

## ABSTRACT

SHEN, SHITIAN. Improving Learning & Reducing Time: A Constrained Action Based Reinforcement Learning Approach. (Under the direction of Min Chi.)

Intelligent Tutoring Systems (ITSs) have been shown to be highly effective at improving student learning in real classrooms through scaffolding, adaptive supports and contextualized feedback to individual learners. ITSs generally need to decide *what* to teach such as which problem is assigned to students and *how* to teach such as whether present a problem as a problem solving or a worked example, with the guide of *pedagogical strategy*, which maps students' behavior information and current learning context into an optimal decision. However, inducing an effective pedagogical strategy (policy) is challenging due to the lack of learning theory for decision-making in the ITSs and modeling for the relation between the pedagogical decision and student learning. There is a clear need to advance data-driven approaches to address this challenge.

Specifically, we applied the data-driven approaches to deal with a particular type of pedagogical decision: problem solving (PS) vs. worked example (WE) in an ITS named Deep Thought, which teaches undergraduate students logic proof and strictly controls the learning content to be equivalent to individual students. When solving a PS, students are required to complete a problem with tutor's support such as hint, while students are provided with a problem as well as an expert solution step by step when doing a WE. In this work, we construct the effective pedagogical strategy in an offline manner using three different Reinforcement Learning (RL) frameworks including tabular Markov Decision Process (MDP), Partially Observable MDP (POMDP), and constrained action-based POMDP (CAPOMDP).

We explored four aspects of the RL framework: 1) *state representation*: presenting students' learning behavior and learning context based on either the set of selected features or the belief state space where each state is associated with the probability; 2) *reward function*: investigating the impact of immediate and delayed rewards on the effectiveness of policies, and detecting the effectiveness of policies using learning gain and time as reward separately; 3) *policy execution*: comparing the effectiveness of stochastic policy execution with that of the deterministic execution; 4) *action-based constraints*: investigate the impact of constraints on the effectiveness of policies.

A series of experiments is conducted to empirically evaluate the RL policies and to compare their effectiveness with the Random baseline. Results indicate that there is an aptitude-treatment interaction (ATI) effect, where a particular type of students are sensitive to the policies in that the RL policy can significantly improve their learning performance comparing with Random baseline, while other students are not sensitive in that they can always learn regardless of policies. Furthermore, we run several statistical analysis to identify which type of PS and WE patterns has a significant positive or negative impact on student learning.

© Copyright 2019 by Shitian Shen

All Rights Reserved

Improving Learning & Reducing Time: A Constrained Action Based  
Reinforcement Learning Approach

by  
Shitian Shen

A dissertation submitted to the Graduate Faculty of  
North Carolina State University  
in partial fulfillment of the  
requirements for the Degree of  
Doctor of Philosophy

Computer Science

Raleigh, North Carolina

2019

APPROVED BY:

---

Tiffany Barnes

---

Jonathan Rowe

---

Dennis Bahler

---

Min Chi  
Chair of Advisory Committee

## **ACKNOWLEDGEMENTS**

This material is based upon work supported by the National Science Foundation under Grant No. 1432156 “Educational Data Mining for Individualized Instruction in STEM Learning Environment”, 1651909 “CAREER: Improving Adaptive Decision Making in Interactive Learning Environment”, and 1726550 “Integrated Data-driven Technologies for Individualized Instruction in STEM Learning Environments”.

# TABLE OF CONTENTS

|   |             |
|---|-------------|
| <b>LIST OF TABLES</b> .....   | <b>vi</b>   |
| <b>LIST OF FIGURES</b> .....  | <b>viii</b> |
| <b>Chapter 1 Introduction</b> .....                                   | <b>1</b>    |
| 1.1 Pedagogical Decision: Worked Example vs. Problem Solving .....    | 2           |
| 1.2 Reinforcement Learning Frameworks .....                           | 2           |
| 1.2.1 Tabular MDP Framework .....                                     | 3           |
| 1.2.2 POMDP Framework .....   | 3           |
| 1.2.3 CAPOMDP Framework .....   | 4           |
| 1.3 Contributions .....   | 4           |
| 1.4 Outline of the Thesis .....                                       | 6           |
| <b>Chapter 2 Related Work</b> .....                                   | <b>7</b>    |
| 2.1 Pedagogical Decisions: Worked Example vs. Problem Solving .....   | 7           |
| 2.2 Applying RL into Educational Domain .....                         | 9           |
| 2.2.1 Markov Decision Process (MDP) .....                             | 9           |
| 2.2.2 Partially Observable Markov Decision Process (POMDP) .....      | 10          |
| 2.2.3 Deep RL Framework .....   | 10          |
| 2.2.4 Summarization of RL Applications in Educational Domain .....    | 11          |
| 2.3 Constrained Reinforcement Learning .....                          | 11          |
| 2.4 Aptitude Treatment Interaction Effect .....                       | 12          |
| <b>Chapter 3 Markov Decision Process</b> .....                        | <b>14</b>   |
| 3.1 Introduction .....  | 14          |
| 3.2 MDP Framework .....   | 15          |
| 3.2.1 Policy Induction .....  | 16          |
| 3.2.2 Policy Evaluation .....   | 16          |
| 3.2.3 Stochastic Policy Execution .....                               | 17          |
| 3.3 Pedagogical Decisions in a Logic Tutor: Deep Thought .....        | 18          |
| 3.3.1 Overview of Deep Thought .....                                  | 18          |
| 3.3.2 Two Training Datasets: <i>DT-Imme</i> and <i>DT-Delay</i> ..... | 20          |
| 3.3.3 Feature Set .....   | 20          |
| 3.4 Feature Selection on the MDP Framework .....                      | 21          |
| 3.4.1 Related Work For Feature Selection in RL .....                  | 22          |
| 3.4.2 Five Correlation Metrics .....                                  | 22          |
| 3.4.3 Correlation-based Feature Selection Approaches .....            | 24          |
| 3.4.4 PreRL-FS Approach .....   | 25          |
| 3.4.5 Ensemble Approach .....   | 26          |
| 3.4.6 Comparison Results for Feature Selection Approaches .....       | 26          |
| 3.5 Experiments Overview .....  | 28          |
| 3.5.1 Research Questions .....  | 28          |
| 3.5.2 Reinforcement Learning Policies .....                           | 29          |

|                  |  |           |
|------------------|--|-----------|
| 3.5.3            | Experiments Overview . . . . .   | 29        |
| 3.5.4            | ATI effect: Splitting Students Based on Response Time . . . . .                    | 30        |
| 3.5.5            | Statistic Analysis . . . . .   | 31        |
| 3.6              | Four Experiments . . . . .   | 31        |
| 3.6.1            | Experiment 1: Preliminary Feature Selection . . . . .                              | 31        |
| 3.6.2            | Experiment 2: Ensemble Feature Selection & Immediate vs. Delayed Rewards . . . . . | 34        |
| 3.6.3            | Experiment 3: Low Correlation-based Feature Selection . . . . .                    | 38        |
| 3.6.4            | Experiment 4: Stochastic Policy Execution . . . . .                                | 41        |
| 3.6.5            | Conclusions of Experiments . . . . .   | 42        |
| 3.7              | Post-hoc Comparisons . . . . .   | 43        |
| 3.7.1            | Global Median Split . . . . .  | 44        |
| 3.7.2            | The Impact of Feature Selection on RL Policies . . . . .                           | 45        |
| 3.7.3            | Problem Solving vs. Worked Example under policies . . . . .                        | 47        |
| 3.8              | Conclusions, Limitations, & Discussion . . . . .                                   | 49        |
| <b>Chapter 4</b> | <b>Partially Observable Markov Decision Process . . . . .</b>                      | <b>52</b> |
| 4.1              | Introduction . . . . .   | 52        |
| 4.2              | POMDP Framework . . . . .  | 53        |
| 4.2.1            | Feature Transformation . . . . .   | 54        |
| 4.2.2            | Belief State Estimation . . . . .  | 54        |
| 4.2.3            | POMDP Policy Induction . . . . .   | 55        |
| 4.3              | Experiments Overview & Research Questions . . . . .                                | 55        |
| 4.3.1            | Induced Policies . . . . .   | 55        |
| 4.3.2            | Research Questions . . . . .   | 56        |
| 4.3.3            | Empirical Evaluation . . . . .   | 56        |
| 4.4              | Experiment 1: POMDP with Selected Features . . . . .                               | 57        |
| 4.4.1            | Participants & Conditions . . . . .  | 57        |
| 4.4.2            | Results . . . . .  | 57        |
| 4.4.3            | Conclusion & Discussion . . . . .  | 58        |
| 4.5              | Experiment 2: POMDP with a wide range of features . . . . .                        | 58        |
| 4.5.1            | Participants & Conditions . . . . .  | 58        |
| 4.5.2            | Results . . . . .  | 58        |
| 4.5.3            | Conclusion & Discussion . . . . .  | 60        |
| 4.6              | Post-hoc Comparison . . . . .  | 60        |
| 4.6.1            | Across Six Groups . . . . .  | 60        |
| 4.6.2            | POMDP vs. MDP Framework . . . . .  | 62        |
| 4.6.3            | Stochastic vs. Deterministic Policy Execution . . . . .                            | 62        |
| 4.6.4            | Post-hoc Discussion . . . . .  | 64        |
| 4.7              | Conclusions, Limitations, & Discussion . . . . .                                   | 64        |
| <b>Chapter 5</b> | <b>Constrained Action-based POMDP . . . . .</b>                                    | <b>65</b> |
| 5.1              | Introduction . . . . .   | 65        |
| 5.2              | Problem Statement . . . . .  | 67        |
| 5.3              | Constrained Action-based POMDP (CAPOMDP) . . . . .                                 | 68        |
| 5.3.1            | Factored State Representation . . . . .  | 69        |

|  |  |            |
|--|--|------------|
| 5.3.2  | Reward Function  | 69         |
| 5.3.3  | Policy Induction   | 70         |
| 5.4  | Experiment Setup   | 70         |
| 5.4.1  | Procedure & Evaluation                                     | 70         |
| 5.4.2  | Training Corpus  | 71         |
| 5.4.3  | Policy Execution   | 72         |
| 5.5  | Experiment Overview & Research Questions                   | 72         |
| 5.5.1  | Induced Policies   | 72         |
| 5.5.2  | Research Questions   | 73         |
| 5.6  | Experiment 1: Improving Learning Gain Using CAPOMDP        | 74         |
| 5.6.1  | Participants & Conditions                                  | 74         |
| 5.6.2  | Experiment 1 Results                                       | 75         |
| 5.6.3  | Experiment 1 Conclusion                                    | 78         |
| 5.7  | Experiment 2: Reducing Total Time Using CAPOMDP            | 78         |
| 5.7.1  | Participants & Conditions                                  | 78         |
| 5.7.2  | Experiment 2 Result  | 79         |
| 5.7.3  | Experiment 2 Conclusion                                    | 82         |
| 5.8  | Post-hoc Comparisons                                       | 83         |
| 5.8.1  | Participants & Conditions                                  | 83         |
| 5.8.2  | Post-hoc Comparison Results                                | 84         |
| 5.8.3  | Conclusion of Post-hoc Comparisons                         | 88         |
| 5.9  | Log Analysis   | 88         |
| 5.9.1  | Impact of Action-based Constraints on Learning Performance | 88         |
| 5.9.2  | Relation between Behaviors to Learning Outcomes            | 90         |
| 5.9.3  | Impact of Pedagogical Strategy on Behavior                 | 93         |
| 5.9.4  | Summarization of Log Analysis                              | 97         |
| 5.10   | Conclusion & Discussion                                    | 97         |
| <b>Chapter 6 Conclusions and Future Work</b> |  | <b>99</b>  |
| 6.1  | Conclusions  | 99         |
| 6.2  | Limitation   | 100        |
| 6.3  | Future Work  | 101        |
| <b>BIBLIOGRAPHY</b>                          |  | <b>103</b> |

## LIST OF TABLES

|            |  |    |
|------------|--|----|
| Table 2.1  | Reinforcement Learning Applications in Educational Domain . . . . .                | 11 |
| Table 3.1  | Reinforcement Learning Policies in Four Experiments . . . . .                      | 29 |
| Table 3.2  | Overview of Experiments . . . . .  | 30 |
| Table 3.3  | <i>MDP-ECR</i> Policy . . . . .  | 32 |
| Table 3.4  | Pre-test and Transfer Post-test in Experiment 1 . . . . .                          | 33 |
| Table 3.5  | Ensemble-Imme Policy . . . . .   | 35 |
| Table 3.6  | Ensemble-Delay Policy . . . . .  | 36 |
| Table 3.7  | Pre-test and Transfer Post-test in Experiment 2 . . . . .                          | 37 |
| Table 3.8  | Pairwise Contrasts on Adjusted Transfer Post-test in Experiment 2 . . . . .        | 38 |
| Table 3.9  | WIG-Low Policy . . . . .   | 39 |
| Table 3.10 | Pre-test and Transfer Post-test in Experiment 3 . . . . .                          | 40 |
| Table 3.11 | Pre-test and Transfer Post-test in Experiment 4 . . . . .                          | 42 |
| Table 3.12 | Size of each group in post-hoc comparisons . . . . .                               | 44 |
| Table 3.13 | Pre-test and Transfer Post-test Score across Experiment 1-3 . . . . .              | 45 |
| Table 3.14 | Pairwise Contrasts on Adjusted Transfer Post-test in Post-hoc Comparison . . . . . | 46 |
| Table 3.15 | PS and WE Counts and Comparisons for each Policy across Experiment 1-3 . . . . .   | 48 |
|            |  |    |
| Table 4.1  | Implemented policies in two experiments for POMDP study . . . . .                  | 55 |
| Table 4.2  | Overview of Experiments . . . . .  | 56 |
| Table 4.3  | Mean and SD of learning performance for each condition in Experiment 1 . . . . .   | 57 |
| Table 4.4  | Mean and SD of Behavior Variables for each condition in Experiment 1 . . . . .     | 58 |
| Table 4.5  | Mean and SD of learning performance for each condition in Experiment 2 . . . . .   | 59 |
| Table 4.6  | Mean and SD of behavior variables for each condition in Experiment 2 . . . . .     | 59 |
| Table 4.7  | learning Performance In Post-hoc comparison For POMDP Study . . . . .              | 61 |
| Table 4.8  | Learning performance for each framework in post-hoc comparison . . . . .           | 63 |
| Table 4.9  | Learning performance for each policy execution in post-hoc comparison . . . . .    | 63 |
|            |  |    |
| Table 5.1  | Implemented policies in two experiments for CAPOMDP study . . . . .                | 73 |
| Table 5.2  | Overview of Experiments . . . . .  | 74 |
| Table 5.3  | Participants and Conditions in Experiment 1 . . . . .                              | 75 |
| Table 5.4  | Learning Performance in Experiment 1 . . . . .                                     | 76 |
| Table 5.5  | Learning Performance of HighIC Groups in Experiment 1 . . . . .                    | 77 |
| Table 5.6  | Learning Performance of LowIC Groups in Experiment 1 . . . . .                     | 77 |
| Table 5.7  | Participants and Conditions in Experiment 2 . . . . .                              | 79 |
| Table 5.8  | Learning Performance in Experiment 2 . . . . .                                     | 80 |
| Table 5.9  | Learning Performance of HighIC Groups by condition in Experiment 2 . . . . .       | 81 |
| Table 5.10 | Learning Performance of LowIC Groups by condition in Experiment 2 . . . . .        | 81 |
| Table 5.11 | Participants and Conditions in Post-hoc Comparison . . . . .                       | 83 |
| Table 5.12 | Learning Performance in Post-Hoc Comparison . . . . .                              | 84 |
| Table 5.13 | Learning Performance of HighIC Groups in Post-Hoc Comparison . . . . .             | 85 |
| Table 5.14 | Learning Performance of LowIC Groups in Post-Hoc Comparison . . . . .              | 85 |
| Table 5.15 | Learning Performance of High vs. Low Carry-out . . . . .                           | 89 |

|            |   |    |
|------------|---|----|
| Table 5.16 | Definition, Mean and standard deviation (SD) of Behavior Variables . . . . .  | 91 |
| Table 5.17 | Correlation Tests Between Behavior Variables with Learning Outcomes . . . . . | 92 |
| Table 5.18 | Behavior Variables by Condition . . . . .                                     | 95 |

## LIST OF FIGURES

|             |  |    |
|-------------|--|----|
| Figure 3.1  | Interface for Worked Example . . . . .   | 18 |
| Figure 3.2  | Interface for Problem Solving . . . . .  | 19 |
| Figure 3.3  | DT-Immed . . . . .   | 26 |
| Figure 3.4  | DT-Delay . . . . .   | 26 |
| Figure 3.5  | DT-Immed . . . . .   | 27 |
| Figure 3.6  | DT-Delay . . . . .   | 27 |
| Figure 3.7  | Interaction effect for the adjusted transfer post-test score in Experiment 1 . . | 33 |
| Figure 3.8  | Interaction effect for the adjusted transfer post-test score in Experiment 2 . . | 37 |
| Figure 3.9  | Interaction effect for the adjusted transfer post-test score in Experiment 3 . . | 40 |
| Figure 3.10 | Interaction effect for adjusted transfer post-test score across Experiment 1-3 . | 46 |
| Figure 4.1  | The process of POMDP policy induction . . . . .                                  | 54 |
| Figure 5.1  | The general process of the CAPOMDP policy induction . . . . .                    | 68 |
| Figure 5.2  | Unconstrained policy execution . . . . .   | 72 |
| Figure 5.3  | CAPOMDP policy execution . . . . .   | 72 |

## CHAPTER

# 1

# INTRODUCTION

Intelligent Tutoring Systems (ITSs), as one type of highly interactive e-learning environment, have been widely used in the educational domain [D'M11; Koe97; Van07]. While ITSs hold great promise, they are difficult and expensive to construct and are often brittle and inflexible in their interactions with students. The effective ITSs generally provide step-by-step adaptive support and contextualized feedback to individual learners at run-time [Koe97; Van06], and determine *what* to teach such as which problem is present to students, and *how* to teach such as whether give a hint, whether present a given problem as a worked example. The step-by-step behavior can be viewed as a sequential decision process where at each step the system chooses an appropriate action from a set of options. This decision process is governed by the *pedagogical strategy* or policy which selects the action based upon the given user input and the current learning context [Igl09b].

