A neural network framework for the orbitofrontal cortex during model-based reinforcement learning

Zhewei Zhang[1,2#], Zhenbo Cheng[3#], Zhongqiao Lin[1,2], Chechang Nie[1,2], and Tianming Yang[1*]

[1] Institute of Neuroscience, Key Laboratory of Primate Neurobiology, CAS Center for Excellence in Brain Science and Intelligence Technology, Shanghai Institutes for Biological Sciences, Chinese Academy of Sciences
[2] University of Chinese Academy of Sciences, Beijing 100049, China
[3] Department of Computer Science and Technology, Zhejiang University of Technology, Hangzhou, China

[#]Equal contributions.
*Address correspondence to:
Tianming Yang (tyang@ion.ac.cn)
Institute of Neuroscience
320 Yue Yang Rd.
System Neuroscience Building, Room 302
Shanghai, China 200031
Phone: +86-21-54921737, Fax: +86-21-54921735

Pages: 25
Abstract = 150 words
Figures = 6
Tables = 0

Abstract

Model-based reinforcement learning (mbRL) has been widely used in explaining animal behavior. In mbRL, the model, or the structure of the task, is used to evaluate the associations between actions and outcomes. It has been proposed that the orbitofrontal cortex (OFC) encodes the model during mbRL. However, it is not well understood how the OFC acquires and stores model information. Here, we propose a neural network framework based on reservoir computing. Reservoir networks exhibit heterogeneous and dynamic activity patterns that are suitable to encode task states. The information can be extracted by a linear readout trained with reinforcement learning. We demonstrate how our framework acquires and stores the task state space. The framework exhibits mbRL behavior and its aspects resemble experimental findings of the OFC. Our study provides a theoretical explanation of how the OFC may contribute to mbRL and a new approach to understanding the neural mechanism underlying mbRL.

Introduction

Even the simplest reinforcement learning (RL) algorithm captures the essence of operant conditioning in psychology and animal learning (Rescorla & Wagner, 1972). That is, actions that are rewarded tend to be repeated more frequently; actions that are punished are more likely to be avoided. However, it fails to explain animals' behavior in more complicated situations. One particular approach to extending the capabilities of RL algorithms, known as model-based reinforcement learning (mbRL, as contrast to model-free RL, or mfRL), uses the knowledge of the task structure, i.e., the model, to guide the learning (Beck et al., 2008). mbRL is especially successful in explaining the goal-directed learning behavior in complex environments (Dolan & Dayan, 2013; Doll, Simon, & Daw, 2012; Keramati, Smittenaar, Dolan, & Dayan, 2016).

Several studies have investigated the possible brain structures that may be involved in mbRL (Daw, Gershman, Seymour, Dayan, & Dolan, 2011; Glascher, Daw, Dayan, & O'Doherty, 2010; Haber, Kim, Mailly, & Calzavara, 2006; Kennerley, Behrens, & Wallis, 2011; Schultz, Dayan, & Montague, 1997). Notably, the orbitofrontal cortex (OFC) has been hypothesized to represent the task space and encode task states (Wilson, Takahashi, Schoenbaum, & Niv, 2014). Several lesion studies showed that the animals with OFC lesions exhibited deficits acquiring task information for building a task model (Hornak et al., 2004; Izquierdo, Suda, & Murray, 2004; Takahashi et al., 2011). Electrophysiology studies of the OFC have demonstrated that the OFC encodes many aspects of reward information, including reward value (J. L. Jones et al., 2012; Padoa-Schioppa, 2011; Padoa-Schioppa & Assad, 2006; Rudebeck, Mitz, Chacko, & Murray, 2013; Wallis & Miller, 2003), probability (Kennerley & Wallis, 2009), risk (O'Neill & Schultz, 2015), information value (Blanchard, Hayden, & Bromberg-Martin, 2015), abstract rules (Wallis, Anderson, & Miller, 2001), and strategies (Tsujimoto, Genovesio, & Wise, 2011). Yet, it is not well understood how models themselves may be encoded and represented by a neural network, and what sort of neuronal firing properties we expect to find in neurophysiological experiments. Furthermore, we do not know how to teach a model-agnostic neural network to acquire the structure of the task just based on trial and error.

The recent development of reservoir computing may provide a solution (Buonomano & Maass, 2009; Laje & Buonomano, 2013; Maass, Natschlager, & Markram, 2002). Reservoir networks are recurrent networks with fixed connections. Within a reservoir network, neurons are randomly and sparsely connected. Importantly, the internal states of a reservoir exhibit rich temporal dynamics, which represents a nonlinear transformation of its input history and can be very useful for encoding task state sequences. The information encoded by the network can be extracted with a linear output, which can be trained during learning. Reservoir networks have been shown to exhibit dynamics similar to that observed in the prefrontal cortex (Barak, Sussillo, Romo, Tsodyks, & Abbott, 2013; Cheng, Deng, Hu, Zhang, & Yang, 2015; Enel, Procyk, Quilodran, & Dominey, 2016).

In the current study, we demonstrate with two commonly learning paradigms how a reservoir network may achieve mbRL by encoding task states without prior knowledge of task structures. Task event sequences, including reward events, are provided as inputs to the network. A simple yet biologically feasible reward-dependent Hebbian learning algorithm is used to adjust its output weights. We show that our framework can solve problems with different task structures and exhibits mbRL behavior previously reported in animals and humans. We further demonstrate the similarity between the reservoir network and the OFC. Manipulations to our network reproduce the behavior of animals with OFC lesions. The reservoir neurons' response patterns resemble characteristics of the OFC neurons reported from previous electrophysiological experiments.

Taken together, these results suggest a simple mechanism that naturally leads to the acquisition of task structure and therefore supports mbRL. Finally, we propose some future experiments that may be used to test our model.

Results

We describe our results in three parts. We start with using our network to model a classical reversal learning task. We take advantage of its simplicity to explain the principles behind the framework. Then we show such a framework may be applied to more complex scenarios, using a two-stage decision task as an example. Finally, we demonstrate how the network framework may be used to describe experimental findings in the OFC during value-based decision making.

Reversal Learning

In a classical reversal learning task, the animals have to keep track of the reward contingency of two choice options that may be reversed during a test session (Izquierdo et al., 2004; B. Jones & Mishkin, 1972). Normal animals were found to learn reversals faster and faster, which has been used as an indication of mbRL (Wilson et al., 2014). The mbRL behavior was however found to be impaired in animals with OFC lesions or with lesions that contained fibers passing near the OFC (Izquierdo et al., 2004; Rudebeck, Saunders, Prescott, Chau, & Murray, 2013). These animals were not able to learn reversals faster and faster when they were repeatedly tested. The learning impairments could be explained by mfRL (Wilson et al., 2014).

Our neural network framework consists of a state encoding layer (SEL), which is a reservoir network. It receives three inputs and generates two outputs (Fig 1a). The three inputs to the SEL are the two choice options A and B, together with a reward input that indicates whether the choice yields a reward or not in the current trial. The outputs represent choice actions A and B for the next trial. We use the neural activity of the SEL at the end of the input to determine the SEL's output.

The framework is able to reproduce animals' behavior. The number of error trials that takes for the framework to achieve the performance threshold, which is set at 93% in the initial learning and at 80% in the subsequent reversals, decreases as the model goes through more and more reversals (Fig 1b). Interestingly, a learning deficit similar to that found in OFC-lesion animals is observed if we remove the reward input to the SEL (Fig 1b). As the OFC and its neighboring brain areas such as the ventromedial prefrontal cortex (vmPFC) are known to receive both the sensory inputs and reward inputs from sensory and reward circuitry in the brain, removing the reward input from our model mimics the situation where the brain has to learn without functioning structures in or near the OFC.

