**Supplementary Materials**

**Experiment 2b**

???

**Regression analyses**

In order to capture trial-by-trial variation in the magnitude of the reward obtained on the setup trial, we regressed participants’ critical trial choices on the reward using a logistic mixed-effects model, estimating both random intercepts and random slopes at the subject level. (Following past research (CITE), this model approximates the value representation of a prediction error update mechanism as the most recently observed reward. In simulations presented below we validate this approximation.)

All models had one regressor: the value of the reward obtained on setup trials. The reward regressor was grand mean centered. The dependent variable was participant choice on the subsequent critical trials, coded as 1 if participants selected the shared-goal action, and 0 otherwise. Thus a positive coefficient indicates that participants were more likely to select the shared-goal action following higher reward on the setup trial.

In order to achieve convergence, models did not allow correlation between the random slope and random intercept. We determined whether the regressor increased the model’s likelihood enough to justify inclusion by calculating a null model with the regressor removed, and comparing models using a likelihood ratio test. All mixed-effects analyses were conducted in R (54), making use of the lme4 linear mixed effects package (55).

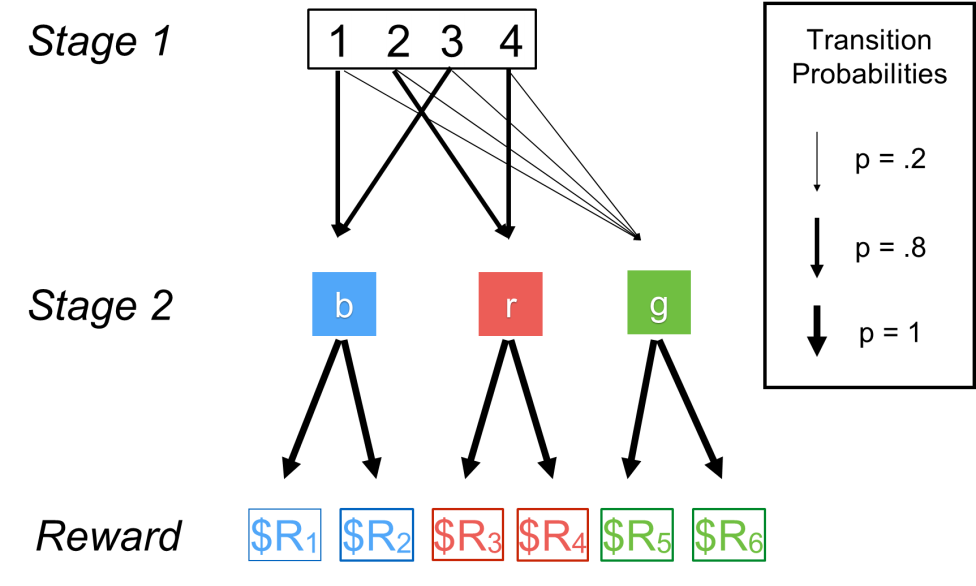
In each experiment, the reward obtained on the setup trial significantly predicted choice. The parameter estimates and significance tests for the mixed-effects models are presented in SM Table 1 below. β is the coefficient of the reward regressor, χ2 is the statistic value in the likelihood ratio test, and p is the significance level of the likelihood ratio test.

|  |  |  |  |
| --- | --- | --- | --- |
| Experiment | β | χ2 | p |
| 1a |  |  |  |
| 1b |  |  |  |
| 2a |  |  |  |
| 2b |  |  |  |

SM Table 1: Parameter estimates and significant tests for the mixed-effects models.

**Computational model**

We specified a computational model of learning and choice that includes model-free goal learning alongside traditional model-based and model-free control. Using this model to generate simulated data in our task, we show that our observed results are obtained only when the model includes model-free goal learning. By comparing our mechanism’s performance and computational efficiency to that of traditional mechanisms, we also show that our mechanism balances model-based accuracy with model-free efficiency.

We generated simulated data for Experiment 1b. The task is a Markov decision process with ten states: The initial Stage 1 state, three Stage 2 states, and six reward states (SM Figure 1). The Stage 1 state had four possible actions, only two of which were available on any given trial. Each Stage 2 state had two available actions which led to reward states. The rewards were randomly generated for each agent by the same process as in the behavioral tasks. Each agent completed 175 trials. Although agents made choices in both Stage 1 and 2, we focus exclusively on the Stage 1 choice because it effectively juxtaposes the three mechanisms.

SM Fig 1: The Markov decision task used in the simulations. Squares represent states and arrows represent actions.

