

---

# Learning to select computations

---

Anonymous

*This citation format is weird -  
check that it is the  
correct one for AAAI.*

## Abstract

Efficient use of limited computational resources is essential to intelligence. Selecting computations optimally according to rational metareasoning would achieve this, but rational metareasoning is computationally intractable. Inspired by psychology and neuroscience, we propose the first learning algorithm for approximating the optimal selection of computations, named Bayesian meta-level policy search (BMPS). We derive a general, sample-efficient reinforcement learning algorithm for learning to select computations from the insight that the value of computation lies between the myopic value of computation and the value of perfect information. We evaluate the performance of BMPS against two state-of-the-art methods for approximate metareasoning—the meta-greedy heuristic and the blinkered policy—on three increasingly difficult metareasoning problems: metareasoning about when to terminate computation, metareasoning about how to choose between multiple actions, and metareasoning about planning. Across all three domains, BMPS achieved near-optimal performance and significantly outperformed the meta-greedy heuristic. The blinkered policy performed on par with BMPS in metareasoning about decision-making, but it is not directly applicable to metareasoning about planning where BMPS outperformed both the meta-greedy heuristic and a generalization of the blinkered policy. Our results are a step towards building self-improving AI systems that can learn to make optimal use of their limited computational resources to efficiently solve complex problems in real-time.

## Introduction

The human brain is the best example of an intelligent system we have so far. One feature that sets it apart from current AI is its remarkable computational efficiency, which enables people to effortlessly solve intractable computational problems for which artificial intelligence either under-performs humans or requires superhuman computing power and training time [Lake et al.2016]. For instance, to defeat Garry Kasparov 3.5–2.5, Deep Blue had to evaluate 200 000 000 positions per second, whereas Kasparov was able to perform at almost the same level by evaluating only 3 positions per second [Campbell, Hoane, and Hsu2002, IBM Research1997]. This ability to make efficient use of limited computational resources is the essence of intelligence [Russell and Wefald1991c]. People accomplish this feat by being very selective about when to think, what to think about, choosing computations adaptively, and terminating deliberation when its expected benefit falls below its cost [Gershman, Horvitz, and Tenenbaum2015, Lieder and Griffiths2017, Payne, Bettman, and Johnson1988].

The theory of rational metareasoning was developed to recreate such intelligent control over computation in machines [Russell and Wefald1991a, Hay et al.2012]. In principle, rational metareasoning can be used to always select those computations that make optimal use of the agent's finite computational resources. Unfortunately, the computational complexity of exact rational metareasoning is prohibitive [Hay et al.2012]. Recent research suggests that the human mind circumvents this computational challenge by learning to select computations through *metacognitive* reinforcement learning [Krueger, Lieder, and Griffiths2017, Lieder and Griffiths2017, Wang et al.2017]. Concretely, people appear to learn to predict the value of alternative cognitive operations from features of the task, their current belief state, and the cognitive operations themselves. If humans learn to metareason through metacognitive reinforcement learning, then it should be possible to build intelligent systems that learn to metareason as efficiently as people.

In this paper, we propose Bayesian meta-level policy search (BMPS), the first algorithm for learning how to metareason and evaluate it against existing methods for approximate metareasoning on three increasingly more complex problems. We find that BMPS performs significantly better than previous approximations to rational metareasoning and conclude with a discussion of future directions towards self-improving AI systems and potential applications.

## Background

### Metareasoning

If reasoning seeks an answer to the question “what should I do?”, metareasoning seeks an answer to the question “how should I decide what to do?”. The theory of rational metareasoning [Russell and Wefald1991a, Russell and Subramanian1995] frames this problem as selecting computations so as to maximize the sum of the rewards of resulting decisions minus the costs of the computations involved. Concretely, one can formalize reasoning as a metalevel Markov decision process (metalevel MDP) and metareasoning as solving that MDP [Hay et al.2012]. A metalevel MDP

$$M_{\text{meta}} = (\mathcal{B}, \mathcal{A}, T_{\text{meta}}, r_{\text{meta}}) \quad (1)$$

is a Markov decision process [Puterman2014] where the actions  $\mathcal{A}$  are cognitive operations, the states  $\mathcal{B}$  encode the agent’s beliefs, and the transition function  $T_{\text{meta}}$  describes how cognitive operations change the beliefs.  $\mathcal{A}$  includes computations  $\mathcal{C}$  that update the belief, as well as a special metalevel action  $\perp$  that terminates deliberation and initiates acting on the current belief. A belief state  $b$  encodes a probability distribution over parameters  $\theta$  of a model of the domain. The parameters  $\theta$  determine the utility of acting according to a policy  $\pi$ , that is  $U_{\pi}^{(\theta)}$ . For one-shot decisions,  $U_{\pi}^{(\theta)}$  is the expected reward of a taking a single action. In sequential decision-problems,  $U_{\pi}^{(\theta)} = V_{\pi}^{(\theta)}(s)$  is the expected sum of rewards the agent will obtain by acting according to policy  $\pi$  if the environment has the characteristics encoded by  $\theta$ . Since  $b$  encodes the agent’s belief about  $\theta$ , its subjective utility  $\hat{U}_{\pi}^{(b)}$  of acting according to  $\pi$  is  $\mathbb{E}_{\theta \sim b}[U_{\pi}^{(\theta)}]$ .

The metalevel reward function  $r_{\text{meta}}$  captures the cost of thinking [Shugan1980] and the external reward  $r$  the agent expects to receive from the environment. The computations  $\mathcal{C}$  have no external effects, thus they always incur a negative reward  $r_{\text{meta}}(b, c) = -\text{cost}(c)$ . In the problems studied below, all computations that deliberate have the same cost, that is  $\text{cost}(c) = \lambda$  for all  $c \in \mathcal{C}$  whereas  $\text{cost}(\perp) = 0$ . An external reward is received only when the agent terminates deliberation and makes a decision based on the current belief state  $b$ . To reduce the variance of this reward signal, the

metalevel reward of terminating deliberation is defined as the expectation of the external reward, that is

$$r_{\text{meta}}(b, \perp) = \max_{\pi} \hat{U}_{\pi}^{(b)} = \max_{\pi} \mathbb{E}_{\theta \sim b} [U_{\pi}^{(\theta)}]. \quad (2)$$

Early work on rational metareasoning [Russell and Wefald1991a] defined the optimal way to select computations as maximizing the value of computation (VOC), that is

$$\arg \max_c \text{VOC}(c, b), \quad (3)$$

where  $\text{VOC}(c, b)$  is the expected improvement in decision quality that can be achieved by performing computation  $c$  in belief state  $b$  and continuing optimally, minus the cost of the optimal sequence of computations [Russell and Wefald1991a]. When no computation has positive value, the policy terminates computation and executes the best object-level action, thus  $\text{VOC}(\perp, b) = 0$ . Using the formalism of metalevel MDPs [Hay et al.2012], this definition can be rewritten as

$$\text{VOC}(c, b) = Q_{\text{meta}}^*(b, c) - r_{\text{meta}}(b, \perp), \quad (4)$$

and the optimal selection of computations can be expressed as the optimal metalevel policy  $\pi_{\text{meta}}^*(b) = \arg \max_c Q_{\text{meta}}^*(b, c)$ .

