Empirically Evaluating the Effectiveness of POMDP vs. MDP Towards the Pedagogical Strategies Induction

Shitian Shen, Behrooz Mostafavi, Collin Lynch,
Tiffany Barnes, and Min Chi

Department of Computer Science, North Carolina State University,
Raleigh, North Carolina 27695, US
{sshen,bzmostaf,cflynch,tmbarnes,mchi}@ncsu.edu

Abstract. The effectiveness of Intelligent Tutoring Systems (ITSs) often significantly depends upon their *pedagogical strategies*, the policies used to decide what action to take next in the face of alternatives. In this work, we evaluate two general RL frameworks for policy induction, POMDP &. MDP, and two alternative policy execution models, stochastic & deterministic, through two empirical studies where they are compared against a random yet reasonable baseline policy. Results show that when the contents are controlled to be equivalent, effective RL-induced policies can improve students' learning significantly more than the random baseline and POMDP is more suitable for the task of pedagogical strategy induction than MDP. Post-hoc comparisons suggest that while no significant difference is found between deterministic and random policy execution, stochastic execution is more effective than random execution.

Keywords: Reinforcement Learning, POMDP, MDP, stochastic, ITS

1 Introduction

Reinforcement Learning (RL) offers one of the most promising approaches to applying data-driven decision-making to improve student learning in Interactive Learning Environments. RL algorithms are designed to induce effective policies that determine the best action for an agent to take in any given situation so as to maximize a cumulative reward. Intelligent Tutoring Systems (ITSs) are a type of highly interactive e-learning environment that facilitates learning by providing step-by-step support and contextualized feedback to individual students [6, 19]. These step-by-step behaviors can be viewed as a sequential decision process where at each step the system chooses an action (e.g. give a hint, show an example) from a set of options. *Pedagogical strategies* are policies that are used to decide what action to take next in the face of alternatives.

In recent years a number of researchers have applied RL to induce effective pedagogical polices for ITSs [2, 4, 7, 13]: some apply Markov Decision Processes

(MDPs) thus treating the user-system interactions as fully observable processes [8, 17] while others utilize partially-observable MDPs (POMDPs) [14, 20, 21] to account for hidden states. However, so far as we know, no prior research has directly compared and empirically evaluated the effectiveness of POMDPs and MDPs for the induction of pedagogical strategies. In this work, we focus on pedagogical decisions and report on two empirical studies that are designed to compare POMDPs vs. MDPs directly from two aspects: *state-space representation* and *policy execution*.

State space representation: In RL success depends upon using an effective state representation. When a student trains on an ITS, there are many factors that *may* affect whether or not they benefit from the experience. Thus, applying RL to induce effective pedagogical policies is often complicated by the fact that the state space may be large and continuous. In this work, for example, we consider 133 state features. This severely limits the effectiveness of tabular MDP methods. Most of the prior work with MDPs has relied on smaller set of predefined state representations which consist of features suggested by the learning literature. While the literature provides useful guidance on what factors to consider, there are many alternatives, some of which are ITS-specific. Moreover, many of the relevant factors such as motivation, affect, and prior knowledge, cannot be observed directly nor are they described explicitly. POMDPs, on the other hand, model unobserved factors by using a belief state space. Thus POMDPs for ITSs can explicitly represent two sources of uncertainty: non-determinism in the control process and partial observability of the students' knowledge levels. In the former case the outcome of the tutorial actions and the students' knowledge levels are represented by a probability distribution, and in the latter case, the underlying knowledge levels are observed indirectly via incomplete or imperfect observations. In short, using the belief state space gives POMDP two potential advantages over MDPs: better handling of uncertainty in the state representation, and the ability to incorporate a large range of state features. As a result, we believe that POMDPs will be more effective than tabular MDPs for ITSs.

Policy execution: Most of the prior research on RL for ITSs has used *deterministic* policy execution. That is, when evaluating the effectiveness of RL-induced policies, the system would strictly carry out the actions determined by the policies. In this work, we explore *stochastic* policy execution where at each decision point there is a small probability that the system will deviate from the policy and take a randomly-selected action. We argue that stochastic execution *can* be more effective than deterministic execution for two reasons. First, stochastic execution continues to explore the state space. If the policy is sub-optimal, there is a chance to try other actions and thus students' learning would not be limited. Second, in cases where the decision is crucial, stochastic execution ensures the policy is followed (see section 3.3 for details). Thus if the policy is optimal, students can still benefit from it. We empirically compare stochastic and deterministic policy execution for both the POMDP and MDP frameworks and we hypothesize that stochastic execution can be more effective.