However, inducing pedagogical strategies in ITSs is challenging. On one hand, the relation between a pedagogical decision of the tutor and students' learning outcomes such as learning gain, can not be immediately observed. On the other hand, each decision affects the student's subsequent behaviors and performance, which also has an impact on the tutor's next decision. Therefore, the effectiveness of the current decision depends upon the effectiveness of subsequent decisions, and this iterative process cannot be easily solved by directly optimizing an objective function. One of potential solutions is to construct a set of hand-crafted rules. However, it's not capable of the adaptive pedagogical decision-making due to the limitation that there is no validated theory of decision-making in ITSs, and these rules are often project-specific and are rarely evaluated. There

is a clear need to advance data-driven approaches for pedagogical decision-making.

## 1.1 Pedagogical Decision: Worked Example vs. Problem Solving

In this work, we induce the pedagogical strategy in an ITS, called Deep Thought (DT) [MB17], which teaches undergraduate students logic proof in “Discrete Mathematics” course at NCSU. DT contains a total of seven learning phases or levels covering different knowledge components. Our goal is to induce an effective policy to deal with a particular type of pedagogical decision in DT: whether to provide students with a **Worked Example (WE)** of a problem or to ask them to engage in **Problem Solving (PS)**. When providing a WE, the tutor will show an expert solution to a problem step by step, while students are required to complete a problem independently with tutor’s support such as hint when solving a PS. Since students are required to solve the same set of problems either in PS or WE, the contents of DT are strictly controlled to be equivalent, which makes the pedagogical strategy induction task more challenging. One possible reason is that there is limited space for the pedagogical strategy to improve students’ learning performance comparing with the baseline strategy. Although there are many different theories about when and how to provide WE or PS, widespread consensus does not exist. This is why we chose to take a data-driven approach to induce the policy. More importantly, little evidence has been presented to date demonstrating that ITSs have effective pedagogical skills or that their pedagogical decisions have a direct impact on student learning when the content is controlled to be equivalent.

## 1.2 Reinforcement Learning Frameworks

Reinforcement Learning (RL) is a robust and well-established approach for data-driven decision-making in interactive environments. In recent years, a number of researchers have begun applying RL to induce effective pedagogical policies for ITSs [Chi11; Dor15; Koe13; RL15], including *offline* methods, where policy is induced from the static training corpus [Sin00; Chi11], and *online* approaches, where policy is estimated and updated continuously from real or simulated user interactions on the run time [Lev00; Bec00]. In this work, we mainly focus on *offline policy induction* for two reasons. First, there is no pre-defined state representation, so we need to construct the state space from a training corpus. Second, collecting student-system interaction data in ITSs is expensive, so we repeatedly use a training corpus to induce and to improve policy in an offline manner. Specifically, we apply three different RL frameworks including Tabular Markov Decision Process (MDP) [Lev00; Sin02] and Partially Observable Markov Decision Process (POMDP) [Roy00; WY07; Zha01] and Constrained Action-based POMDP (CAPOMDP).

### 1.2.1 Tabular MDP Framework

We explore the tabular MDP framework from three aspects including 1) the *reward function*: induce policies using either immediate or delayed reward; 2) the *state representation*: propose the correlation-based feature selection approaches to construct various discrete state space based on different set of discretized features for policy induction; and 3) *policy execution*: execute the policy deterministically or stochastically on the run time of tutor.

We conduct four experiments from Spring 2015 to Fall 2016 in order to answer three research questions based on the tabular MDP framework: 1) Does the immediate reward facilitate the MDP framework to induce a more effective pedagogical policy than the delayed reward ? 2) Does the correlation-based feature selection approach have a positive impact on the effectiveness of the policy ? 3) Can stochastic policy execution further improve the effectiveness of the induced policy as compared to deterministic execution?

We note that the tabular MDP framework has two weaknesses. First, tabular MDPs are not able to efficiently deal with high dimensional feature spaces including both discrete and continuous variables. In particular, there is no prior knowledge about the appropriate structure of state representations for learning pedagogical policies in the ITS domain. Although we implement a feature selection approach to construct the state space, we can't guarantee that the state space and the transitions among states are able to fully express students' learning behaviors and learning process, respectively. Second and more important, many other factors such as motivation, affect, prior knowledge and proficiency, are useful for decision-making, but they can neither be observed directly nor described explicitly, and thus cannot be included in the MDP framework.

### 1.2.2 POMDP Framework

Different from tabular MDP, POMDP generates a belief state space, which comprises probability distributions over latent states. Specifically, POMDP assigns probability to each latent state given the observation with a wide range of features at each time step [Pin03; Pin06]. Consequently, POMDP is able to deal with a large set of features, which can be efficiently transferred into a belief state space. Although the belief state is hard to interpret, prior research has verified that it is useful to track students' mastery of Knowledge Components [Cle16] and knowledge level [Man14; Raf16]. Therefore, we hypothesize that the POMDP framework can induce an effective policy.

We first induce a POMDP policy and an MDP policy given the same set of selected features using immediate reward. Additionally, we use the full power of POMDP to generate the POMDP policy given a wide range of features using immediate reward. We conduct two empirical studies from Fall 2016 to Spring 2017 for comparing the effectiveness of POMDP against MDP and Random policies. Although prior research focuses on policy induction either using POMDP or using MDP, as far as we know, there is rare work which empirically compares POMDP and MDP. Furthermore, we

also explore the POMDP policy execution: deterministic vs. stochastic. Specifically, we have three research questions: 1) Does POMDP induce more effective policy than MDP and Random given a limited feature set ? 2) Is the POMDP policy, induced given a wide range of features, more effective than the MDP and Random policies ? 3) Can stochastic POMDP policy execution further improve the effectiveness of the POMDP policy as compared to deterministic execution?

### 1.2.3 CAPOMDP Framework

It is worthwhile to mention that both POMDP and MDP frameworks are used to handle the standard RL scenario and to induce the unconstrained policies. Different from POMDP and MDP, we construct the constrained action-based POMDP (CAPOMDP) framework by integrating the action-based constraints into the POMDP framework, to deal with a *constrained action-based* RL (CARL) scenario, which involves the additional action-based constraints such as a maximum number of times that an agent may take a specific action. For instance, the action-based constraints in DT are presented as: the last problem on each level must be done in PS, and prior to reaching that problem the students must complete at least one PS and one WE. DT did not allow students stay in the extreme situation where they always solve PS or WE in a level. In this scenario, the early decisions impose special constraints on the future actions. In other words, the available actions for an agent at any given situation are governed not only by the current state but also by prior decisions. Consequently, when deciding the next action, the agent should take these constraints into account. Therefore, we apply CAPOMDP to induce the constrained policy.

Furthermore, we induce two types of constrained policies:  $CAPOMDP_{LG}$  using learning gain as the immediate reward for improving students' learning performance, and  $CAPOMDP_{Time}$  using time as the immediate reward for reducing students' time on task. We conduct two empirical studies from Fall 2017 to Spring 2018 for comparing the effectiveness of  $CAPOMDP_{LG}$  and  $CAPOMDP_{Time}$  with three baselines including the POMDP policy induced by the full power of POMDP as mentioned above, a Deep Reinforcement Learning induced policy and the random policy. There are three research questions: 1) can  $CAPOMDP_{LG}$  outperform baseline policies in terms of learning gain? 2) can  $CAPOMDP_{Time}$  significantly reduce students' time compared with the baseline? 3) Do action-based constraints hurt the effectiveness of unconstrained policies?

## 1.3 Contributions

In short, we extensively explore the tabular MDP and POMDP frameworks and propose CAPOMDP to induce the pedagogical strategies for PS vs. WE decision making, and conduct extensive empirical experiments to investigate the effectiveness of induced RL policies, which are evaluated based on students' learning outcomes such as post-test score, learning gain, and time. Overall, our main

contributions are summarized as follows:

- We find that a consistent Aptitude Treatment Interaction (ATI) effect [CS77a; Sno91] exists across studies: certain students are less sensitive to the induced policies in that they achieve a similar learning performance regardless of policies employed, whereas other students are more sensitive in that their learning is highly dependant on the effectiveness of the policies.
- We induce MDP policies based on immediate and delayed rewards respectively and detect that immediate reward facilitates tabular MDP to induce a more effective policy than delayed one through empirical experiments.
- We propose correlation-based feature selection approaches for state representation in the tabular MDP framework, and empirically verify that MDP policies outperform a random baseline in terms of students' learning performance.
- While previous research mainly execute the RL-induced policies deterministically, we explore both deterministic and stochastic policy execution, and empirical results suggest that the stochastic can be more effective than deterministic execution for both tabular MDP and POMDP policies.
- As of the time of our work on this subject, this is the first study to compare and empirically evaluate the effectiveness of POMDP vs. tabular MDP policies.
- We propose the CAPOMDP framework, a constrained action-based reinforcement learning framework, to induce the policy considering the action-based constraints in our tutor.
- We induce two different CAPOMDP policies:  $CAPOMDP_{LG}$  and  $CAPOMDP_{Time}$  using learning gain and time as rewards respectively, and verify that they are more effective than baselines in that  $CAPOMDP_{LG}$  can significantly improve students' learning performance; and  $CAPOMDP_{Time}$  can significantly reduce total time that students spent in the tutor.

## 1.4 Outline of the Thesis

The rest of the thesis is organized as follows:

**Chapter 2:** This chapter presents related work required to comprehend the rest of the thesis.

**Chapter 3:** This chapter presents the application of the MDP framework for pedagogical strategy induction. The MDP policies are constructed based on the set of features selected through different feature selection approaches. Four experiments are conducted to investigate the effectiveness of the MDP policies.

**Chapter 4:** This chapter shows the application of the POMDP framework. Two empirical studies are conducted to compare the effectiveness of the POMDP policies with that of the MDP and the Random policies.

**Chapter 5:** In this chapter, the CAPOMDP framework is proposed to deal with the constrained action-based RL problem. The CAPOMDP policies are induced using either learning gain or time as reward. Two empirical studies are conducted to compare the effectiveness of the CAPOMDP policies against the POMDP, Deep RL and the Random policies.

**Chapter 6:** This chapter concludes the thesis and presents the limitations of the implemented RL frameworks. This is followed by the future enhancements of the application of RL in the ITSs.

## CHAPTER

# 2

## RELATED WORK

### 2.1 Pedagogical Decisions: Worked Example vs. Problem Solving

A great deal of research has investigated the impacts of worked examples (WE) and problem solving (PS) on student learning [MI11; McL14; Naj14; Sal10]. During PS, students are given a training problem which they must solve independently or with partial assistance, while during WE, students are shown a detailed solution to the problem.

In 2008, McLaren et al. [McL08] compared WE-PS pairs with PS-only, where every student was given the same 10 training problems. Students in the PS-only condition were required to solve every problem while students in the WE-PS condition were given 5 example-problem pairs. Each pair consists of an initial worked example problem followed by tutored problem solving. They found no significant difference in learning performance between the two conditions; however, the WE-PS group spent significantly less time on task than the PS group.

McLaren et al. [MI11] found similar results in two subsequent studies in 2011, which compared learning gains and time on task for high school chemistry students given 10 identical problems in three conditions: WE, PS, and WE-PS pairs. There were no significant differences among the three groups in terms of learning gains, but the WE group spent significantly less time on task than the other two conditions, and no significant time on task difference was found between the PS and WE-PS conditions. A follow-up 2014 study compared four conditions: WE, tutored PS, untutored PS, and Erroneous Examples (EE) in high school stoichiometry [McL14]. Students in the EE con-

dition were given *incorrect* worked examples containing between 1 and 4 errors and were tasked with correcting them. Again the authors found no significant differences among the conditions in terms of learning gains, and as before the WE students spent significantly less time than the other groups. More specifically, for time on task they found that:  $WE < EE < untutored\ PS < tutored\ PS$ . WE students took only 30% of the total time of the tutored PS students.

The advantages of WE were also demonstrated in another study in the domain of electrical circuits [VG11]. In that study, Van et al. compared four conditions: WE, WE-PS pairs, PS-WE pairs (problem-solving followed by an example problem), and PS only. Their results showed that the WE and WE-PS students significantly outperformed the other two groups, and no significant difference was found among four conditions in terms of time on task. Additionally, Razzaq et al. [RH09] designed an experiment on comparing worked examples vs. problem solving in an ITS that teaches mathematics. They found that more proficient students benefit more from WE when controlling for time, while less proficient students benefit more from PS.

Some existing theories of learning suggest that when deciding whether to present PS or WE, a tutor should take into account several factors, including the students' current knowledge model. Vygotsky [Vyg78] coined the term "zone of proximal development" (ZPD) to describe the space between abilities that a student may display independently and those that they may display with support. He hypothesized that the most learning occurs when students are assigned tasks within their ZPD. In other words, the task should neither be so simple that they can achieve it independently or trivially, nor so difficult that they simply cannot make progress even with assistance. We expect, based upon this theory, that if students are somewhat competent in all the knowledge needed for solving a problem, the tutor should present the problem as a PS, and provide help only if the students fail so that they can practice their knowledge. If students are completely unfamiliar with the problem, however, then the tutor should present the problem as a WE. Brown et al. [Bro89] describe a progression from WE to PS following their "model, scaffold & fade" rubric. [KA07] by contrast defined an "assistance dimension", which includes PSs and WEs. The level of assistance a tutor should provide may be resolved differently for different students and should be adaptive to the learning environment, the domain materials used, the students' knowledge level, their affect state and so on. Typically, these theories are considerably more general than the specific decisions that ITS designers must make, which makes it difficult to tell if a specific pedagogical strategy is consistent with the theory. This is why we wish to derive pedagogical policy for PS/WE directly from empirical data.

Finally, compared with all previous studies in which the PSs and WEs are generally designed by domain experts or expert-like system developers, in this work both PSs and WEs are constructed through Mostafavi and Barnes data-driven approach using previous students' log files [MB17]. In short, prior research on WE and PS has shown that WE can be as or more effective than PS or alternating PS with WE, and the former can take significantly less time than the latter two [McL14;

Ren02; MI11; Mos15]. As opposed to previous work, which involves hard-coded rules for providing PS or WE, we apply the Reinforcement Learning approach to induce the pedagogical strategy which explicitly indicates how to make decisions given the current state related to students' learning process, the learning context in the tutor, as well as the predefined constraints.

## 2.2 Applying RL into Educational Domain

### 2.2.1 Markov Decision Process (MDP)

MDP [Lit94; SB98a] is a widely used reinforcement learning framework in educational applications. Beck et al. [Bec00] investigated temporal difference learning to induce pedagogical policies that would minimize the time students spend on completing problems in AnimalWatch, an ITS that teaches arithmetic to grade school students. They used simulated students in the training phase of their study and used time as an immediate reward given that student's time can be assessed at each step. In the test phase, the new AnimalWatch with induced pedagogical policy was empirically compared with the original version. They found that the policy group spent significantly less time per problem than their non-policy peers.