Neurons in the SEL, as expected from a typical reservoir network, show highly heterogeneous response patterns. Some neurons are found to encode the stimulus identity, some neurons encode reward, and others show mixed tuning (Fig 2a). A principal component analysis (PCA) based on the population activity shows that the network can distinguish all four possible task states: choice *A* rewarded, choice *A* not rewarded, choice *B* rewarded, and choice *B* not rewarded (Fig2b).

The ability to distinguish these states is essential for learning. To understand the mbRL behavior exhibited by our model, we study how neurons with different selectivity contribute to the learning (Fig 2c). We find that readout weights of the neurons that are selective to the combination of stimulus and reward inputs (e.g. *AR* and *BR*) are mostly affected by the learning. The difference between the weights of their connections to the outputs *A* and *B* keeps growing despite repeated reversals. In contrast, the weights of the output connections of pure stimulus-selective neurons only wiggle around the baseline between reversals.

The difference between these two groups of neurons explains why our network achieves mbRL only when the reward input is available. Let us first consider the *AR* neurons, which are selective for the situation when choice *A* leads to reward. In these *A*-rewarded blocks, the connections between the *AR* neurons and the DML neuron of choice *A* are strengthened. When the reward contingency is reversed and now choice *A* leads to no reward, the connections between the *AR* neurons and choice *A* are not affected very much. That is because the group of *AN* neurons instead of the *AR* neurons that are activated in the blocks when choice *A* is not rewarded. As the result, the connections between the *AN* neurons and the DML neuron of choice *B* are strengthened and the connections between the *AN* neurons and the DML neuron of choice *A* are weakened. When the reward contingency is flipped again, the connections between the *AR* neurons and the DML neuron of choice *A* are strengthened further. This way, the learning is never erased by the reversals, and the network learns faster and faster. In comparison, let us now consider the *A* neurons, which encode only the sensory inputs and are activated whenever input *A* is present. In the *A*-rewarded blocks, the connections between the *A* neurons and the DML neuron of choice *A* is strengthened. In *B*-rewarded blocks, the connections between the *A* neurons and the DML neuron of choice *A* is however weakened when the network chooses *A* and gets no reward, and the learning in the previous block is reversed. Thus, the output connections of *A* neurons only fluctuate around the baseline with the reversals. They do not contribute much to the learning, and the overall behavior of the network is mostly driven by neurons that are activated by the combination of reward input and sensory inputs. Removing *R* deactivates these neurons and leads to the model-free behavior.

Two-stage Markov decision task

We further test our network model with a two-stage decision making task. The task is similar to the Markov decision task used previously in several human fMRI studies (Glascher et al., 2010). In this task, the subjects have to choose between two options *A1* and *A2*. Their choices then lead to two intermediate outcomes *B1* and *B2* at different but fixed probabilities. The choice of *A1* more likely leads to *B1*, and the choice of *A2* is more likely followed by *B2*. Importantly, the final reward is contingent only on these intermediate outcomes, and the contingency is reversed across blocks (Fig 3a). Thus, the probability of getting a reward is higher for *B1* in one block and becomes lower in the next block. The probabilistic association between the initial choices and the intermediate outcomes never changes. The subjects are not informed of the structure of the task, and they have to figure out the best option by tracking not only the reward outcomes but also the intermediate outcomes.

We keep our framework mostly the same as in the previous task. Here, we have two additional input units that reflect the intermediate outcomes (Fig 3b). To demonstrate our framework's capability of encoding sequential events, the input units are activated sequentially in our simulations as they are in the real experiment (Fig 3c). We also add a *non-reward* input unit whose activity is set to 1 when a reward is not obtained at the end of a trial. The additional non-reward input facilitates learning but does not change the results qualitatively.

For a mfRL strategy, the probability of repeating the previous choice only depends on the reward outcome. The probability of repeating the previous choice is higher when a reward is obtained than when no reward is obtained. The intermediate outcome is ignored. However, for a mbRL strategy, this is no longer the case. For example, consider the situation when the subject initially chooses *A1*, the intermediate outcome happens to be *B2*, and a reward is obtained. If the subject understands *B2* is an unlikely outcome of choice *A1* (rare), but a likely outcome of choice *A2* (common), a reward obtained after the rare event *B2* should actually motivate the subject to switch from the previous choice and choose *A2* the next time. The subject should always choose the option that is more likely to lead to the intermediate outcome that is currently associated with a reward.

To quantify the model-based learning behavior, we first evaluate the impact of the previous trial's outcome on the current trial. We classify all trial outcomes into four categories: common-rewarded (*CR*), common-unrewarded (*CN*), rare-rewarded (*RR*) and rare-unrewarded (*RN*). Here, common and rare indicate whether the intermediate outcome is the more likely outcome of the chosen option or not. Glascher et al (Glascher et al., 2010) showed that the mbRL led to a higher probability of repeating the previous choice in the *CR* and *RN* conditions. This is also what we observe in our network model's behavior (Fig 4a).

To illustrate how the network acquires the model, we define the model-based index, which represents the tendency of model-based behavior (see the *Method*). The model-based index grows larger as the training goes on (Fig 4b). It indicates that the network learns the structure of the task gradually and transits to the model-based behavior from an initially model-free behavior. Similar to our findings in the first task, the SEL without the reward input does not show this transition (Fig 4b). We further quantify the contributions of mbRL and mfRL to the network behavior using a model fitting procedure previously described by Glascher et al. (Glascher et al., 2010), and the network without the reward input shows a significantly smaller weight for mbRL, suggesting it is worse at picking up the task structure (Fig 4c).

Again, a PCA on the SEL population activity shows that the SEL distinguishes different task states (Fig 4d). Because of the structure of the task in which the contingency between the first stage options and the intermediate outcomes is fixed, the network only needs to find out the current reward contingency of the intermediate outcomes. We found that the learning picks out the most relevant neurons that encode the contingency between the intermediate outcomes and the reward outcomes (*B1R*, *B2R*, etc.). Their connection weights to the DML neurons show better and better differentiation of the two choices throughout the training (Fig 4e). In contrast, the weights of neurons that encode the association between the first stage options and the reward outcomes (*A1R*, *A2R*, etc.) are less differentiated. These results suggest that the network acquires the task structure as the result of training.

Value representation by the OFC

Previous electrophysiology studies have shown that OFC neurons encode value during economic choices (Padoa-Schioppa & Assad, 2006; Wallis & Miller, 2003). Among these value encoding neurons, studies have identified multiple classes of neurons encoding a variety of information, including the value of individual offers (offer value), the value of the chosen option (chosen value), and the identity of the chosen option (chosen identity) (Cai & Padoa-Schioppa, 2014; Padoa-Schioppa, 2013).