In short, our goal is to fully evaluate the effectiveness of POMDPs vs. tabular MDPs for the induction of pedagogical strategies and the effectiveness of *stochastic* vs. *deterministic* policy execution through two studies. In both studies, we employ a simple baseline pedagogical policy where the system *randomly* decides whether to present the next problem as Worked Example (WE) or as Problem Solving (PS). Because both PS and WE are always considered to be *reasonable* educational intervention in our learning context, we refer to such policy as *random yet reasonable* policy or *random* in the following.

2 Related Work

MDP for ITSs. Iglesias et al. [5] applied MDP-based RL in an ITS that teaches students database design. Their goal was to provide students with direct navigational support through the system's content. They showed that while students using the induced policies had more effective usage behaviors than their non-policy peers, there was no substantive difference in student learning performance. Chi et al. [10] applied the Value Iteration approach to induce pedagogical policies in a physics ITS. They found that the induced RL policy did not outperform the random policy. Similarly, Shen et al.[16] utilized the MDP framework to induce policies based upon both immediate and delayed rewards in a rule-based ITS for deductive logic. They found no significant difference in learning performance between the immediate-reward, delayed-reward, and random policies. In short, most prior work on the application of MDP to ITSs has found no significant learning differences between the induced RL policies and baseline random policies. One potential explanation for this is that MDP relies on a small set of pre-defined state representations, which may not fully represent the real interactive learning environments.

POMDPs for ITSs. Mandel et al. [9] applied POMDPs in combination with a feature compression approach that can handle a wide range of state features to induce policies for an educational game. Their results showed that the induced POMDP policies outperformed both random and expert-designed polices in both simulated and empirical evaluations. Additionally, Rafferty et al. [12] applied POMDP to represent students' latent knowledge through combining graphical models for concept learning with interpreted belief states in the domain of alphabet arithmetic. They found that the POMDP policies significantly outperformed the random policy. Similarly, Clement et al. [3] constructed a student model to track students' individual mastery of each knowledge component, and then combined it with POMDP to induce instructional policies. Their results showed that the POMDP policies outperformed the theory-based policies in terms of students' knowledge levels and time on task.

In short, prior research on applying POMDPs to induce pedagogical strategies has shown that POMDPs can be more effective than both random baseline policies and expert-designed or domain theory-based policies. However, as far as we know, none of these studies has directly compared and empirically evaluated

POMDP and MDP frameworks, nor have they investigated whether stochastic execution will be more effective than deterministic execution in this context.

3 Methods

3.1 MDP vs. POMDP

An **MDP** can be defined as a 4-tuple $\langle S, A, T, R \rangle$, where S denotes the observable state space, defined by a set of features that represent the interactive learning environment; A denotes the space of possible actions for the agent to execute; T represents the transition probability where $p(s, a, s')$ is the probability of transiting from state s to state s' by taking action a. Finally, the reward function R represents the immediate or delayed feedback: $r(s, a, s')$ denotes the expected reward of transiting from state s to state s' by taking action a. The optimal policy π^* for an MDP can be generated via dynamic programming approaches, such as Value Iteration. This algorithm operates by finding the optimal value for each state $V^*(s)$, which is the expected discounted reward that the agent will gain if it starts in s and follows the optimal policy to the goal. Generally speaking, $V^*(s)$ can be obtained by the optimal value function for each state-action pair $Q^*(s, a)$ which is defined as the expected discounted reward the agent will gain if it takes an action a in a state s and follows the optimal policy to the end. The optimal state value $V^*(s)$ and value function $Q^*(s, a)$ can be obtained by iteratively updating $V(s)$ and $Q(s, a)$ via equations 1 and 2 until they converge:

$$Q(s, a) := \sum_{s'} p(s, a, s') \left[r(s, a, s') + \gamma V_{t-1}(s') \right] \tag{1}$$

$$V(s) := \max_{a} Q(s, a) \tag{2}$$

where $0 \leq \gamma \leq 1$ is a discount factor. When the process converges, the optimal policy π^* can be induced corresponding to the optimal Q-value function $Q^*(s, a)$, represented as:

$$\pi^*(s) = argmax_a Q^*(s, a) \tag{3}$$

In the context of an ITS, this induced policy represents the pedagogical strategy by specifying tutorial actions using the current state.