### Approximations to rational metareasoning

Previous work [Russell and Wefald1991a, Lin et al.2015] has approximated rational metareasoning by the meta-greedy policy

$$\pi_{\text{greedy}}(b) = \arg \max_c \text{VOC}_1(c, b), \quad (5)$$

where

$$\begin{aligned} \text{VOC}_1(c, b_t) = & \mathbb{E} [r_{\text{meta}}(B_{t+1}, \perp) | b_t, c_t] \\ & + r_{\text{meta}}(b_t, c) - r_{\text{meta}}(b_t, \perp), \end{aligned} \quad (6)$$

is the myopic value of computation [Russell and Wefald1991b]. The meta-greedy policy selects each computation assuming that it will be the last computation. This is optimal when the improvement from each additional computation is less than that from the previous one, but deliberation is too little when this assumption is violated. For instance, when the outcome of simulating an action is binary, either outcome of a single simulation may be insufficient to change the agent’s decision—rendering the myopic VOC negative—whereas two or more simulations might change it and thus have a positive VOC when the stakes are sufficiently high.

Hay et al. (2012) approximated rational metareasoning by combining the solutions to smaller metalevel MDPs that formalize the problem of deciding how to decide between one object-level action and the expected return of its best alternative. Each of these smaller metalevel MDPs includes only the computations for reasoning about the expected return of the corresponding object-level action. While this *blinkered* approximation is more accurate than the meta-greedy policy, it is also significantly less scalable and not directly applicable to metareasoning about planning.

It has been proposed that people approximate optimally selecting individual computations by metareasoning over a small subset of all possible sequences of computations [Milli, Lieder, and Griffiths2017]. The solution to this simplified problem can be approximated efficiently [Lieder and Griffiths2017], but this approximation neglects the sequential nature of selecting individual computations.

To our knowledge these are the main approximations to rational metareasoning. Hence, to date, there appears to be no accurate and scalable method for solving general metalevel MDPs.

## Metalevel reinforcement learning

It has been proposed that metareasoning can be made tractable by learning an approximation to the value of computation [Russell and Wefald1991a]. However, despite some preliminary steps in this direction [Harada and Russell1998, Lieder et al.2014, Lieder, Krueger, and Griffiths2017] and related work on meta-learning [Smith-Miles2009, Schaul and Schmidhuber2010, Thornton et al.2013, Wang et al.2017], learning to approximate bounded optimal information processing remains an unsolved problem in artificial intelligence.

Previous research in cognitive science suggests that people circumvent the intractability of metareasoning by learning a metalevel policy from experience [Lieder and Griffiths2017, Cushman and Morris2015, Krueger, Lieder, and Griffiths2017]. At least in some cases, the underlying mechanism appears to be model-free reinforcement learning (RL) [Cushman and Morris2015, Krueger, Lieder, and Griffiths2017, Wang et al.2017]. This suggests that model-free reinforcement learning might be a promising approach to solving metalevel MDPs. To our knowledge, this approach is yet to be explored in artificial intelligence. Here, we present a proof-of-concept that near-optimal metalevel policies can be learned through metalevel RL.

## BMPS: A meta-level RL algorithm for selecting computations

According to rational metareasoning, one should continue to reason until none of the available computations has a positive VOC. Until then, one should always choose the computation that confers the highest improvement in decision-quality net its cost. While the improvement in decision quality contributed by a computation  $c$  under the optimal continuation is generally intractable to compute, it can be bounded. Figure 1 illustrates that if the expected decision quality improves monotonically with the number of computations, then the improvement achieved by the optimal sequence of computations should lie between the advantage of deciding immediately after the first computation over making a decision without it (Russell & Wefald, 1991) and the benefit of obtaining perfect information about all actions [Howard1966]. The former is given by the myopic value of information<sup>1</sup>, that is

$$\text{VOI}_1(c, b_t) = \mathbb{E}_{B_{t+1}|b_t, c} \left[ \max_{\pi} \hat{U}_{\pi}^{(B_{t+1})} \right] - \max_{\pi} \hat{U}_{\pi}^{(b)}. \quad (7)$$

The latter is given by the value of perfect information about all actions, that is

$$\text{VPI}_{\text{all}}(b) = \mathbb{E}_{\theta \sim b} \left[ \max_{\pi} U_{\pi}^{(\theta)} \right] - \max_{\pi} \hat{U}_{\pi}^{(b)}. \quad (8)$$

In problems with many possible actions, this upper bound can be very loose, and the VOC may be closer to the value of knowing the value functions of the policies  $\Pi_c$  about

<sup>1</sup>The  $\text{VOI}_1$  defined here is equal to the myopic VOC defined by Russell and Wefald (1991) plus the cost of the computation.

the  
is  
not  
clear.  
↓

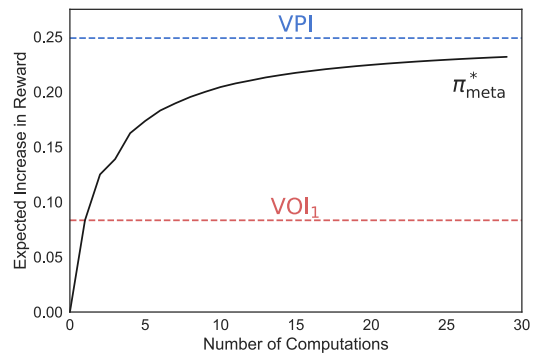


Figure 1: Expected performance in metareasoning about how to choose between three actions increases monotonically with the number of computations, asymptoting at the value of perfect information (VPI). Consequently, the value of executing a single computation must lie between the myopic value of information ( $\text{VOI}_1$ ) and the VPI.

whose returns the computation  $c$  is informative, that is

$$\text{VPI}_A(b, c) = \mathbb{E}_{\theta \sim b} \left[ \max \left( \mathcal{U}_c^{(\theta)} \cup \hat{\mathcal{U}}_c^{(b)} \right) \right] - r_{\text{meta}}(b, \perp), \quad (9)$$

where  $\mathcal{U}_c^{(\theta)} = \{U_\pi^{(\theta)} : \pi \in \Pi_c\}$  are the unknown utilities of the policies that computation  $c$  is informative about, and  $\hat{\mathcal{U}}_c^{(b)} = \{\hat{U}_\pi^{(b)} : \pi \notin \Pi_c\}$  is the set of the expected utilities of all policies that  $c$  is not informative about. This definition generalizes the value of perfect information about a single action [Dearden, Friedman, and Russell1998] to policies.

