# Faster Teaching by POMDP Planning

Anna N. Rafferty[1], Emma Brunskill[1],
Thomas L. Griffiths[1], and Patrick Shafto[2]

[1] University of California, Berkeley, CA 94720, USA
[2] University of Louisville, KY 40292, USA

**Abstract.** Both human and automated tutors must infer what a student knows and plan future actions to maximize learning. Though substantial research has been done on tracking and modeling student learning, there has been significantly less attention on planning teaching actions and how the assumed student model impacts the resulting plans. We frame the problem of optimally selecting teaching actions using a decision-theoretic approach and show how to formulate teaching as a partially-observable Markov decision process (POMDP) planning problem. We consider three models of student learning and present approximate methods for finding optimal teaching actions given the large state and action spaces that arise in teaching. An experimental evaluation of the resulting policies on a simple concept-learning task shows that framing teacher action planning as a POMDP can accelerate learning relative to baseline performance.

## 1 Introduction

When assisting a student, a teacher must both diagnose a student's understanding and use a teaching policy for deciding on the best pedagogical action to take next. There has been substantial interest in the cognitive science, education, and intelligent tutoring systems communities in modeling and tracking student learning. In particular, there have been a number of results demonstrating the benefit of taking a Bayesian probabilistic approach (see, e.g., [4, 6, 7, 17]). However, there has been much less work on how to compute an automated teaching policy that leverages a probabilistic learner model in order to achieve a long-term teaching objective, which is the focus of this paper.

We use a probabilistic, sequential, decision-theoretic approach to compute individualized teaching policies. More specifically, we employ a Bayesian probabilistic representation over the learner's (hidden) knowledge, and embed this within a powerful framework known as a partially-observable Markov decision process (POMDP) [14]. Given a learning objective and a set of models describing the learning process, POMDPs provide a framework for computing an optimal teaching policy that maximizes the objective. Though POMDPs are related to other decision-theoretic approaches used in previous education research, they are more powerful in two key respects. First, POMDPs can use sophisticated models of learning, rather than assuming learners' understanding can be directly observed or approximated by a large number of features (as in [1, 5]). Second,

in contrast to approaches that only maximize the immediate benefit of the next action [6, 10], POMDPs reason about both the immediate learning gain and the long-term benefit to the learner after a particular activity.

Though POMDPs offer an appealing theoretical framework, there are often significant obstacles to practical implementation. Specifically, planning teaching requires modeling learning, and richer, more realistic models of learning lead to computational challenges for planning. In this paper we develop an approach for computing approximate POMDP policies, which makes it feasible to use these policies with human learners. In addition, we examine three different models of concept learning, and demonstrate how, given the same learning objective, these lead to qualitatively different teaching policies. We explore the impact of these varying policies in an example concept-learning task. While there exist a few recent papers exploring the use of POMDPs to compute teaching policies [2, 3, 9, 16], to our knowledge ours is the first paper to demonstrate with human learners that POMDP planning results in more efficient learning than baseline performance and the first to explore the impact of different models of learning on the computed policies.

## 2 Modeling Teaching as a POMDP

POMDP planning is used to compute an optimal conditional policy for selecting actions to achieve a goal, in absence of perfect information about the state of the world. Briefly, a POMDP consists of a tuple $\langle S, A, Z, p(s'|s,a), p(z|s,a), r(s,a), \gamma \rangle$ where $S$ is a set of states $s$, $A$ is a set of actions $a$, and $Z$ is a set of observations $z$ [14]. The transition model $p(s'|s,a)$ gives the probability of transitioning from state $s$ to state $s'$ after taking action $a$. The observation model $p(z|s,a)$ indicates the probability of an observation $z$ given that action $a$ is taken in state $s$. The planner's probability distribution over the current state is the *belief state* and can be updated using Bayesian filtering. The cost model $r(s,a)$ specifies the cost of taking action $a$ in state $s$, and the discount factor $\gamma$ represents the relative harm of immediate costs versus delayed costs. POMDP planning computes a policy that specifies which action to take, given a belief state, in order to minimize the expected sum of (discounted) future costs.