Iglesias and her colleagues applied online Q-learning with time as the immediate reward to generate a policy in RLATES, an intelligent educational system that teaches students database design [Igl09b; Igl09a; Igl03]. The goal of inducing the policy was to provide students with direct navigation support through the system's content and to help them learn more efficiently. They also used simulated students in the training phase and evaluated the induced policy by comparing the performance of both simulated and real students using RLATES with that of other students using IGNATES, which provides indirect navigation support without RL. Their results showed that students using RLATES spent significantly less time than students using IGNATES, but there was no significant difference in students' level of knowledge, evaluated by the exam.

Martin et al. [MA04] applied a model-based RL method with delayed reward to induce policies that would increase the efficiency of hint sequencing on Wayang Outpost, a web-based ITS which prepares students for the mathematics section of the Scholastic Aptitude Test. During the training phase, the authors used a student model to generate the training data for inducing the policies. In the test phase, the induced RL policies were tested on a simulated student model and students' performance was evaluated by learning level, a customized score function. The results showed that simulated students following RL policies achieved a significantly better learning level than the non-policy group.

Additionally, Chi et al. [Chi11] applied a model-based RL method with delayed reward to induce pedagogical policies to improve the effectiveness of Cordillera, an Intelligent Natural Language Tutoring System that teaches students college physics. They collected an exploratory corpus by train-

ing human students on the ITS that makes random decisions. Their empirical evaluation showed the induced policies were significantly more effective than the previous policies based on students' normalized learning gain (NLG).

### **2.2.2 Partially Observable Markov Decision Process (POMDP)**

POMDP [Jaa95; KS98] is another widely used framework in educational domains. Different from the MDP framework where the state space is constructed by a set of observable features, the POMDP framework uses a belief state space to model the unobserved factors, such as students' knowledge level and proficiency. Mandel et al. [Man14] combined a feature compression approach that can handle a large range of state features with POMDP to induce policies for an educational game. The induced policies with the immediate reward outperformed both random and expert-designed policies in both simulated and empirical evaluations.

Rafferty et al. [Raf16] applied POMDP to represent students' latent knowledge by combining embedded graphical models for concept learning with interpreted belief states in the domain of alphabet arithmetic. They applied POMDP to induce policies using time as the reward, with a goal of reducing the expected time for learners to comprehend concepts. They evaluated policies using simulated and real-world studies and found that the POMDP-based policies significantly outperformed a random policy.

Clement et al. [Cle16] constructed models to track students' individual mastery of each Knowledge Component in their foreign language learning ITS. They combined POMDP with the student models to induce teaching policies using learning gain as the immediate reward. The results of a series of simulated studies showed that the POMDP policies outperformed the learning theory-based policies in terms of students' knowledge levels on task. Similarly, Whitehill et al. [WM17] implemented POMDP to induce a teaching policy with the purpose of minimizing the expected time in their ITS. The belief state of POMDP is constructed based on a modified student model which hypothesized that students cannot always fully absorb the examples and only partially update their belief state. They conducted a real-world study and verified that the POMDP policy performed favorably compared to two hand-crafted teaching policies.

### **2.2.3 Deep RL Framework**

Wang et al. [Wan17] applied a deep RL framework for personalizing interactive narratives in an educational game. They designed the rewards based on normalized learning gain (NLG) and found that, in simulation studies, the students with the Deep RL policy achieved a higher NLG score than with the linear RL agent. Furthermore, Narasimhan et al. [Nar15] implemented a Deep Q-Network (DQN) approach in a text-based strategy game, where the state is represented by a Long Short-Term Memory (LSTM) network, and the Q value is approximated by a multi-layered neural network, and

the reward is designed based on the performance of game quest. Using simulations, they found that the deep RL policy significantly outperformed the random policy in terms of quest completion.

**Table 2.1** Reinforcement Learning Applications in Educational Domain

| Framework | Prior Work                | Reward    | Experiment       | Evaluation     |
|-----------|---------------------------|-----------|------------------|----------------|
| MDP       | Beck et al. [Bec00]       | Immediate | Simulation       | Time           |
|           | Iglesias et al. [Igl09a]  | Immediate | Simulated & Real | Time & Perform |
|           | Martin et al. [MA04]      | Delay     | Simulation       | Perform        |
|           | Chi et al. [Chi11]        | Delay     | Laboratory       | Perform        |
| POMDP     | Mandel et al. [Man14]     | Immediate | Simulated & Real | Performance    |
|           | Rafferty et al. [Raf16]   | Immediate | Simulated & Real | Time           |
|           | Clement et al. [Cle16]    | Immediate | Simulation       | Performance    |
|           | Whitehill et al. [WM17]   | Immediate | Real             | Time           |
| Deep RL   | Wang et al. [Wan17]       | Immediate | Simulation       | Performance    |
|           | Narasimhan et al. [Nar15] | Immediate | Real             | Performance    |

### 2.2.4 Summarization of RL Applications in Educational Domain

Table 2.1 summarizes the related work about the RL applications in the educational domain. Although tabular MDP has the explainable state space and explicit rules which map a state into an action, it is challenging to construct an effective state space with the limited set of features, which directly impact the effectiveness of MDP policies. While both POMDP and Deep RL have been shown to be highly effective in many real-world applications with high dimensional feature spaces, they generally require a great deal of training data, especially Deep RL. More importantly, it is often hard to interpret the induced POMDP and Deep RL policies.

Compared with previous research, we extensively explore both MDP and POMDP frameworks from three aspects including reward function, state representation, and policy execution. We also leverage the constraints into POMDP framework. Furthermore, we evaluate the effectiveness of induced RL policies using a series of the empirical experiments conducted in real classroom settings.

## 2.3 Constrained Reinforcement Learning

In general, prior research on constrained RL has focused on inducing the optimal policy subject to constraints such as *safety* and *cost*. For example, systems that physically interact with humans need to satisfy the basic safety parameters or engage in risk avoidance [Ach17]. Similarly, Lee et al. [Lee17]

applied Bayesian RL algorithm to make the robot reach a target position as quickly as possible while avoiding dangerous places (say a crater) that might render them irretrievable. Garcia et al. [GF12] proposed the safe RL algorithm to explore the safe state space in dangerous and continuous control tasks.

The constrained RL problem can be transformed into the normal RL problem by assigning the negative or high cost for the particular actions that trigger constraints. Williams et al. [WY07] proposed the POMDP-based spoken dialogue system needs to successfully complete the task while minimizing the length of a dialogue by assigning the negative reward to the action which extended the length of the dialogue. Similarly, Hanheide et al. [Han17] assigned a unique cost to each action and applied the POMDP framework to induce the policy in a robot planning task, and tried to minimize the cost of a policy, which is the linear combination of the cumulative cost for the executed actions and the cumulative reward for the goal state. Sanner et al. [San10] introduced the Relational Dynamic Influence Diagram Language to present the factored MDP or POMDP frameworks, and directly hard-coded constraints for each pair of states and actions to maintain the situation that the agent reached a legal state and executed a legal action at each step.

Some of prior work specified a *cost* function, which is similar to the reward function and directly solves the constrained RL problem. For example, both Dolgov et al. [DD05] and Altman et al. [Alt99] applied the constrained MDP framework to induce the policy subject to the upper bound of the cumulative cost generated from the cost function. Furthermore, Kim et al. [Kim11] proposed the point-based value iteration approach to induce the policy based upon the constrained POMDP framework. Poupart et al. [Pou15] applied the linear programming approach to approximate the value functions of the state for both reward and cost. So far as we know, no prior work has directly sought to address the action-based constraints in the context of ITS.

## 2.4 Aptitude Treatment Interaction Effect

Previous work shows that the aptitude treatment interaction (ATI) effect commonly exists in many real-world studies. More formally, the ATI effect states that instructional treatments are more or less effective to individual learners depending on their abilities [CS77b]. For example, Kalyuga et al. [Kal03] empirically evaluated the effectiveness of worked examples (WE) vs. problem solving (PS) on student learning in programmable logic. Their results show that WE is more effective for inexperienced students while PS is more effective for experienced learners.

Moreover, D’Mello et al. [D’M10] compared two versions of ITSs: one is an affect-sensitive tutor which selects the next problem based on students’ affective and cognitive states combined, while the other is the original tutor which selects the next problem based on students’ cognitive states alone. An empirical study shows that there is no significant difference between two version of tutors for students with high prior knowledge. However, there was a significant difference for students

with low prior knowledge: those who trained on the affect-sensitive tutor had significantly higher learning gain than their peers using the original tutor.

Chi et al. [CV10] investigated the ATI effect in the domain of probability and physics, and their results showed that the high incoming competence students can learn regardless of instructional interventions, while for students with low incoming competence, those who follow more effective instructional interventions learned significantly more than those following less effective interventions. In our prior work, it is consistently shown that for pedagogical decisions on WE vs. PS, certain learners are always less sensitive in that their learning is not affected, while others are more sensitive to variations in different policies. For example, Shen et al. [SC16b] trained students in an ITS for logic proofs, then divided students into the Fast and Slow groups based on time, and found that the Slow groups are more sensitive to the pedagogical strategies while the Fast groups are less sensitive.

In our prior work, it is consistently shown that for pedagogical decisions on WE vs. PS, certain learners are always less sensitive in that their learning is not affected, while others are more sensitive to variations in different policies. For example, [SC16b] trained students in an ITS for logic proofs, then divided students into the Fast and Slow groups based on time, and found that the Slow groups are more sensitive to the pedagogical strategies while the Fast groups are less sensitive.

# MARKOV DECISION PROCESS

## 3.1 Introduction

We explore the tabular MDP framework from three aspects including *reward function*, *state representation*, and *policy execution*. This chapter is modified from papers published in [SC16a; She18b].

**Reward Function.** In general, real-world RL applications often contain two types of rewards: immediate reward, which is the immediate feedback after taking an action, and delayed reward, which is the reward received later after taking more than one action. The longer rewards are delayed, the harder it becomes to assign credit or blame to particular actions or decisions. On the other hand, learning short-term performance boosts may not result in long-term learning gains. Thus, in this work we explore both immediate and delayed rewards in our policy induction, and empirically evaluate the impact of the induced policies on student learning. Our results show that using immediate rewards can be more effective than using delayed rewards.

**State Representation.** For RL, as with all machine learning tasks, success depends upon an effective state representation. When a student interacts with an ITS, there are many factors that might determine whether the student learns well from the ITS, yet many other factors are not well understood. To make the RL problem tractable, our approach is to begin with a large set of features to which a series of feature-selection methods are applied to reduce them to a tractable subset. In this work, we apply a series of correlation-based feature selection methods to RL: first we explored the option of selecting the next feature that is the *most correlated (High option)* to the currently

selected feature set and then the option of selecting the *least correlated (Low option)*. The correlation in our case is calculated between features. Additionally, the high correlation-based option is commonly used for supervised learning where the features that are most highly correlated with the output labels, are often selected [YP97; LL06; CS14; Kop15]. Section 3.4.6 shows that indeed the high-correlation option outperformed two baseline methods: the random baseline and also the best feature selection explored in our previous work [Chi11]. However, for our dataset, the high option-selected features tend to be homogeneous. Different from the supervised learning tasks, we hypothesize that it is more important to have heterogeneous features in RL that can grasp different aspects of learning environments. Therefore, we also explore the low correlation-based option for feature selection with a goal to increase the diversity of the selected feature set. To do so, we select the next feature that is the least correlated with the current selected features to add to the the state representation. Our results show that the low correlation-based option significantly outperformed not only the high option but also the other two baselines.

**Policy Execution.** In most of the prior work with RL in ITSs, deterministic policy execution is used. 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. We argue that stochastic execution can be more effective than deterministic execution because if the RL induced policy is sub-optimal, under the stochastic policy execution, it would still be possible for the system to carry out the optimal action; whereas if the induced policy is indeed optimal, our approach will make sure that when the decisions are crucial, the stochastic policy execution would behave like deterministic policy execution in that the optimal action will be carried out (see section 3.2.3 for details). We empirically evaluate the effectiveness of the stochastic policy execution but our results show that there is a ceiling effect.

The rest of this chapter is arranged as follows: Section 3.2 describes the reinforcement learning framework and Markov Decision Process. Section 3.3 describes the tutorial decisions, Deep Thought tutor, our training data and state representation. Section 3.4 describes five correlation metrics and then introduces our proposed feature selection methods. Section 3.5 presents the overview of our four empirical studies and research questions. Section 3.6 reports experimental results for each of the four experiments. Section 3.7 presents our post-hoc comparison results. Finally, we summarize our conclusions, limitations in Section 3.8.

## 3.2 MDP Framework

The Markov Decision Process (MDP) is one of the most widely used RL frameworks. In general, an MDP is 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 transitioning 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 transitioning from state  $s$  to state  $s'$  by taking action  $a$ . Since we apply the tabular MDP framework, reward function  $R$  and transition probability table  $T$  can be easily estimated from the training corpus. The goal of an MDP is to generate the deterministic policy  $\pi : s \rightarrow a$  that maps each state onto an action.

### 3.2.1 Policy Induction

Once the tuple  $\langle S, A, T, R \rangle$  is set, the optimal policy  $\pi^*$  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 3.1 and 3.2 until they converge:

$$Q(s, a) := \sum_{s'} p(s, a, s') [r(s, a, s') + \gamma V_{t-1}(s')] \quad (3.1)$$

$$V(s) := \max_a Q(s, a) \quad (3.2)$$

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

$$\pi^*(s) = \operatorname{argmax}_a Q^*(s, a) \quad (3.3)$$

where  $\pi^*$  is the deterministic policy that maps a given state into an action. In the context of an ITS, this induced policy represents the pedagogical strategy by specifying tutorial actions using the current state.

### 3.2.2 Policy Evaluation

The effectiveness of the MDP policy is estimated by Expected Cumulative Reward (ECR) [TL08; Chi11]. The ECR of a policy  $\pi$  is calculated by the average over the value function of initial states. It's defined as:

$$ECR(\pi) = \sum_i \frac{N_i}{N} \times V^\pi(S_i) \quad (3.4)$$

Where  $N$  denotes the number of trajectories in the training corpus, i.e. the total number of the

initial states; and  $N_i$  denotes the number of states  $S_i$  as the initial states in the training corpus. In our case, the trajectories have a finite time horizon. Thus, ECR evaluates the expected reward of the initial states. The higher the ECR value of a policy, the better the policy is expected to perform.

### 3.2.3 Stochastic Policy Execution

The crucial part of stochastic policy execution is to assign a probability to each action. Note that in a policy  $\pi$ , each action  $a$  for a particular state  $s$  is associated with a Q-value, called  $Q^\pi(s, a)$  calculated by using equation 3.1 in Section 3.2. Thus, we transform  $Q^\pi(s, a)$  into probability  $p^\pi(s, a)$  by the *softmax function* [SB98b], shown as follows:

$$p^\pi(s, a) = \frac{e^{\tau \cdot Q^\pi(s, a)}}{\sum_{a'} e^{\tau \cdot Q^\pi(s, a')}} \quad (3.5)$$

Here  $\tau$  is a positive parameter, which controls the variance of probabilities for the state and action pair. Generally speaking, when  $\tau \rightarrow 0$ , the stochastic policy execution is close to random decision-making. When  $\tau \rightarrow +\infty$ , the stochastic policy execution becomes deterministic. In order to determine the optimal  $\tau$ , we use Importance Sampling [PS02] which can mathematically evaluate the effectiveness of policies with different  $\tau$  values. Specifically, Importance Sampling (*IS*) of a policy  $\pi$  is formulated as follows:

$$IS(\pi|\mathcal{D}) = \frac{1}{N_{\mathcal{D}}} \sum_{i=1}^{N_{\mathcal{D}}} \left[ \prod_{t=1}^{L^i} \frac{p^\pi(s_t^i, a_t^i)}{p^d(s_t^i, a_t^i)} \cdot \left( \sum_{t=1}^{L^i} \gamma^{t-1} r_t^i \right) \right] \quad (3.6)$$

Where  $N_{\mathcal{D}}$  denotes the number of trajectories in the training corpus  $\mathcal{D}$ ;  $L^i$  is the length of the  $i$ th trajectory;  $s_t^i$ ,  $a_t^i$  and  $r_t^i$  are the state, action and reward at the  $t$ th time step of the  $i$ th trajectory respectively;  $p^d(s_t^i, a_t^i)$  is the probability of taking the action  $a_t^i$  for the state  $s_t^i$ , calculated based on the other policy  $d$ , which generates the training corpus  $\mathcal{D}$ . In our case, the decision in the training corpus is randomly decided, thus  $p^d(s_t^i, a_t^i)$  always equal to 0.5. In general, the higher value of  $IS(\pi|\mathcal{D})$ , the better policy  $\pi$  is supposed to be.

We explore a wide range of  $\tau$  and find that the optimal value of  $\tau$  is 0.06 for the MDP-based policies. Moreover, it is important to note that based on Equation 3.5, 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.