Critically, the myopic value of information ( $\text{VOI}_1$ ), the VPI about all actions, and the  $\text{VPI}_A$  can all be computed efficiently or efficiently approximated by Monte-Carlo integration [Hammersley2013]. BMPS thus approximates the expected improvement in decision quality gained by a computation by linearly interpolating between its myopic VOI and the value of perfect information, that is

$$\text{VOC}(c, b) \approx w_1 \cdot \text{VOI}_1(c, b) + w_2 \cdot \text{VPI}_{\text{all}}(b) + w_3 \cdot \text{VPI}_A(b, c) - w_4 \cdot \text{cost}(c), \quad (10)$$

with the constraints that  $w_1, w_2, w_3 \in [0, 1]$ ,  $w_1 + w_2 + w_3 = 1$ , and  $w_4 \in [1, h]$  where  $h$  is an upper bound on how many computations can be performed. Below we propose an algorithm through which the agent can learn these weights from experience.

Since the VOC defines the optimal metalevel policy (Equation 3), we can approximate the optimal policy by plugging our VOC approximation (Eq. 10) into Equation 3. This yields

$$\pi_{\text{meta}}(b; \mathbf{w}) = \arg \max_c \{w_1 \cdot \text{VOI}_1(c, b) + w_2 \cdot \text{VPI}_{\text{all}}(b) + w_3 \cdot \text{VPI}_A(b, c) - w_4 \cdot \text{cost}(c)\}. \quad (11)$$

The parameters  $\mathbf{w}$  of this policy are estimated by maximizing the expected return  $\mathbb{E}[\sum_t r_{\text{meta}}(b_t, \pi_{\text{meta}}(b_t; \mathbf{w}))]$ . Together with the constraints on the weights stated above, this effectively reduces the intractable problem of solving metalevel MDPs to a simple 3-dimensional optimization problem. There are many ways this optimization problem could be solved. Since estimating the expected return for a given weight vector can be expensive, we use Bayesian optimization (BO) [Mockus2012] to optimize the weights in a sample efficient manner.

The novelty of BMPS lies in leveraging machine learning to approximate the solution to metalevel MDPs and in the discovery of features that make this tractable. As far as we know, BMPS is the first general approach to metalevel RL.

In the following sections, we validate the assumptions of BMPS, evaluate its performance on increasingly complex metareasoning problems, compare it to existing methods, and discuss potential applications.

## Evaluation of BMPS in simulations

We evaluate how accurately BMPS can approximate rational metareasoning against two state-of-the-art ~~approximations~~ <sup>methods</sup>—the meta-greedy policy and the blinkered approximation—on three increasingly difficult metareasoning problems: deciding when to stop thinking, deciding how to decide, and deciding how to plan.

### 1. Metareasoning about when to stop deliberating

How long should an agent deliberate before answering a question? Our evaluation mimics this problem for a binary prediction task (e.g., “Will the price of the stock go up or down?”). Every deliberation incurs a cost and provides probabilistic evidence  $X_t \sim \text{Bernoulli}(p)$  in favor of one outcome or the other. At any point the agent can stop deliberating and predict the outcome supported by the majority of its deliberations so far. The agent receives a reward of  $+1$  if its prediction is correct, or incurs a loss of  $-1$  if it is incorrect. The goal is to maximize the expected reward of this one prediction minus the cost of computation.

**Metalevel MDP:** We formalize the problem of deciding when to stop thinking as a metalevel MDP  $M_{\text{meta}} = (\mathcal{B}, \mathcal{A}, T_{\text{meta}}, r_{\text{meta}})$  where each belief state  $(\alpha, \beta) \in \mathcal{B}$  defines a beta distribution over the probability  $p$  of the first outcome. The metalevel actions  $\mathcal{A}$  are  $\{c_1, \perp\}$  where  $c_1$  refines the belief by sampling, and  $\perp$  terminates deliberation and predicts the outcome that is most likely according to the current belief. The transition probabilities for sampling are defined by the agent’s belief state, that is  $T_{\text{meta}}((\alpha, \beta), c_1, (\alpha + 1, \beta)) = \frac{\alpha}{\alpha + \beta}$  and  $T_{\text{meta}}((\alpha, \beta), c_1, (\alpha, \beta + 1)) = \frac{\beta}{\alpha + \beta}$ . Predicting (executing  $\perp$ ) always transitions to a terminal state. The reward function  $r_{\text{meta}}$  reflects the cost of computation ( $r_{\text{meta}}(b, c_1) = -\lambda$ ) and the probability of making the correct prediction ( $r_{\text{meta}}(b, \perp) = +1 \cdot p_{\text{correct}}(\alpha, \beta) - 1 \cdot (1 - p_{\text{correct}}(\alpha, \beta))$ ) where  $p_{\text{correct}}(\alpha, \beta) = \max\{\frac{\alpha}{\alpha + \beta}, \frac{\beta}{\alpha + \beta}\}$ . We set the horizon to  $h = 30$ , meaning that the agent can perform at most 30 computations before making a prediction.

Since there is only one object-level action (i.e., to predict the outcome that appears most likely) the VPI about all actions is identical to the VPI for a single action. When reporting on this problem, we will thus not distinguish between them and use the term VPI instead. For the same reason, the blinkered approximation is equivalent to solving the problem exactly.

*I think it's also worth noting that this can be done as an application of machine learning like a stochastic RL setting as well as blinkered as a single global optimization.*



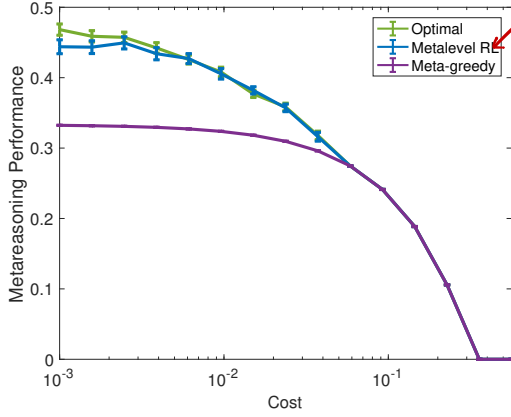


Figure 2: Results of performance evaluation on the problem of metareasoning about when to terminate deliberation.