Many teaching tasks can be easily formalized within this framework. We model the learner's knowledge as a state $s$. The transition model then describes how teaching actions stochastically change the learner's knowledge, and the observation model indicates the probability that a learner will give a particular response to a tutorial action, such as a question, based on her current understanding. We will shortly describe several alternate learner models that employ different state representations, transition models, and observation models.

In the remainder of the paper, we consider how this framework can be applied in a concept learning task. In such a task, we set the cost for each action to be the expected amount of time for the learner to complete the activity, and when the learner knows the correct concept, the action cost drops to zero. As a consequence, the computed policies select actions to minimize the expected time

for the learner to understand the concept. The space of tutorial actions may vary widely based on the domain being taught. Within concept-learning, it is natural to consider three types of actions: *examples*, *quizzes*, and *questions with feedback*. *Example* and *quiz* actions are equivalent to the *elicit* and *tell* pedagogical actions that have been used previously in intelligent tutoring systems [5]. The resulting POMDP can be used to find the optimal policy for teaching the learner the concept, taking into account the learner's responses.

## 3 Learner Models

We consider three learner models, inspired by the cognitive science literature, that correspond to restrictions of Bayesian learning. While the models we describe are only rough approximations of human concept learning, we will show that they are still sufficient to enable us to compute better teaching policies.

**Memoryless Model:** We first consider a model in which the learner's knowledge state is the single concept she currently believes is correct, similar to a classic model of concept learning proposed by Restle [11]. In this model, the learner does not explicitly store any information previously seen. If an action is a *quiz* action, or if the provided evidence in an *example* or *question with feedback* action is consistent with the learner's current concept, then her state stays the same. If the action contradicts the current concept, the learner transitions to a state consistent with that action, with probability proportional to the prior probability of that concept. The observation model is deterministic: when asked to provide an answer to an equation, the learner provides the answer consistent with her current beliefs. This model underestimates human learning capabilities, and thus provides a useful measure of whether POMDP planning can still accelerate learning when a pessimistic learner model is used.

**Discrete Model with Memory:** The key limitation of our first model is its lack of memory of past evidence. A more psychologically plausible model is one in which learners maintain a finite memory of the past $M$ actions. Like the memoryless model, this model assumes that the learner stores her current guess at the true concept, and this guess is updated only when information is shown that contradicts the guess. In this case, the learner shifts to a concept that is consistent with the current evidence and all evidence in the $M$-step history. The transition probability is again proportional to the initial concept probability, and the observation model is deterministic based on the learner's current guess.

**Continuous Model:** A more complex, but natural, view of learning is that the learner maintains a probability distribution over multiple concepts [15]. In this case the state is a $|C|$-dimensional, continuous-valued vector that sums to 1, where $C$ is the set of possible concepts. The state space $S$ is an infinite set of all such vectors, the simplex $\Delta_{|C|}$. The transition function assumes that for *quiz* actions, each state transitions deterministically to itself. For *example* and *question with feedback* actions, state dimensions for concepts that are inconsistent with the provided information are set to zero. The full joint transition probability is then re-normalized. The observation model assumes the learner gives answer $a_n$

to a question with probability equal to the amount of probability she places on concepts that have $a_n$ as the correct answer for this question.

To improve the robustness of our policies to the coarse learner models we employ, all models include two extra parameters, $\epsilon_t$ and $\epsilon_p$. $\epsilon_t$ corresponds to the probability that the learner ignores a given teaching action, resulting in the learner not transitioning to a new concept, while $\epsilon_p$ corresponds to the probability that the learner produces an answer inconsistent with her current guess.

## 4 Finding Policies

Our goal is to compute a policy that selects the best action given a distribution over the learner's current knowledge state, the belief state. Offline POMDP planners compute in advance a policy for each belief in the set of potential beliefs.[3] However, since this set grows exponentially with the number of states, offline approaches cannot scale to the large size of common teaching domains. We instead turn to online POMDP forward search techniques, which have proven promising in other large domains (see [13] for a survey). We compute the future expected cost associated with taking different actions from the current belief state by constructing a forward search tree of potential future outcomes. This tree is constructed by interleaving branching on actions and observations. After the tree is used to estimate the value of each action for the current belief, the best pedagogical action is chosen. The learner then responds to the action, and this response, plus the action chosen, is used to update the belief representing the new distribution over the learner's knowledge state. We then construct a new forward search tree to select a new action for the updated belief.

