasic windows (Bonnefond et al. 2017).

We also hypothesize that if the dual-roles issue is indeed a problem, then neurons in prefrontal cortical (PFC) regions implicated in goal-directed control (Genovesio et al., 2012b; Passingham and Wise 2012) should encode separate task phase information. There is indeed evidence that the same neurons activate in planning and action execution phases of a task (Mushiake et al. 2006). Additionally, core variables underlying goal-based planning are found distributed across PFC regions such as orbitofrontal and anterior cingulate (ACC) or dorsolateral PFC which have been shown to represent planning involving world models (Schuck et al. 2016), predicted future states (Wimmer and Büchel 2019), and inferring what goals are valuable (Jones et al. 2012) and distance to goals (Juechems et al. 2019). Physiologically, these same PFC areas are also exhibit cross-frequency coupling across the cortex (Voloh et al. 2015). The prediction then is that GOLSA computations involving gating such as goal-gradient and their interaction with drives should be represented across PFC regions and the hippocampus, especially the orbitofrontal cortex. Empirical evidence showing distance to reward based on spatial goals is encoded in the orbitofrontal and hippocampus supports this intuition (Wikenheiser et al. 2016; Gauthier and Tank 2018).

Other proposals for how neural circuits solve planning tasks with world models have aimed to implement MBRL directly, while altogether sidestepping the key oscillatory gating and inhibition mechanisms in GOLSA. A core example is Friedrich and Lengyel's (2016) spiking neuron network realization of model-based RL. Their neurons represented state-action (SA) conjunctions and synaptic weights representing transition probabilities (T(s, a, s'). However, they note that the same neurons involved in planning must later be activated during action akin to the dual-roles operation of GOLSA's goal- or current-state to gradient projection. Their network, though, has no plausible method of determining its past and current state but knowledge of the current state is required in action selection. Their model relies crucially on external, nonneural signals about critical information for learning and task phases making the underlying neural processes supporting their model biologically uncertain. Additionally, Friedrich and Lengyel (2016) argued that goal-gradient spreading activation models that encode goal distance like ours could not handle cases with multiple goals and re-valuation. However, we show this mechanism works during several types of replanning and for decision-making with multiple goals or time-dependent drives. Empirically, there is evidence that the ACC and OFC represent distance to goals similar to our backwards gradient spreading activation (Juechems et al. 2019). This is further bolstered by indications that OFC represents a cognitive map that may be prioritized towards goals or valuable states (Wikenheiser et al. 2016) These comparisons suggest that GOLSA's representations, utilization of oscillatory gating, and backwards goal-gradient mechanism align better with current neurophysiology and imaging compared to MBRL network models.

Notably, GOLSA is not the first model to deal with the dual-roles problem. The Leabra framework (O'Reilly and Munakata 2019), like the GOLSA model, uses a two-phase system to generate activation and then learning signals for updating the network weights. Additionally, GOLSA utilization of oscillations differs from Leabra because it uses them for more than a clocking of phases, but also acts to gate the inflow of information between layers, e.g., switching between actual vs. desired state transitions. A takeaway from both ours and the Leabra framework providing solutions to the dual-roles problems suggests an important biological constraint that should be considered in assessing the biological plausibility of algorithmic models.

Extensions to GOLSA

In its present form, the model can only navigate discrete state spaces. One solution to extend GOLSA's capacity is to tessellate the continuous space into discrete regions or future GOLSA iterations could employ a distributed representational frameworks or function approximation that offer a superior level of capacity without needing to expand the number of states (Eliasmith and Anderson 2003). Another issue is large state spaces can distort the gradient, which is necessary for model function. For example, in big spaces, the unit activity from interconnected units will become a lot smaller given the representations of firing span 0 to 1. This closeness of activity in close states makes differentiating connections susceptible to noise. One solution is solution is building a hierarchical structure or temporal abstractions such as options in RL (Sutton et al. 1999). This would allow planning over sub-tasks with

their own gradient and would drastically reduce the state-space size.

