Which Temporal Difference Learning Algorithm Best Reproduces Dopamine Activity in a Multi-choice Task?

Jean Bellot[1], Olivier Sigaud[1], and Mehdi Khamassi[1,2]

[1] Institut des Systemes Intelligents et de Robotique (ISIR), Universite Pierre et Marie Curie (UPMC), 4 place Jussieu, 75005 Paris, France
[2] UMR 7222, Centre National de la Recherche Scientifique (CNRS), France
{jean.bellot,olivier.sigaud,mehdi.khamassi}@isir.upmc.fr

Abstract. The activity of dopaminergic (DA) neurons has been hypothesized to encode a reward prediction error (RPE) which corresponds to the error signal in *Temporal Difference (TD) learning* algorithms. This hypothesis has been reinforced by numerous studies showing the relevance of *TD learning* algorithms to describe the role of basal ganglia in classical conditioning. However, recent recordings of DA neurons during multi-choice tasks raised contradictory interpretations on whether DA's RPE signal is action dependent or not. Thus the precise TD algorithm (i.e. Actor-Critic, Q-learning or SARSA) that best describes DA signals remains unknown. Here we simulate and precisely analyze these TD algorithms on a multi-choice task performed by rats. We find that DA activity previously reported in this task is best fitted by a *TD error* which has not fully converged, and which converged faster than observed behavioral adaptation.

Keywords: dopamine, reinforcement learning, reward prediction error, behavioral adaptation, instrumental conditioning.

1 Introduction

The work of Wolfram Schultz and colleagues during the 90s has highlighted the link between the information carried by the activity of dopaminergic (DA) neurons and the error signal computed by *Temporal Difference (TD) learning* algorithms [1,2,3]. However, most experiments involved *Pavlovian conditioning*, where the animal remains passive during the 2 seconds delay between the stimulus and the reward. In contrast, several different *TD learning* algorithms have been proposed with a different way of encoding the choice of actions (i.e. Actor-Critic, Q-learning, SARSA) and which cannot be discriminated based on these data [4].

More recent studies have focused on DA activity during multi-choice tasks, where animals learn to perform the right actions in order to obtain reward [5,6]. This enables to investigate whether the RPE signal in DA neurons is action-dependent or only depends on the conditioned stimuli. With these recent

T. Ziemke, C. Balkenius, and J. Hallam (Eds.): SAB 2012, LNAI 7426, pp. 289–298, 2012.

protocols, one can compare the ability of different *TD learning* algorithms to reproduce the activity patterns of DA neurons. However, these studies convey contradictory conclusions.

In [5], monkeys had to choose among two conditioned stimuli presented on a screen, each stimulus being associated to reward with a different probability. This time, the RPE carried by recorded DA neurons appeared to depend on the action the animal would subsequently perform. This RPE signal appeared to be consistent with the SARSA algorithm.

In [6], rats had to choose between two wells delivering two different rewards (large versus small reward; or delayed versus immediate reward). In each trial, an odor (conditioned stimulus) was presented, carrying the information enabling to identify which reward was available in each well. The progressive learning of stimulus-reward associations and changes in these associations enabled to analyze the type of RPE that was encoded by DA neurons during the task. The authors found that DA neurons are encoding an error depending on the value of the current best option, not matter whether the rat would subsequently perform that option or select the wrong option. Such type of RPE is compatible with the Q-LEARNING algorithm.

Thus, the conclusions of both studies are inconsistent. But none of them did attempt to compare DA activity with empirical simulations of the algorithms. Here, we simulate diverse basic *TD learning* algorithms in order to determine which of them best reproduces the results obtained by [6]. We first describe the model used to simulate the multi-choice task studied in [6] and the *TD learning* algorithms. Then we focus on reproducing the behavioral data and the dopamine activity depending on the meta-parameters of the algorithms.

2 Material and Methods: Computational Model

In this section, we first describe the computational model used to simulate the task of [6]. Then we present the three *TD learning* algorithms introduced in [4].

2.1 Modelling the Blocks

We have modelled the experiments of [6] with a Markov Decision Process (MDP) (see Fig.1): in a given block of trials one well (right or left) delivers the best reward, the other contains the worst one (big vs small in the size condition; immediate versus delayed in the delay condition). After nosepoking, the animal receives one of three odors: odor 1 informs it that only the left well delivers reward; with odor 2 only the right well is rewarding; odor 3 indicates a free-choice trial where the animal is rewarded everywhere but needs to find the best reward.