While forward search solves some of the computational issues in finding a policy, the cost of searching the full tree is $O((|A||Z|)^H)$, where $H$ is the task horizon (i.e., the number of sequential actions considered), and requires an $O(|S|^2)$ operation at each node. This is particularly problematic as the size of the state space may scale with complexity of the learner model: the memoryless model has a state space of size $|C|$, while the discrete model with memory has state space of size $|C||A|^M$ and the continuous model has an infinite state space. To reduce the number of nodes we must search through, we take a similar approach to [12] and restrict the tree by sampling only a few actions. Additionally, we limit $H$ to control the depth of the tree and use an evaluation function at the leaves.

Since the belief state in the continuous model is a distribution over an infinite set of states, we approximate the belief state for this model to make inference tractable. We represent the belief state as a weighted set of probabilistic particles and update these particles based on the transition and observation models (see [8] for more about this technique, known as *particle filtering*). If no particles are consistent with the current observation, we reinitialize the belief state with two particles: one with a distribution induced by rationally updating the prior using all previous evidence and one with a uniform distribution.

[3] Most state-of-the-art offline algorithms try to compute a policy over a subset of the reachable subspace, but this is still typically a very large number of beliefs.

## 5 Empirically Testing Optimized Teaching Policies

POMDP planning provides a way to select actions optimally with respect to a particular learning objective. However, given the simplifications made for computational tractability and that our learner models only approximate true learners, it is necessary to empirically test whether this framework results in more efficient learning. We demonstrate its effectiveness by teaching learners "alphabet arithmetic," a concept-learning task in which letters are mapped to numbers. While this task is artificial, it provides a preliminary evaluation of POMDP planning for problem selection and shares several important characteristics with real teaching domains: it is rich enough that learners may have misconceptions and that we expect some teaching policies to be more effective than others.

In alphabet arithmetic, learners infer a mapping from letters to numbers from a set of equations using letters. For *example* actions, learners are shown an equation where two distinct letters sum to a numerical answer. For instance, $A$ could be mapped to 0 and $B$ to 1, and one might show the learner the equation $A + B = 1$. *Quiz* actions leave out the numerical answer and ask the learner to give the correct sum. *Questions with feedback* combine these two actions. We assume learners have a uniform prior over mappings.

### 5.1 Methods

**Participants.** A total of 40 participants were recruited online and received a small amount of monetary compensation for their participation.
**Stimuli.** All participants were randomly assigned three mappings between the letters $A$–$F$ and the numbers 0–5. These mappings were learned in succession.
**Procedure.** Participants were assigned to either the *control condition*, in which teaching actions for all mappings were chosen randomly, or to the *experimental condition*. Each participant in the experimental condition experienced all three of the teaching policies in random order, one for each mapping learned. The experiment consisted of a sequence of teaching and assessment phases. In each teaching phase, a series of three teaching actions was chosen based on condition. After each teaching phase, participants completed an assessment phase in which they were asked to give the number to which each letter corresponded. Teaching of a given mapping terminated when the participant completed two consecutive assessment phases correctly or when 40 teaching phases had been completed. Within all phases, the equations the participant had seen were displayed on-screen, and participants could optionally record their current guesses about which letter corresponded to which number.
**Computing policies.** We estimated the median time to complete each action type from the control participants: *example* actions took 7.0s, *quiz* actions took 6.6s, and *question with feedback* actions took 12s. These values were the cost for each action in the experimental condition. When computing the action values within the forward search tree, we set the cost for a leaf node to be the probability of not passing the assessment phase multiplied by $10 \cdot \min_a r(a)$, a scaling of the minimum future cost.

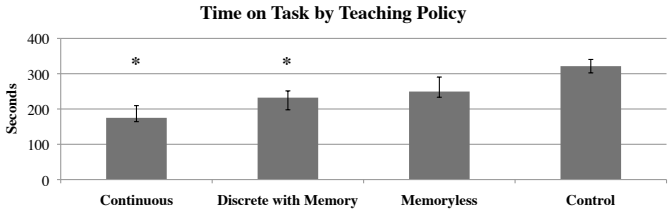


**Fig. 1.** Median time to learn each mapping, by policy type; error bars correspond to bootstrapped 68% confidence intervals (equivalent to one standard error). Asterisks indicate that the policies based on the continuous model and the discrete model with memory result in significantly faster learning than the control.