POMDPs are an extension of MDPs, defined by a 7-tuple $\langle$ *S, A, R,* P_h*,* P_o*, B, prior* $\rangle$, where A and R have the same definitions as in MDPs. S represents the *hidden* state space $\{s_1, s_2..., s_K\}$. P_h denotes the transition probability where $p_h(s_i, a, s_j)$ is the probability of transiting from the hidden state s_i to s_j by taking the action a. P_o is the conditional observation probability where $p_o(o, a, s)$ is the probability of the observation o given a hidden state s and the action a. $Prior$ denotes the prior probability distribution of hidden states. B denotes the

belief state space, where $B_t(s_k) = p(s_k|o_{1:t})$ is the probability of the underlying state s_k at a particular time t given the historical observation sequence $o_{1:t}$ and we have:

$$B_t(s_k) = \frac{1}{Z} \sum_{s_i} B_{t-1}(s_i) p_h(s_i, a_t, s_k) p_o(o_t, a_t, s_k) \tag{4}$$

where Z is the normalization factor. In this work we use an Input-Output Hidden Markov Model (IOHMM) [1] to construct the belief state space. In particular, for a given action (input) and observable variables (output) at a time step, its corresponding belief state is calculated by equation 4 via the forward-backward algorithm. Note that the observable variables can be represented in different ways which we will discuss in Sec 3.2. We treat the last observation in each trajectory as the end state in order to evaluate the transition probability from hidden states to the terminal states. The IOHMM parameters are estimated through Expectation-Maximization (EM) algorithm. When training the IOHMM, we run the EM algorithm 10 times with different randomly assigned initial parameter settings in order to avoid local optima and treat the highest likelihood as the global maximum.

The POMDP policy induction procedure can be divided into three steps. First, we transform the training corpus into the hidden state space through the Viterbi algorithm. Second, we implement Q-learning to estimate the Q-values for each hidden state and action pair: (s, a). Third, we estimate the Q value of belief state b and action a at time step t as:

$$Q_t(b, a) = \sum_{s} B_t(s) \cdot Q(s, a) \tag{5}$$

Thus, $Q_t(b, a)$ is a linear combination of the $Q(s, a)$ for each hidden state with its corresponding belief $B_t(s)$. When the process converges, π^* is induced by taking the optimal action a at time t associated with the highest $Q_t(b, a)$.

3.2 Feature Transformation – FAMD

As part of the data pre-processing step, we apply Factor Analysis for Mixed Data (FAMD) to transform our original state feature space which contains both continuous and categorical variables into a principle subspace while maintaining the majority of the relevant information and removing redundancy. When applying FAMD, we standardize the continuous variables and transform categorical variables into a complete disjunctive table which is then scaled by the equation: $x_d^{'} = (x_d - w_d)/\sqrt{w_d}$, where x_d denotes a dimension in the disjunctive table, and $w_d = \frac{1}{N} \sum_{i=1}^{N} x_{di}$. Here w_d refers to the mean of the corresponding x_d. This scaling method balances the impact of variable types on the subsequent analysis. After the features are scaled, we apply Principle Component Analysis (PCA) on the scaled space to extract the important components.

3.3 Policy Execution: Stochastic vs. Deterministic

Once the policies have been induced from either the MDP or POMDP frameworks, most existing ITSs execute them deterministically. That is, the tutor always selects the action with the highest Q-value given the current state. In stochastic policy execution, we generalize this approach by using a *softmax function* [18] which transforms the Q-values into the probability of taking an action a in the current state s by the acceptance probability function:

$$p(s,a) = \frac{e^{\tau \cdot Q(s,a)}}{\sum_{a' \in A} e^{\tau \cdot Q(s,a')}} \tag{6}$$

Here τ is a positive parameter, which controls the variance of probabilities for the state and action pair. In general, when $\tau \to 0$, the stochastic policy execution is close to random decision-making. When $\tau \to +\infty$, the stochastic policy execution becomes deterministic. In order to determine the optimal τ, we use Importance Sampling [11] which can mathematically evaluate the effectiveness of policies with different τ values and the best value of τ is 0.06 for both the MDP- and POMDP-based RL. Moreover, it is important to note that based on equation 6, for a given state s: if the Q-value of the optimal action a^* is much higher than the Q-values of other alternative suboptimal actions, then the stochastic policy execution becomes deterministic in that the probability of carrying out the optimal action would be closer to 1; if the difference between them is small, then the stochastic policy execution becomes closer to random.