Here we show that our framework may explain this apparent heterogeneous value encoding in the OFC. Here we model a two-alternative economic choice task by providing two inputs to the SEL, representing the value of each option (Fig 5a). The framework can reproduce the choice behavior of monkeys (Fig 5b)(Padoa-Schioppa & Assad, 2006). Then we study the selectivity of the SEL neurons. We find not only neurons that encode the value of each option (offer value neurons, middle panel in Fig 6a), but also neurons that encode the value of the chosen option (chosen value neurons, left panel in Fig 6a). Furthermore, a proportion of neurons show the selectivity for the choice as previously reported (chosen identity neurons, right panel in Fig 6a). We classify the neurons in the reservoir network into 10 categories as described in Padoa-Schioppa and Assad (Padoa-Schioppa & Assad, 2006). Interestingly, we are able to find neurons in 9 of the 10 categories (Fig 6b, c). The only missing category (neurons encoding other/chosen value) was also very rare in the experimental data. Although the proportions of neurons encoding each category are not an exact copy of the experimental data, but the similarity is apparent. This is surprising given that we do not tune the internal connections of the SEL to the task. The heterogeneity is naturally expected from a reservoir network, but it takes much more effort to explain with recurrent network models that have a well-defined structure (Daie, Goldman, & Aksay, 2015; Rustichini & Padoa-Schioppa, 2015).

Discussion

So far, we have shown that a simple reservoir-based network model may exhibit model-based learning behavior. The more interesting question is that why the network is capable of doing so and how this network model may help us to understand the functions of the OFC.

We place a reservoir network as the centerpiece of our model. Reservoir networks are large, distributed, nonlinear dynamical recurrent neural networks with fixed weights. Because of recurrent networks' complicated dynamics, they are especially useful in modeling temporal sequences including languages (Rodriguez, 2001; Suykens, Vandewalle, & Moor, 1996). They have been shown to be Turing equivalent (Kilian & Siegelmann, 1996) and capable of approximating arbitrary dynamical systems (Funahashi & Nakamura, 1993). In our model, the reservoir network encodes a combinatory of inputs that constitutes the task state space. States are encoded by the activities of the reservoir neurons, and the learned action values are represented by the weights of the readout connections. We show that a reinforcement learning algorithm is capable of solving the relatively simple tasks in this study. However, it has been shown that reinforcement learning is in general not very efficient for extracting information from reservoir networks. A possible solution is to introduce additional layers to help with the readout (Cheng et al., 2015).

It is important to note that reward events must also be provided as an input to allow mbRL. Including reward events allows the network to establish associations between sensory stimuli and rewards, thus facilitates model-based learning. Removing reward inputs to the reservoir leads to a mfRL behavior. Although reward modulates neural activities almost everywhere in the cortex, the OFC is unique in its role of encoding the association between sensory stimuli and rewards. Removing the reward input to the reservoir mimics the situation when animals cannot rely on such an association to learn tasks. In this case, the reservoir is still perfectly functional in terms of encoding task events other than rewards. We hypothesize this simulates the situation when animals have to depend on their other memory structures in the brain – such as hippocampus or other medial temporal lobe structures – for learning. The importance of the reward input to the reservoir explains the key role that the OFC plays in mbRL.

Several recent studies reported that selective lesions in the OFC did not reproduce the behavior deficits in reversal learning previously seen if the fibers passing through or near the OFC were spared (Rudebeck, Saunders, et al., 2013). Since these fibers probably carry the reward information from the midbrain areas, these results do not undermine the importance of reward inputs. Presumably, when the lesion is limited to the OFC, the projections that carrying the reward information were still available to or might even be redirected to other neighboring prefrontal structures, including ventromedial prefrontal cortex, which might take over the role of the OFC and contribute to mbRL in animals with selective OFC lesions.

There are several reasons why we choose reservoir networks to construct our model. First reason is that we would like to pair our network model with reinforcement learning. Reservoir networks have fixed internal connections; the training occurs only at the readout. The number of parameters is thus much smaller, which could be important for efficient reinforcement learning. Generality is another benefit offered by reservoir networks. Because the internal connections are fixed, we can use the same network to solve a different problem by just training a different readout. The reservoir can be seen as a general-purpose task state representation network. Lastly, our results as well as several other studies show that neurons in reservoir networks – although with untrained connections weights – show properties similar to that observed in the real brain (Barak et al., 2013; Cheng et al., 2015; Sussillo & Abbott, 2009), suggesting local plasticity may not play a role as important as previously thought.

The fact that the internal connections are fixed in a reservoir network means that the selectivities of the reservoir neurons are also fixed. This may seem at odds with the experimental findings of many OFC neurons shifting their encodings rapidly during reversals (Rolls, Critchley, Mason, & Wakeman, 1996). However, these observations may be interpreted differently. The neurons that were found to have different responses during reversals might be in fact encoding rewards. On the other hand, there is evidence that OFC neurons with inflexible encodings during reversals might be more important for mbRL behavior (Schoenbaum, Saddoris, & Stalnaker, 2007).

The performance of our network depends on several factors. First, it is important that reservoir should be able to distinguish between different task states. The number of possible task states may be only 4 or 8 as in our examples, or may be impossibly large even if the number of inputs increases only slightly. The latter is due to the infamous combinatorial explosion problem. One may alleviate the problem by introducing learning in the reservoir to weed out irrelevant combinations. Second, the dynamics of the reservoir should allow information to be maintained long enough until the decision is made. The recent developed gated recurrent neural networks may provide a solution with units that may maintain information for long periods (Chung, Gulcehre, Cho, & Bengio, 2014). Third, the model exhibits substantial variability between runs, suggesting the initialization may impact its performance. Further investigation is needed to make the model more robust.

Our model makes several testable predictions. First, because of the reservoir structure, the inputs from the same source should be represented evenly in the network. For example, in a visual task, different visual stimuli should be represented at roughly the same strength in the OFC, even if their visual salience may be drastically different. Second, we should be able to find neurons encoding all relevant task parameters in the network. Third, reducing the number of inputs may make the network to be more efficient in certain tasks. This may seem counter-intuitive. But removing inputs reduces the number of states that the network has to encode, thus improves learning efficiency for tasks that do not require those additional states. For example, if we remove the reward input to the SEL, which is essential for model-based learning, the network should however be more efficient at model-free learning. Indeed, animals with OFC lesions were found to perform better than control animals when reward history was not important (Riceberg & Shapiro, 2012).

In summary, our framework does not intend to be a complete model of how the OFC works. Instead of creating a complete neural network solution of mbRL or the OFC, which is improbable at the moment, we are aiming at the modest goal of providing a proof of concept that approaches the critical problem of how to acquire the model in mbRL. By demonstrating the network's similarity to the experimental findings in the OFC, our study opens up new possibilities in future investigation.

Materials and Methods

Neural Network Model

The model is composed of three layers: an input layer (IL), a state encoding layer (SEL), and a decision-making output layer (DML) (Fig. 1a).

The units in the input layer represent the identities of sensory stimuli and the reward obtained.

The input neurons are sparsely connected to the SEL units. The connection weights $w_i^{(1)}$ are set to 0 with a probability of p_{IR}=0.2. Nonzero weights are assigned independently from a standard uniform distribution [0, 1].

In the SEL, there are N= 500 neurons. The neurons in the SEL are connected with a low probability p=0.1 and the connections are randomly and independently set from a Gaussian distribution with zero mean and a variance of $g^2/(p*N)$, where the gain g acts as the control parameter in the SEL. Connections in the SEL could be both positive and negative.