**Evaluation procedure:** We evaluated the potential of BMPS in two steps: First, we performed a regression analysis to evaluate whether the proposed features are sufficient to capture the value of computation. Second, we tested whether the proposed features are sufficient to learn a near-optimal metalevel policy. The metalevel RL agent learns the weights  $w$  of the policy defined in Equation 11 that maximize expected return through Gaussian process Bayesian optimization. We ran 500 iterations of optimization, estimating the expected return of the policy entailed by the probed weight vector by its average return across 2500 episodes. The performance of learned policy was evaluated on an independent test set of 3000 episodes.

To perform these evaluations, we first established the ground truth by solving the metalevel MDP with backward induction [Puterman2014].

**Results:** First, linear regression analyses confirmed that three simple features ( $\text{VOI}_1(c, b)$ ,  $\text{VPI}(c, b)$ , and  $\text{cost}(c)$ ) are sufficient to capture between 90.8% and 100.0% of the variance in the value of computation for performing a simulation ( $\text{VOC}(b, c_1)$ ) across different states  $b$  depending on the cost of computation. Concretely, as the cost of computation increased from 0.001 to 0.1 the regression weights shifted from  $0.76 \cdot \text{VPI} + 0.46 \cdot \text{VOI}_1 - 4.5 \cdot \text{cost}$  to  $0.00 \cdot \text{VPI} + 1.00 \cdot \text{VOI}_1 - 1.00 \cdot \text{cost}$  and the explained variance increased from 90.8% to 100.0%. The explained variance and the weights remained the same for costs greater than 0.1. Supplementary Figure 1 illustrates this fit for  $\lambda = 0.02$ .

Second, we found that the  $\text{VOI}_1$  and the  $\text{VPI}$  features are sufficient to learn a near-optimal metalevel policy. As shown in Figure 2, the performance of metalevel RL policy was at most 5.19% lower than the performance of the optimal metalevel policy across all costs. The difference in performance was largest for the lowest cost  $\lambda = 0.001$

( $t(2999) = 3.75, p = 0.0002$ ) and decreased with increasing cost so that there was no statistically significant performance difference between BMPS and the optimal metalevel policy for costs greater than  $\lambda = 0.0025$  (all  $p > 0.15$ ). The policy learned with BO performed between 6.78% and 35.8% better than the meta-greedy policy across all costs where the optimal policy made more than one observation (all  $p < 0.0001$ ) and 20.3% better on average ( $t(44999) = 42.4, p < 10^{-15}$ ).

## 2. Metareasoning about decision-making

How should an agent allocate its limited decision-time across estimating the expected utilities of multiple alternatives? To evaluate how well BMPS can solve this kind of problem, we evaluate it on the *Bernoulli metalevel probability model* introduced by Hay et al. (2012). This problem differs from the previous one in two ways. First, instead of having only a single object-level action (i.e., make a prediction), there are now  $k \geq 2$  object-level actions. Second, instead of making a prediction and being rewarded for its accuracy, the agent chooses an action  $a_i$  and receives a payoff that is sampled from its outcome distribution, that is  $r(s, a_i) \sim \text{Bernoulli}(\theta_i)$  where  $\theta_i$  is the action’s unknown reward probability. This problem differs from the standard multi-armed bandit problem in two ways: First, the agent takes only a single object-level action and thus receives only one external reward. Second, the agent is equipped with a simulator that it can use to estimate the reward probabilities  $\theta_1, \dots, \theta_k$  via sampling; simulated outcomes do not count towards the agent’s reward, but each simulation has a cost.

**Metalevel MDP:** The Bernoulli metalevel probability model is a metalevel MDP  $M_{\text{meta}} = (\mathcal{B}, \mathcal{A}, T_{\text{meta}}, r_{\text{meta}}, h)$  where each belief state  $b$  defines  $k$  Beta distributions over the reward probabilities  $\theta_1, \dots, \theta_k$  of the  $k$  possible actions. Thus  $b$  can be represented by  $((\alpha_1, \beta_1), \dots, (\alpha_k, \beta_k))$  where  $b(\theta_i) = \text{Beta}(\theta_i; \alpha_i, \beta_i)$  for all  $1 \leq i \leq k$ . For the initial belief state  $b_0$  these parameters are  $\alpha_i = \beta_i = 1$  for all  $1 \leq i \leq k$ . The metalevel actions  $\mathcal{A}$  are  $\{c_1, \dots, c_k, \perp\}$  where  $c_i$  simulates action  $a_i$  and  $\perp$  terminates deliberation and executes the action with the highest expected return, that is action  $\arg \max_i \frac{\alpha_i}{\alpha_i + \beta_i}$ . The metalevel transition probabilities ( $T_{\text{meta}}(b_t, c_i, b_{t+1})$ ) encode that performing computation  $c_i$  increments  $\alpha_i$  with probability  $\frac{\alpha_i}{\alpha_i + \beta_i}$  and increments  $\beta_i$  with probability  $\frac{\beta_i}{\alpha_i + \beta_i}$ . The metalevel reward function  $r_{\text{meta}}(b, c)$  is  $-\lambda$  for  $c \in \{c_1, \dots, c_k\}$  and  $r_{\text{meta}}(b, \perp) = \max_i \frac{\alpha_i}{\alpha_i + \beta_i}$ . Finally, the horizon  $h$  is the maximum number of metalevel actions that can be performed and the last metalevel action has to be to terminate deliberation and take action ( $\perp$ ).

**Evaluation procedure:** We evaluated BMPS on Bernoulli metalevel probability problems with  $k \in \{2, \dots, 5\}$  object-level actions, a horizon of  $h = 25$ , and computational costs ranging from  $10^{-4}$  to  $10^{-1}$ . We evaluated the performance of metalevel RL against the optimal metalevel policy and three alternative approximations: the meta-greedy heuristic [Russell and Wefald1991a], the blinkered approximation [Hay et al.2012], and the metalevel policy that always deliberates as much as possible. We trained the metalevel RL policy with Bayesian optimization as described above, but with 100 iterations of 1000 episodes each. To combat the possibility of overfitting, we evaluated the average returns of the five best weight vectors over 5000 more episodes and selected the one that performed best. The optimal metalevel policy and the blinkered policy were computed using backward induction [Puterman2014]. We evaluated the performance of each policy by its average return across 2000 episodes for each combination of computational cost and number of object-level actions.

**Results:** An analysis of variance confirmed that these four methods differed significantly in their performance ( $F(4, 279932) = 123078.5, p < 10^{-15}$ ). We found that the policy obtained by metalevel RL attained 99.2% of optimal performance (0.6540 vs. 0.6596,  $t(1998) = -6.98, p < 0.0001$ ) and significantly outperformed the meta-greedy heuristic (0.60,  $t(1998) = 86.9, p < 10^{-15}$ ) and the full-deliberation policy (0.20,  $t(1998) = 475.2, p < 10^{-15}$ ). The performance of BMPS (0.6540) and the blinkered approximation (0.6559) differed by only 0.29%.