We implemented model-free goal learning with the options framework (CITE), a common framework for hierarchical policy abstraction. Under this framework, an “option” is a flexible policy which terminates upon attainment of a goal state. In our task, we defined two options: one which terminates at blue (denoted *O1*), and the other at red (*O2*).

An agent using these options faces two challenges. It must choose an option, and then successfully attain the chosen option’s goal state. Our proposed mechanism addresses the first challenge by maintaining a model-free value for each option, denoted *V(Oi)*. The values are initialized to zero and updated after every trial by:

where *Oi­* is the chosen option,*­ r* is the received reward, and *α* is a learning rate. Agents then choose options probabilistically in proportion to their value by entering the values into a softmax function (with temperature parameter *β*):

Once agents choose an option, they use model-based planning to achieve that option’s goal state. Agents maintained the transition probabilities from each Stage 1 action *a* to each Stage 2 state *s*, denoted *T(a,s)*. Since participants were told these probabilities explicitly and had extensive practice with them, we assumed that agents knew the correct transition probabilities. (Our results do not change if agents learn the probabilities from experience.) Agents selected action *a\** which maximized the likelihood of attaining the goal state *gi* of their chosen option *Oi*:

For comparison, agents also implemented a fully model-based hierarchical controller. The model-based controller was identical to the model-free goal learner except in its option evaluation mechanism. The model-based controller maintained values of each of the six reward states, denoted *V(sj)*, the transition probabilities from each Stage 2 action *a* to each reward state *sj*, denoted *T(a,sj)*, and the set of available actions in each Stage 2 goal state, denoted *A(gi)*.It calculated option value *V(Oi)* according to:

This model-based option evaluation mechanism is more accurate than our model-free mechanism because it disregards any rewards obtained from transitions to the green states. However, it comes at the computational cost of evaluating each possible Stage 2 pathway. That cost is minor in our simplified task, but in real-world scenarios it could be prohibitive.

Finally, to ensure that our results could only be the product of model-free *goal* learning, we implemented a traditional model-free action learner. We used Q-learning, a common model of human learning and decision making (CITE). Agents maintained a value for each state-action pair, denoted *Q(s,a)*. After choosing action *a* in state *s* and transitioning to state *s’*, agents updated their state-action pair values by temporal difference learning with learning rate *α*:

We included eligibility traces, so the prediction error was applied to every previously chosen state-action pair utilized in that trial with decay parameter λ. Agents chose actions by entering the *Q* values into a softmax function with the same temperature parameter *β*.

Since agents maintained three separate controllers, each suggesting different actions, we mixed these action suggestions probabilistically to produce a final action selection. Let *aMFG*, *aMB*, and *aMF* represent the actions chosen by the model-free goal, model-based, and model-free mechanisms respectively. Then agents chose action *a* according to:

where *wMFG* and *wMB* represent the weighting of the agent towards model-free goal and model-based control, respectively. (Our results do not change if agents instead choose actions by weighing the action values of the three controllers together before entering them into a single softmax function.)

Thus, agents were characterized by five parameters: *α* (the learning rate), *λ* (the eligibility trace), *β* (the softmax temperature), *wMFG* (the model-free goal weight), and *wMB* (the model-based weight). Each agent’s parameters were randomly sampled as follows. *α* was sampled from a uniform distribution from 0 to 1, which we denote as U(0,1). *λ* was sampled from U(.5,1). *β* was sampled from U(.5,1.5). For the weights, three variables –  ,,and – were sampled from U(0,1), and then and . We generated 200 agents per simulation, and analyzed agents’ behavior by the same process as in the behavioral tasks.

*Results*

In the simulation with model-free goal learning, agents chose the shared-goal action XX% of the time after a reward and XX% of the time after a punishment (NUMBERS). The mixed-effects model on same-type trials estimated a model-free goal coefficient of .063 (NUMBERS), and was preferred to a null model (NUMBERS). In contrast, when agents did not perform model-free goal learning (e.g. ), agents showed no difference in behavior following a reward versus a punishment (NUMBERS). Analysis by mixed effect models similarly showed null results (NUMBERS).

We also compared the performances of agents who exhibited only model-free goal selection (), model-based control (), or model-free control (). As predicted, our mechanism accumulated more total reward on the task than a pure model-free mechanism (NUMBERS), but less than a pure model-based mechanism (NUMBERS), suggesting that our mechanism balances the accuracy of model-based approaches with the computational efficiency of model-free approaches.