Each neuron in the SEL is described by an activation variable x_i for i = 1, 2, ..., N, which is initialized with a normal distribution $N(0, \sigma_{\mathrm{ini}}^2)$ at the beginning of each trial. x_i is updated at each time step (dt = 1ms) as follows:

$$\tau\frac{dx_i}{dt} = -x_i + g\sum_{j=1}^{N} w_{ij}y_j + w_i^{(1)}I + \sigma_{noise}dW_i \qquad (1)$$

where τ represents the time constant, w_{ij} is the synaptic weight between neurons i and j, dW_i stands for the white noise, and σ_{noise} is its variance. The firing rate y_i of neuron i is a function of the activation variable x_i relative to a minimal firing rate y_{min}=0 and the maximal rate y_{max}=1:

$$\mathrm{y} = \begin{cases} y_0 + y_0 tanh(x/y_0) & x \le 0 \\ y_0 + (y_{max} - y_0) * tanh(x/(y_{max} - y_0)) & x > 0 \end{cases} \qquad (2)$$

Here y_0 = 0.1 is the baseline firing rate.

The SEL neurons project to the DML. The two competing neurons in the DML represent the two choices respectively. The total input of neuron k in the DML is

$$v_k = \sum_i w_{ik}^{(2)} y_i \qquad \text{for } k = 1, 2 \qquad (3)$$

where $w_{ik}^{(2)}$ is the weight of the synapse between neuron i in the SEL circuit and neuron k in the DML. The synaptic weights between the SEL and DML are randomly initialized with uniform distribution [0, 1], and normalized to keep the squared sum of synaptic weights projecting to the same DML unit equal to 1.

The synaptic weights between the SEL and DML are updated based on the reward outcome during the training phase. The stochastic choice behavior of our model is described by a softmax function:

$$p_k = \frac{e^{-\beta v_k}}{\sum_l e^{-\beta v_l}} \tag{4}$$

where p_k represents the probability for choosing the choice a_k, and the other choice is chosen with probability 1- p_k. β adjusts the competition strength of two choices, and v_k is the input of the DML unit k. The firing rate of the unit k, y_k, is set to 1 if choice a_k is chosen, otherwise it is set to 0.

Reinforcement Learning

At the end of each trial, the weights between the SEL and the DML neurons are updated. The plastic weights in eq (3) in trial n+1 are updated as follows:

$$w_{ik}^{(2)}(n+1) = w_{ik}^{(2)}(n) + \Delta w_{ik} \tag{5}$$

The update term Δw_{ik} depends on the reward prediction error and the responses of the neurons in the SEL circuit and DML:

$$\Delta w_{ik} = \eta(r - E[r])(y_i - y_{th})y_k \tag{6}$$

where η is the learning rate, and r is the reward. $E[r]$ denotes the expected value. When the reward r is larger than E[r], the connections between the SEL neurons whose firing rate is above the threshold y_{th} and the neurons in the DML would be strengthened, and the connections between the neurons whose firing rate is below y_{th} and the neurons in the DML would be weakened. After each update, the weights $w_{ik}^{(2)}(n)$ are normalized:

$$w_{ik}^{(2)}(n) = \frac{w_{ik}^{(2)}(n)}{\sqrt{\sum_{i=1}^{N}[w_{ik}^{(2)}(n)]^2}} \tag{7}$$

so that the vector length of $w_{ik}^{(2)}(n)$ remains constant. The normalization stops the weights from growing infinitely (Royer & Pare, 2003).

Behavior Task

Reversal learning

The network has to choose between two options. One option leads to a reward, and the other does not. The stimulus-reward contingency is reversed every 100 trials. The criterion for learning is set to 28 correct trials in 30 successive trials for the initial learning and 24 correct trials in 30 successive trials for subsequent reversals.

The input layer units represent the identities of the two options and the reward. An option unit's response is set to 1 for if the corresponding option is chosen in the current trial, otherwise it is set to 0. The reward unit's response is set to 1 if the choice is rewarded in the current trial. The output of the network indicates its choice for the next trial. The network parameters are set as follows. Time constant τ = 100ms, Network gain g=2, training threshold y_{th} = 0.4, temperature parameter β = 4, learning rate η = 0.001, noise gain σ_{noise} =0.01, initial noise gain σ_{ini} = 0.01.

The selectivity of neurons in the SEL is determined at the time point when the decision is made. A unit is defined as selective to a certain input or a combination of inputs if its responses are significantly higher under the condition when the input or all inputs of the combination are set to 1 than when they are set to 0.

Two-stage Markov decision task

The network has to make a choice between options *A1* and *A2*. *A1* leads to intermediate outcome *B1* at the probability of 80%, and *B2* at the probability of 20%. Vice versa, option *A2* leads to *B2* at the probability of 80%, and *B1* at a lower probability of 20%. The contingency between options (*A1*, *A2*) and intermediate outcomes (*B1*, *B2*) is fixed. Initially, *B1* leads to a reward at the probability of 80% and *B2* leads to reward at the probability of 20%. The reward contingency is reversed every 50 trials.

The input layer contains 6 units, representing the identities of two first stage options *A1* and *A2*, two intermediate outcome *B1* and *B2*, and the reward and non-reward conditions, respectively. The activity of option unit *A1* or *A2* is set to 1 when the respective option is chosen. The activity of intermediate outcome unit *B1* or *B2* is set to 1 when the respective intermediate outcome is presented. The reward unit's activity is set to 1 when a reward is obtained, and the non-reward unit's activity is set to 1 when no reward is obtained. The units are activated sequentially, reflecting the sequential nature of the task. The *A* units are activated between 200 and 700ms after a trial starts, the *B* units between 700 and 1200ms, and the reward units between 1200 and 1700ms.

The output of the network indicates its choice. The network parameters are set as follows. Time constant τ = 500ms, Network gain g=2.25, training threshold y_{th} = 0.2, temperature parameter β = 2, learning rate η = 0.001, noise gain σ_{noise} =0.01, initial noise gain σ_{ini} = 0.01.

The selectivity of neurons in the SEL is determined at the time point when the decision is made. There are 8 conditions in this task, namely *A1B1R*, *A1B1N*, *A2B1R*, *A2B1N*, *A1B2R*, *A1B1N*, *A2B2R*, and *A2B2N*. For example, *A1B1R* indicates the condition when *A1* is chosen, intermediate outcome *B1* is presented, and a reward is obtained. A neuron's preferred condition is the condition under which its activity is the largest and significantly higher than its activity under any other conditions. Then the neurons are grouped into different categories based on their preferred conditions. The neurons in category *A1R* are the neurons whose preferred conditions are *A1B1R*, *A1B2R*, *A2B1N* and *A2B2N*. All the preferred conditions of the neurons in category *A1R* provide evidence for associating *A1* with the reward. Similarly, the preferred conditions of the neurons in the category *B1N* are *A1B1N*, *A1B2R*, *A2B1N* and *A2B2R*. They provide evidence that *B1* is not associated with the reward.

Model-based model fitting

In order to test the model-based learning, we fit our data based on the model introduced by Daw et al. (Daw et al., 2011). The model fits the behavioral results with a mixture of model-free and model-based learning algorithm. In our simplified task, the network makes only one choice in each trial. The inverse temperature parameters β_1 is set to 2, which is also used to produce simulated behavioral choices. The parameter p, which captures the tendency for perseveration and switching, is set to 0, although all conclusions still hold when p is allowed to vary. The free parameters relevant in our task are α_1, α_2, λ and w. α_1 and α_2 are the learning rates in the model-free and model-based learning algorithms, respectively. The eligibility λ represents how large proportion of credit from the reward can be given to the first states and actions in our task paradigm. w is the weight for model-based learning. When w equals 1, the behavior is purely model-based. When w equals 0, the behavior is purely model-free. The fitting is done by a maximum likelihood estimation procedure.