Figure 3b shows the metareasoning performance of each method as a function of the number of options. We found that the performance of BMPS scaled well with the size of the decision problem. For each number of options, the relative performance of the different methods was consistent with the results reported above.

Figure 3a shows the methods' average performance as a function of the cost of computation. An ANOVA confirmed that the effect of the metareasoning method differed across different costs of computation ( $F(24, 279932) = 64401.4, p < 10^{-15}$ ). BMPS outperformed the meta-greedy heuristic for costs smaller than 0.03 (all  $p < 10^{-15}$ ), and the full-deliberation policy for costs greater than 0.0003 (all  $p < 0.005$ ). For costs below 0.0003, the blinkered policy performed slightly better than BMPS (all  $p < 0.0005$ ). For all other costs both methods performed at the same level (all  $p > 0.1$ ), with the exception of the cost  $\lambda = 0.01$ , for which BMPS outperformed the blinkered approximation ( $t(1998) = 3.2, p = 0.001$ ). Additionally, for costs larger than 0.01, performance of BMPS becomes indistinguishable from the optimal policy's performance (all  $p > 0.24$ ).

Finally, as illustrated in Supplementary Figure 2, we found

that BMPS learned surprisingly quickly, usually discovering near-optimal policies in less than 10 iterations.

### 3. Metareasoning about planning

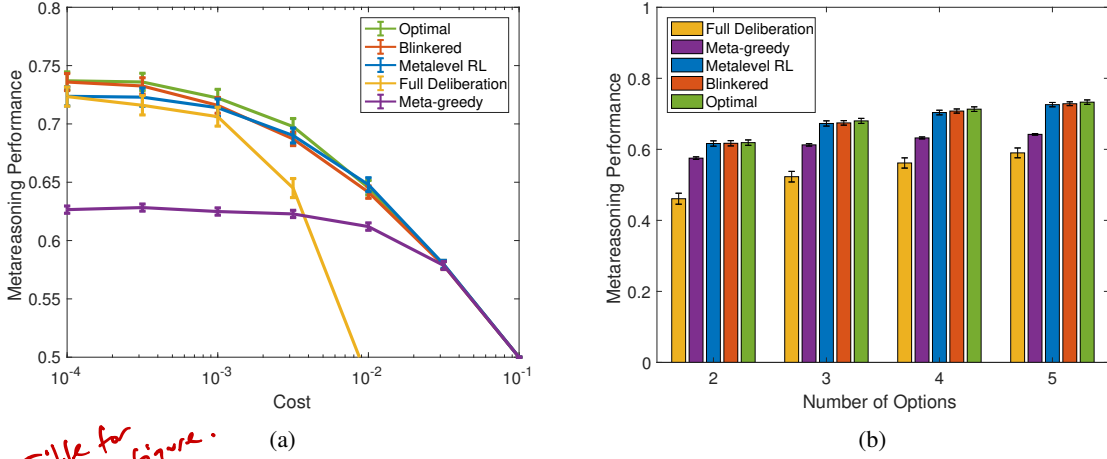
Having evaluated BMPS on problems of metareasoning about how to make a one-shot decision, we now evaluate its performance at deciding how to plan. To do so, we define the *Bernoulli metalevel tree*, which generalizes the Bernoulli metalevel probability model by replacing the one-shot decision between  $k$  options by a tree-structured sequential decision problem that we will refer to as the *object-level MDP*. The transitions of the object-level MDP are deterministic and known to the agent. The reward associated with each of  $K = 2^{h+1} - 1$  states in the tree is deterministic, but initially unknown;  $r(s, a, s_k) = \theta_k \in \{-1, 1\}$ . The agent can uncover these rewards through reasoning at a cost of  $-\lambda$  per reward. When the agent terminates deliberation, it executes a policy with maximal expected utility. Unlike in the previous domains, this policy entails a sequence of actions rather than a single action.

**Metalevel MDP:** The Bernoulli metalevel tree is a metalevel MDP  $M_{\text{meta}} = (\mathcal{B}, \mathcal{A}, T_{\text{meta}}, r_{\text{meta}})$  where each belief state  $b$  encodes one Bernoulli distribution for each transition's reward. Thus,  $b$  can be represented as  $(p_1, \dots, p_K)$  such that  $b(\theta_k = 1) = p_k$  and  $b(\theta_k = -1) = 1 - p_k$ . The initial belief  $b_0$  has  $p_k = 0.5$  for all  $k$ . The metalevel actions are defined  $\mathcal{A} = \{c_1, \dots, c_K, \perp\}$  where  $c_k$  reveals the reward at state  $k$  and  $\perp$  selects the path with highest expected sum of rewards according to the current belief state. The transition probabilities  $T_{\text{meta}}(b_t, c_k, b_{t+1})$  encode that performing computation  $c_k$  sets  $p_k$  to 1 or 0 with equal probability (unless  $p_k$  has already been updated, in which case  $c_k$  has no effect). The metalevel reward function is defined  $r_{\text{meta}}(b, c) = -\lambda$  for  $c \in \{c_1, \dots, c_K\}$ , and  $r_{\text{meta}}((p_1, \dots, p_K), \perp) = \max_{t \in \mathcal{T}} \sum_{k \in t} \mathbb{E}[\theta_k | p_k]$  where  $\mathcal{T}$  is the set of possible trajectories  $t$  through the environment, and  $\mathbb{E}[\theta_k | p_k] = 2p_k - 1$  is the expected reward attained at state  $s_k$ .

**The recursively blinkered policy** The blinkered policy of Hay et al. (2012) was defined for problems where each computation informs the value of only one action. This assumption of "independent actions" is crucial to the efficiency of the blinkered approximation because it allows the problem to be decomposed into one independent subproblem for each action. When there are few computations associated with each action, each subproblem can be efficiently solved.

Critically, the Bernoulli metalevel tree violates the assumption of independent actions. This is because here "actions" are policies, and the reward at each state affects the values of all policies visiting that state. One can still apply the blinkered policy in this case, approximating the value of a

I never  
if you  
take a  
then in this  
order.