4 Deep Thought & Procedure

Deep Thought (DT) is a data-driven ITS used in undergraduate-level Discrete Mathematics (DM) course at North Carolina State University (NCSU). DT provides students with a graph-based representation of logic proofs which allows students to solve problems by adding rule applications (represented as nodes). The system automatically verifies proofs and provides immediate feedback on logical errors. Every problem in DT can be presented in the form of either Worked Example (WE) or Problem Solving (PS). In WE (shown in Figure 1), students are given a detailed example showing the expert solution for the problem or were shown the best step to take given their current solution state. In PS (shown in Figure 2), by contrast, students are tasked with solving the same problem using the ITS or completing an individual problem-solving step. By focusing on the pedagogical decisions of choosing WE vs. PS, which would allow us to strictly control the content to be *equivalent* for all students.

The problems in DT are organized into six strictly ordered levels and in each level students are required to complete 3–4 problems. In the **pre-test**(level 1), all participants receive the same set of PS problems and students performance in this level is used to measure their incoming competence. In the following five training levels 2–6, before the students proceed to a new problem, the system followed the corresponding RL-induced or random policies to decide whether to

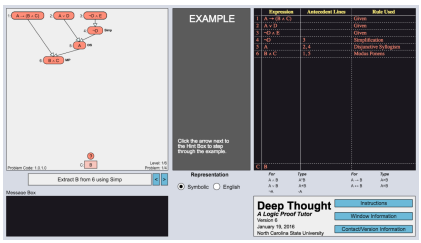

Fig. 1. Worked Example (WE) GUI

Fig. 2. Problem Solving (PS) GUI

present it as PS or WE. The last question on each level is a PS without DT's help and thus functioned as a mini-post-test for evaluating students' knowledge on the concepts of that level. Since the bulk of the relevant content is covered in levels 3–6, the student scores on these four levels are used as our **post-test** to measure their post-training performance. More specifically, we calculated the score as $posttest = \sum_{i=3}^{6} LevelScore(i)/4$. In addition to the pre- and post-test scores, we also evaluated students performance based on **Normalized Learning Gain** ($NLG = \frac{posttest - pretest}{100 - pretest}$) where 100 is the maximum post-test score. In the following, it is important to note that due to class constraints the pre- and post-tests covered different concepts and were collected at different times: the pre-test occurred in a single session before the policies were employed, while the post-test scores were collected at the end of later levels. Therefore the two scores cannot be directly aligned.

4.1 Training Corpus

Our training corpus was collected in the Fall 2014 and Spring 2015 semesters. All students used the same ITS, followed the same general procedure, studied the same training materials, and worked through the same set of training problems. The only substantive difference was the presentation of the materials, WE or PS, randomly decided.

The training dataset contained the interaction logs of 306 students and the average number of problems solved by students was 23.7 and the average time that students spent in the tutor was 5.29 hours. From the interaction logs, we extracted a total of 133 state feature variables, 59 discrete and 74 continuous, to represent the students' behaviors and the interactive learning environment. In addition, we calculated student level scores based on their performance on last problem in each of levels 1–6. For the sake of simplicity, level scores were normalized to $[0, 100]$ and reward functions were defined as the difference between the current and previous level scores.

4.2 Research Questions and Study Overview

In this work, we investigated two primary research questions: 1) Is POMDP more effective than MDP for pedagogical strategy induction in our ITS? and 2)

Is stochastic policy execution more effective than deterministic execution? We conducted two empirical experiments, involving the following **six policies**:

1. **MDP-det:** MDP using 6 features selected via RL-based feature selection methods in [15]; deterministic policy execution.
2. **MDP-sto:** MDP using the same 6 features above; stochastic.
3. **POMDP-det:** POMDP using the same 6 features above; deterministic.
4. **FPOMDP-det:** POMDP using FAMD on 133 features; deterministic.
5. **FPOMDP-sto:** POMDP using FAMD on 133 features; stochastic.
6. **Random:** Random yet reasonable decision (baseline).