Model-based index

Inspired by the factorial analysis from Daw et al. (Daw et al., 2011), we define a MB index (eq.8) to quantify the tendency of repeating the choice in the last trial under different situations. The combination of the two reward outcomes and the two intermediate outcomes, common and rare, gives us four possible outcomes: common-rewarded (CR), common-unrewarded (CN), rare-rewarded (RR) and rare-unrewarded (RN). In the model-based learning, the agent is more likely to repeat the previous choice if the last trial is a CR or an RN trial. Higher MB index means that the behavioral pattern is more similar to the mbRL behavior.

$$\text{MB index} = \frac{p(stay|CR)+p(stay|RN)-p(stay|CN)-p(stay|RR)}{p(stay|CR)+p(stay|RN)+p(stay|CN)+p(stay|RR)} \qquad (8)$$

Value-based economic choice task

Unlike the two previous paradigms, both options in this paradigm lead to a reward. Two input units represent the rewards associated with the two options, respectively. The input strength is proportional to reward magnitude. In our simulations, the reward A is valued twice as much as reward B for the same reward magnitude. The relative value preference between the two options is not provided as an input to the network directly, but used in calculating the expected value. The value of the reward is defined as the product of the relative value and the reward magnitude.

The activity of the input unit, f(t), is described by the following equations (Rustichini & Padoa-Schioppa, 2015).

$$\mathrm{g}(t) \;=\; {}^{1}\!\big/_{((1+\exp(-(t-475)/30))\,*\,(1+\exp((t-700)/100)))} \qquad (9)$$

$$\mathrm{f}(t) \;=\; {}^{(\mathrm{mag}_{\mathrm{r}_i}\,-\,\min(\mathrm{mag\_r}_i))\,*\,\mathrm{g(t)}}\!\Big/_{(\max(\mathrm{mag\_r}_i)\,-\,\min(\mathrm{mag\_r}_i))\,*\,\max(\mathrm{g}(t))} \qquad (10)$$

Here t is the time in the unit of ms within a trial, $\mathrm{mag}_{\mathrm{r}_i}$ is the magnitude of the reward type i in each trial, $\max\left(\mathrm{mag}_{\mathrm{r}_i}\right)$ is the maximal reward magnitude of reward type i within the block, and $\min(\mathrm{mag\_r}_i)$ represents the minimal reward magnitude of reward type i, which is always

0 in our simulations. The expected value is the sum of the product of the probability of choosing the option and corresponding reward magnitude.

$$E(r) = p_1(\gamma * m_1) + p_2 m_2 \tag{11}$$

where p_i and m_i are the probability of choosing option *i* and its reward magnitude, and γ=2 is the relative value preference between the two reward options. Only the data from the trials after 6000 trials training are included for the analyses. The network parameters are set as follows. Time constant τ = 100ms, Network gain g=2.5, training threshold y_{th} = 0.2, temperature parameter β = 4, learning rate η = 0.005, noise gain σ_{noise} =0.05, initial noise gain σ_{ini} = 0.2.

As in Padoa-schioppa and Assad (Padoa-Schioppa & Assad, 2006), the following variables are defined for further analysis: total value (the sum of the value of two options), chosen value (the value of the chosen option), other value (the value of the unchosen option), value difference (chosen-other value), value ratio (other/chosen value), offer value (the value of the one option), chosen juice (the identity of the chosen option), and value A chosen (the value of the option A when option A is chosen).

We use an analysis similar to that in Padoa-schioppa and Assad (Padoa-Schioppa & Assad, 2006) to study the selectivity of SEL units during the post-offer period (0-500ms after the stimulus onset). Linear regressions are applied to each variable to fit the neural responses in this time window for each SEL unit separately. A variable is considered to explain the response of a neuron if the slope of a fitting linear function is significantly different from zero.

Acknowledgements: This work is supported by the CAS Hundreds of Talents Program and Science and Technology Commission of Shanghai Municipality (15JC1400104) to T. Y., and by Public Projects of Zhejiang Province (2016C31G2020069) and the 3rd Level in Zhejiang Province "151 talents project" to Z. C. We thank Yu Shan and Xiao-Jing Wang for discussions and comments during the study. The authors declare no competing financial or nonfinancial interests.

References

Barak, O., Sussillo, D., Romo, R., Tsodyks, M., & Abbott, L. F. (2013). From fixed points to chaos: three models of delayed discrimination. *Prog Neurobiol, 103*, 214-222. doi:10.1016/j.pneurobio.2013.02.002

Beck, J. M., Ma, W. J., Kiani, R., Hanks, T., Churchland, A. K., Roitman, J., . . . Pouget, A. (2008). Probabilistic population codes for Bayesian decision making. *Neuron, 60*(6), 1142-1152. doi:10.1016/j.neuron.2008.09.021

Blanchard, T. C., Hayden, B. Y., & Bromberg-Martin, E. S. (2015). Orbitofrontal cortex uses distinct codes for different choice attributes in decisions motivated by curiosity. *Neuron, 85*(3), 602-614. doi:10.1016/j.neuron.2014.12.050

Buonomano, D. V., & Maass, W. (2009). State-dependent computations: spatiotemporal processing in cortical networks. *Nat Rev Neurosci, 10*(2), 113-125. doi:10.1038/nrn2558

Cai, X., & Padoa-Schioppa, C. (2014). Contributions of orbitofrontal and lateral prefrontal cortices to economic choice and the good-to-action transformation. *Neuron, 81*(5), 1140-1151. doi:10.1016/j.neuron.2014.01.008

Cheng, Z., Deng, Z., Hu, X., Zhang, B., & Yang, T. (2015). Efficient reinforcement learning of a reservoir network model of parametric working memory achieved with a cluster population winner-take-all readout mechanism. *J Neurophysiol, 114*(6), 3296-3305. doi:10.1152/jn.00378.2015

Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. *ArXiv e-prints, 1412*. Retrieved from http://adsabs.harvard.edu/abs/2014arXiv1412.3555C

Daie, K., Goldman, M. S., & Aksay, E. R. (2015). Spatial patterns of persistent neural activity vary with the behavioral context of short-term memory. *Neuron, 85*(4), 847-860. doi:10.1016/j.neuron.2015.01.006

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*(6), 1204-1215. doi:10.1016/j.neuron.2011.02.027

Dolan, R. J., & Dayan, P. (2013). Goals and habits in the brain. *Neuron, 80*(2), 312-325. doi:10.1016/j.neuron.2013.09.007

Doll, B. B., Simon, D. A., & Daw, N. D. (2012). The ubiquity of model-based reinforcement learning. *Curr Opin Neurobiol, 22*(6), 1075-1081. doi:10.1016/j.conb.2012.08.003

Enel, P., Procyk, E., Quilodran, R., & Dominey, P. F. (2016). Reservoir Computing Properties of Neural Dynamics in Prefrontal Cortex. *PLoS Comput Biol, 12*(6), e1004967. doi:10.1371/journal.pcbi.1004967

Funahashi, K., & Nakamura, Y. (1993). Approximation of Dynamical-Systems by Continuous-Time Recurrent Neural Networks. *Neural Networks, 6*(6), 801-806. doi:Doi 10.1016/S0893-6080(05)80125-X

Glascher, J., Daw, N., Dayan, P., & O'Doherty, J. P. (2010). States versus rewards: dissociable neural prediction error signals underlying model-based and model-free reinforcement learning. *Neuron, 66*(4), 585-595. doi:10.1016/j.neuron.2010.04.016