At each block change, there is a shift in the well that delivers the best reward and the animal learns the new place-reward association. In this work, we only present simulations of free-choice trials (where odor 3 is presented) which are crucial to discriminate between the competing algorithms.

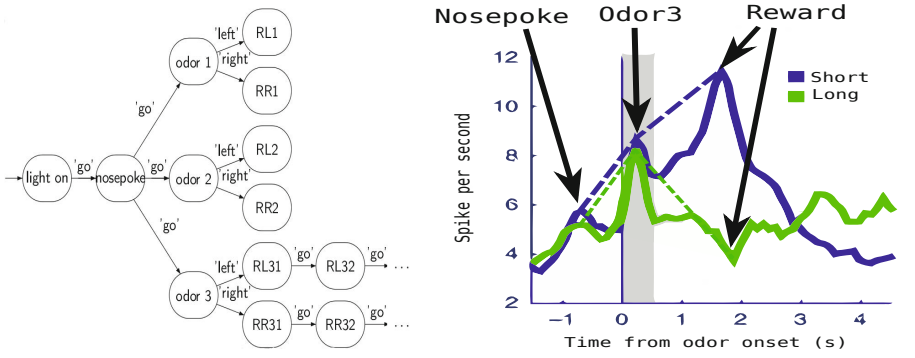

Fig. 1. Modelling the state of the task used in [6]. Left: Markov Decision Process used to model the task; RL31, Reward Left following odor 3; RR31, Reward Right following odor 3. The other states represent the delay. Right: State decomposition illustrated on DA activity reported by Roesch. We use the RPE calculated by the different algorithms at the three different states : 'nosepoke', 'odor3' and 'RL31' or 'RR31' (depending on the choice of the algorithm) to fit the DA activity extracted from the right graph.

The DA activity in Fig. 1 Right was obtained in this task by averaging the DA activity over all trials once the performance of rats went above 50%. The high response to the odor cue, independent from the future choice, is interpreted by the authors as consistent with a *TD error* calculated by Q-LEARNING [6]. However, since the behavioral performance converge quickly, the RPE should also have converged towards 0 at the time of the reward, as observed in Schultz's work [1]. This is not the case of this DA activity. It rather looks like an error that has not converged yet. Thus we simulate here the three concurrent TD-learning algorithms to empirically evaluate the nature of the RPE signal encoded by this DA activity.

2.2 Studied Algorithms

Our study of the performance of all algorithms is focused on the match between the evolution of the *TD error* and the DA activity. We compare three algorithms: Q-LEARNING, SARSA and ACTOR-CRITIC. Q-LEARNING and SARSA are based on the same principles. They update for each (s, a) pair a Q-table that stores the utility expectation for performing action a in state s. The ACTOR-CRITIC architecture contains a critic, i.e. a model of the value function V that stores the utility expectation from each state s, and an actor, the policy P which associates to any (s, a) pair the probability of choosing action a in state s (i.e. $P(a|s)$).

These value functions are updated from the *TD error* δ using $\forall f \epsilon \{Q, V, P\}$ $f_{t+1} = f_t + \alpha\delta_t$. But the computation of the *TD error* differs depending on the algorithm. The update rule is:

- Q-LEARNING: $\delta_t = r_{t+1} + \gamma \max_a(Q(s_{t+1}, a)) - Q(s_t, a_t)$
- SARSA: $\delta_t = r_{t+1} + \gamma Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)$
- ACTOR-CRITIC: $\delta_t = r_{t+1} + \gamma V(s_{t+1}) - V(s_t)$

For action selection, we use a *softMax* policy which chooses an action with a probability proportional to the value of this action: $P(a|s_t) = \frac{\exp(\beta Q(s_t,a))}{\sum_b \exp(\beta Q(s_t,b))}$.

3 Reproduction of Behavioral Results

With the model described above we first fit the behavioral data from [6] to determine which set of parameters can better reproduce the learning dynamics of the rats.