Because only positive weight changes were learned in the goal-gradient layer, this impedes GOLSA easily adapting to changing environmental topology and planning with stochastic transitions. In the results, we focused on replanning under changes, and instituted environment changes by zeroing out gradient and adjacency weights for blocked states. However, an online solution is to signal the difference between desired and actual state transition representations and send that signal back to the state representations in the goal-gradient and adjacent-states layer as an active forgetting signal. Despite future improvements, GOLSA provides a model framework goal-directed planning and control that is algorithmically motivated and unique, provides a biologically feasible systems-level explanation of human planning behavior and mechanisms that RL models have yet to deliver, and maps onto expected representations of single neurons in hippocampal and striatal model-based control networks.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s42113-020-00095-7.

Acknowledgments We thank A. Ramamoorthy for helpful discussions on the manuscript.

Funding JWB was supported by NIH R21 DA040773.

References

Alvernhe, A., Save, E., & Poucet, B. (2011). Local remapping of place cell firing in the Tolman detour task. *The European Journal of Neuroscience, 33*, 1696–1705. https://doi.org/10.1111/j.1460-9568.2011.07653.x.

Attias, H. (2003). Planning by probabilistic inference. In *Proc. of the 9th Int. Workshop on Artificial Intelligence and Statistics*.

Averbeck, B. B., Chafee, M. V., Crowe, D. A., & Georgopoulos, A. P. (2002). Parallel processing of serial movements in prefrontal cortex. *Proceedings of the National Academy of Sciences of the United States of America, 99*, 13172–13177. https://doi.org/10.1073/pnas.162485599.

Badre, D., & Nee, D. E. (2018). Frontal cortex and the hierarchical control of behavior. *In Trends in Cognitive Sciences., 22*, 170–188. https://doi.org/10.1016/j.tics.2017.11.005.

Balleine, B. W., & O'Doherty, J. P. (2010). Human and rodent homologies in action control: corticostriatal determinants of goal-directed and habitual action. *In Neuropsychopharmacology., 35*, 48–69. https://doi.org/10.1038/npp.2009.131.

Behrens, T. E. J., Muller, T. H., Whittington, J. C. R., Mark, S., Baram, A. B., Stachenfeld, K. L., & Kurth-Nelson, Z. (2018). What is a cognitive map? Organizing knowledge for flexible behavior. *In Neuron, 100*, 490–509. https://doi.org/10.1016/j.neuron.2018.10.002.

Beierholm, U., Guitart-Masip, M., Economides, M., Chowdhury, R., Düzel, E., Dolan, R., & Dayan, P. (2013). Dopamine modulates reward-related vigor. *Neuropsychopharmacology., 38*, 1495–1503. https://doi.org/10.1038/npp.2013.48.

Berger, H. (1929). Uber das Elektroenkephalogramm des Menschen. *Archiv Fur Psychatrie, 87*, 527–570.

Berke, J. D. (2018). What does dopamine mean? *In Nature Neuroscience., 21*, 787–793. https://doi.org/10.1038/s41593-018-0152-y.

Bertsekas, D. P. (2010). Dynamic programming and optimal control 3rd Edition , Volume II by Chapter 6 Approximate dynamic programming approximate dynamic programming. *Control., 10*(1), 1.141.6891.

Bonnefond, M., Kastner, S., & Jensen, O. (2017). Communication between brain areas based on nested oscillations. *ENeuro., 4*, ENEURO.0153–ENEU16.2017. https://doi.org/10.1523/ENEURO.0153-16.2017.

Botvinick, M. M., & Cohen, J. D. (2014). The computational and neural basis of cognitive control: charted territory and new frontiers. *Cognitive Science, 38*, 1249–1285. https://doi.org/10.1111/cogs.12126.

Canolty, R. T., & Knight, R. T. (2010). The functional role of cross-frequency coupling. *Trends in Cognitive Sciences, 14*(11), 506–515. https://doi.org/10.1016/j.tics.2010.09.001.

Caplan, J. B., Madsen, J. R., Schulze-Bonhage, A., Aschenbrenner-Scheibe, R., Newman, E. L., & Kahana, M. J. (2003). Human θ oscillations related to sensorimotor integration and spatial learning. *Journal of Neuroscience, 23*(11), 4726–4736.

Cavanagh, J. F., & Frank, M. J. (2014). Frontal theta as a mechanism for cognitive control. In Trends in cognitive sciences (Vol. 18, Issue 8, pp. 414–421). https://doi.org/10.1016/j.tics.2014.04.012.