Haber, S. N., Kim, K. S., Mailly, P., & Calzavara, R. (2006). Reward-related cortical inputs define a large striatal region in primates that interface with associative cortical connections, providing a substrate for incentive-based learning. *J Neurosci, 26*(32), 8368-8376.

doi:10.1523/JNEUROSCI.0271-06.2006

Hornak, J., O'Doherty, J., Bramham, J., Rolls, E. T., Morris, R. G., Bullock, P. R., & Polkey, C. E. (2004). Reward-related reversal learning after surgical excisions in orbito-frontal or dorsolateral prefrontal cortex in humans. *J Cogn Neurosci, 16*(3), 463-478. doi:10.1162/089892904322926791

Izquierdo, A., Suda, R. K., & Murray, E. A. (2004). Bilateral orbital prefrontal cortex lesions in rhesus monkeys disrupt choices guided by both reward value and reward contingency. *J Neurosci, 24*(34), 7540-7548. doi:10.1523/JNEUROSCI.1921-04.2004

Jones, B., & Mishkin, M. (1972). Limbic lesions and the problem of stimulus--reinforcement associations. *Exp Neurol, 36*(2), 362-377.

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*(6109), 953-956. doi:10.1126/science.1227489

Kennerley, S. W., Behrens, T. E., & Wallis, J. D. (2011). Double dissociation of value computations in orbitofrontal and anterior cingulate neurons. *Nat Neurosci, 14*(12), 1581-1589. doi:10.1038/nn.2961

Kennerley, S. W., & Wallis, J. D. (2009). Evaluating choices by single neurons in the frontal lobe: outcome value encoded across multiple decision variables. *Eur J Neurosci, 29*(10), 2061-2073. doi:10.1111/j.1460-9568.2009.06743.x

Keramati, M., Smittenaar, P., Dolan, R. J., & Dayan, P. (2016). Adaptive integration of habits into depth-limited planning defines a habitual-goal-directed spectrum. *Proc Natl Acad Sci U S A*. doi:10.1073/pnas.1609094113

Kilian, J., & Siegelmann, H. T. (1996). The dynamic universality of sigmoidal neural networks. *Information and Computation, 128*(1), 48-56. doi:DOI 10.1006/inco.1996.0062

Laje, R., & Buonomano, D. V. (2013). Robust timing and motor patterns by taming chaos in recurrent neural networks. *Nat Neurosci, 16*(7), 925-933. doi:10.1038/nn.3405

Maass, W., Natschlager, T., & Markram, H. (2002). Real-time computing without stable states: a new framework for neural computation based on perturbations. *Neural Comput, 14*(11), 2531-2560. doi:10.1162/089976602760407955

O'Neill, M., & Schultz, W. (2015). Economic risk coding by single neurons in the orbitofrontal cortex. *J Physiol Paris, 109*(1-3), 70-77. doi:10.1016/j.jphysparis.2014.06.002

Padoa-Schioppa, C. (2011). Neurobiology of economic choice: a good-based model. *Annu Rev Neurosci, 34*, 333-359. doi:10.1146/annurev-neuro-061010-113648

Padoa-Schioppa, C. (2013). Neuronal origins of choice variability in economic decisions. *Neuron, 80*(5), 1322-1336. doi:10.1016/j.neuron.2013.09.013

Padoa-Schioppa, C., & Assad, J. A. (2006). Neurons in the orbitofrontal cortex encode economic value. *Nature, 441*(7090), 223-226. doi:10.1038/nature04676

Rescorla, R. A., & Wagner, A. R. (1972). A theory of Pavlovian conditioning: Variations in the effectiveness of reinforcement and nonreinforcement. In A. H. Black & W. F. Prokasy (Eds.), *Classical conditioning II: Current research and theory* (pp. 64-99). New York: Appleton-Century-Crofts.

Riceberg, J. S., & Shapiro, M. L. (2012). Reward stability determines the contribution of orbitofrontal cortex to adaptive behavior. *J Neurosci, 32*(46), 16402-16409. doi:10.1523/JNEUROSCI.0776-12.2012

Rodriguez, P. (2001). Simple recurrent networks learn context-free and context-sensitive languages by counting. *Neural Comput, 13*(9), 2093-2118. doi:10.1162/089976601750399326

Rolls, E. T., Critchley, H. D., Mason, R., & Wakeman, E. A. (1996). Orbitofrontal cortex neurons: role in olfactory and visual association learning. *J Neurophysiol, 75*(5), 1970-1981.

Royer, S., & Pare, D. (2003). Conservation of total synaptic weight through balanced synaptic depression and potentiation. *Nature, 422*(6931), 518-522.

Rudebeck, P. H., Mitz, A. R., Chacko, R. V., & Murray, E. A. (2013). Effects of amygdala lesions on reward-value coding in orbital and medial prefrontal cortex. *Neuron, 80*(6), 1519-1531. doi:10.1016/j.neuron.2013.09.036

Rudebeck, P. H., Saunders, R. C., Prescott, A. T., Chau, L. S., & Murray, E. A. (2013). Prefrontal mechanisms of behavioral flexibility, emotion regulation and value updating. *Nat Neurosci, 16*(8), 1140-1145. doi:10.1038/nn.3440

Rustichini, A., & Padoa-Schioppa, C. (2015). A neuro-computational model of economic decisions. *J Neurophysiol, 114*(3), 1382-1398. doi:10.1152/jn.00184.2015

Schoenbaum, G., Saddoris, M. P., & Stalnaker, T. A. (2007). Reconciling the roles of orbitofrontal cortex in reversal learning and the encoding of outcome expectancies. *Ann N Y Acad Sci, 1121*, 320-335. doi:10.1196/annals.1401.001

Schultz, W., Dayan, P., & Montague, P. R. (1997). A neural substrate of prediction and reward. *Science, 275*(5306), 1593-1599.

Sussillo, D., & Abbott, L. F. (2009). Generating coherent patterns of activity from chaotic neural networks. *Neuron, 63*(4), 544-557.

Suykens, J. A. K., Vandewalle, J., & Moor, B. L. R. d. (1996). *Artificial neural networks for modelling and control of non-linear systems*. Boston: Kluwer Academic Publishers.

Takahashi, Y. K., Roesch, M. R., Wilson, R. C., Toreson, K., O'Donnell, P., Niv, Y., & Schoenbaum, G. (2011). Expectancy-related changes in firing of dopamine neurons depend on orbitofrontal cortex. *Nat Neurosci, 14*(12), 1590-1597. doi:10.1038/nn.2957

Tsujimoto, S., Genovesio, A., & Wise, S. P. (2011). Comparison of strategy signals in the dorsolateral and orbital prefrontal cortex. *J Neurosci, 31*(12), 4583-4592. doi:10.1523/JNEUROSCI.5816-10.2011

Wallis, J. D., Anderson, K. C., & Miller, E. K. (2001). Single neurons in prefrontal cortex encode abstract rules. *Nature, 411*(6840), 953-956. doi:10.1038/35082081

Wallis, J. D., & Miller, E. K. (2003). Neuronal activity in primate dorsolateral and orbital prefrontal cortex during performance of a reward preference task. *Eur J Neurosci, 18*(7), 2069-2081.