Experiment 1 compared the MDP-det, POMDP-det and Random policies and Experiment 2 compared the FPOMDP-sto, FPOMDP-det, MDP-sto and Random policies. Since all students were drawn from the same target population and no significant difference was found between the two Random groups in Experiment 1 and 2, we conducted a post-hoc comparison across two experiments.

5 Experiment 1 (Exp1)

105 undergraduate students who enrolled in the DM course at NCSU in Fall 2016 were randomly assigned to one of three conditions: MDP-det ($N = 45$), POMDP-det ($N = 35$), Random ($N = 30$).

Learning performance. The middle columns in Table 1 show the mean (and SD) for students' corresponding learning performance in Exp1. A one-way ANOVA test showed no significant difference among the three conditions on the pre-test score: $F(2, 102) = 0.33$, $p = 0.72$. Much to our surprise, no significant difference was found on the post-test score and the NLG. Table 1 shows, although POMDP-det had slightly higher post-test score and NLG than MDP-det, the differences were not significant. More surprisingly, both POMDP-det and MDP-det scored lower than Random on the NLG even though the differences were not significant. Note that all of the three conditions' NLGs were not high and it suggested that all three policies may not be very effective.

Behaviors. The last three columns in Table 1 show the total time (in hours) that students spent in DT, and the number of PSs and WEs that were determined by the policies. Note that there were extra 9 problems determined by hard-coded pre-defined rules rather than polices. A one-way ANOVA test showed no significant difference on time among the three conditions nor on the number of WE. But there was significant difference on the number of PSs: $F(2, 102) = 6.82$, $p = .001$. The Tukey HSD test [1] suggested that POMDP-det solved significantly less PSs than both MDP-det ($p = .012$) and Random ($p = .003$).

Discussion. Consistent with prior research on applying MDP to ITSs, Exp1 showed that the MDP policy performed no better than the baseline Random policy. However, unlike prior research on applying POMDP to ITSs, our POMDP policy did not outperform the Random policy. One possible explanation is that

[1] Post hoc comparisons using the Tukey HSD test with Bonferroni correction

Table 1. Learning Performance and Behaviors in Experiment 1

	Learning Performance			Behaviors		
Policy	pre-test	post-test	NLG	Time	#PS	#WE
POMDP-det	40.46(20.89)	49.01(16.37)	0.042(0.50)	3.0(1.8)	5.1(1.7)	6.3(0.9)
MDP-det	41.98(21.21)	45.28(16.21)	-0.094(0.56)	3.7(1.9)	6.2(1.8)	5.9(1.2)
Random	37.83(21.75)	48.0(15.01)	0.066(0.51)	3.4(2.3)	6.4(1.5)	5.8(1.3)

the state representation was limited. In Exp1, we strictly controlled the state features used in MDP to be the same as the observation space for POMDP so that the primary difference between two models was that POMDP used the belief state space. As our MDP policy was not effective, simply adding the belief state space did not make a substantive difference. In other words, by limiting the number of state features in POMDP, we may restricted its full power. Therefore in Experiment 2, we expanded POMDP by generating the belief state space with a large range of features and then compared it against MDP and Random.

6 Experiment 2 (Exp2)

181 students enrolled in the DM course at NCSU in the Spring 2017 semester were randomly assigned into four conditions: FPOMDP-sto ($N = 45$), FPOMDP-det ($N = 46$), MDP-sto ($N = 46$), and Random ($N = 44$).

6.1 Results

The middle columns in of Table 2 show the students' learning performance in Exp2. No significant difference was found among the four conditions on pre-test score: $F(3, 179) = 2.01, p = 0.11$.

Post-test Score & NLG. A one-way ANCOVA test using the pre-test score as a covariate showed a significant difference among the four conditions on the post-test score: $F(3, 178) = 3.87, p = .010$. Similarly, a one-way ANOVA test showed a significant difference among them on the NLG: $F(3, 179) = 4.47, p = .004$. The Tukey HSD tests showed that on the post-test score, FPOMDP-sto scored significantly higher than Random: $p = .008$ with the effect size $d^2 = 0.62$ and FPOMDP-det scored marginally significant higher than Random $p = .090$, $d = 0.41$; on the NLG, the two FPOMDP conditions performed closely and both scored significantly higher than Random: $p = .009$, $d = 0.63$ for FPOMDP-sto and $p = .011$, $d = 0.65$ for FPOMDP-det respectively. Finally, no significant difference was found either between the two FPOMDP conditions and MDP-sto, or MDP-sto and Random.