3.1 Methods

In order to reproduce the behavioral results of [6], the simulated agent learns during 30 trials of a block where the best reward is on the left (block 1 or 4) using the above algorithms. From these trials, the agent learns the block (it goes more often to the best reward). After this initial learning stage, a block change occurs where the side of the best reward is reversed (right instead of left). The behavior of the agent is compared to that of the animals from 15 trials before the block change to 30 trials after. The performance is computed as the number of left choices. It is averaged over 100 simulated agents.We test different meta-parameter sets of the algorithms:

- *max_iter*: 100
- α: from 0.1 to 0.9 with 0.1 steps,
- β: from 0.1 to 0.9 with 0.1 steps,
- γ: $\epsilon[0.6, 0.7, 0.8, 0.9]$.
- After empirical tuning, we set the reward value to 5.

The obtained results are compared to those of [6] (see Fig. 2) by minimizing the squared error between a set of points over the curves. The points are extracted from the curves of [6] in the *size* case and the *delay* case. Then we look for the set of meta-parameters that optimize the match in both cases (see Fig. 2 for the *delay* case).

3.2 Results

Figure 2 Left shows the reproduction of the behavior with Q-LEARNING, SARSA and ACTOR-CRITIC. We obtain the following optimal meta-parameter sets for the different algorithms:

- Q-LEARNING: $\alpha = 0.3, \beta = 0.3, \gamma = 0.7$
- SARSA: $\alpha = 0.3, \beta = 0.3, \gamma = 0.7$
- ACTOR-CRITIC: $\alpha = 0.7, \beta = 0.8, \gamma = 0.7$

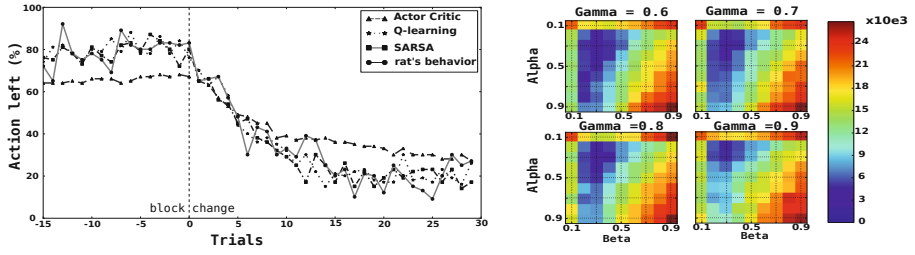

Fig. 2. Reproducing the behavior of the rat for the *delay* case with Q-LEARNING, SARSA and ACTOR-CRITIC. Left: reproduction of the behavior using the parameters obtained from the best fit with the behavioral data of Roesch et al. Right: squared error as a function of α, β and γ illustrated for the case of Q-LEARNING.

Figure 2 Right shows the squared error as a function of α, β and γ for the *delay* case (*size* case not shown). One can see that *Qlearning* and SARSA show a similar sensitivity to the parameters. The error is lower for α and β close to 0.3. ACTOR-CRITIC requires a larger α and β. As can be seen on Fig 2 , the smallest error for ACTOR-CRITIC is obtained for a β value near the tested limit (0.9). Thus we additionally test higher β values for ACTOR-CRITIC (1, 1.5 and 2). This does not change the results: the same parameter set enables the ACTOR-CRITIC model to give the best compromise between fitting the behavior during the delay condition and fitting the behavior during the size condition ($\alpha = 0.7$, $\beta = 0.8$ and $\gamma = 0.7$).

4 Comparing DA Activity with *TD Error*

In this section, we describe how we compare the DA activity to the *TD error* depending on the algorithms' meta-parameters.

4.1 Methods

Based on the parameters obtained from the behavioral results, we then investigate whether the *TD error* computed in simulation can match the DA activity observed in rats. The idea is that, if DA activity reflects the RPE signal of the learning algorithm by which rats learn the task, algorithms tuned to fit the behavior should display the same pattern of activity as responses of DA neurons.

Thus we compare the reported DA activity with the *TD error* computed by the different algorithms. We fit three states of our MDP with three points of the experimental curve (see Fig. 1 Right), corresponding to moments where the rat: (1) touches the port, *nosepokes*; (2) receives an odor, *odor*; (3) receives the reward, *reward*.

The DA activity and the *TD error* do not share a common scale. Thus, we minimize with least square the difference $||(a\delta_s + b) - R_s||^2$ where R_s is the experimental DA activity in state s and δ_s is the average *TD error* computed in s over the different trials. Thus we have: $\delta_s = \frac{1}{n}\Sigma_{e=0}^{n}\delta_s(e)$, where n is the number of considered trials and $\delta_s(e)$ is the *TD error* computed from the e^{th} trial in s. The (a, b) pair is determined with the least square method.