Wilson, R. C., Takahashi, Y. K., Schoenbaum, G., & Niv, Y. (2014). Orbitofrontal cortex as a cognitive map of task space. *Neuron, 81*(2), 267-279. doi:10.1016/j.neuron.2013.11.005

Figures

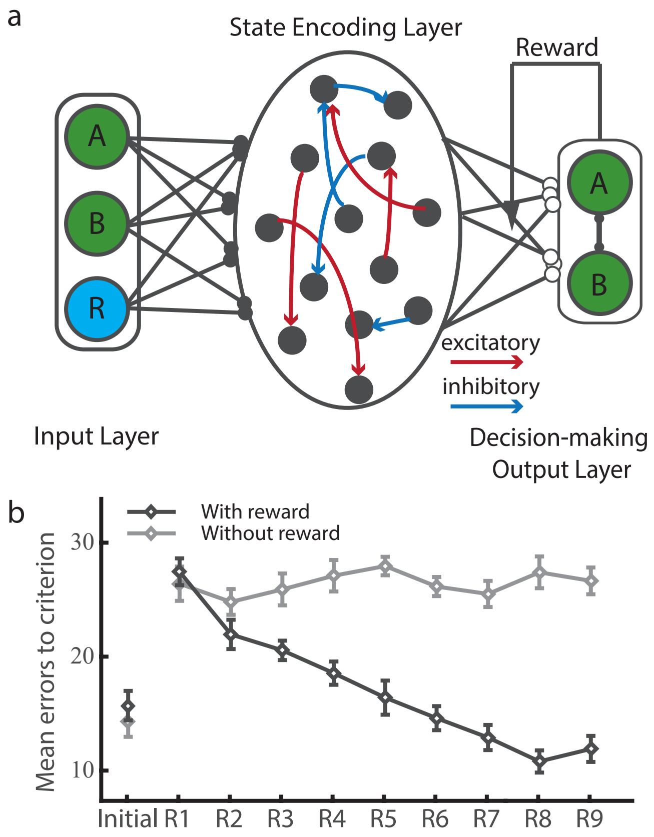

Figure 1. (a) The schematic diagram of the model. The network is composed of three parts: input layer (IL), the state encoding layer (SEL) and the decision-making output layer (DML). (b) The number of the error trials made before the network achieves the performance threshold. The dark line indicates the performance of the network with the reward input; the light line indicates the performance of the network without the reward input as a model for animals of OFC lesions.

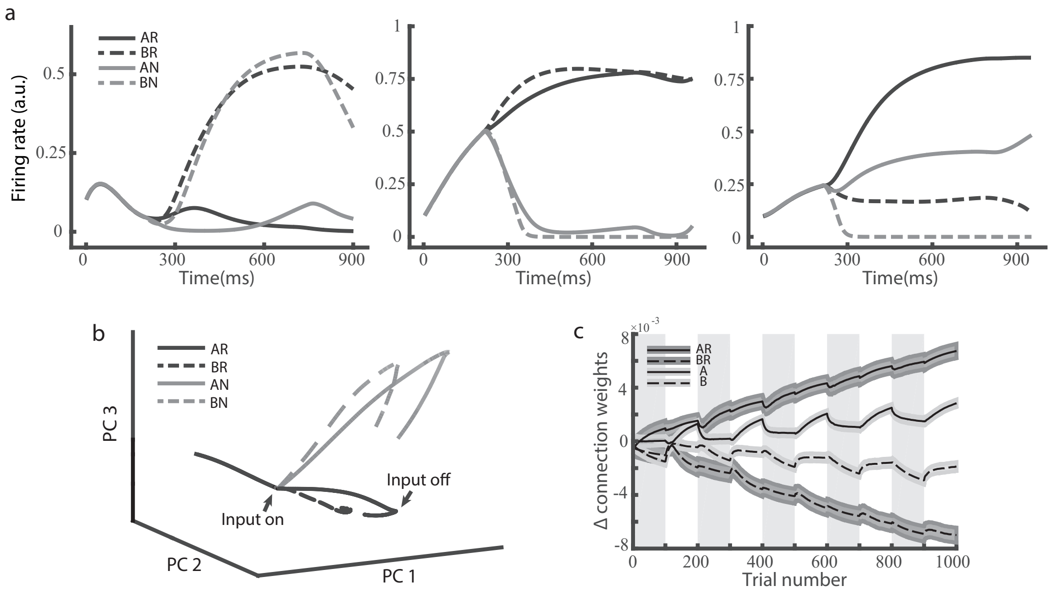

Figure 2. (a) Selectivity of three example neurons in the reservoir network. Input units are set to 1 from 200ms to 700ms. Left panel: an example neuron that encodes choice options; middle panel: an example neuron that encodes reward outcomes; right panel: an example neuron with mixed selectivity. (b) PCA on the network population activity. The network states are plotted in the space spanned by the first 3 PCA components. The activities in different conditions are differentiated after the cue onset. (c) The difference between the connection weights between SEL neurons and the DML unit *A* and DML unit *B*. The SEL neurons are grouped according to their selectivities. For example, *AR* represents the group of neurons that respond most strongly when the input units *A* and *R* are both activated. The gray and white area indicates the blocks in which the option *A* and the option *B* leads to the reward, respectively.

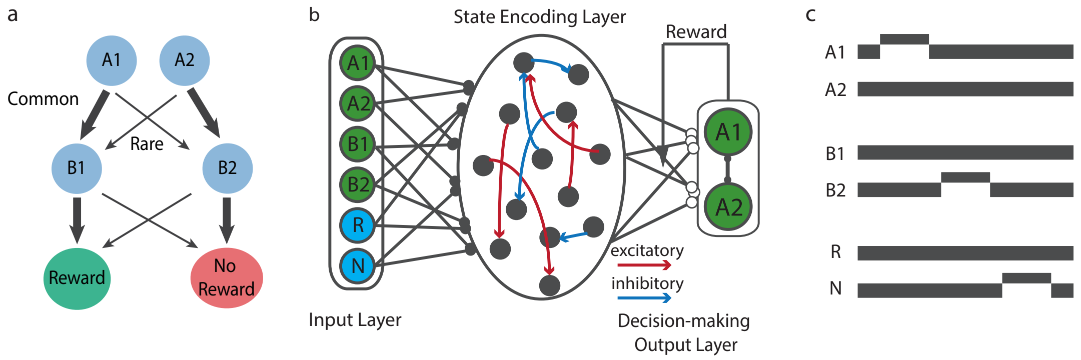

Figure 3. (a) Task structure of the two-stage Markov decision task. Two options *A1* and *A2* are available, they lead to two intermediate outcomes *B1* and *B2* at different probabilities. The width of the arrows indicates the transition probability. Intermediate outcomes *B1* and *B2* lead to rewards at different probability, but the reward contingency of the intermediate outcomes is reversed between blocks. (b) The schematic diagram of the model. It is similar to Fig 1a. The only difference is that there are more input units. (c) Units in the input layer are activated sequentially. In the example trial, option *A1* is chosen, *B2* is presented, and no reward is obtained.