To summarize, Exp2 shows that FPOMDP-sto > Random on both post-test score and NLG, and FPOMDP-det > Random on the NLG. Similar to Exp1,

[2] Cohen's d is defined as the mean learning gain of the experimental group minus the mean learning gain of the control group, divided by the groups' pooled standard deviation.

Table 2. Learning Performance and Behaviors in Exp2

	Learning Performance			Behaviors		
Policy	pre-test	post-test	NLG	Time	#PS	#WE
FPOMDP-sto	45.03(22.29)	**60.58(21.55)**	**0.253(0.56)**	2.4(1.1)	8.6(1.7)	5.1(0.5)
FPOMDP-det	35.03(20.55)	55.09(15.03)	**0.247(0.49)**	2.0(0.8)	7.8(2.1)	5.5(0.9)
MDP-sto	41.86(23.20)	52.18(21.07)	0.042(0.71)	2.2(1.1)	4.1(1.3)	7.7(1.3)
Random	45.04(23.99)	48.23(18.01)	-0.175(0.77)	2.0(0.9)	6.3(1.3)	5.9(0.9)

no significant difference was found between MDP-sto and Random on either post-test score or NLG.

Behaviors. The last three columns in Table 2 show the total time (in hours) that students spent and the number of PSs and WEs which were determined by policies. A one-way ANOVA test showed that there was no significant difference among the four conditions on time. Additionally, one-way ANOVA tests showed a significant difference on the number of PSs: $F(3, 179) = 69.79, p < .000$ and WEs: $F(3, 179) = 69.69, p < .000$. Specifically, MDP-sto solved the least PS but the most WE among the four conditions. The Tukey HSD test indicated that FPOMDP-sto solved significantly more PSs than MDP-sto ($p < .000$) and Random ($p < .000$) respectively. Thus, we concluded that students under various conditions indeed performed significantly different. However, it is hard to directly make a connection between the behaviors and the learning performance. Further research is needed to assess this in detail.

6.2 Conclusions

While Exp1 shows that POMDP-det is no more effective than either MDP-det or Random, Exp2 indicates that by incorporating a large range of both continuous and categorical state features into the POMDP framework through FAMD, the FPOMDP-det and FPOMDP-sto polices outperform the Random policy while the MDP-sto policy does not. Therefore, our results support that the POMDP framework is more suitable for the task of pedagogical strategy induction than the tabular MDP framework because the former is able to handle a large range of state features.

7 Post-Hoc Comparisons

In both Exp1 and Exp2, the students were drawn from the same target population and were assigned to each condition randomly, thus providing the most rigorous test of our hypotheses. In this section, we conducted a post-hoc comparison across the two experiments in the hope that this wider view will shed some light on our main results. All of the participants were enrolled in the experiments with the same method but in different semesters. One-way ANOVA tests indicated that there was no significant difference between the two Random groups on the pre-test score: $F(1, 72) = 1.73$, $p = 0.19$, the post-test score:

Table 3. Comparisons Results Across Experiment 1 and 2

Measure	Com-POMDP	Com-MDP	Com-Random	Com-Stochastic	Com-Deterministic
pre-test	40.08(21.92)	41.92(22.17)	42.12(23.23)	43.46(22.68)	38.26(21.03)
post-test	**57.87(18.71)**	48.97(19.17)	48.14(16.75)	**56.43(21.61)**	50.53(16.26)
NLG	**0.251(0.53)**	-0.021(0.65)	-0.076(0.68)	**0.14(0.65)**	0.08(0.55)

$F(1, 72) = .003$, $p = 0.95$, and the NLG: $F(1, 72) = 2.28$, $p = 0.14$. Thus, for the purposes of this analysis, we combined the two Random groups into a single Random group ($N = 74$). Therefore, we have a total of six groups as described in section 4.2. A one-way ANOVA test showed no significant difference among the six groups on the pre-test score: $F(6, 281) = 1.19$, $p = 0.31$.