### 3.3 Pedagogical Decisions in a Logic Tutor: Deep Thought

#### 3.3.1 Overview of Deep Thought

**Deep Thought** (DT) is a data-driven ITS used in the undergraduate-level Discrete Mathematics (DM) course at North Carolina State University (NCSU) [MB17]. DT provides students with a graph-based representation of logic proofs which allows students to solve problems by applying logic rules to derive new logical statements, represented as nodes. The system automatically verifies proofs and provides immediate feedback on rule application (but not strategy) 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 3.1), students are given a detailed example showing the expert solution for the problem or were shown the best next step to take given their current solution state. In PS (shown in Figure 3.2), by contrast, students are tasked with solving the same problem using the ITS or completing an individual problem-solving step. Focusing on the pedagogical decisions of choosing WE vs. PS allows us to strictly control the content to be *equivalent* for all students.

The screenshot shows the interface for a Worked Example in the Deep Thought Logic Proof Tutor. On the left, a proof graph displays six nodes: 1 ( $A \rightarrow (B \wedge C)$ ), 2 ( $A \vee D$ ), 3 ( $\neg D \wedge E$ ), 4 ( $\neg D$ ), 5 ( $A$ ), and 6 ( $B \wedge C$ ). Arrows indicate dependencies: 1 and 2 lead to 3; 3 and 4 lead to 5; 5 and 4 lead to 6. Node 4 is labeled 'Simp' and node 6 is labeled 'MP'. Below the graph, a hint box contains a question mark and the text 'Extract B from 6 using Simp'. The interface also includes a 'Representation' section with radio buttons for 'Symbolic' (selected) and 'English', and a table of logical symbols. At the bottom right, there are buttons for 'Instructions', 'Window Information', and 'Contact/Version Information', along with the text 'Deep Thought A Logic Proof Tutor Version 6 January 19, 2016 North Carolina State University'.

| Expression                     | Antecedent Lines | Rule Used             |
|--------------------------------|------------------|-----------------------|
| 1 $A \rightarrow (B \wedge C)$ |                  | Given                 |
| 2 $A \vee D$                   |                  | Given                 |
| 3 $\neg D \wedge E$            |                  | Given                 |
| 4 $\neg D$                     | 3                | Simplification        |
| 5 $A$                          | 2, 4             | Disjunctive Syllogism |
| 6 $B \wedge C$                 | 1, 5             | Modus Ponens          |

Figure 3.1 Interface for Worked Example

All of the hints that students receive for PS in DT are data-driven. Specifically, next-step hints for a PS are constructed by using previous successful student solutions which include the current proof state, and by matching current expressions in the proof. The hint presented at the current proof state guides the student to the most frequent next step that had resulted in successful completion of the proof given that proof state [Sta13], and is given to the student below the proof con-



Figure 3.2 Interface for Problem Solving

instruction window on the left hand side of the tutor (shown in Figure 3.2). The hints are in the format of “Use expression X and expression Y to derive expression Z using rule”. Students are given the opportunity to request hints on-demand by clicking the “Get Hint” button next to the dialogue box; however, if students stay in the current proof state for longer than the median step time of that problem or a maximum of 30 seconds, DT automatically presents the available hint. The WEs were constructed in a similar manner, where the most efficient (shortest-path) solution of the current proof from previous student solutions was used for a step-by-step presentation of the proof with procedurally constructed instructions given to the student below the proof window (Figure 3.2). At each step, the instructions for constructing the next step are presented in the same format as the next-step hints until the conclusion is reached.

The problems in DT are organized into six strictly ordered levels with 3–4 problems per level. Level 1 functions as a *pre-test* in that all participants receive the same set of PS problems. In the five training levels 2–6, before the students proceed to a new problem, the system follows the corresponding RL-induced or random policies to decide whether to present the problem as PS or WE. The last question on each level is a PS without DT’s help and thus functions as a quiz for evaluating students’ knowledge of the concepts of that level. After completing the entire training in DT, students take an in-class exam, referred to as the *transfer post-test*.

While students’ pre-test and transfer post-test scores are used for evaluating student learning performance, *it is important to note that the two scores cannot be directly aligned to measure student learning gains in that they cover different domain concepts*. On the other hand, a significant correlation was found between students’ pre-test and transferred post-test scores:  $r = 0.17, n =$

241,  $p = .005$ . Therefore, when comparing the transfer post-test scores in the following, we used ANCOVA tests with pre-test scores as the covariate.

Finally, note that when inducing RL policies using training data set, reward functions are generated based on level scores because there was a significant positive correlation between students' level scores and transfer post-test scores. Given that the ultimate goal of the DT tutor is to improve students' performance in the real classroom exam, the transfer post-test score, in the following we used the transfer post-test score to evaluate students' learning performance and to investigate the effectiveness of pedagogical policies.

### 3.3.2 Two Training Datasets: *DT-Imme* and *DT-Delay*

Our training dataset was collected in the Fall 2014 and Spring 2015 semesters, with a total of 306 students involved. All students were trained on DT where whether to present the next problem as a WE or a PS was *randomly* decided. The average number of problems solved by students was 23.7 and the average time that each student spent in the tutor was 5.29 hours. In addition, we calculated students' level scores based on their performance on the last problem in each of levels 1–6. For the sake of simplicity, level scores were normalized to  $[0, 100]$ . If the students quit the tutor during the training, we assigned a strong negative reward, -300 in this case, on the last problem they attempted. Furthermore, the **immediate reward** was defined as the difference between the current and previous level scores, and the **delayed reward** was defined as the difference of the level scores between level 1 and 6. From the interaction logs, we represent each observation using a high-dimensional feature space introduced in the following section. Combing observation with two types of rewards, we construct two different types of training datasets named *DT-Imme* and *DT-Delay* respectively.

### 3.3.3 Feature Set

A total of 133 state features were extracted from the DT log files. They include 45 categorical features and 78 continuous features that can be grouped into five categories listed as follows:

1. **Autonomy (AM)**. This category relates to the amount of student work done. For example, *interaction* denotes the cumulative number of student clickstream interactions and *hintCount* denotes the number of times a student clicked the hint button during problem solving. There are a total of 12 features in the AM category, including 8 categorical and 4 continuous features.
2. **Temporal Situation (TS)**. This category encodes the time-related information about the work process. For example, *avgTime* denotes the average time taken per problem, and *TotalPSTime* denotes the total time for solving a particular problem. There are a total of 13 continuous features in the TS category.

3. **Problem Solving (PS).** This category encodes information about the current problem solving context. For example, *probDiff* is the difficulty of the current solved problem; *NewLevel* indicates whether the current solved problem is in a new level in the tutor. There are a total of 30 features in the PS category, including 13 categorical and 17 continuous features.
4. **Performance (PM).** This category describes information about the student’s performance during problem solving. For example, *RightApp* denotes the number of correct rule applications. There are a total of 36 features in the PM category, including 24 categorical and 12 continuous features.
5. **Student Action (SA).** This category is a tutor-specific category for DT. It evaluates the statistical measurement of students’ behavior. For instance, *actionCount* denotes the number of non-empty-click actions that students take; *AppCount* denotes the number of clicks for derivation of a logical expression. There are a total of 32 continuous features in the SA category.

Before feature selection and policy induction, we discretized the continuous features by exploring k-means clustering first and then a simple median split. The latter is conducted only if we failed to get balanced clusters from the former. More specifically, the general discretization process is 1) Given a continuous feature, we start by using k-means with k equal to 5 and generate the clusters; 2) if the size of the clusters are not balanced, we reduce the value of K, until balanced clusters are constructed; 3) Finally, if k equals 1, we use median split to discretize the feature.

In this work, we focus on applying different feature selection approaches to generate a small set of features to construct the state space in a tabular MDP framework. By doing so, we can shed some light on what the most important features are for decision-making on PS vs. WE. Moreover, when applying RL in real-world scenarios, we may not always have the full computation power to track all of the features at once. Next, we describe the feature selection approaches in Section 3.4.

### 3.4 Feature Selection on the MDP Framework

One of the biggest challenges of applying the tabular MDP framework into DT is the high dimensional feature space. In the state representation, each state is a vector representation composed of a number of state features and thus the state space grows exponentially in the number of state features, which would cause a data sparsity problem (the available data is too little to cover each state in the state space) and would exponentially increase the computational complexity. On the other hand if only including a small set of features, while existing learning literature and theories give helpful guidance on state representation, we argue that such guidance is often considerably more general than the specific state features chosen. For example, to describe a student’s knowledge level, we can use "Percentage Correct" defined as the number of the correct student entries

divided by the total number of the student entries, or "Number of Correct" defined as the number of the correct student entries, or "Number of Incorrect" defined as the number of the incorrect student entries and so on. When making specific decisions about including a feature on student knowledge level in the state, for example, it is often not clear which of these features should be included. Therefore a more general state representation approach is needed. To this end, this project began with a large set of features to which a series of feature-selection methods were applied to reduce them to a tractable subset.

### 3.4.1 Related Work For Feature Selection in RL

Much previous work on feature selection for RL mainly focused on model-free RL. Model-free algorithms learn a value function or policy directly from the experience while interacting with the agent. [KN09] applied Least-Squares Temporal Difference (LSTD) with *Lasso* regularized items to approximate the value function as well as to select an effective feature subset. Similarly, [Kel06] applied LSTD to approximate a value function and select a feature subset by implementing *Neighborhood Component Analysis* to decompose approximation error, which can be used to evaluate the efficacy of the feature subset. [Bac09] explored the penalization of an approximation function by using *Multiple Kernel learning*. Additionally, [Wri12] proposed the feature selection embedded in a neuro-evolutionary function which approximates the value function, and they selected each feature based on its contribution to the evolution of network topology.

[Chi11] previously investigated 10 feature selection methods, called *RLpre-FS* (Sec. 3.4.4). These methods were implemented to derive a set of various policies, where features are mostly selected based on the single feature's performance or covariance in training data. The results showed there was no consistent winner and in some particular cases these methods perform no better than the random baseline method.

Different from prior work, our features are selected based on the correlations through two steps: 1) a new feature is selected based on its correlation with the current "optimal" subset of features; 2) for different sets of state features, the same  $A$ ,  $R$  and training data are used for estimating  $T$  when applying MDP to induce policies and ECR is used to evaluate the induced policies.

### 3.4.2 Five Correlation Metrics

Our feature selection methods involve five correlation metrics. The first four are commonly used in supervised learning and here we will investigate whether they can be effectively applied for feature selection in RL. We propose the fifth metric, called Weighted Information Gain (WIG), by combining the first four metrics and adapting them based on the characteristics of our data sets. More specifically, we have:

1. Chi-squared (CHI) [Zib07]: a statistical test used to identify the independence between the

two variables: whether the distribution of a categorical variable differs significantly from another categorical variable.

2. Information gain (IG) [LL06]: measures how much information we would gain about a variable  $Y$  if knowing another variable  $X$ . It is calculated as:

$$IG(Y, X) = H(Y) - H(Y|X) \quad (3.7)$$

where  $H(\cdot)$  is the entropy function – measuring the uncertainty of a variable.  $IG(Y, X)$  evaluates how the uncertainty of a variable  $Y$  would change from knowing the variable  $X$ . To some extent, it can also be treated as a type of correlation between  $X$  and  $Y$ . Note that IG has the bias towards the variable with a large number of distinct values.

3. Symmetrical uncertainty (SU)[YL03]: it is defined as:

$$SU(Y, X) = \frac{H(Y) - H(Y|X)}{H(X) + H(Y)} \quad (3.8)$$

SU evaluates the correlation between two variables  $Y$  and  $X$  by normalizing  $IG(Y, X)$ . SU compensates for the weakness of IG by considering the uncertainty of both variables  $X$  and  $Y$  in the denominator.

4. Information gain ratio (IGR) [Ken83]: is the ratio of information gain to the intrinsic information, which is the entropy of conditional information. IGR can be represented as:

$$IGR(Y, X) = \frac{H(Y) - H(Y|X)}{H(X)} \quad (3.9)$$

Compared with SU, IGR only considers the uncertainty of variable  $X$  in the denominator.

5. Weighted Information gain (WIG) is proposed as:

$$WIG(Y, X) = \frac{H(Y) - H(Y|X)}{(H(Y) + H(X)) \cdot H(X)} \quad (3.10)$$

WIG can be seen as a combination of IG, SU and IGR. Compared to SU, WIG sets more weight on  $X$  by multiplying  $H(X)$  in the denominator while compared to IGR, WIG normalizes IG by considering the uncertainty of both variables  $X$  and  $Y$ .

In our application, each of the five correlation metrics is used for evaluating the correlation between the current selected feature set  $Y$  with a new feature  $X$ . For each metric we explore two options: The High option is to select the next feature that is **most correlated** to the currently selected feature set whereas the Low option is to select the **least correlated** feature. As described above,

the high correlation-based option is commonly used for supervised learning where the features that are most highly correlated with the output labels are often selected [YP97; LL06; CS14; Kop15]. However, for RL, the high option-selected features tend to be homogeneous. Different from the supervised learning tasks, we hypothesize that it is more important to have heterogeneous features in RL that can grasp different aspects of learning environments. Therefore, we also explore the low correlation-based option for feature selection with a goal to increase the diversity of the selected feature set. As a result, we have 10 correlation-based methods named: CHI-high, IG-high, SU-high, IGR-high, WIG-high, CHI-low, IG-low, SU-low, IGR-low, and WIG-low. Our goal is to investigate which option is better: high vs. low, and which of the five correlation metric performs the best.

### 3.4.3 Correlation-based Feature Selection Approaches

---

#### Algorithm 1 Correlation-based Feature Selection Algorithm

---

**Require:**  $\Omega$ : Feature space;  $\mathcal{D}$ : Training data;  $\mathcal{N}$ : Maximum number of selected features

**Ensure:**  $\mathcal{S}^*$ : Optimal feature set

```

1: for  $f_i$  in  $\Omega$  do
2:    $ECR_i \leftarrow \text{CALCULATE-ECR}(\mathcal{D}, f_i)$ 
3: end for
4: Add  $f^*$  with highest  $ECR$  to  $\mathcal{S}^*$ 
5: while  $\text{SIZE}(\mathcal{S}^*) < \mathcal{N}$  do
6:   for  $f_i$  in  $\Omega - \mathcal{S}^*$  do
7:      $C_i \leftarrow \text{CALCULATE-CORRELATION}(\mathcal{S}^*, f_i, m)$ 
8:   end for
9:    $\mathcal{F} \leftarrow \text{SELECTTOP}(C, 5, \text{reverse})$  ▷ Select top 5 features based on correlation metrics
10:  for  $f_i$  in  $\mathcal{F}$  do
11:     $ECR_i \leftarrow \text{CALCULATE-ECR}(\mathcal{D}, \mathcal{S}^* + f_i)$ 
12:  end for
13:  Replace  $\mathcal{S}^*$  by  $\mathcal{S}^* + f_i$  with highest  $ECR$ 
14: end while

```

---

Algorithm 1 shows the process of our correlation-based feature selection method. It contains three major parts. In the first part (lines 1–4), the algorithm constructs MDPs for every single feature in  $\Omega$ , induces a single-feature policy and calculates its  $ECR$  (defined in Sect. 3.4). Then the feature with highest  $ECR$  is added to the current optimal feature set  $\mathcal{S}^*$ . In the second part (lines 6–9), the algorithm follows a forward step-wise feature selection procedure in that, given the currently selected feature set  $\mathcal{S}^*$ , it selects the next feature based on the five correlation metrics described above. More specifically, it first calculates the correlations between  $\mathcal{S}^*$  with each feature  $f_i \in \Omega - \mathcal{S}^*$  using a specific correlation metric  $m$ , ranks the results, and then selects the top 5 features with the

highest correlations for high-option or the bottom 5 lowest features for low options, decided by the Boolean variable *reverse* in line 9. These features are selected to form a feature pool  $\mathcal{F}$ . In the third part (lines 10–13), the current  $\mathcal{S}^*$  is combined with each feature  $f_i \in \mathcal{F}$  to induce a policy and *Calculate-ECR* function calculates the *ECR* of the induced policy. Then  $\mathcal{S}^* + f_k$ , the combination that produces the policy with the highest *ECR*, will be the new  $\mathcal{S}^*$  for the next round. The algorithm will terminate when the size of an optimal feature set reaches maximum number  $\mathcal{N}$ .

### 3.4.4 PreRL-FS Approach

Chi et al. [Chi11] developed a series of feature selection approaches, referred as *PreRL-FS* in the following. They can be grouped into three categories: 1) four ECR-based methods, which use ECR, Upper-Bound, Lower-Bound of ECR, or Hedge value of the single-feature policy as the feature selection criteria. In particular, the Upper-Bound and Lower-Bound of ECR refer to the 95% confidence interval for ECR, and Hedge is defined as  $Hedge = ECR / (Upper\ Bound - Lower\ Bound)$ ; 2) two PCA-based methods, which select features that are highly correlated with principal components; and 3) four ECR & PCA-based methods, the combination of the former two approaches. The results indicated that the *four ECR-based* methods outperformed the other two types of approaches in terms of ECR.