Coon, W. G., Gunduz, a., Brunner, P., Ritaccio, a. L., Pesaran, B., & Schalk, G. (2016). Oscillatory phase modulates the timing of neuronal activations and resulting behavior. *NeuroImage, 133*, 294–301. https://doi.org/10.1016/j.neuroimage.2016.02.080.

Daw, N. D., & Dayan, P. (2014). The algorithmic anatomy of model-based evaluation. *Philosophical Transactions of the Royal Society, B: Biological Sciences, 369*, 20130478. https://doi.org/10.1098/rstb.2013.0478.

Daw, N. D., Niv, Y., & Dayan, P. (2005). Uncertainty-based competition between prefrontal and dorsolateral striatal systems for behavioral control. *Nature Neuroscience, 8*, 1704–1711. https://doi.org/10.1038/nn1560.

Daw, N. D., Gershman, S. J., Seymour, B., Dayan, P., & Dolan, R. J. (2011). Model-based influences on humans' choices and striatal prediction errors. *Neuron., 69*, 1204–1215. https://doi.org/10.1016/j.neuron.2011.02.027.

Dayan, P. (1993). Improving generalization for temporal difference Learning: The Successor Representation. *Neural Computation*. https://doi.org/10.1162/neco.1993.5.4.613.

Dayan, P. (2009). Goal-directed control and its antipodes. *Neural Networks, 22*, 213–219. https://doi.org/10.1016/j.neunet.2009.03.004.

Dayan, P., & Berridge, K. C. (2014). Model-based and model-free Pavlovian reward learning: revaluation, revision, and revelation. *In Cognitive, Affective and Behavioral Neuroscience., 14*, 473–492. https://doi.org/10.3758/s13415-014-0277-8.

De Wit, S., & Dickinson, A. (2009). Associative theories of goal-directed behaviour: a case for animal-human translational models. *Psychological Research Psychologische Forschung, 73*, 463–476. https://doi.org/10.1007/s00426-009-0230-6.

Dickinson, A., & Balleine, B. (1994). Motivational control of goal-directed action. *Animal Learning & Behavior, 22*, 1–18. https://doi.org/10.3758/BF03199951.

Djikstra, E. W. (1959). A note on two problems in Connexion with graphs. *Numerische Mathematik*.

Dolan, R. J., & Dayan, P. (2013). Goals and habits in the brain. *In Neuron., 80*, 312–325. https://doi.org/10.1016/j.neuron.2013.09.007.

Doll, B. B., Simon, D. A., & Daw, N. D. (2012). The ubiquity of model-based reinforcement learning. *In Current Opinion in Neurobiology., 22*, 1075–1081. https://doi.org/10.1016/j.conb.2012.08.003.

Ekstrom, A. D., Kahana, M. J., Caplan, J. B., Fields, T. A., Isham, E. A., Newman, E. L., & Fried, I. (2003). Cellular networks underlying human spatial navigation. *Nature., 425*, 184–188. https://doi.org/10.1038/nature01964.

Eliasmith, C., & Anderson, C. H. (2003). *Neural engineering: computation, representation, and dynamics in neurobiological systems*. MIT Press.

Epstein, R. A., Patai, E. Z., Julian, J. B., & Spiers, H. J. (2017). The cognitive map in humans: spatial navigation and beyond. *In Nature Neuroscience., 20*, 1504–1513. https://doi.org/10.1038/nn.4656.

Friedrich, J., & Lengyel, M. (2016). Goal-directed decision making with spiking neurons. *The Journal of Neuroscience, 36*, 1529–1546. https://doi.org/10.1523/JNEUROSCI.2854-15.2016.

Gaussier, P., Revel, A., Banquet, J. P., & Babeau, V. (2002). From view cells and place cells to cognitive map learning: processing stages of the hippocampal system. *Biological Cybernetics, 86*, 15–28. https://doi.org/10.1007/s004220100269.

Gauthier, J. L., & Tank, D. W. (2018). A dedicated population for reward coding in the Hippocampus. *Neuron., 99*, 179–193.e7. https://doi.org/10.1016/j.neuron.2018.06.008.

Genovesio, A., Tsujimoto, S., & Wise, S. P. (2012a). Encoding goals but not abstract magnitude in the primate prefrontal cortex. *Neuron., 74*, 656–662. https://doi.org/10.1016/j.neuron.2012.02.023.