*Title for figure.*

Figure 3: (a) Metareasoning performance as a function of the cost of computation. Error bars enclose 95% confidence intervals. (b) Metareasoning performance (i.e. expected reward of chosen option minus cost of the decision process) of alternative methods on the Bernoulli metalevel probability model as a function of the number of actions. Error bars enclose 95% confidence intervals.

computation  $c_k$  by assuming that future computations will be limited to  $\mathcal{E}_{c_k}$ , the set of computations that are informative about *any* of the policies the initial computation is relevant to. However, for large trees, this only modestly reduces the size of the initial problem. This suggests a recursive generalization: Rather than applying the blinkered approximation once and solving the resulting subproblem exactly, we recursively apply the approximation to the resulting subproblems. Finally, to ensure that the subproblems decrease in size monotonically, we remove from  $\mathcal{E}_{c_k}$  the computations about rewards on the path from the agent’s current state to the state  $s_k$  inspected by computation  $c_k$  and call the resulting set  $\mathcal{E}'_{c_k}$ . Thus, we define the *recursively blinkered policy* as  $\pi^{\text{RB}}(b) = \arg \max_c Q^{\text{RB}}(b, c)$  with  $Q^{\text{RB}}(b_t, \perp) = r_{\text{meta}}(b_t, \perp)$  and

$$Q^{\text{RB}}(b_t, c_t) = \mathbb{E} \left[ r_{\text{meta}}(b_t, c_t) + \max_{c_{t+1} \in \mathcal{E}'_{c_t}} Q^{\text{RB}}(B_{t+1}, c_{t+1}) \right] \quad (12)$$

**Evaluation procedure:** We evaluated each method’s performance by its average return over 5000 episodes for each combination of tree-height  $h \in \{2, \dots, 6\}$  and computational cost  $\lambda \in \{2^{-7}, \dots, 2^0\}$ . To facilitate comparisons across planning problems with different numbers of steps, we measured the performance of meta-level policies by their expected return divided by the tree-height.

We trained the metalevel RL policy with Bayesian optimization as described above, but with 100 iterations of 1000 episodes each. To combat the possibility of overfitting, we evaluated the average returns of the three best weight vectors over 2000 more episodes and selected the one that performed best.

For metareasoning about how to plan in trees of height 2 and 3, we were able to compute the optimal metalevel policy using dynamic programming. But for larger trees, computing the optimal metalevel policy would have taken significantly longer than 6 hours and was therefore not undertaken.

**Results:** We first compared BMPS with the optimal policy for  $h \in \{2, 3\}$ , finding that it attained 98.4% of optimal performance (0.367 vs. 0.373,  $t(159998) = -2.87p < 10^{-15}$ ). An ANOVA of the performance of the approximate policies confirmed that the metareasoning performance differed significantly across the four methods we evaluated ( $F(3, 799840) = 4625010; p < 10^{-15}$ ), and that the magnitude of this effect depends on the height of the tree ( $F(12, 799840) = 1110179, p < 10^{-15}$ ) and the cost of computation ( $F(21, 799840) = 1266582, p < 10^{-15}$ ).

Across all heights and costs, BMPS achieved a metareasoning performance of 0.392 units of reward per object-level action, thereby outperforming the meta-greedy heuristic (0.307,  $t(399998) = 72.84, p < 10^{-15}$ ), the recursively blinkered policy (0.368,  $t(399998) = 20.77, p < 10^{-15}$ ), and the full-deliberation policy (−1.740,  $t(399998) = 231.18, p < 10^{-15}$ ).

As shown in Figure 4a, BMPS performed near-optimally across all computational costs, and its advantage over the meta-greedy heuristic and the tree-blinkered approximation was largest when the cost of computation was low, whereas its benefit over the full-deliberation policy increased with increasing cost of computation.

Figure 4b shows that the performance of BMPS scaled very well with the size of the planning problem, and that its ad-

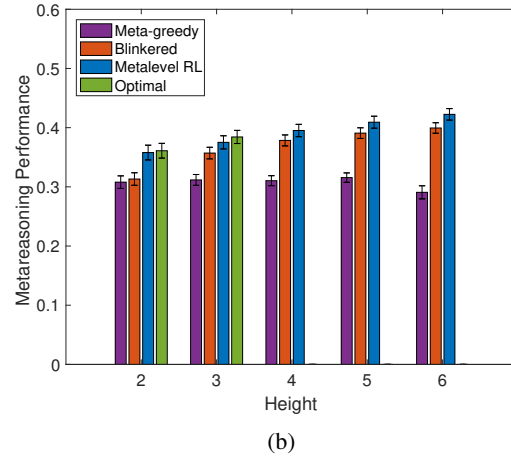
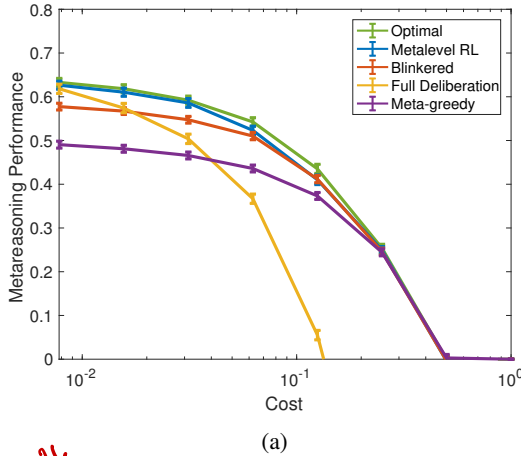


Figure 4: (a) Metareasoning performance as a function of computation cost on a Bernoulli tree of height three. Metareasoning performance is the average reward earned per object-level state visited. (b) Metareasoning performance as a function of tree height. The optimal policy is only shown for heights at which it can be computed in under six hours. The full observation policy is not shown because its performance is negative for all heights.

vantage over the meta-greedy heuristic increased with the height of the tree.

### When is metareasoning worthwhile?

The purpose of metareasoning is to properly allocate limited computational resources. Unfortunately, the computational cost of rational metareasoning is often much higher than the benefits of any improvements that could be made to object-level reasoning [Hay et al.2012]. Our approximate metareasoning method is more efficient, but it remains to be shown under what circumstances its benefits outweigh its costs.

We reasoned that metareasoning should be most likely to pay off when object-level computations are very costly and the stakes are high. For instance, during severe weather, important, potentially life-or-death decisions—such as which cities to evacuate in the face of an approaching tornado—must be evaluated on a limited number of computationally intense weather simulations [Baumgart et al.2008].

In this section we apply BMPS to this real-world problem. We consider an emergency manager who must determine whether to evacuate a number of cities in the tornado's likely path. To make these decisions, she must estimate the risks that each city will be severely hit, using fine-grained weather simulations to inform her decisions. Each object-level computations (i.e., which city to focus the next weather simulation on) has to be chosen so as to minimize both the risk of false positive errors (evacuating cities that are safe) and the risk of false negative errors (failing to evacuate a city that will be hit by the tornado).