Post-test Scores & NLGs Across Six Groups. A one-way ANCOVA test using the pre-test score as a covariate found a significant difference among the six groups on the post-test score: $F(5, 281) = 4.65$, $p = .000$. Similarly, a one-way ANOVA test found a significant difference among the six groups on the NLG: $F(5, 282) = 3.13$, $p = .009$. Specifically, the Tukey HSD tests found that on the post-test, both FPOMDP-sto and FPOMDP-det scored significantly higher than MDP-det: $p = .001$, $d = 0.81$ and $p = .037$, $d = 0.63$ respectively; FPOMDP-sto also scored significantly higher than the combined Random: $p = .004$, $d = 0.64$ and marginally significantly higher than POMDP-det: $p = .072$, $d = 0.60$. Additionally, the Tukey HSD tests found that on NLG, both FPOMDP-sto and FPOMDP-det scored marginally significantly higher than MDP-det: $p = .085$, $d = 0.62$ and $p = .052$, $d = 0.65$ respectively; FPOMDP-sto scored significantly higher than the combined Random: $p = .032$, $d = 0.53$, while FPOMDP-det scored marginally significantly higher than the combined Random: $p = .098$, $d = 0.54$. No significant difference was found for other pairs of conditions. Thus, the post-hoc comparisons across six groups on post-test score found that FPOMDP-sto, FPOMDP-det > MDP-det and FPOMDP-sto > Random.

Group Combination. To compare POMDP vs. MDP, we combined FPOMDP-sto and FPOMDP-det as the combined POMDP group ($N = 93$), and MDP-sto and MDP-det as the combined MDP group ($N = 86$). Note that POMDP-det was not included in the combined POMDP group because POMDP-det did not use the full power of the POMDP framework of incorporating a large range of state features into consideration. Moreover, to compare Stochastic vs. Deterministic, we combined FPOMDP-sto and MDP-sto as the combined Stochastic group (N= 93), and FPOMDP-det and MDP-det as the combined Deterministic group (N=86). Table 3 shows the pre- and post-test scores, and the NLGs of the combined POMDP, the combined MDP, the combined Random, the combined Stochastic, and the combined Deterministic groups respectively.

7.1 POMDP vs. MDP Framework

A one-way ANOVA test indicated no significant difference on pre-test score among the combined POMDP, the combined MDP and the combined Random

groups. A one-way ANCOVA test using the pre-test score as a covariate on the post-test score showed that there was a significant difference among the three groups: $F(2, 249) = 8.85$, $p = .000$. Similarly, a one-way ANOVA test found that there was a significant difference among them on the NLG: $F(2, 250) = 6.97$, $p = .002$. Tukey HSD tests showed that the combined POMDP significantly outperformed the combined MDP on the post-test score: $p = .001$, $d = 0.47$ and the NLG: $p = .011$, $d = 0.46$; and the combined POMDP also significantly outperformed the combined Random on the post-test score: $p = .000$, $d = 0.55$ and the NLG: $p = .009$, $d = 0.54$. However, no significant difference was found between the combined MDP and the combined Random on either the post-test score or the NLG. These results are consistent with Exp2 results in that POMDPs are more suitable than MDPs for pedagogical policy induction.

7.2 Stochastic vs. Deterministic Policy Execution

A one-way ANOVA test showed no significant difference on the pre-test score among the combined Stochastic, the combined Deterministic, the combined Random groups. A one-way ANCOVA test using the pre-test score as a covariate on the post-test score found a significant difference among the three groups: $F(2, 249) = 5.04$, $p = .007$. A one-way ANOVA test found a marginally significant difference among them on the NLG: $F(2, 250) = 2.78$, $p = .064$. More specifically, while no significant difference was found between the combined Deterministic group and the combined Random on either the post-test score or the NLG, the Tukey HSD tests showed that the combined Stochastic group scored significantly higher than the combined Random: $p = .005$, $d = 0.43$ on the post-test score and marginally significantly on the NLG: $p = .056$, $d = 0.32$. In short, our post-hoc comparisons suggested that while no significant difference was found between deterministic and random policy execution, stochastic policy execution can be more effective than random execution.

8 Conclusions and Future Work

We empirically evaluate the effectiveness of POMDP vs. tabular MDP for the induction of pedagogical strategies and *stochastic* vs. *deterministic* for the policy execution. Our results show that: POMDPs can be more effective than tabular MDPs but their effectiveness does not appear to stem from the belief state space *alone*. Rather it stems from the POMDPs' ability to incorporate a larger range of state features. Furthermore, our results suggest that while no significant difference is found between deterministic and random policy execution, stochastic policy execution can outperform random execution. This may be caused by the fact that our RL-induced policies may still be sub-optimal. Stochastic policy execution provides a chance for the system to explore alternative actions and to obtain better performance, while deterministic policy execution cannot. In future work, we will empirically compare the effectiveness of POMDP with the continuous MDP using the same large set of features, in order to verify whether the belief state space representation is a crucial advantage of POMDP.