We set $\epsilon_t$, the probability of ignoring a teaching action, and $\epsilon_p$, the probability of making a production error when answering a question, by finding the values that maximized the log likelihood under a given model of the data from the control condition.[4] For forward planning, we limited the lookahead horizon to two and stopped planning after three seconds.[5] There was delay of three seconds between actions in all conditions to allow time for planning.

### 5.2 Results

We compared the number of phases as well as the time participants took to learn each mapping. Initial inspection showed that the distribution of learning times exhibited a long right tail, so we analyzed results using medians, which are more robust than means to outliers and non-symmetric distributions. There was no significant within-subjects difference in the amount of time or number of phases to learn the first, second, or third mapping (Kruskal-Wallis $p > 0.8$).

Overall, participants taught by POMDP planning took significantly fewer phases to learn each mapping than participants in the control condition (3 phases versus 4, Kruskal-Wallis $p < 0.00005$) and also took significantly less time per mapping (232 seconds versus 321 seconds, Kruskal-Wallis $p < 0.001$); see Figure 1. Planned pairwise comparisons show that all of the POMDP policies resulted in fewer phases to completion than the control, and all POMDP policies but the policy from the memoryless model resulted in significantly faster learning.

Differences in policies occurred based on the learner model used; see Figure 2 for part of one policy. The policy from the memoryless learner model repeats specific example actions more often than the other policies since previous actions

[4] The calculation was performed using the EM algorithm for the two discrete models and using a forward filtering approximation for the continuous model. We found the following values: memoryless model: $\epsilon_t = 0.15$ and $\epsilon_p = 0.019$; discrete model with memory: $\epsilon_t = 0.34$ and $\epsilon_p = 0.046$; and continuous model: $\epsilon_t = 0.14$ and $\epsilon_p = 0.12$.

[5] Policies for the first 9 actions were precomputed with 10 actions sampled at each level. Later actions were precomputed by sampling the following number of actions at each level: 7 and 6 actions for the memoryless model; 8 and 8 actions for the discrete model with memory; and 4 and 3 actions for the continuous model. 16 particles were used for the continuous model, and $M = 2$ for the discrete model with memory.

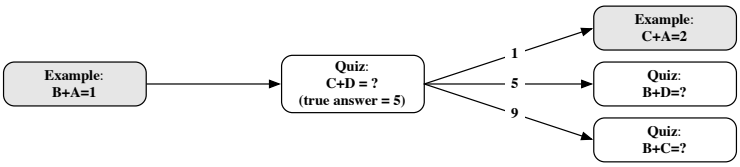


**Fig. 2.** Part of a policy from the discrete model with memory. Possible student answers to the quiz are indicated on the arrows; some are omitted. Based on the student's response, the action after the quiz may correct a misconception, try to better misdiagnose the cause of an incorrect answer, or continue quizzing to try to detect a misconception.

are not stored in memory. The fact that this model did not significantly decrease time to learn suggests that using too pessimistic of a model may be detrimental for problem selection. Overall, policies for this model also asked more questions (39% of actions) than policies for the other models (about 10% of actions). This is because the state of a memoryless learner after an example is known with less certainty since it is constrained only to be consistent with the last example.

Policies for both the discrete model with memory and the continuous model began with six independent equations that fully specify the mapping. This is the policy one might have hand-crafted to teach this task, demonstrating that despite approximations in planning, the POMDP planner finds reasonable teaching policies. Each of the policies for these two models gives examples until there is a high probability the learner is in the correct state, and then asks quiz questions, which are less costly than examples, to detect misconceptions.