### Methods

We model the above emergency management scenario as follows: There is a finite amount of time  $T$  until evacuation decisions about  $k$  cities will have to be made. For each city  $i$ , the weather service can run a fine grained, stochastic simulation of how it will be impacted by the approaching tornado ( $c_i$ ). Each simulation is scored based on whether the simulated impact would warrant an evacuation or not. Consequently, the belief state and the transition function of the corresponding meta-level MDP are the same as in the Bernoulli meta-level probability model, with Beta distributions tracking the probability that the tornado will cause evacuation-warranting damage in each city. The only difference is in the final decision, and thus the meta-level reward function. Rather than choosing a single option, the agent must make an independent binary decision about whether to evacuate each city. Evacuation has a cost,  $\lambda_{\text{evac}} = -1$ , but failing to evacuate a heavily-hit city has a much larger cost,  $\lambda_{\text{fn}} = -20$ . Thus the meta-level reward function is

$$r_{\text{meta}}(b, \perp) = \sum_{1 \leq i \leq k} \max \left\{ \frac{\alpha_i}{\alpha_i + \beta_i} \cdot \lambda_{\text{fn}}, \lambda_{\text{evac}} \right\} \quad (13)$$

In contrast to previous simulations, we now explicitly consider the cost of meta-reasoning. The decision time  $T$  has to be allocated between reasoning about the cities and meta-reasoning about which city to reason about. More formally,

$$T = n_{\text{sim}} \cdot (t_{\text{MR}} + t_{\text{sim}}), \quad (14)$$

where  $n_{\text{sim}}$  is the number of simulations run,  $t_{\text{MR}}$  is the amount of time it takes to choose one simulation to run (i.e. by meta-reasoning), and  $t_{\text{sim}}$  is the amount of time it

also, when you can compute a meta-level policy once, but have to solve object-level problems many times...

redefine  $\alpha, \beta$  for this context.



takes to run one simulation. Thus, for given values of  $t_{\text{MR}}$  and  $t_{\text{sim}}$  the number of simulations that can be performed is

$$n_{\text{sim}} = \left\lfloor \frac{T}{t_{\text{MR}} + t_{\text{sim}}} \right\rfloor, \quad (15)$$

where  $\lfloor x \rfloor$  is the largest integer that is smaller than  $x$ . Note that the meta-policy learned by BMPS can be trained offline, and thus training time does not factor into the above equation. For the simulations reported below, we used a single BMPS policy optimized for  $k = 20$  and  $n_{\text{sim}} = 50$  because we found the weights to be relatively insensitive to these parameters.

To assess if BMPS could be useful in practice, we compare the utility of evacuation decisions made by the BMPS meta-policy to those made by a baseline meta-policy that uniformly distributes simulations across the  $k$  cities. Because the BMPS policy has  $t_{\text{MR}} > 0$  while the baseline policy has  $t_{\text{MR}} \approx 0$ , BMPS will typically run fewer simulations. (How many fewer depends on the relative values of  $t_{\text{sim}}$  and  $t_{\text{MR}}$ .) Thus, BMPS must make up for a reduced number of computations by choosing more valuable computations.

## Results

We evaluated the BMPS policy and the uniform computation policy on the tornado problem with  $T = 24$  hours,  $k \in \{10, 30\}$  cities, and a range of simulation times  $t_{\text{sim}} \in [2^{-1}, 2^4]$  hours. For each policy and parameter setting we estimate utility as the mean return over 5000 rollouts.

Empirically, we found that  $t_{\text{MR}} \approx 1$  ms for  $k = 10$  and  $t_{\text{MR}} \approx 3$  ms for  $k = 30$ . Thus, even with a conservative estimate of  $t_{\text{MR}} = 0.001$  hours, metareasoning would cost at most one simulation ( $n_{\text{sim}} \leq 100$  for all considered parameter values). Consequently, our simulations revealed that diverting some of the computational resources to metareasoning was beneficial regardless of how long a tornado simulation might take and the number of cities being considered (see Figure 5).

In follow-up simulations we found that the benefit of metareasoning also extends to situations where object-level reasoning is much less costly as long as the number of object-level computations is sufficiently high relative to the amount of available time (see Figure 6 in Supplementary Material). Overall, we found that the benefit of metareasoning increases with the costliness of object-level reasoning and the complexity of the decision problem relative to the agent’s computational budget.

*While this is a single hypothetical scenario, we believe it illustrates that in conclusion, there are many situations in which our method makes metareasoning so efficient that its costs are outweighed by its benefits.*

## Discussion

We have introduced a new approach to solving the foundational problem of rational metareasoning: metalevel RL. This approach applies algorithms from RL to metalevel MDPs to learn a policy for selecting computations. Our results show that BMPS can outperform the state of the art methods for approximate metareasoning. While we illustrated this approach using a policy search algorithm based on Bayesian optimization, there are many other RL algorithms that could be used instead, including policy gradient algorithms, actor-critic methods, and temporal difference learning with function approximation.

Since BMPS approximates the value of computation as a linear combination of the myopic VOI and the value of perfect information, it can be seen as a generalization of the meta-greedy approximation [Lin et al.2015, Russell and Wefald1991c]. It is the combination of these features with RL that makes BMPS tractable and powerful. Metalevel RL works well across a wider range of problems than previous approximations because it reduces arbitrarily complex metalevel MDPs to low-dimensional optimization problems.

BMPS could be used to find the optimal algorithms that specific computational architectures should use to solve a specific class of problems as efficiently as possible. We predict that metalevel RL will enable significant advances in artificial intelligence and its applications. In the long view, metalevel RL may become a foundation for self-improving AI systems that learn how to solve increasingly more complex problems increasingly more efficiently. To facilitate this advance, future work will apply deep RL [Mnih et al.2015] to metalevel MDPs to discover efficiently computable features for predicting the value of computation. This will overcome the limitation that computing the value of perfect information takes a non-negligible amount of time. We have shown that the benefits of metareasoning with our method already more than outweigh its computational costs in scenarios where the object-level computations are very expensive. It might

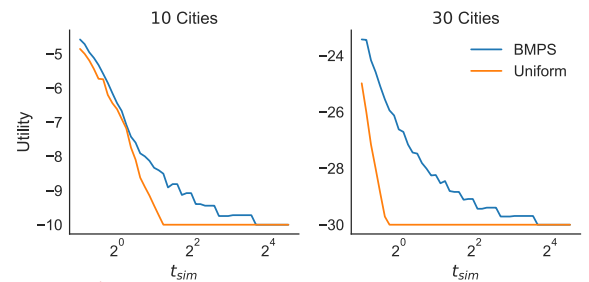


Figure 5: Benefit of metareasoning in the tornado evacuation scenario as a function of the duration of tornado simulations ( $t_{\text{sim}}$ ) in hours.

therefore benefit practical applications that involve complex large-scale simulations, active learning problems, hyperparameter search, and the optimization of functions that are very expensive to evaluate. Finally, BMPS could also be applied to derive rational process models of human cognition.