Genovesio, A., Tsujimoto, S., & Wise, S. P. (2012b). Encoding goals but not abstract magnitude in the primate prefrontal cortex. *Neuron., 74*, 656–662. https://doi.org/10.1016/j.neuron.2012.02.023.

Gershman, S. J. (2018). The successor representation: its computational logic and neural substrates. *The Journal of Neuroscience, 38*, 7193–7200. https://doi.org/10.1523/JNEUROSCI.0151-18.2018.

Goel, V., & Grafman, J. (1995). Are the frontal lobes implicated in "planning" functions? *Interpreting data from the Tower of Hanoi. Neuropsychologia., 33*, 623–642. https://doi.org/10.1016/0028-3932(95)90866-P.

Guerguiev, J., Lillicrap, T. P., & Richards, B. A. (2017). Towards deep learning with segregated dendrites. *ELife., 6*. https://doi.org/10.7554/eLife.22901.

Guitart-Masip, M., Huys, Q. J. M., Fuentemilla, L., Dayan, P., Duzel, E., & Dolan, R. J. (2012). Go and no-go learning in reward and punishment: interactions between affect and effect. *NeuroImage., 62*, 154–166. https://doi.org/10.1016/j.neuroimage.2012.04.024.

Hart, P. E., Nilsson, N. J., & Raphael, B. (1968). A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics., 4*, 100–107. https://doi.org/10.1109/TSSC.1968.300136.

Herrmann, C. S., Munk, M. H. J., & Engel, A. K. (2004). Cognitive functions of gamma-band activity: memory match and utilization. *Trends in Cognitive Sciences, 8*(8), 347–355. https://doi.org/10.1016/j.tics.2004.06.006.

Hull, C. L. (1943). Principles of behavior. An introduction to behavior theory. *The Journal of Philosophy, 40*, 558. https://doi.org/10.2307/2019960.

Huys, Q. J. M., Eshel, N., O'Nions, E., Sheridan, L., Dayan, P., & Roiser, J. P. (2012). Bonsai trees in your head: how the pavlovian system sculpts goal-directed choices by pruning decision trees. *PLoS Computational Biology, 8*. https://doi.org/10.1371/journal.pcbi.1002410.

Ivey, R., Bullock, D., & Grossberg, S. (2011). A neuromorphic model of spatial lookahead planning. *Neural Networks, 24*, 257–266. https://doi.org/10.1016/j.neunet.2010.11.002.

Jones, J. L., Esber, G. R., McDannald, M. A., Gruber, A. J., Hernandez, A., Mirenzi, A., & Schoenbaum, G. (2012). Orbitofrontal cortex supports behavior and learning using inferred but not cached values. *Science., 338*, 953–956. https://doi.org/10.1126/science.1227489.

Juechems, K., & Summerfield, C. (2019). Where does value come from? *In Trends in cognitive sciences, 23*, 836–850. https://doi.org/10.1016/j.tics.2019.07.012.

Juechems, K., Balaguer, J., Herce Castañón, S., Ruz, M., O'Reilly, J. X., & Summerfield, C. (2019). A network for computing value equilibrium in the human medial prefrontal cortex. *Neuron., 101*, 977–987.e3. https://doi.org/10.1016/j.neuron.2018.12.029.

Keramati, M., & Gutkin, B. (2014). Homeostatic reinforcement learning for integrating reward collection and physiological stability. *ELife., 3*. https://doi.org/10.7554/eLife.04811.

Klausberger, T., Somogyi, P. (2008). Neuronal diversity and temporal dynamics: the unity of hippocampal circuit operations. *Science, 321*(5885), 53–57. https://doi.org/10.1126/science.1149381.

Knoblock, C. A. (1990). *Learning abstraction hierarchies for problem solving*. In AAAI (pp. 923–928). Chicago.

Lau, B., & Glimcher, P. W. (2008). Value representations in the primate striatum during matching behavior. *Neuron., 58*, 451–463. https://doi.org/10.1016/j.neuron.2008.02.021.

Levine, S. (2018). Reinforcement learning and control as probabilistic inference: Tutorial and review. arXiv preprint arXiv:1805.00909.