4.2 Results

The curves in [6] are obtained by averaging the DA activity over all trials once the performance of rats is over 50%. In the *delay* case, this happens after the fifth trial after the block changed (see Fig. 2 Left). The *TD error* should have converged towards 0 over learning. This is not the case of the DA activity in [6]. It rather looks like an error that has not converged yet because the reported response of DA neurons to rewards does not vanish with learning. Thus we look for a temporal window where the *TD error* may behave like the DA activity recorded in [6]. More precisely, we vary the number of trials considered in our average on δ_s so as to match the DA activity as well as we can. Fig. 3 Left shows the squared error as a function of this number of trials with Q-LEARNING.

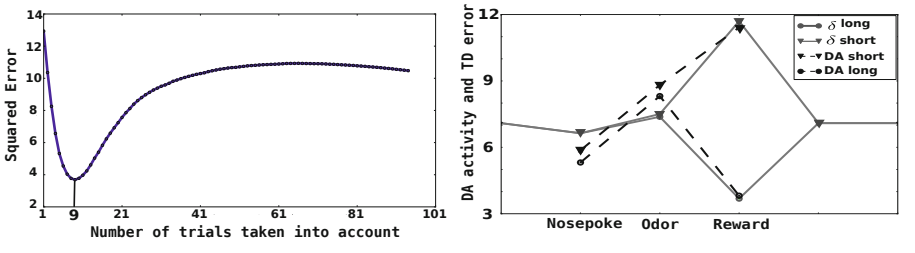

Fig. 3. Left: Evolution of the error obtained when fitting DA activity with the *TD error* in function of the number of trials taken into account in the calculation of the averaged RPE. Right: Best fit of the DA activity recorded in [6] during the delay case, from the *TD error* computed with the Q-LEARNING algorithm and averaged over the first 9 trials (minimizing the error as shown in Left).

The results are consistent with the conclusions of [6], since they show that, in the *delay* case, Q-LEARNING is the algorithm that best matches the DA activity. However, this is the case only if we just consider the 9 first trials after the performance got over 50%. When we take all the 20 *free-choice* trials used in [6], the error is much larger. The same applies to SARSA. But SARSA has different errors for the two odors, which is not observed in DA activity.

In the ACTOR-CRITIC case, the high fitting error is mainly due to the mismatch at the nosepoke state. Like other algorithms, the ACTOR-CRITIC's RPE

signal which best fits DA activity is produced by a learning process that has already converged (even more strongly converged due to the high α, see Fig. 2). Thus the RPE signal is almost flat and requires a high amplification factor a to be compared to DA activity. This high a amplifies the noise at the nosepoke state.

In Table 1, we report the error of each algorithm as well as the optimal number of trials n to be considered when computing the average *TD error*. In the *size* case, none of the algorithms obtains a low error. Thus this case is reproduced worse than the *delay* case.

Table 1. Squared error (e^2) and percentage of error (e%) obtained with the different algorithms with respect to the value in [6]. This latter value is computed as $e\% = \frac{1}{n}\Sigma_i \frac{|d_i - s_i|}{d_i}$ where d_i is the value on [6]'s curve at instant i, s_i is the value obtained in simulation and n is the number of points.

	size			delay		
Algorithm	n	e^2	$e\%$	n	e^2	$e\%$
Qlearning	4	8.2	14.6%	9	3.8	10.7%
SARSA	3	8.2	14.5%	3	9	16.9%
ACTOR-CRITIC	1	10.1	15.5%	41	8.3	16%

In summary, the results show that although Q-LEARNING obtains better results, as expected by [6], the three algorithms, when fitted on the rat's behavior, have too much converged to reproduce the observed pattern of responses of DA neurons. This suggests that the rate with which behavior is adapted may be different from the rate with which the RPE signal encoded by DA neurons is learned, as if their responses were not tightly linked to the behavior. To assess this simple interpretation, we next test the algorithms after releasing the constraint on the fit with the behavior.

4.3 Optimization of Parameters over DA Activity Only

So far, we have used the same parameters for reproducing behavioral results and DA activity, considering that the behavior of the rat was directly driven by the *TD error*. This assumption was consistent with other studies in the literature [7,8]. Nevertheless, from Fig. 2, it is clear that the DA activity cannot be fitted with a *TD error* that has converged, whereas the behavior itself has converged.