## References

- [Baumgart et al.2008] Baumgart, L. A.; Bass, E. J.; Philips, B.; and Kloesel, K. 2008. Emergency management decision making during severe weather. *Weather and Forecasting* 23(6):1268–1279.
- [Campbell, Hoane, and Hsu2002] Campbell, M.; Hoane, A. J.; and Hsu, F.-h. 2002. Deep blue. *Artificial intelligence* 134(1-2):57–83.
- [Cushman and Morris2015] Cushman, F., and Morris, A. 2015. Habitual control of goal selection in humans. *Proceedings of the National Academy of Sciences* 112(45):13817–13822.
- [Dearden, Friedman, and Russell1998] Dearden, R.; Friedman, N.; and Russell, S. 1998. Bayesian Q-learning. In *AAAI*, 761–768.
- [Gershman, Horvitz, and Tenenbaum2015] Gershman, S. J.; Horvitz, E. J.; and Tenenbaum, J. B. 2015. Computational rationality: A converging paradigm for intelligence in brains, minds, and machines. *Science* 349(6245):273–278.
- [Hammersley2013] Hammersley, J. 2013. *Monte Carlo methods*. Springer Science & Business Media.
- [Harada and Russell1998] Harada, D., and Russell, S. 1998. Meta-level reinforcement learning. In *NIPS'98 Workshop on Abstraction and Hierarchy in Reinforcement Learning*.
- [Hay et al.2012] Hay, N.; Russell, S.; Tolpin, D.; and Shimony, S. 2012. Selecting computations: Theory and applications. In de Freitas, N., and Murphy, K., eds., *Uncertainty in Artificial Intelligence: Proceedings of the Twenty-Eighth Conference*. P.O. Box 866 Corvallis, Oregon 97339 USA: AUAI Press.
- [Howard1966] Howard, R. A. 1966. Information value theory. *IEEE Transactions on systems science and cybernetics* 2(1):22–26.
- [IBM Research1997] IBM Research. 1997. Kasparov vs Deep Blue: A contrast in styles. [research-web.watson.ibm.com/deepblue/meet/html/d.2.shtml](http://research-web.watson.ibm.com/deepblue/meet/html/d.2.shtml). Retrieved on August 29 2017.
- [Krueger, Lieder, and Griffiths2017] Krueger, P. M.; Lieder, F.; and Griffiths, T. L. 2017. Enhancing metacognitive reinforcement learning using reward structures and feedback. In *Proceedings of the 39th Annual Conference of the Cognitive Science Society*. Cognitive Science Society.
- [Lake et al.2016] Lake, B. M.; Ullman, T. D.; Tenenbaum, J. B.; and Gershman, S. J. 2016. Building machines that learn and think like people. *arXiv preprint arXiv:1604.00289*.
- [Lieder and Griffiths2017] Lieder, F., and Griffiths, T. 2017. Strategy selection as rational metareasoning. *Psychological Review*.
- [Lieder et al.2014] Lieder, F.; Plunkett, D.; Hamrick, J. B.; Russell, S. J.; Hay, N.; and Griffiths, T. 2014. Algorithm selection by rational metareasoning as a model of human strategy selection. In *Advances in Neural Information Processing Systems*, 2870–2878.
- [Lieder, Krueger, and Griffiths2017] Lieder, F.; Krueger, P. M.; and Griffiths, T. L. 2017. An automatic method for discovering rational heuristics for risky choice. In *Proceedings of the 39th Annual Meeting of the Cognitive Science Society*. Austin: Cognitive Science Soc.
- [Lin et al.2015] Lin, C. H.; Kolobov, A.; Kamar, E.; and Horvitz, E. 2015. Metareasoning for planning under uncertainty. In *Proceedings of the 24th International Conference on Artificial Intelligence*, 1601–1609. AAAI Press.
- [Milli, Lieder, and Griffiths2017] Milli, S.; Lieder, F.; and Griffiths, T. L. 2017. When does bounded-optimal metareasoning favor few cognitive systems? In *AAAI Conference on Artificial Intelligence*, volume 31. AAAI Press.
- [Mnih et al.2015] Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A. A.; Veness, J.; Bellemare, M. G.; Graves, A.; Riedmiller, M.; Fidjeland, A. K.; Ostrovski, G.; et al. 2015. Human-level control through deep reinforcement learning. *Nature* 518(7540):529–533.
- [Mockus2012] Mockus, J. 2012. *Bayesian approach to global optimization: theory and applications*, volume 37. Springer Science & Business Media.
- [Payne, Bettman, and Johnson1988] Payne, J. W.; Bettman, J. R.; and Johnson, E. J. 1988. Adaptive strategy selection in decision making. *Journal of Experimental Psychology: Learning, Memory, and Cognition* 14(3):534.
- [Puterman2014] Puterman, M. L. 2014. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons.
- [Russell and Subramanian1995] Russell, S. J., and Subramanian, D. 1995. Provably bounded-optimal agents. *Journal of Artificial Intelligence Research* 2:575–609.
- [Russell and Wefald1991a] Russell, S., and Wefald, E. 1991a. Principles of metareasoning. *Artificial Intelligence* 49(1-3):361–395.
- [Russell and Wefald1991b] Russell, S., and Wefald, E. 1991b. Principles of metareasoning. *Artificial intelligence* 49(1-3):361–395.
- [Russell and Wefald1991c] Russell, S. J., and Wefald, E. 1991c. *Do the right thing: studies in limited rationality*. Cambridge, MA: MIT press.
- [Schaul and Schmidhuber2010] Schaul, T., and Schmidhuber, J. 2010. Metalearning. 5(6):4650. revision 91489.
- [Shugan1980] Shugan, S. M. 1980. The cost of thinking. *Journal of consumer Research* 7(2):99–111.
- [Smith-Miles2009] Smith-Miles, K. A. 2009. Cross-disciplinary perspectives on meta-learning for algorithm selection. *ACM computing surveys* 41(1).

[Thornton et al.2013] Thornton, C.; Hutter, F.; Hoos, H. H.; and Leyton-Brown, K. 2013. Auto-WEKA: Combined selection and hyperparameter optimization of classification algorithms. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 847–855. New York: ACM.

[Wang et al.2017] Wang, J. X.; Kurth-Nelson, Z.; Tirumala, D.; Soyer, H.; Leibo, J. Z.; Munos, R.; Blundell, C.; Kumaran, D.; and Botvinick, M. 2017. Learning to reinforcement learn. In *Proceedings of the 39th Annual Conference of the Cognitive Science Society*. Cognitive Science Society.

## Supplementary Material

[INCLUDE FIGURE HERE -FALK]

Figure 6: Benefit of metareasoning over allocating computation uniformly as a function of the cost of object-level reasoning, the number of computations, and the total amount of available time.