---

#### Algorithm 2 Ensemble Feature Selection Algorithm

---

**Require:**

$\Omega$ : Feature space;  $\mathcal{D}$ : Training data;  $\mathcal{N}$ : Maximum number of selected features;  
 $\mathcal{M}$ : A set of feature selection approaches.

**Ensure:**  $\mathcal{S}^*$ : Optimal feature set

```

1: for  $f_i$  in  $\Omega$  do
2:    $ECR_i \leftarrow \text{CALCULATE-ECR}(\mathcal{D}, f_i)$ 
3: end for
4: Add  $f^*$  with highest ECR to  $\mathcal{S}^*$ 
5: while  $\text{SIZE}(\mathcal{S}^*) < \mathcal{N}$  do
6:    $\mathcal{F} \leftarrow \emptyset$ 
7:   for  $Method_k$  in  $\mathcal{M}$  do
8:      $\mathcal{F}_k \leftarrow \text{SELECT-FEATURE}(\mathcal{D}, \Omega - \mathcal{S}^*, Method_k)$ 
9:      $\mathcal{F} \leftarrow \mathcal{F} \cup \mathcal{F}_k$ 
10:  end for
11:  for  $f_i$  in  $\mathcal{F}$  do
12:     $ECR_i \leftarrow \text{CALCULATE-ECR}(\mathcal{D}, \mathcal{S}^* + f_i)$ 
13:  end for
14:  Replace  $\mathcal{S}^*$  by  $\mathcal{S}^* + f_i$  with highest ECR
15: end while

```

---

### 3.4.5 Ensemble Approach

Algorithm 2 shows the basic process of our ensemble feature selection procedure, which is similar to that of correlation-based methods. The major difference is in the second part (lines 6–10). Our ensemble approach explored a total of 12 feature selection methods are referred to as  $\mathcal{M}$  in Algorithm 2: the four *ECR-based* methods which are the better methods among the *PreRL-FS* approaches and the eight out of the 10 proposed correlation-based methods (WIG-high and WIG-low were excluded here because they were not explored when we first explored the ensemble approach). More specifically, the ensemble approach integrates the features  $\mathcal{F}_k$  generated from each of feature selection method  $Method_k$  in  $\mathcal{M}$  and generates a relatively large feature pool  $\mathcal{F}$ . The maximum size of  $\mathcal{F}$  can be up to 70, but often much smaller because of the overlapping of feature sets generated from different methods. Note that the feature pool is still much larger than any of our 10 correlation-based methods, which is 5. After generating the feature pool, the ensemble method carries out the same procedure, the third part (lines 11–13), as the correlation-based methods described above. Although the ensemble method has a relatively high computational complexity, it has a wider exploration of the feature space by integrating different types of feature selection methods.

### 3.4.6 Comparison Results for Feature Selection Approaches

We explore three categories of feature selection approaches: *PreRL-FS*, ensemble, and high- and low- correlation-based approaches and compare them against a random feature selection baseline. We use ECR (Section 3.4) to theoretically evaluate the effectiveness of the MDP policies, which indirectly verify the effectiveness of feature selection approaches. Note that ECR is calculated based on the induced MDP policies and the two training datasets: DT-Immed and DT-Delay (Section 3.3.2).

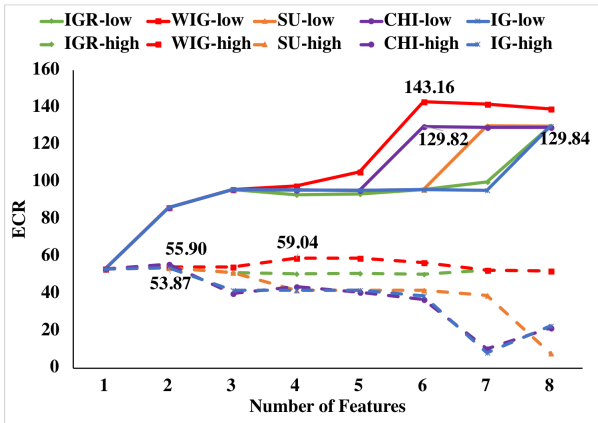


Figure 3.3 DT-Immed

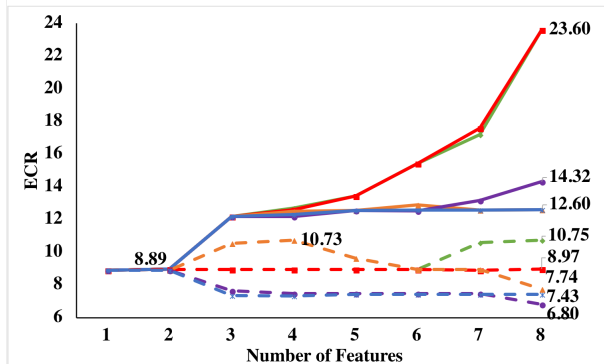


Figure 3.4 DT-Delay

**High VS Low Correlation-based Approaches.** Figure 3.3 and Figure 3.4 show the ECR values of 10 correlation-based methods on DT-Immed and DT-Delay respectively, where the y-axis represents the value of the ECR of the induced policy given the selected features, and the x-axis denotes the number of features (maximum is 8). Note that all feature selection methods start in the same place at  $x = 1$  except the random method. This is because all methods will initially select the feature with the best ECR of single-feature policy. However, ECR values vary dramatically as the number of selected features increases. The solid line indicates the performance of the low correlation-based approaches and the dotted line denotes the performance of the high correlation-based version. In addition, the ECR value of policies using immediate reward is much higher than that of policies using the delayed reward.

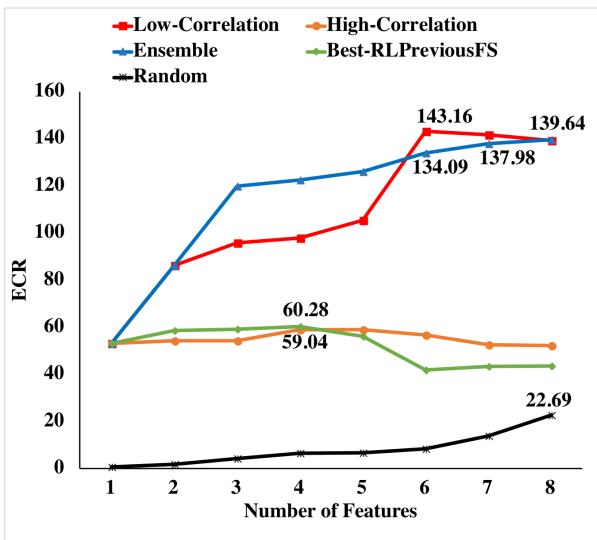


Figure 3.5 DT-Immed

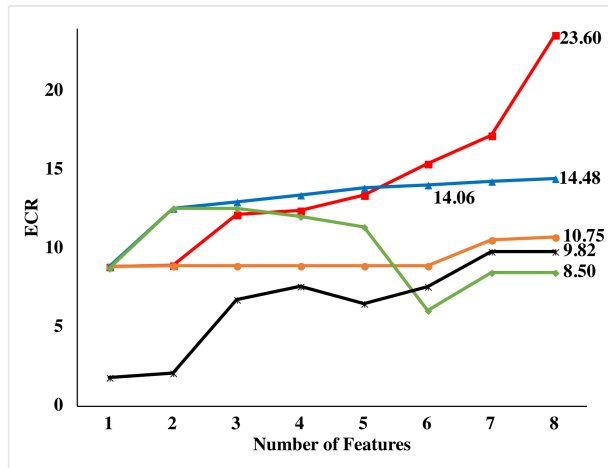


Figure 3.6 DT-Delay

The results show that for each of the five correlation metrics, the low correlation-based option significantly outperforms the high correlation-based option. For the *DT-Immed* dataset, the *ECR* of WIG-low is 143.16, while *ECR* of WIG-High is only 59.04. Similarly, the *ECR*s of CHI-low and CHI-High are 129.82 vs. 55.90. The average percent increase for the low correlation methods over the high correlation methods is 75.35%, the maximum percent increase is 142.48%, and the minimum percent increase is 17.24%. To summarize, our results show that low correlation is more suitable for the MDP framework than high correlation, and indirectly illustrate that the high variety of the feature space had a positive impact on the effectiveness of the induced policies. The same pattern was found in the *DT-Delay* dataset.

**Five Correlation Metrics.** Figures 3.3 and 3.4 show that WIG is the consistently highest performer in that it has the best ECR for both DT-Immed and DT-Delay datasets. CHI performed well in DT-Immed dataset while IGR performed well in DT-Delay dataset. In short, our proposed WIG performed best among all the five correlation metrics.

**Overall Comparison.** Figures 3.5 and 3.6 show the overall comparison among all methods on DT-Immed and DT-Delay data respectively. Particularly, with the purpose of simplicity, for both low and high correlation-based methods and the PreRL-FS methods, we selected the best method from each category. In other words, the figures present a comparison among the five methods including the best of five Low-correlations, the best of five High-correlations, ensemble, the best of PreRL-FS, and the random approach. Results show that, as expected, the random method performs worst across the two datasets. In addition, the best of the high correlation-based methods outperforms random and Best-RLPreviousFS approaches when the number of features is above 5. The best of the low correlation-based methods outperforms other methods. In general, the best low correlation-based method outperforms the best of *PreRL-FS* by an average of 43.87% and outperforms the ensemble method by an average of 9.05%. In addition, the ensemble method improves over the best of *PreRL-FS* by an average of 36.46%. The value of ECR does not always rise as the the number of features increases. The ECR of the low-correlation approach decreases a lot when increasing the number of features from 6 to 8. The ECR of the ensemble method seems to converge when the number of features is more than 6 for both two training datasets. The ECR of the best of *PreRL-FS* decreases when the number of features is more than 4.

In summary, based on ECR results we can rank five categories of methods as Low correlation-based > Ensemble > High correlation-based  $\approx$  PreRL-FS  $\gg$  Random. In particular, the WIG-Low approach performs best among all implemented approaches.

## 3.5 Experiments Overview

### 3.5.1 Research Questions

In this work, we investigate the effectiveness of RL-induced policies using the MDP framework from three aspects: state representation using different feature selections, reward function, and policy execution options. For each aspect, we have a corresponding research question and thus our three research questions are listed as follows:

- **Q1 (State):** Can effective feature selection methods empirically improve the effectiveness of the induced policy?
- **Q2 (Reward):** Does immediate reward facilitate the MDP framework to induce a more effective pedagogical policy than delayed reward?

- **Q3 (Execution):** Can stochastic policy execution be more effective than deterministic policy execution?

### 3.5.2 Reinforcement Learning Policies

Table 3.1 lists the five RL policies induced for investigating the three research questions above. All five policies were induced using the MDP framework but involved different types of feature selection methods (the second column), reward function (the third column), and/or policy execution (the fourth column). The last column shows that the ECR of the RL-induced policies. More specifically, *MDP-ECR* is induced by using MDP with the best PreRL-FS feature selection approach; *Ensemble-Imme* and *Ensemble-Delay* are two policies induced with the ensemble feature selection approach using immediate and delayed reward respectively; and *WIG-det* and *WIG-sto* were both induced using WIG with the low-correlation option for feature selection, and the main difference is that the former is executed deterministically while the latter is executed stochastic. Note that because *WIG-sto* is a stochastic policy and because ECR can only be calculated for a deterministic policy, the ECR of *WIG-sto* is listed as “NA”.

**Table 3.1** Reinforcement Learning Policies in Four Experiments

| Policy                | Feature Selection | Reward    | Execution     | ECR    |
|-----------------------|-------------------|-----------|---------------|--------|
| <i>MDP-ECR</i>        | ECR-based         | Immediate | Deterministic | 60.28  |
| <i>Ensemble-Imme</i>  | Ensemble          | Immediate | Deterministic | 137.98 |
| <i>Ensemble-Delay</i> | Ensemble          | Delay     | Deterministic | 14.06  |
| <i>WIG-det</i>        | Low Corre-based   | Immediate | Deterministic | 143.16 |
| <i>WIG-sto</i>        | Low Corre-based   | Immediate | Stochastic    | NA     |

Note: ECR is only used for evaluating the deterministic policies

### 3.5.3 Experiments Overview

Four experiments, one per semester from the Spring of 2015 to the Fall of 2017, were conducted to empirically evaluate the impact of the three aspects on the effectiveness of the five RL-induced policies described above. In each experiment, we compared one or two RL policies against the Random yet reasonable baseline policy. Table 3.2 shows the overview of the four experiments and the corresponding research questions.

**Table 3.2** Overview of Experiments

| Experiment   | Policies   | Research Question |            |               |
|--------------|--|-------------------|------------|---------------|
|              |  | Q1(State)         | Q2(Reward) | Q3(Execution) |
| Experiment 1 | <i>MDP-ECR vs. Random</i>                          | ✓                 |            |               |
| Experiment 2 | <i>Ensemble-Imme vs. Ensemble-Delay vs. Random</i> | ✓                 | ✓          |               |
| Experiment 3 | <i>WIG-det vs. Random</i>                          | ✓                 |            |               |
| Experiment 4 | <i>WIG-sto vs. Random</i>                          | ✓                 |            | ✓             |

### 3.5.4 ATI effect: Splitting Students Based on Response Time

Overall, results across the four experiments consistently exhibit an ATI effect. That is, rather than all students, only certain students' learning is significantly affected by the pedagogical decisions on PS vs. WE. In the following, they are referred to as the *Responsive* group and by contrast, we refer to other students as the *Unresponsive* group. It is often not clear which group of the students are more sensitive to the induced policy due in part to the fact that we do not fully understand why such differences exist. In this work, we split *Responsive* and *Unresponsive* groups based upon some measurement of incoming competence.

One common way to measure students' incoming competence is to use their pre-test scores. Across the four experiments, all of the students received the same initial training at Level 1 and our results showed that students' pre-test scores indeed reflect their incoming competence in that a significant positive correlation between students' pre-test scores and transfer post-test scores:  $r = 0.17, n = 241, p = .005$ . However, our initial analysis also showed that student pre-test scores are also skewed in that we can't generate the balanced groups by splitting students based on their pre-test scores. For example, in Experiment 1, a medium split on pretest scores would divide the Random group into 16 High pretest group vs. 6 in the Low pretest group. Similarly, in Experiment 3, the WIG-det group is divided into: WIG-det-High (N=31), WIG-det-Low (N=14). The WIG-det-High has twice as many students as WIG-det-Low.

On the other hand, ever since the mid-1950s, response time has been used as a preferred dependent variable in cognitive psychology [Tho86]. It has primarily been used to assess student learning because response time can indicate how active and accessible student knowledge is. For example, it is shown that response time reveals student proficiency [SS02] and there was a significant negative correlation between the students' average response time and their final exam scores taken at the end of the semester [GEB]. With the advent of computerized testing, more and more researchers have begun to use response-time as a learning performance measurement [SS02]. Inspired by these prior work, we explored to use the average time in Level 1 (*avgTime*) to split students which indeed

consistently generated more balanced groups across all four experiments. Therefore, in the following studies, students were split using *avgTime*, and we will mainly focus on comparing students' transfer post-test scores by using pre-test scores as the covariate to measure students learning improvement on the tutor.

To summarize, in each of the following experiments, students are divided into *Responsive* and *Unresponsive* groups by a median split on their response time at Level 1. Since each experiment has a slightly different median value and criteria for splitting, there is no general definition for the *Responsive* and *Unresponsive* groups. In the post-hoc comparison, we combined all of the experiments and used a global median split to check whether our results would still hold.

### 3.5.5 Statistic Analysis

In the following analyses, we run several different types of statistical tests to evaluate student performance with a focus on their transfer post-test scores. Although students' pre-test scores were not used to split students into *Responsive* and *Unresponsive* groups, they are used as the covariate in ANCOVA when comparing students' transfer post-test scores.

To confirm that the assumptions of ANCOVA were met, for each experiment ANOVA tests were performed and indicated that there is no significant difference on pre-test score among different treatment groups. In addition, two-way ANOVA tests for each experiment using group and pre-test as factors show that there is no significant interaction effect on transfer post-test score. These results indicate that the pre-test covariate and treatment group variable are independent and that the relationship between the covariate and treatment group variable is the same across groups. Thus, the assumptions of ANCOVA are met, and we report ANCOVA results for the transfer post-test scores.

## 3.6 Four Experiments

### 3.6.1 Experiment 1: Preliminary Feature Selection

Experiment 1 was conducted in the Spring of 2015. We compared two policies: an MDP policy and a Random baseline policy. Our research question in Experiment 1 is Q1 (State): Can effective feature selection methods empirically improve the effectiveness of the induced policy?