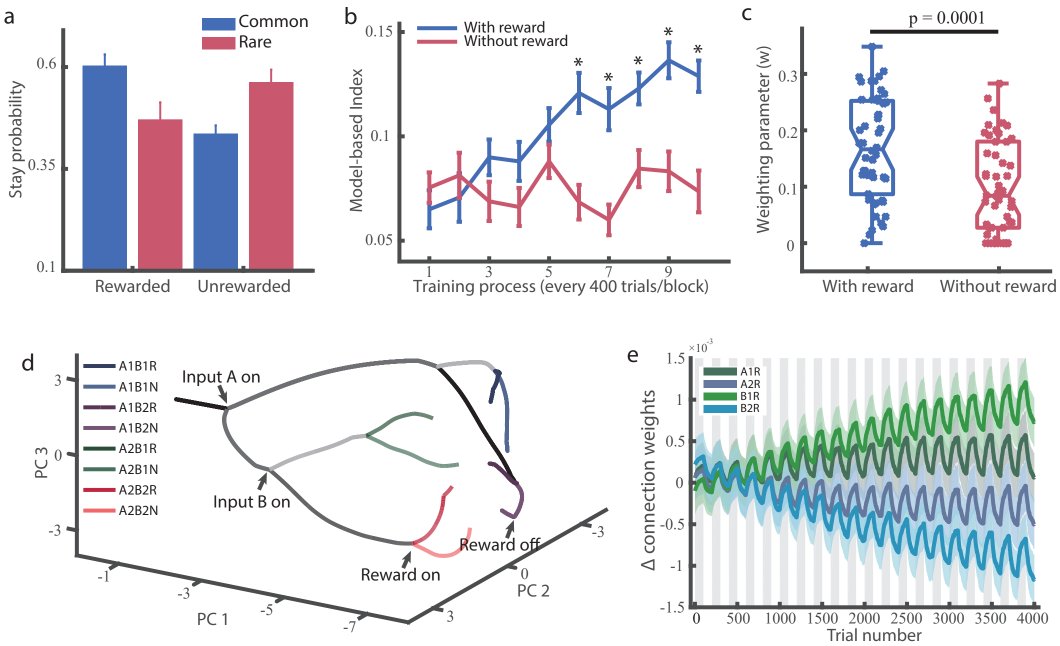

Figure 4. (a) Factorial analysis of choice behavior. The agent is more likely to repeat the choice under the conditions common-rewarded (*CR*) and rare-unrewarded (*RN*) than under the conditions common-unrewarded (*CU*) and rare-rewarded (*RR*). (b) MB index keeps growing in the intact network (blue line), but stays at a low level when the network is without its reward input (red line). (c) Fitting the behavioral performance with a mixture of model-free and model-based algorithms. The weight parameter *w* for model-based learning is significantly larger for the intact network (blue data points) than the network without its the reward input (red data points). Each data point represents a simulation run. (d) PCA on the network population activity. The network states are plotted in the space spanned by the first 3 PCA components. The network can distinguish all 8 different states. (e) The weight differences between the connections between SEL neurons and the DML unit *A1* and DML unit *A2*. Similar to Fig 2c. The gray and white areas indicate the blocks in which intermediate outcome *B1* is more likely to lead to a reward and the blocks in which *B2* is more likely to lead to a reward, respectively.

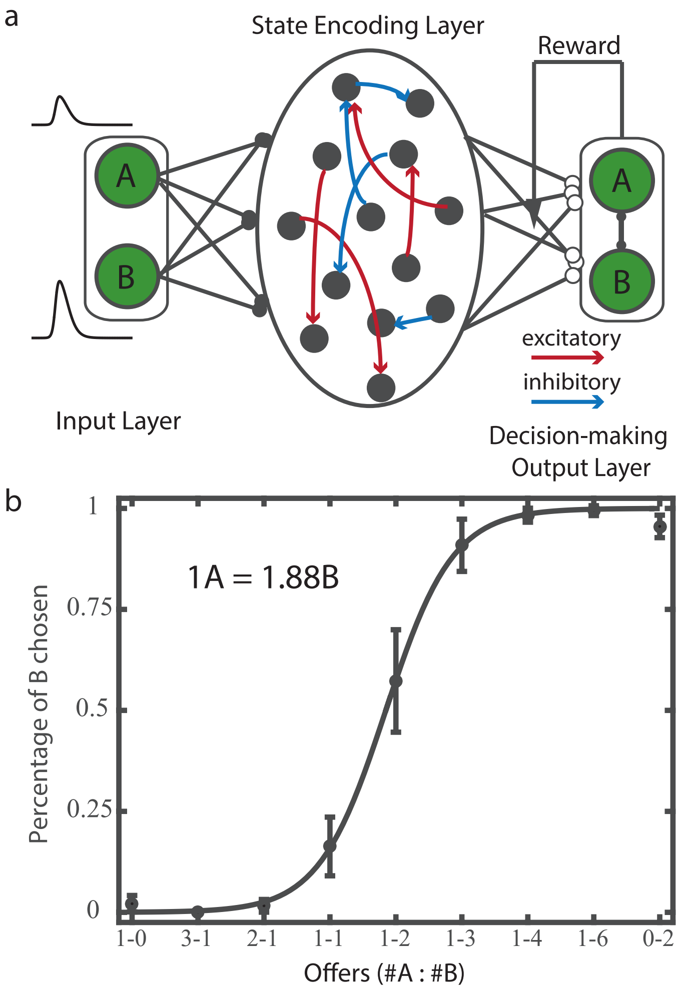

Figure 5. (a) The schematic diagram of the model. The input neurons' responses are not a step function as in the previous paradigms, illustrated in the left side of the panel. (b) Choice pattern. The relative value preference calculated based on the network behavior is indicated on the top left, and the actual relative value preference used in the simulation is $1A = 2B$.

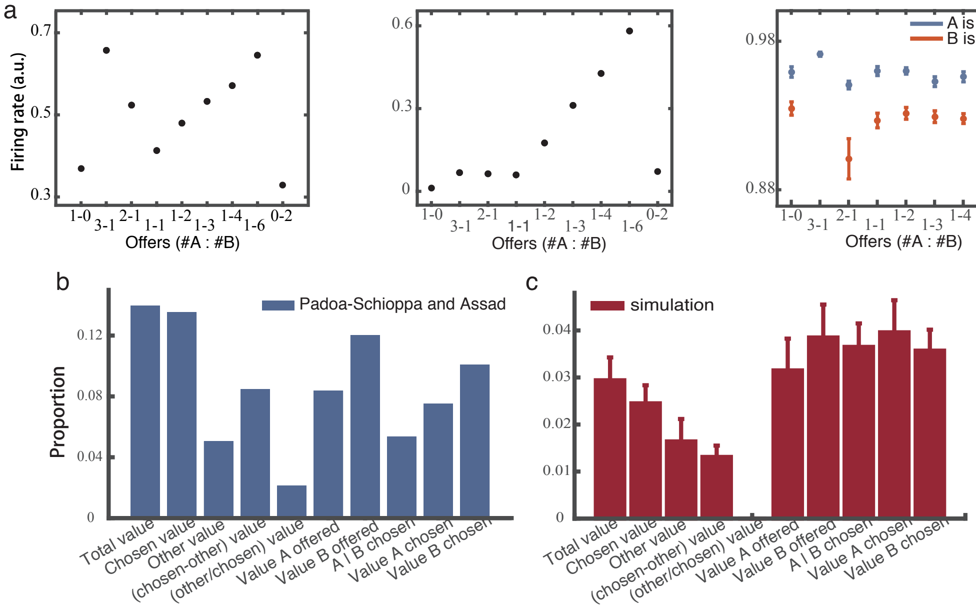

Figure 6. (a) Three example neurons in the SEL. Left panel: a neuron that encodes chosen value; middle panel: a neuron that encodes offer value; right panel: a neuron that encodes chosen juice. (b) The proportions of the neurons with different selectivities from a previous experimental study (Padoa-Schioppa & Assad, 2006). (c) The proportions of the neurons in our reservoir network with different selectivities.