In order to test the assumption that DA activity may reflect a learning dynamics slower than the one reflected in behavior, we now fit this activity with the model without constraining the meta-parameters on the behavioral data. However, we restrict the matching process to the trials where the behavior of the rat is above 50% of correct choice. Thus we cannot avoid at least a minor influence of the behavior in this fitting process.

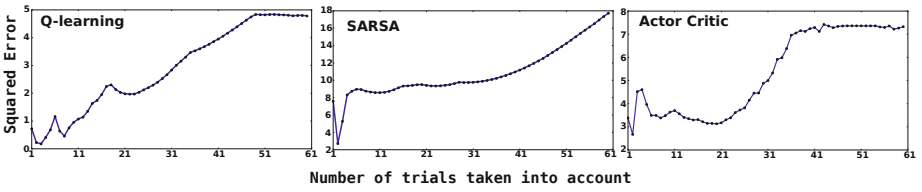

Fig. 4. Squared error as a function of the number of trials, between the DA activity from [6] and the *TD error* computed from different algorithms with free parameters. Left: Q-LEARNING, Middle: SARSA, Right: ACTOR-CRITIC.

Under this new condition, we obtain a better fit than previously (see Fig.4). These results show a large difference between the algorithms, in terms of their capability to reproduce the DA activity as a function of the parameters. As previously, Q-LEARNING can fit DA data. SARSA cannot do so as well as Q-LEARNING. Finally, ACTOR-CRITIC obtains a better performance than for previous results, performing comparably with Q-LEARNING. Indeed, if we only consider the 20 first trials (corresponding to the number of *free-choice* used in [6]), then the error and the corresponding meta-parameters are given in Table 2.

Table 2. Meta-parameters when fitting only with DA activity

	Qlearning				SARSA				ACTOR-CRITIC			
	α	β	γ	error	α	β	γ	error	α	β	γ	error
size	0.8	0.9	0.6	7.2	0.1	0.1	0.9	8.1	0.2	0.9	0.7	9.6
delay	0.1	0.6	0.9	2.0	0.1	0.5	0.9	9.5	0.1	0.1	0.9	3.4

Globally, one can see that to get a minimal error with respect to DA activity with the three algorithms in this task, the meta-parameters have to differ from the ones used to match behavioral results. In particular, the learning rate must be lower so that the value does not converge too quickly. One can conclude that the DA activity is compatible with a *TD error* computed by Q-LEARNING or ACTOR-CRITIC in the *delay* case. SARSA is a much less likely candidate algorithm with these data. In the *size* case, the three algorithms still cannot reproduce DA activity satisfyingly (see Table 2) which are discussed below.

5 Discussion

In this work, we have tried to fit DA activity observed during a multi-choice task with various RL algorithms. The starting hypothesis, initially resulting from DA recordings in passive monkeys, was that the response of these neurons would encode an RPE similar to the error signal used in RL [1].

Here we studied the link between the information carried by DA neurons recorded by [6] and the error computed by Q-LEARNING, SARSA and ACTOR-CRITIC algorithms in a task where animals are to perform active action selection. We found that none of these algorithms could satisfyingly reproduce the observed patterns of responses when keeping the behavioral parameters. However, when the learning rate of behavioral adaptation and the learning rate of the adaptation of the expected value were dissociated, we found that Q-LEARNING and ACTOR-CRITIC could both reproduce DA activity during the *delay* condition while SARSA could not.

The *size* condition remains problematic because after a block change, DA activity reported by [6] does not reflect the reversal of the contingencies: instead of displaying a negative RPE when the reward is worst than previously expected, the response of DA neurons to the small reward remains high. This pattern prevents the standard RL algorithms from reproducing neural activity.

Another important issue is the global increase of DA activity along the trial, getting higher when time gets close to reward delivery (see Fig. 6 in [6]). At first glance, this could look like a learned value function instead of an RPE. This possible confusion between value and RPE is reflected by the frequent usage of the term "value" instead of "RPE" in the original article [6]. It could be interesting to see whether the simulated value function of the tested algorithm can contribute to the reproduction of DA activity in this task. However, this would be inconsistent with the now well established theory that the phasic responses of DA neurons encode an RPE [1,9,10,11,12,13].