For Experiment 1, we only explored the PreRL-FS feature selection approaches, and among them, the ECR-based approach using the lower bound of ECR as the selection criteria performed the best. In the following, we refer to the induced policy as the *MDP-ECR* policy. Table 3.3 shows the definition of the four selected features (left) and the corresponding policy (right). The row denotes the value of the first two features while the column denotes the value of the last two features. For example, when the four features  $f_1:f_2:f_3:f_4$  is 0:0:0:0 (the top-left cell), the decision is a PS (black

cell). Overall, the *MDP-ECR* policy contains 11 pedagogical rules that propose a PS (black cells) and 5 rules that propose a WE (white cells).

**Table 3.3** *MDP-ECR* Policy

|  |   |     |     |     |     |     |     |  |  |  |  |     |  |  |  |  |     |  |  |  |  |     |  |  |  |  |
|--|---|-----|-----|-----|-----|-----|-----|--|--|--|--|-----|--|--|--|--|-----|--|--|--|--|-----|--|--|--|--|
| <b>NextClickWE</b> ( $f_1$ ): Number of next step clicks in a Worked Example         | <div style="display: flex; flex-direction: column; align-items: center;"> <div style="margin-bottom: 5px;">Last two features</div> <div style="margin-bottom: 5px;"><math>f_{13}:f_{14}</math></div> <table border="1" style="border-collapse: collapse; text-align: center;"> <tr> <td></td> <td>0:0</td> <td>0:1</td> <td>1:0</td> <td>1:1</td> </tr> <tr> <td>0:0</td> <td style="background-color: black;"></td> <td style="background-color: black;"></td> <td style="background-color: white;"></td> <td style="background-color: white;"></td> </tr> <tr> <td>0:1</td> <td style="background-color: black;"></td> <td style="background-color: black;"></td> <td style="background-color: white;"></td> <td style="background-color: white;"></td> </tr> <tr> <td>1:0</td> <td style="background-color: black;"></td> <td style="background-color: black;"></td> <td style="background-color: black;"></td> <td style="background-color: black;"></td> </tr> <tr> <td>1:1</td> <td style="background-color: black;"></td> <td style="background-color: white;"></td> <td style="background-color: black;"></td> <td style="background-color: black;"></td> </tr> </table> </div> |     | 0:0 | 0:1 | 1:0 | 1:1 | 0:0 |  |  |  |  | 0:1 |  |  |  |  | 1:0 |  |  |  |  | 1:1 |  |  |  |  |
|  |   | 0:0 | 0:1 | 1:0 | 1:1 |     |     |  |  |  |  |     |  |  |  |  |     |  |  |  |  |     |  |  |  |  |
| 0:0  |   |     |     |     |     |     |     |  |  |  |  |     |  |  |  |  |     |  |  |  |  |     |  |  |  |  |
| 0:1  |   |     |     |     |     |     |     |  |  |  |  |     |  |  |  |  |     |  |  |  |  |     |  |  |  |  |
| 1:0  |   |     |     |     |     |     |     |  |  |  |  |     |  |  |  |  |     |  |  |  |  |     |  |  |  |  |
| 1:1  |   |     |     |     |     |     |     |  |  |  |  |     |  |  |  |  |     |  |  |  |  |     |  |  |  |  |
| <b>TotalWETime</b> ( $f_2$ ): Total time for solving a Worked Example                |   |     |     |     |     |     |     |  |  |  |  |     |  |  |  |  |     |  |  |  |  |     |  |  |  |  |
| <b>symbolicRepnCount</b> ( $f_3$ ): Number of problems using symbolic representation |   |     |     |     |     |     |     |  |  |  |  |     |  |  |  |  |     |  |  |  |  |     |  |  |  |  |
| <b>difficultProbCount</b> ( $f_4$ ): Number of solved difficult problems             |   |     |     |     |     |     |     |  |  |  |  |     |  |  |  |  |     |  |  |  |  |     |  |  |  |  |
| Note: Black: PS, White: WE   |   |     |     |     |     |     |     |  |  |  |  |     |  |  |  |  |     |  |  |  |  |     |  |  |  |  |

### 3.6.1.1 Experiment 1: Participants & Conditions

DT was assigned to 67 undergraduate students as one of their regular homework assignments. Completion of the tutor was required for full credit. Students were randomly assigned to the two conditions: Random ( $N = 22$ ) and *MDP-ECR* ( $N = 45$ ). Because all of students followed the random policy when collecting our training data for RL in previous years, we assign more students to the *MDP-ECR* condition to evaluate the effectiveness of RL-induced policies.

Results of Experiment 1 show that there is no significant difference between the *MDP-ECR* and Random on either pre-test ( $F(1, 65) = 1.81, p = 0.18$ ) or transfer post-test ( $F(1, 65) = 0.46, p = 0.50$ ). However, once we did a median split on students based on the students' "average response time on level 1", our results show that students whose *level1-avgstepTime*  $< 7.1$  sec, are more sensitive to the effectiveness of pedagogical strategies than their peers whose *level1-avgstepTime*  $\geq 7.1$  sec. In the following, we refer the former as the Responsive group and the latter as the Unresponsive group. By combining Policy {*MDP-ECR*, Random} with Type {Responsive, Unresponsive}, we have a total of four groups including: Random-Resp ( $N = 9$ ), Random-Unres ( $N = 13$ ), *MDP-ECR-Resp* ( $N = 23$ ) and *MDP-ECR-Unres* ( $N = 22$ ). Pearson's Chi-squared test showed that there was no significant difference on the distribution of Unresponsive vs. Responsive between two policies ( $\chi^2(1, N = 67) = 0.27, p = 0.59$ ).

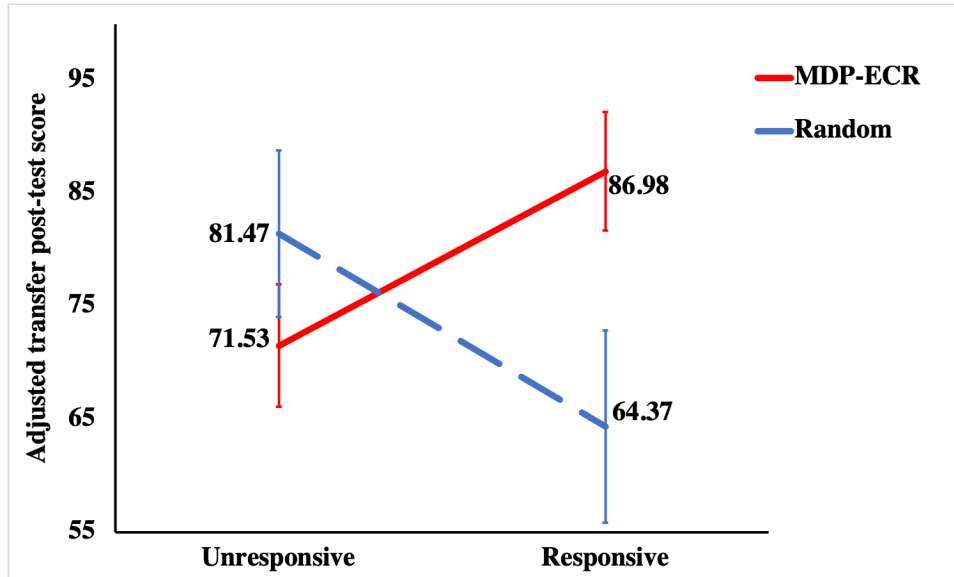
### 3.6.1.2 Experiment 1: Results

Table 3.4 presents the mean and SD for students' corresponding learning performance in Experiment 1. Despite the fact that students are randomly assigned, *Random-Unres* significantly out-

**Table 3.4** Pre-test and Transfer Post-test in Experiment 1

| Type  | Pre-Test Score |                     | Transfer Post-Test Score |              |
|-------|----------------|---------------------|--------------------------|--------------|
|       | MDP-ECR        | Random              | MDP-ECR                  | Random       |
| Resp  | 58.06(29.92)   | 48.59(35.64)        | <b>87.50(16.38)</b>      | 69.88(34.43) |
| Unres | 53.87(33.17)   | <b>86.15(20.42)</b> | 69.94(28.54)             | 79.54(23.73) |
| Total | 56.11(31.19)   | 67.37(34.24)        | 79.31(24.27)             | 74.71(29.28) |

performs all other groups on the pre-test according to results of ANOVA tests:  $F(1, 20) = 9.01, p = .007$  for *Random-Resp*,  $F(1, 33) = 8.82, p = .006$  for *MDP-ECR-Unres*,  $F(1, 34) = 6.37, p = .016$  for *MDP-ECR-Resp*, probably due to the small sample size in the random groups. Despite *Random-Unres* out-performance, no significant difference is found on the pre-test score either between *MDP-ECR* and *Random* (two columns):  $F(1, 65) = 1.81, p = 0.18$ , or between Responsive and Unresponsive (two rows):  $F(1, 65) = 1.26, p = 0.27$ . Furthermore, a possible explanation for a high pre-test score of *Random-Unres* is that *Random-Unres*, considered as the *high proficiency* students, can always learn regardless of teaching policies and are less sensitive to the learning environment [CS77b; Chi11].



**Figure 3.7** Interaction effect for the adjusted transfer post-test score in Experiment 1

**Transfer Post-Test Score.** A two-way ANCOVA test, using Policy and Type as the two factors and

the pre-test score as covariate shows that there is a significant interaction effect on their transfer post-test scores:  $F(1, 62) = 5.39, p = .023$ , but no significant main effect of either Policy or Type. Figure 3.7 depicts the cross-over interaction between Policy and Type on the *adjusted transfer post-test score*, which is the transfer post-test score adjusted by the linear regression (ANCOVA) model built to describe the relation between the pre- and transfer post-test score.

Planned contrasts using Tukey’s adjustment reveal a significant difference between the two Responsive groups in that *MDP-ECR-Resp* scored significantly higher adjusted transfer post-test than *Random-Resp*,  $t(62) = 2.26, p = .027$ , while there is no significant difference between two Unresponsive groups.

### 3.6.1.3 Experiment 1: Conclusions & Limitations

In summary, we find a significant interaction effect in that MDP-ECR benefits the Responsive students significantly more than the Unresponsive students, while no such difference was found between the Responsive and Unresponsive groups under the Random policy. However, one important limitation of Experiment 1 is that the *Random-Unres* group has significant higher pre-test score than all other groups. Therefore, in Experiment 2, we repeat the general procedure of Experiment 1 but explored correlation based and ensemble-based feature selection methods and also explore both Immediate and Delayed rewards.

### 3.6.2 Experiment 2: Ensemble Feature Selection & Immediate vs. Delayed Rewards

Experiment 2 was conducted in the Fall of 2016 and it investigated on two research questions:

- **Q1 (State):** Can effective feature selection methods indeed empirically improve the effectiveness of the induced policy?
- **Q2 (Reward):** Does the immediate reward facilitate the MDP framework to induce a more effective pedagogical policy than the delayed reward?

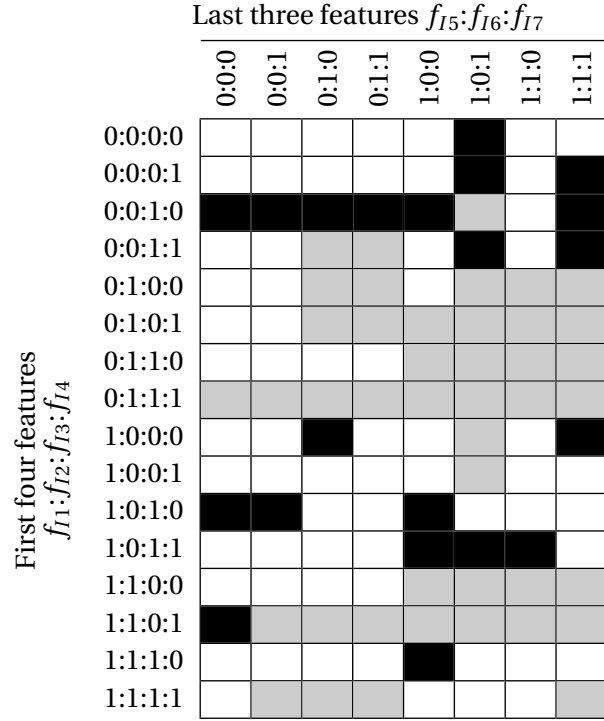
In Experiment 2, we applied the ensemble feature selection (Sec. 3.4.5) to select a small subset of features from the original 133 features for inducing two policies, named *Ensemble-Imme* and *Ensemble-Delay*, from the two training datasets *DT-Imme* and *DT-Delay* respectively (Sec. 3.3.2). More specifically, *Ensemble-Imme* involves seven features and *Ensemble-Delay* policy involves six features. Table 3.5 and 3.6 display the selected features as well as the corresponding policies. In the tables, black cells denote **PS** actions, white cells denote **WE** actions, and gray cells denote that no policy is induced due to the absence of the state in the training data. Generally speaking, the *Ensemble-Imme* policy prefers WE over PS as it contains 65 rules for WE vs. 21 rules for PS; while *Ensemble-Delay* policy prefers PS over WE as it has 48 rules for PS and 18 for WE. Additionally, while Figure 3.5 shows that the ensemble feature selection with eight features would result in a higher

**Table 3.5** Ensemble-Imme Policy

---

|  |
|--|
| <b>TotalPSTime</b> ( $f_1$ ): Total time for solving a problem                       |
| <b>NewLevel</b> ( $f_2$ ): Whether the current solved problem is in a new level      |
| <b>WrongApp</b> ( $f_3$ ): Number of incorrect application of rules                  |
| <b>TotalWETime</b> ( $f_4$ ): Total time for working on a worked example             |
| <b>UseCount</b> ( $f_5$ ): Number of different types of applied rules                |
| <b>AppCount</b> ( $f_6$ ): Number of clicks for derivation                           |
| <b>NumProbRule</b> ( $f_7$ ): Number of expected distinct rules for a solved problem |

---



ECR policy than the policy with seven features, we still used the latter here because 1) the ECRs of the two policies are actually very close; and 2) the seven-feature policy is less complicated and had less "none-mapping" from state to action (the gray color cells) compared with the eight-feature policy. For similar reasons, we determined the *Ensemble-Delay* policy to be six features.

### 3.6.2.1 Experiment 2: Participants and Conditions

A total of 106 students participated in Experiment 2 and were randomly assigned into three conditions: *Random* ( $N = 30$ ), *Ensemble-Imme* ( $N = 38$ ) and *Ensemble-Delay* ( $N = 38$ ). 94 students completed the assignment, distributed as *Random* ( $N = 27$ ), *Ensemble-Imme* ( $N = 34$ ) and *Ensemble-Delay* ( $N = 33$ ). Pearson's Chi-squared test yields no significant correlation between completion rate and condition ( $\chi^2(2, N = 106) = .012, p = .994$ ).

The last row in Table 3.7 presents the mean and SD for students' corresponding learning performance in Experiment 2. No significant difference was found among the three policies on either pre-test ( $F(2, 91) = 0.04, p = 0.96$ ) or transfer post-test ( $F(2, 91) = 1.33, p = 0.27$ ). Furthermore, similar as Experiment 1, we use the median of "average response time on level 1" (median(*level1-avgstepTime*) = 8.01 sec) to split students in Experiment 2. Different from Experiment 1, it was shown that students whose *level1-avgstepTime* < 8.01 sec are less sensitive to the effectiveness of

**Table 3.6** Ensemble-Delay Policy

|   | Last three features $f_{D4}:f_{D5}:f_{D6}$ |       |       |       |       |       |       |       |
|---|--|-------|-------|-------|-------|-------|-------|-------|
|   | 0:0:0                                      | 0:0:1 | 0:1:0 | 0:1:1 | 1:0:0 | 1:0:1 | 1:1:0 | 1:1:1 |
| <b>stepTimeDev</b> ( $f_1$ ): Step time deviation                                     | 0:0:0                                      | 0:0:1 | 0:1:0 | 0:1:1 | 1:0:0 | 1:0:1 | 1:1:0 | 1:1:1 |
| <b>probDiff</b> ( $f_2$ ): Difficulty of the current solved problem                   | 0:0:0                                      | 0:0:1 | 0:1:0 | 0:1:1 | 1:0:0 | 1:0:1 | 1:1:0 | 1:1:1 |
| <b>symbolicRCount</b> ( $f_3$ ): Number of whole problems for symbolic representation | 0:0:0                                      | 0:0:1 | 0:1:0 | 0:1:1 | 1:0:0 | 1:0:1 | 1:1:0 | 1:1:1 |
| <b>actionCount</b> ( $f_4$ ): Number of non-empty-click actions taken by students     | 0:0:0                                      | 0:0:1 | 0:1:0 | 0:1:1 | 1:0:0 | 1:0:1 | 1:1:0 | 1:1:1 |
| <b>SInfoHintCount</b> ( $f_5$ ): Number of System Information Hint requests           | 0:0:0                                      | 0:0:1 | 0:1:0 | 0:1:1 | 1:0:0 | 1:0:1 | 1:1:0 | 1:1:1 |
| <b>NSClickCountWE</b> ( $f_6$ ): Number of next step clicks in Worked Examples        | 0:0:0                                      | 0:0:1 | 0:1:0 | 0:1:1 | 1:0:0 | 1:0:1 | 1:1:0 | 1:1:1 |

pedagogical strategies than those whose  $level1-avgstepTime \geq 8.01$  sec, and thus we refer the former as the Unresponsive group and the latter as the Responsive group.