Liu, D., & Todorov, E. (2007). Evidence for the flexible sensorimotor strategies predicted by optimal feedback control. *The Journal of Neuroscience, 27*, 9354–9368. https://doi.org/10.1523/JNEUROSCI.1110-06.2007.

Maass, W. (2000). On the computational power of winner-take-all. *Neural Computation, 12*, 2519–2535. https://doi.org/10.1162/089976600300014827.

Momennejad, I., Russek, E. M., Cheong, J. H., Botvinick, M. M., Daw, N. D., & Gershman, S. J. (2017). The successor representation in human reinforcement learning. *Nature Human Behaviour, 1*, 680–692. https://doi.org/10.1038/s41562-017-0180-8.

Mushiake, H., Saito, N., Sakamoto, K., Itoyama, Y., & Tanji, J. (2006). Activity in the lateral prefrontal cortex reflects multiple steps of future events in action plans. *Neuron., 50*, 631–641. https://doi.org/10.1016/j.neuron.2006.03.045.

Niv, Y., Joel, D., & Dayan, P. (2006). A normative perspective on motivation. *Trends in Cognitive Sciences, 10*, 375–381. https://doi.org/10.1016/j.tics.2006.06.010.

O'Keefe, J., & Nadel, L. (1978). *The hippocampus as a cognitive map*. Oxford: Clarendon Press.

O'Keefe, J., Burgess, N., Donnett, J. G., Jeffery, K. J., & Maguire, E. A. (1998). Place cells, navigational accuracy, and the human hippocampus. *Philosophical Transactions of the Royal Society, B: Biological Sciences, 353*, 1333–1340. https://doi.org/10.1098/rstb.1998.0287.

O'Reilly, R. C. (2020). Unraveling the mysteries of motivation. In Trends in Cognitive Sciences. https://doi.org/10.1016/j.tics.2020.03.001, 24, 425, 434.

O'Reilly, R. C., & Munakata, Y. (2019). Computational explorations in cognitive neuroscience. *In Computational Explorations in Cognitive Neuroscience*. https://doi.org/10.7551/mitpress/2014.001.0001.

O'Reilly, R., Hazy, T. E., Mollick, J. A., Mackie, P., & Herd, S. A. (2014). Goal-driven cognition in the brain: a computational framework. *arXiv: Neurons and Cognition*.

Palva, J. M., Monto, S., Kulashekhar, S., & Palva, S. (2010). Neuronal synchrony reveals working memory networks and predicts individual memory capacity. *Proceedings of the National Academy of Sciences of the United States of America, 107*(16), 7580–7585. https://doi.org/10.1073/pnas.0913113107.

Passingham, R., & Wise, S. (2012). The neurobiology of the prefrontal cortex: anatomy, evolution, and the origin of insight. **OUP Oxford**.

Piray, P., & Daw, N. D. (2020). Linear reinforcement learning: flexible reuse of computation in planning, grid fields, and cognitive control. *bioRxiv*. https://doi.org/10.1101/856849.

Poucet, B., Lenck-Santini, P. P., Hok, V., Save, E., Banquet, J. P., Gaussier, P., & Muller, R. U. (2004). Spatial navigation and hippocampal place cell firing: the problem of goal encoding. *In Reviews in the Neurosciences., 15*, 89–107. https://doi.org/10.1515/REVNEURO.2004.15.2.89.

Roesch, M. R., Calu, D. J., & Schoenbaum, G. (2007). Dopamine neurons encode the better option in rats deciding between differently delayed or sized rewards. *Nature Neuroscience, 10*(12), 1615–1624. https://doi.org/10.1038/nn2013.

Roux, F., & Uhlhaas, P. J. (2014). Working memory and neural oscillations: alpha-gamma versus theta-gamma codes for distinct WM information? *Trends in Cognitive Sciences, 18*(1), 16–25. https://doi.org/10.1016/j.tics.2013.10.010.

Russek, E. M., Momennejad, I., Botvinick, M. M., Gershman, S. J., & Daw, N. D. (2017). Predictive representations can link model-based reinforcement learning to model-free mechanisms. *PLoS Computational Biology, 13*, e1005768. https://doi.org/10.1371/journal.pcbi.1005768.