The alternative interpretation that we propose and whose plausibility is confirmed by our empirical simulations is that the DA signal recorded by [6] may correspond to an RPE that has not yet fully converged while the animals behavior has already converged. In our simulations, Q-LEARNING and ACTOR-CRITIC algorithms could fit DA activity during the delay condition when the simulated error signal was averaged only during early learning trials. The fact that we cannot fit DA activity when considering all post-learning trials seems to reveal that the observed choice behavior of the animal has a different dynamics of adaptation than the learning process encoded by DA neurons. First, this suggests that instead of only reporting the averaged post learning DA activity, showing the trial-by-trial evolution of DA's response accross learning may be more informative, and may lead to different conclusions on which algorithm among Actor-Critic, Q-learning and SARSA best describes the activity. Second, this suggests that the observed behavior is not the direct consequence of a unique learning system that we suppose relies on the recorded DA activity. This could indicate the presence of a second parallel decision system which speeds up the behavioral adaptation: the behavior would result from the influence of a cortically-driven fast learning process while the slower habitual learning subserved via DA neurons in the striatum would take more time to converge. This idea would be consistent with the proposal of dual decision-making systems subserving parallel learning processes for goal-directed behaviors and habits in mammals [8,14]:

a fast model-based RL system combined with a slower model-free system such as TD-learning algorithms studied here.

In future work, it would be interesting to test the ability of such dual system model to explain both behavioral adaptation and DA activity reported in [6]. A simpler alternative explanation that we could compare would be that the Actor and the Critic underlying behavioral adaptation in this task may have different learning rates.

Acknowledgements. We are very grateful to Matthew R. Roesch and Geoffrey Schoenbaum for useful discussions. This work was supported by: L'Agence Nationale de la Recherche ANR-11-BSV4-0006 "LU2" (Learning Under Uncertainty) project.

References

1. Schultz, W., Dayan, P., Montague, P.R.: A neural substrate of prediction and reward. Science 275(5306), 1593–1599 (1997)
2. Hollerman, J.R., Schultz, W.: Dopamine neurons report an error in the temporal prediction of reward during learning. Nat. Neurosci. 1(4), 304–309 (1998)
3. Schultz, W.: Predictive reward signal of dopamine neurons. Journal of Neurophysiology 80(1), 1–27 (1998)
4. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction. The MIT Press (March 1998)
5. Morris, G., Nevet, A., Arkadir, D., Vaadia, E., Bergman, H.: Midbrain dopamine neurons encode decisions for future action. Nat. Neurosci. 9(8), 1057–1063 (2006)
6. Roesch, M.R., Calu, D.J., Schoenbaum, G.: Dopamine neurons encode the better option in rats deciding between differently delayed or sized rewards. Nat. Neurosci. 10(12), 1615–1624 (2007)
7. Tanaka, S.C., Doya, K., Okada, G., Ueda, K., Okamoto, Y., Yamawaki, S.: Prediction of immediate and future rewards differentially recruits cortico-basal ganglia loops. Nature Neuroscience 7(8), 887–893 (2004)
8. Daw, N.D., Niv, Y., Dayan, P.: Uncertainty-based competition between prefrontal and dorsolateral striatal systems for behavioral control. Nat. Neurosci. 8(12), 1704–1711 (2005)
9. Bayer, H.M., Glimcher, P.W.: Midbrain dopamine neurons encode a quantitative reward prediction error signal. Neuron 47(1), 129–141 (2005)
10. Niv, Y., Daw, N.D., Dayan, P.: Choice values. Nature Neuroscience 9(8), 987–988 (2006)
11. Daw, N.D.: Dopamine: at the intersection of reward and action. Nat. Neurosci. 10(12), 1505–1507 (2007)
12. Niv, Y., Schoenbaum, G.: Dialogues on prediction errors. Trends in Cognitive Sciences 12(7), 265–272 (2008)
13. Matsumoto, M., Hikosaka, O.: Two types of dopamine neuron distinctly convey positive and negative motivational signals. Nature 459(7248), 837–841 (2009)
14. Keramati, M., Dezfouli, A., Piray, P.: Speed/Accuracy Trade-Off between the habitual and the Goal-Directed processes. PLoS Comput. Biol. 7(5), e1002055 (2011)