By combining Policy with Type {Responsive, Unresponsive}, we have a total of six groups including three Unresponsive groups: *Random-Unres* ( $N = 15$ ), *Ensemble-Imme-Unres* ( $N = 16$ ), *Ensemble-Delay-Unres* ( $N = 15$ ); and three Responsive groups: *Random-Resp* ( $N = 12$ ), *Ensemble-Imme-Resp* ( $N = 18$ ), and *Ensemble-Delay-Resp* ( $N = 18$ ). Pearson’s chi-squared test shows that there is no significant difference in the distribution of Unresponsive vs. Responsive among the three conditions,  $\chi^2(1, N = 94) = .681, p = .711$ .

### 3.6.2.2 Experiment 2: Results

Table 3.7 presents the mean and SD for students’ corresponding learning performance. One-way ANOVA tests show that there is no significant difference on the pre-test score either among the three policies {*Ensemble-Imme*, *Ensemble-Delay*, *Random*},  $F(2, 91) = 0.04, p = 0.96$ , or among the three Unresponsive groups,  $F(2, 43) = 0.14, p = 0.87$ , or among the three Responsive groups,  $F(2, 45) = 0.65, p = 0.53$ . Additionally, there is a significant difference between Responsive and Unresponsive: the former scores significantly higher than the latter on the pre-test score,  $F(1, 92) = 7.33, p = .008$ .

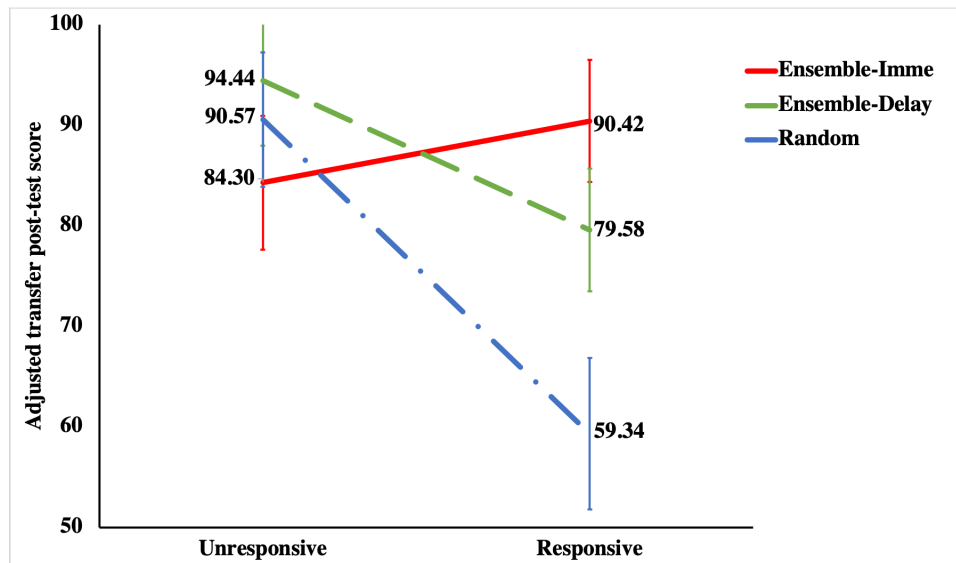
**Transfer Post-Test Score.** A two-way ANCOVA test, using Policy {*Ensemble-Imme*, *Ensemble-Delay*, *Random*} and Type {*Responsive*, *Unresponsive*} as two factors and the pre-test score as covariate, shows that there is a significant main effect of Type,  $F(1, 87) = 4.45, p = .037$ , and a significant interaction effect on transfer post-test scores,  $F(2, 87) = 3.90, p = .024$ . Figure 3.8 presents

**Table 3.7** Pre-test and Transfer Post-test in Experiment 2

| Type  | Pre-Test Score |                |              | Transfer Post-Test Score |                     |              |
|-------|----------------|----------------|--------------|--------------------------|---------------------|--------------|
|       | Ensemble-Imme  | Ensemble-Delay | Random       | Ensemble-Imme            | Ensemble-Delay      | Random       |
| Resp  | 62.20(31.84)   | 68.17(30.81)   | 74.76(23.90) | <b>90.97(24.36)</b>      | 81.25(31.43)        | 62.24(40.16) |
| Unres | 54.11(37.27)   | 48.27(27.71)   | 49.56(30.47) | 83.33(22.36)             | <b>92.38(10.64)</b> | 88.75(22.43) |
| Total | 58.52(34.11)   | 58.81(30.65)   | 60.76(30.07) | 87.50(23.43)             | 86.49(24.34)        | 76.96(33.67) |

the cross-over interaction between Policy and Type on the *adjusted transfer post-test score*, which is the transfer post-test score adjusted by the linear regression model built to describe the relation between the pre- and transfer post-test score.

Table 3.8 presents the results of contrast tests using Tukey’s adjustment for multiple comparisons. Results indicate that while there is no significant difference among three Unresponsive groups, *Ensemble-Imme-Resp* achieved significantly higher adjusted transfer post-test score than *Random-Resp*:  $p = 0.01$ .



**Figure 3.8** Interaction effect for the adjusted transfer post-test score in Experiment 2

**Table 3.8** Pairwise Contrasts on Adjusted Transfer Post-test in Experiment 2

| Pairwise Policy Comparison   |                                  | $t(87)$ | $p$ -value |
|------------------------------|----------------------------------|---------|------------|
| Ensemble-Imme- <b>Resp</b>   | vs. Ensemble-Delay- <b>Resp</b>  | -1.26   | 0.75       |
| Ensemble-Imme- <b>Resp</b>   | vs. Random- <b>Resp</b>          | 3.22    | 0.01 *     |
| Ensemble-Delay- <b>Resp</b>  | vs. Random- <b>Resp</b>          | 2.11    | 0.21       |
| Ensemble-Imme- <b>Unres</b>  | vs. Ensemble-Delay- <b>Unres</b> | 1.09    | 0.85       |
| Ensemble-Imme- <b>Unres</b>  | vs. Random- <b>Unres</b>         | -0.67   | 0.98       |
| Ensemble-Delay- <b>Unres</b> | vs. Random- <b>Unres</b>         | 0.42    | 0.99       |

· marginal significant at  $p < 0.1$ ; \* significant at  $p < 0.05$ .

### 3.6.2.3 Experiment 2: Conclusion

Our empirical results suggest that the ATI effect exists in Experiment 2: while no significant difference is found among the three Unresponsive groups, a significant difference is found among the three Responsive groups in that students following the *Ensemble-Imme* policy score significantly higher on the transfer post-test than their peers following the *Random* policy. This suggests that immediate reward can facilitate the MDP framework to induce an effective policy and that the ensemble feature selection approach is able to extract a good subset of features for MDP to induce a more effective policy compared with the Random policy. Finally, since it was shown that the immediate reward is more effective than the delayed reward for policy induction in the MDP framework in Experiment 2, we will only use the immediate reward to induce policy in the following two experiments. <https://www.overleaf.com/project/59fa12c0ebcd6470de670d13>

### 3.6.3 Experiment 3: Low Correlation-based Feature Selection

Experiment 3 was conducted in the Spring of 2017, and the goal was to further investigate the effectiveness of our feature selection methods. So the research question for Experiment 3 is Q1 (State): can effective feature selection methods empirically improve the effectiveness of the induced policy?

Results of feature selection showed that the policy with the highest ECR is induced when WIG-Low is applied and the number of selected features is six (see Figure 3.5 in Section 3.4), so in Experiment 3 we implemented and empirically evaluated the induced *WIG-det* policy. Table 3.9 shows the selected features and *WIG-det* policy, which contains only 9 rules associated with PS but 46 rules for WE.

**Table 3.9** WIG-Low Policy

---

**TotalPSTime** ( $f_1$ ): Total time for solving a problem

**easyProSolved** ( $f_2$ ): Number of easy problems solved

**NewLevel** ( $f_3$ ): Whether current solved problem is in a new level

**avgstepTime** ( $f_4$ ): Average time per step

**hintRatio** ( $f_5$ ): Ratio between hint count and action count

**NumProbRule** ( $f_6$ ): Number of expected rules for the next problem

---

Note: Black: PS, White: WE, Gray: No mapping from state to action

|                                       |       | Last three features $f_{14}:f_{15}:f_{16}$ |       |       |       |       |       |       |       |
|---------------------------------------|-------|--|-------|-------|-------|-------|-------|-------|-------|
|                                       |       | 0:0:0                                      | 0:0:1 | 0:1:0 | 0:1:1 | 1:0:0 | 1:0:1 | 1:1:0 | 1:1:1 |
| First three features<br>$f_1:f_2:f_3$ | 0:0:0 |  |       |       | ■     |       |       |       | ■     |
|                                       | 0:0:1 |  |       | ■     |       |       |       | ■     |       |
|                                       | 0:1:0 |  | ■     |       |       |       |       |       | ■     |
|                                       | 0:1:1 |  | ■     |       | ■     |       | ■     | ■     | ■     |
|                                       | 1:0:0 |  |       |       |       |       |       |       |       |
|                                       | 1:0:1 |  |       | ■     |       |       |       |       |       |
|                                       | 1:1:0 |  | ■     |       |       |       |       |       | ■     |
|                                       | 1:1:1 |  | ■     |       | ■     |       | ■     |       | ■     |

### 3.6.3.1 Experiment 3: Participants and Conditions

A total of 92 students were randomly assigned into two different groups: *Random* ( $N = 45$ ) and *WIG-det* ( $N = 47$ ). In the end, a total of 82 students completed the assignment and were distributed as follows: *Random* ( $N = 38$ ) and *WIG-det* ( $N = 44$ ). Pearson’s chi-squared test revealed no significant relationship between completion rate and condition  $\chi^2(1, N = 92) = .034, p = .852$ .

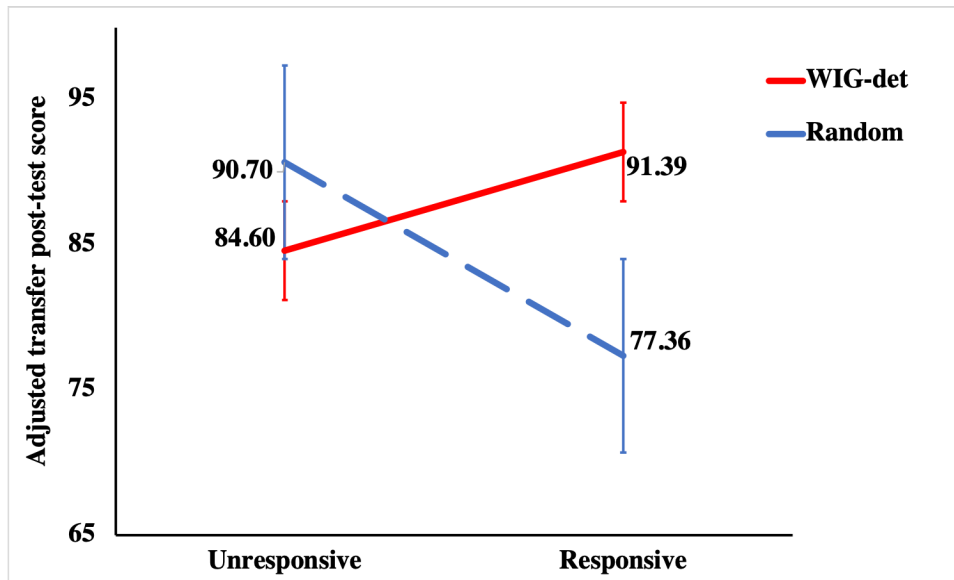
The last row in Table 3.10 shows the mean and SD for either condition’s corresponding learning performance. No significant difference was found between *WIG-det* and *Random* on either pre-test ( $F(1, 80) = 2.02, p = 0.16$ ) or transfer post-test ( $F(1, 80) = 1.74, p = 0.19$ ). Furthermore, as in Experiments 1 and 2, we perform a median split using the “average response time on level 1” (*level1-avgstepTime*) to split students and find that students whose *level1-avgstepTime*  $< 8.34$  sec are less sensitive to the effectiveness of pedagogical strategies while their peers whose *level1-avgstepTime*  $\geq 8.34$  sec are more sensitive to the effectiveness of pedagogical strategies in Experiment 3. In the following section, we refer the former as the Unresponsive group and the latter as the Responsive group. Combining Policy with Type {Responsive, Unresponsive}, we have a total of four groups including two Responsive groups, *Random-Resp* ( $N = 18$ ) and *WIG-det-Resp* ( $N = 22$ ) and two Unresponsive groups, *Random-Unres* ( $N = 20$ ) and *WIG-det-Unres* ( $N = 22$ ). Pearson’s chi-squared test revealed no significant difference in the distribution of Unresponsive vs. Responsive between *Random* and *WIG-det*,  $\chi^2(1, N = 82) = 0, p = .987$ .

**Table 3.10** Pre-test and Transfer Post-test in Experiment 3

| Type  | Pre-Test Score |              | Transfer Post-Test Score |              |
|-------|----------------|--------------|--------------------------|--------------|
|       | WIG-det        | Random       | WIG-det                  | Random       |
| Resp  | 73.13(25.71)   | 63.76(28.02) | <b>91.76(11.47)</b>      | 75.69(25.26) |
| Unres | 77.63(27.52)   | 69.65(27.91) | 85.93(17.35)             | 90.31(17.26) |
| Total | 75.38(26.41)   | 66.85(27.74) | 88.84(14.83)             | 83.38(22.38) |

### 3.6.3.2 Experiment 3: Results

Table 3.10 presents the mean and SD for students' corresponding learning performance in Experiment 3. One-way ANOVA tests show that no significant difference is found on the pre-test score either between *WIG-det* and *Random*,  $F(1, 80) = 2.03$ ,  $p = 0.16$ , or between Responsive and Unresponsive groups,  $F(1, 80) = 0.67$ ,  $p = 0.42$ . Additionally, no significant difference is found either between the two Responsive groups,  $F(1, 40) = 0.87$ ,  $p = 0.36$ , or between the two Unresponsive groups,  $F(1, 38) = 1.21$ ,  $p = 0.28$ .



**Figure 3.9** Interaction effect for the adjusted transfer post-test score in Experiment 3

**Transfer Post-Test Score.** A two-way ANCOVA test using Policy and Type as two factors and the pre-test score as covariate shows that there is no significant main effect of either Policy or Type, but

there is a significant interaction effect on post-test score,  $F(1, 77) = 6.94, p = .010$ . Figure 3.9 depicts the cross-over interaction between Policy and Type on the adjusted transfer post-test score.

Furthermore, planned contrasts using Tukey’s adjustment indicate a significant difference between the two Responsive groups in that *WIG-Resp* achieved the significantly higher adjusted transfer post-test score than *Random-Resp*,  $t(77) = 2.54, p = .013$ , while there is no significant difference between two Unresponsive groups.

### 3.6.3.3 Experiment 3: Conclusion

Again results from Experiment 3 shows that there is an ATI effect. The Unresponsive groups are less sensitive to the policies in that they achieve a similar performance on the transfer post-test scores, while the Responsive groups are more sensitive in that their performances are strongly dependent on the effectiveness of the policy. Specifically, the *WIG-det* policy is more effective than the *Random* policy for the Responsive groups.

### 3.6.4 Experiment 4: Stochastic Policy Execution

In Experiments 1–3, all RL policies were executed deterministically, that is, the action was fully carried out given a state according to the induced RL-policies. However, one classic problem in RL is finding a balance between exploration (discovering more about the world) and exploitation (using what we already know to maximize performance). One approach to improving deterministic policies is to execute them stochastically, where each action is associated with a probability and has a chance to be selected. Therefore, we converted the *WIG-det* policy in Experiment 3 into a stochastic policy, called *WIG-sto*, through the *softmax function* (Section 3.2.3). We conducted Experiment 4 in the Fall of 2017 to investigate two research questions:

- **Q1 (State):** Can effective feature selection methods indeed empirically improve the effectiveness of the induced policy?
- **Q3 (Execution):** Can the stochastic policy execution be more effective than the deterministic policy execution?