References

1. Bengio, Y., Frasconi, P.: An input output hmm architecture. In: Advances in neural information processing systems. pp. 427–434 (1995)
2. Chi, M., VanLehn, K., Litman, D., Jordan, P.: Empirically evaluating the application of reinforcement learning to the induction of effective and adaptive pedagogical strategies. User Modeling and User-Adapted Interaction 21(1-2), 137–180 (2011)
3. Clement, B., Oudeyer, P.Y., Lopes, M.: A comparison of automatic teaching strategies for heterogeneous student populations. In: EDM 16-9th International Conference on Educational Data Mining (2016)
4. Doroudi, S., Holstein, K., Aleven, V., Brunskill, E.: Towards understanding how to leverage sense-making, induction and refinement, and fluency to improve robust learning. International Educational Data Mining Society (2015)
5. Iglesias, A., Martínez, P., Aler, R., Fernández, F.: Reinforcement learning of pedagogical policies in adaptive and intelligent educational systems. Knowledge-Based Systems 22(4), 266–270 (2009)
6. Koedinger, K.R., Anderson, J.R., Hadley, W.H., Mark, M.A.: Intelligent tutoring goes to school in the big city (1997)
7. Koedinger, K.R., Brunskill, E., Baker, R.S., McLaughlin, E.A., Stamper, J.: New potentials for data-driven intelligent tutoring system development and optimization. AI Magazine 34(3), 27–41 (2013)
8. Levin, E., Pieraccini, R., Eckert, W.: A stochastic model of human-machine interaction for learning dialog strategies. IEEE Transactions on speech and audio processing 8(1), 11–23 (2000)
9. Mandel, T., Liu, Y.E., Levine, S., Brunskill, E., Popovic, Z.: Offline policy evaluation across representations with applications to educational games. In: Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems. pp. 1077–1084 (2014)
10. M. Chi, VanLehn, K., Litman, D.J., Jordan, P.W.: Empirically evaluating the application of reinforcement learning to the induction of effective and adaptive pedagogical strategies. User Model. User-Adapt. Interact. 21(1-2), 137–180 (2011)
11. Peshkin, L., Shelton, C.R.: Learning from scarce experience. arXiv preprint cs/0204043 (2002)
12. Rafferty, A.N., Brunskill, E., Griffiths, T.L., Shafto, P.: Faster teaching via pomdp planning. Cognitive science 40(6), 1290–1332 (2016)
13. Rowe, J.P., Lester, J.C.: Improving student problem solving in narrative-centered learning environments: A modular reinforcement learning framework. In: International Conference on Artificial Intelligence in Education. pp. 419–428. Springer (2015)
14. Roy, N., Pineau, J., Thrun, S.: Spoken dialogue management using probabilistic reasoning. In: Proceedings of the 38th Annual Meeting on Association for Computational Linguistics. pp. 93–100. Association for Computational Linguistics (2000)
15. Shen, S., Chi, M.: Aim low: Correlation-based feature selection for model-based reinforcement learning. In: EDM. pp. 507–512 (2016)
16. Shen, S., Chi, M.: Reinforcement learning: the sooner the better, or the later the better? In: Proceedings of the 2016 Conference on User Modeling Adaptation and Personalization. pp. 37–44. ACM (2016)
17. Singh, S., Litman, D., Kearns, M., Walker, M.: Optimizing dialogue management with reinforcement learning: Experiments with the njfun system. Journal of Artificial Intelligence Research 16, 105–133 (2002)

18. Sutton, R.S., Barto, A.G.: Reinforcement learning: An introduction. MIT press (1998)
19. Vanlehn, K.: The behavior of tutoring systems. International journal of artificial intelligence in education 16(3), 227–265 (2006)
20. Williams, J.D., Young, S.: Partially observable markov decision processes for spoken dialog systems. Computer Speech & Language 21(2), 393–422 (2007)
21. Zhang, B., Cai, Q., Mao, J., Chang, E., Guo, B.: Spoken dialogue management as planning and acting under uncertainty. In: INTERSPEECH. pp. 2169–2172 (2001)