Salamone, J. D., & Correa, M. (2012). The mysterious motivational functions of mesolimbic dopamine. *In Neuron., 76*, 470–485. https://doi.org/10.1016/j.neuron.2012.10.021.

Schuck, N. W., Cai, M. B., Wilson, R. C., & Niv, Y. (2016). Human orbitofrontal cortex represents a cognitive map of state space. *Neuron., 91*, 1402–1412. https://doi.org/10.1016/j.neuron.2016.08.019.

Schultz, W., Dayan, P., & Montague, P. R. (1997). A neural substrate of prediction and reward. *Science., 275*, 1593–1599. https://doi.org/10.1126/science.275.5306.1593.

Simon, H. A. (1956). Rational choice and the structure of the environment. *Psychological Review, 63*, 129–138. https://doi.org/10.1037/h0042769.

Stachenfeld, K. L., Botvinick, M. M., & Gershman, S. J. (2017). The hippocampus as a predictive map. *Nature Neuroscience, 20*, 1643–1653. https://doi.org/10.1038/nn.4650.

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

Sutton, R. S., Precup, D., & Singh, S. (1999). Between MDPs and semi-MDPs: a framework for temporal abstraction in reinforcement learning. *Artificial Intelligence, 112*, 181–211. https://doi.org/10.1016/S0004-3702(99)00052-1.

Tai, L. H., Lee, A. M., Benavidez, N., Bonci, A., & Wilbrecht, L. (2012). Transient stimulation of distinct subpopulations of striatal neurons mimics changes in action value. *Nature Neuroscience, 15*, 1281–1289. https://doi.org/10.1038/nn.3188.

Toates. (1986). Motivational Systems. In *Problems in the behavioral sciences*. New York: Cambridge University Press.

Todorov, E. (2009). Efficient computation of optimal actions. *Proceedings of the National Academy of Sciences of the United States of America, 106*, 11478–11483. https://doi.org/10.1073/pnas.0710743106.

Tolman, E. C. (1948). Cognitive maps in rats and men. *Psychological Review, 55*, 189–208. https://doi.org/10.1037/h0061626.

Voloh, B., Valiante, T. A., Everling, S., & Womelsdorf, T. (2015). Theta-gamma coordination between anterior cingulate and prefrontal cortex indexes correct attention shifts. *Proceedings of the National Academy of Sciences of the United States of America, 112*, 8457–8462. https://doi.org/10.1073/pnas.1500438112.

Welsh, M., Cicerello, A., Cuneo, K., & Brennan, M. (1995). Error and temporal patterns in tower of Hanoi performance: cognitive mechanisms and individual differences. *The Journal of General Psychology, 122*, 69–81. https://doi.org/10.1080/00221309.1995.9921223.

Westbrook, J., van den Bosch, R., Maatta, J.I., Hofmans, L., Papadopetraki, D., Cools, R., & Frank, M. J. (2020). Dopamine promotes cognitive effort by biasing the benefits versus costs of cognitive work. *Science, 367*, 1362–1366.

Wikenheiser, A. M., & Schoenbaum, G. (2016). Over the river, through the woods: cognitive maps in the hippocampus and orbitofrontal cortex. *In Nature Reviews Neuroscience., 17*, 513–523. https://doi.org/10.1038/nrn.2016.56.

Wilson, R. C., Takahashi, Y. K., Schoenbaum, G., & Niv, Y. (2014). Orbitofrontal cortex as a cognitive map of task space. *Neuron., 81*, 267–279. https://doi.org/10.1016/j.neuron.2013.11.005.

Wimmer, G. E., & Büchel, C. (2019). Learning of distant state predictions by the orbitofrontal cortex in humans. *Nature Communications, 10*. https://doi.org/10.1038/s41467-019-10597-z.

Wunderlich, K., Smittenaar, P., & Dolan, R. J. (2012). Dopamine enhances model-based over model-free choice behavior. *Neuron., 75*, 418–424. https://doi.org/10.1016/j.neuron.2012.03.042.

Yamagata, T., Nakayama, Y., Tanji, J., & Hoshi, E. (2012). Distinct information representation and processing for goal-directed behavior in the dorsolateral and ventrolateral prefrontal cortex and the dorsal premotor cortex. *The Journal of Neuroscience, 32*, 12934–12949. https://doi.org/10.1523/JNEUROSCI.2398-12.2012.