## 6 Conclusion

In this work, we described how teaching can be modeled within the POMDP framework and demonstrated the effectiveness of POMDP planning experimentally. The experimental results showed that different learner models result in systematically different policies and that the policies for the more complex learner models were more effective. This illustrates that optimal problem selection depends not only on knowledge of the domain but also on one's assumptions about the learner. Computational challenges still exist for using POMDP planning: despite sampling only a fraction of possible actions and using very short horizons, planning took $2-3$ seconds per action. However, we believe further speed ups are possible through more sophisticated ways of constructing the forward search tree (such as in [13]). Despite such challenges, our work demonstrates the potential of POMDP planning to lead to empirical improvements in learning. POMDP planning provides a natural framework for problem selection that can use the many existing learner models developed in the ITS community. One question not addressed by the current work is whether POMDP planning can identify policies that improve upon those chosen by actual teachers. In future work, we would like to investigate this question in more realistic learning situations, and investigate integrating these ideas in existing tutoring systems.

**Acknowledgements.** This work was supported by a National Defense Science and Engineering Graduate Fellowship to ANR, a National Science Foundation Mathemat-

ical Sciences Postdoctoral Fellowship to EB, and National Science Foundation grant number IIS-0845410 to TLG.

## References

1. Barnes, T., Stamper, J.: Toward automatic hint generation for logic proof tutoring using historical student data. In: Proceedings of the 8th International Conference on Intelligent Tutoring Systems (2008)
2. Brunskill, E., Garg, S., Tseng, C., Pal, J., Findlater, L.: Evaluating an adaptive multi-user educational tool for low-resource regions. In: Proceedings of the International Conference on Information and Communication Technologies and Development (2010)
3. Brunskill, E., Russell, S.: RAPID: A reachable anytime planner for imprecisely-sensed domains. In: Proceedings of the 26th Conference on Uncertainty in Artificial Intelligence (2010)
4. Chang, K., Beck, J., Mostow, J., Corbett, A.: A Bayes net toolkit for student modeling in intelligent tutoring systems. In: Proceedings of the 8th International Conference on Intelligent Tutoring Systems (2006)
5. Chi, M., Jordan, P., VanLehn, K., Hall, M.: Reinforcement learning-based feature selection for developing pedagogically effective tutorial dialogue tactics. In: Proceedings of the 1st International Conference on Educational Data Mining (2008)
6. Conati, C., Muldner, K.: Evaluating a decision-theoretic approach to tailored example selection. In: Proceedings of the 20th International Joint Conference on Artificial Intelligence (2007)
7. Corbett, A., Anderson, J.: Knowledge tracing: Modeling the acquisition of procedural knowledge. User Modeling and User-Adapted Interaction 4(4), 253–278 (1995)
8. Doucet, A., de Freitas, N., Gordon, N.: Sequential Monte Carlo Methods in Practice. Springer, New York (2001)
9. Folsom-Kovarik, J., Sukthankar, G., Schatz, S., Nicholson, D.: Scalable POMDPs for diagnosis and planning in intelligent tutoring systems. In: AAAI Fall Symposium on Proactive Assistant Agents (2010)
10. Murray, R., Vanlehn, K., Mostow, J.: Looking ahead to select tutorial actions: A decision-theoretic approach. International Journal of Artificial Intelligence in Education 14(3), 235–278 (2004)
11. Restle, F.: The selection of strategies in cue learning. Psychological Review 69(4), 329–343 (1962)
12. Ross, S., Chaib-draa, S., Pineau, J.: Bayesian reinforcement learning in continuous POMDPs with application to robot navigation. In: Proceedings of the International Conference on Robotics and Automation (2008)
13. Ross, S., Pineau, J., Paquet, S., Chaib-draa, B.: Online planning algorithms for POMDPs. Journal of Artificial Intelligence Research 32(1), 663–704 (2008)
14. Sondik, E.J.: The Optimal Control of Partially Observable Markov Processes. Ph.D. thesis, Stanford University (1971)
15. Tenenbaum, J.: Rules and similarity in concept learning. In: Advances in Neural Information Processing Systems 12 (2000)
16. Theocharous, G., Beckwith, R., Butko, N., Philipose, M.: Tractable POMDP planning algorithms for optimal teaching in "SPAIS". In: IJCAI PAIR Workshop (2009)
17. Villano, M.: Probabilistic student models: Bayesian belief networks and knowledge space theory. In: Proceedings of the Second International Conference on Intelligent Tutoring Systems (1992)