#### 3.6.4.1 Experiment 4: Participants and Results

A total 101 of students were randomly split into two distinct groups, *Random* ( $N = 51$ ) and *WIG-sto* ( $N = 50$ ). In the end, a total of 88 students completed the experiment, distributed as *Random* ( $N = 44$ ) and *WIG-sto* ( $N = 44$ ). Pearson’s chi-squared test shows that no significant relationship exists between completion rate and condition,  $\chi^2(1, N = 101) = 0, p = 1$ .

Table 3.11 presents the mean and SD for students’ corresponding learning performance in Experiment 4. There is no significant difference between *WIG-det* and *Random* on either pre-test,

**Table 3.11** Pre-test and Transfer Post-test in Experiment 4

| Type  | Pre-Test Score |              | Transfer Post-Test Score |              |
|-------|----------------|--------------|--------------------------|--------------|
|       | WIG-sto        | Random       | WIG-sto                  | Random       |
| Resp  | 67.43(30.75)   | 75.25(26.37) | 95.24(12.49)             | 92.79(16.63) |
| Unres | 70.96(25.89)   | 68.44(33.02) | 91.07(14.60)             | 94.04(12.26) |
| Total | 69.11(28.26)   | 72.01(29.58) | 93.25(13.54)             | 93.39(14.55) |

$F(1, 86) = 0.22, p = 0.64$ , or transfer post-test,  $F(1, 86) = 0.02, p = 0.96$ , due to a ceiling effect: about 72.8% of students receive a transfer post-test score of 100. As a result, the *WIG-sto* group scores as high on the transfer post-test as the *Random* group.

Furthermore, as in Experiments 2 and 3, we conduct a median split using the “average response time on level 1” (*level1-avgstepTime*, median = 5.29 sec). Note that this median time is much lower than those used in Experiments 1–3. After splitting, the ceiling effect was found among all four groups of students.

#### 3.6.4.2 Experiment 4: Conclusion

Despite the fact that we used the same DT version, had similar test items in the transfer post-test, and had balanced assignment of students involved in Experiment 4, we found a ceiling effect on the transfer post-test score, which is a significant limitation of Experiment 4. While it is not clear whether the stochastic policy execution would indeed have an effective impact on students’ learning performance, it did show that when conducting empirical studies in this domain, we still face many challenges that need to be addressed, especially how to effectively evaluate the induced policies.

#### 3.6.5 Conclusions of Experiments

We investigated the impact of reward function, state representation, and policy execution on the effectiveness of RL-induced policies using the MDP framework. Four experiments were conducted to compare a series of RL-induced policies with that of a Random policy. With the exception of a ceiling effect found in Experiment 4, an ATI effect is consistently observed across Experiments 1–3 after splitting students into the Responsive and Unresponsive groups using their *level1-avgstepTime*. Specifically, the Unresponsive groups are less sensitive to the effectiveness of policies in that they perform similarly to their random peers regardless of the policies, while the Responsive groups are more sensitive to the RL-induced policies.

For the reward function, we found that using Immediate rewards works more effectively than

using Delayed rewards in Experiment 2, while no significant difference is found between *Ensemble-Delay-Resp* and *Random-Resp*. For policy execution, unfortunately, we can not determine the effectiveness of the stochastic policy execution due to a ceiling effect on transfer post-test scores.

For the state representation, we find that by combining effective feature selection methods with RL, our MDP-induced policies can be more effective than the random policy for Responsive students: for Experiment 1, while no significant difference was found between the *Random-Res* and *Random-Unre* groups, the *MDP-ECR-Resp* group scores significantly higher than the *MDP-ECR-Unres* group. For Experiment 2, while no significant difference is found among the three Unresponsive groups on the transfer post-test scores, the *Ensemble-Imme-Resp* group scores significantly higher than the *Random-Resp* group. For Experiment 3, again while no significant difference is found among the three Unresponsive groups on the transfer post-test scores, the *WIG-det-Res* group scores significantly higher than the *Random-Res* group.

Despite all these findings, for different experiments students are split into Responsive vs. Unresponsive students using different median split values and criteria: for Experiment 1, we have  $level1-avgstepTime < 7.6$  sec as Responsive group vs.  $level1-avgstepTime \geq 7.6$  sec as Unresponsive group; for Experiment 2, we have  $level1-avgstepTime \geq 8.01$  sec as the Responsive group vs.  $level1-avgstepTime < 8.01$  sec as Unresponsive group; and for Experiment 3, we have  $level1-avgstepTime \geq 8.34$  sec as Responsive group vs.  $level1-avgstepTime < 8.34$  sec as Unresponsive group. Therefore, it is not clear whether the same results will hold if we split them using one global median value and criteria. Additionally, different feature selection methods are applied for inducing different MDP policies in Experiments 1–3. Thus we conduct a post-hoc comparison to explore the impact of different feature selection methods on the effectiveness of the induced policies.

### 3.7 Post-hoc Comparisons

In Experiments 1–4, students were drawn from the same target population and all of them were enrolled in experiments with the same method but in different semesters. By assigning students to each condition randomly, it provides the most rigorous test of our hypotheses. In this section, we conduct a post-hoc comparison across the four experiments in the hope that this wider view will shed some light on our main results.

Since all Random students followed the Random policy and trained on the same DT tutor across all four experiments, we expect their performance on both pre-test and transfer post-test will reflect whether our students are indeed similar and whether our transfer post-tests are equivalent from semester to semester. A one-way ANOVA test shows that there is no significant difference on the pre-test score among the four Random groups, however a one-way ANCOVA test on Experiment using the pre-test score as covariate shows there is a significant difference among the four Random groups on the transfer post-test scores,  $F(3, 127) = 3.60$ ,  $p = .015$ . Specifically, post-hoc

Tukey HSD tests show that while no significant difference is found among Random groups across Experiments 1–3, the Random group in Experiment 4 scores significantly higher than Random in Experiment 1,  $t(127) = -2.89, p = .024$ . This suggests that Experiment 4 is significantly different from the first three experiments. Indeed, once we combine the three Random groups across Experiments 1–3 into a large Random group, referred as *Com-Random*, and referring to the Random group in Experiment 4 as *Random4*, a one-way ANCOVA test using the pre-test score as covariate indicates that there is a significant difference,  $F(1, 128) = 8.58, p = .004$ , such that *Random4* ( $M = 93.39, SD = 14.55$ ) scores higher on the transfer post-test than *Com-Random* ( $M = 79.21, SD = 27.96$ ). Therefore, our post-hoc comparison will only involve Experiments 1–3 and involve five groups: ranking from most recent to the oldest, *WIG-det*, *Ensemble-Imme*, *Ensemble-Delay*, *MDP-ECR*, and *Com-Random* groups.

### 3.7.1 Global Median Split

While the ATI effect exists in Experiments 1–3, the Responsive and Unresponsive groups are split in different ways for different experiments. In post-hoc comparisons, we explore consistent splitting criteria and investigate whether the same results will hold. For the global median split, we combine all the students in all policy groups who were in our analysis across Experiments 1–3. Particularly, we find that students whose *level1-avgstepTime* < 8.01 sec are less sensitive to the effectiveness of pedagogical strategies than their peers whose *level1-avgstepTime* ≥ 8.01 sec. In the following section, we refer the former as the Unresponsive group and the latter as the Responsive group.

Combining Policy {*WIG-det*, *Ensemble-Imme*, *Ensemble-Delay*, *MDP-ECR*, *Com-Random*} with Type factor {Responsive, Unresponsive}, we have a total of 10 groups. Table 3.12 shows the number of the students in each group, and a Pearson’s chi-squared test indicates that there is no significant difference in the distribution of Responsive vs. Unresponsive among the five policies,  $\chi^2(4, N = 239) = 3.10, p = 0.54$ .

**Table 3.12** Size of each group in post-hoc comparisons

| Type | Experiment 1<br><i>MDP-ECR</i> | Experiment 2<br><i>Ensemble-Imme</i> | Experiment 2<br><i>Ensemble-Delay</i> | Experiment 3<br><i>WIG-det</i> | Experiments 1,2,3<br><i>Com-Random</i> |
|------|--------------------------------|--------------------------------------|---------------------------------------|--------------------------------|--|
| Ures | 27                             | 15                                   | 16                                    | 20                             | 46                                     |
| Resp | 18                             | 18                                   | 18                                    | 24                             | 41                                     |

In the post-hoc analysis, we compare the three MDP policies against the *Com-Random* policy to determine the impact of the feature selection methods. All three MDP policies (*WIG-det*, *Ensemble-Imme*, and *MDP-ECR*) are induced by applying different feature selection methods with

RL using *immediate* rewards, DT-Imme. Additionally, to determine the impact of the reward function we compared the *Ensemble-Imme* and *Ensemble-Delay* against *Com-Random* since the former two use the same feature selection method. For the impact of the reward function, the same patterns are found in the post-hoc comparison as in Experiment 2: while no significant difference is found among the three Unresponsive groups, the *Ensemble-Imme-Resp* significantly out-performs the *Random-Resp* and no significant difference is found between the *Ensemble-Delay-Resp* and *Random-Resp*. Therefore, in the following, we will focus on exploring the impact of the feature selection on RL-induced policies.

### 3.7.2 The Impact of Feature Selection on RL Policies

Table 3.13 presents the mean and SD for students' pre-test and transfer post-test scores for eight groups of students: four Policies {*WIG-det*, *Ensemble-Imme*, *MDP-ECR*, *Com-Random*}  $\times$  2 Types {Responsive, Unresponsive}. It is important to note that since all students are split using the new global median value, the pre-test and transfer post-test scores are different from those listed in the tables for the individual experiments.

**Table 3.13** Pre-test and Transfer Post-test Score across Experiment 1-3

| Policy        | Pre-Test Score |              | Transfer Post-Test Score |                     |
|---------------|----------------|--------------|--------------------------|---------------------|
|               | Unres          | Resp         | Unres                    | Resp                |
| WIG-det       | 80.23(22.64)   | 71.34(29.04) | 84.53(17.60)             | <b>92.45(11.21)</b> |
| Ensemble-Imme | 54.11(37.27)   | 62.20(31.84) | 83.33(22.37)             | 90.97(24.37)        |
| Com-Random    | 59.37(32.18)   | 71.51(26.25) | 84.10(24.98)             | 73.70(30.34)        |
| MDP-ECR       | 60.48(30.56)   | 49.53(31.82) | 87.96(15.97)             | 66.32(28.93)        |
| Total         | 60.91(31.55)   | 66.24(29.79) | 85.98(20.24)             | 80.12(27.88)        |

**Pre-test scores.** A two-way ANOVA test using Policy and Type as two factors show that there is no significant main effect of Type, no significant interaction effect of Policy and Type, but a significant main effect of Policy on pre-test score:  $F(3, 201) = 3.54, p = .016$ . Specifically, planned contrasts using Tukey's adjustment indicate a significant difference between *WIG-det* and *Com-Random* in that the former achieved the significantly higher pre-test score than the later,  $t(201) = 3.22, p = .009$ , while there is no significant difference for other pair of policies.

**Transfer Post-Test Score.** To take the differences among the eight groups on the pretest into account, we run a two-way ANCOVA test, using Policy and Type as the two factors and the pre-test score as covariate. Results show that there is a significant main effect of Type,  $F(1, 200) = 3.91$ ,

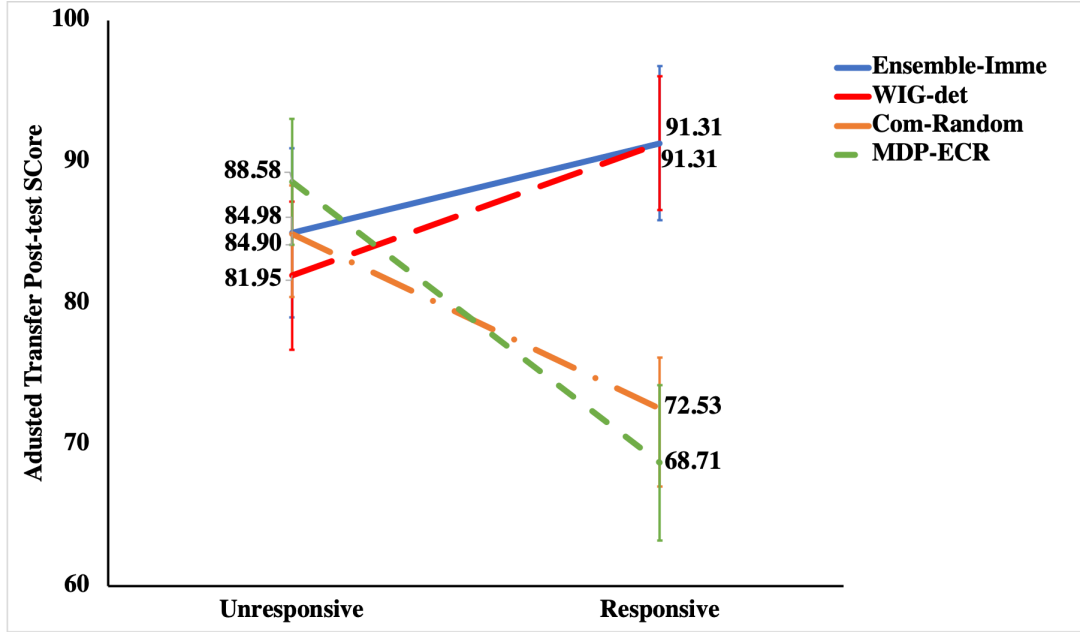


Figure 3.10 Interaction effect for adjusted transfer post-test score across Experiment 1-3

Table 3.14 Pairwise Contrasts on Adjusted Transfer Post-test in Post-hoc Comparison

| Pairwise Policy Comparison  |                                 | <i>t</i> (200) | <i>p</i> -value |
|-----------------------------|---------------------------------|----------------|-----------------|
| WIG-det- <b>Resp</b>        | vs. Random- <b>Resp</b>         | 3.16           | 0.022 *         |
| WIG-det- <b>Resp</b>        | vs. MDP-ECR- <b>Resp</b>        | 3.09           | 0.027 *         |
| Ensemble-Imme- <b>Resp</b>  | vs. Random- <b>Resp</b>         | 2.86           | 0.054 ·         |
| Ensemble-Imme- <b>Resp</b>  | vs. MDP-ECR- <b>Resp</b>        | 2.92           | 0.045 *         |
| WIG-det- <b>Resp</b>        | vs. Ensemble-Imme- <b>Resp</b>  | 0.00           | 1.00            |
| MDP-ECR- <b>Resp</b>        | vs. Random- <b>Resp</b>         | 0.57           | 1.00            |
| WIG-det- <b>Unres</b>       | vs. Random- <b>Unres</b>        | 0.47           | 1.00            |
| WIG-det- <b>Unres</b>       | vs. MDP-ECR- <b>Unres</b>       | 0.96           | 0.99            |
| Ensemble-Imme- <b>Unres</b> | vs. Random- <b>Unres</b>        | 0.01           | 1.00            |
| Ensemble-Imme- <b>Unres</b> | vs. MDP-ECR- <b>Unres</b>       | 0.48           | 1.00            |
| WIG-det- <b>Unres</b>       | vs. Ensemble-Imme- <b>Unres</b> | 0.38           | 1.00            |
| MDP-ECR- <b>Unres</b>       | vs. Random- <b>Unres</b>        | 0.66           | 0.99            |

· marginally significant at  $p < 0.1$ ; \* significant at  $p < 0.05$ .

$p = .049$ , and a significant interaction effect of Policy  $\times$  Type on transfer post-test scores,  $F(3, 200) = 4.22$ ,  $p = .006$ . The interaction is shown in Figure 3.10, which presents the mean and standard error of adjusted transfer post-test score for each group, which is the transfer post-test score adjusted by

the linear regression model built to describe the relation between the pre- and transfer post-test scores.

Table 3.14 presents the results of contrast tests using Tukey's adjustment for multiple comparisons. Results indicate that *WIG-det-Resp* scored significantly higher than *MDP-ECR-Resp* and *Random-Resp*;  $p = .022$  and  $p = .027$  respectively; *Ensemble-Imme-Resp* achieved higher score than *MDP-ECR-Resp* and *Random-Resp*, where the difference is significant  $p = .045$  and marginally significant  $p = .054$  respectively. No significant difference is found between other pairs.

**Conclusion.** We find that the ATI effect exists in the post-hoc comparisons. Specifically, the Unresponsive groups are less sensitive to the effectiveness of policies since they achieve similar transfer post-test scores, whereas the Responsive groups are more sensitive in that their learning performance is significantly dependent on the policy. Specifically, the *WIG-det* policy outperforms the *MDP-ECR* and *Random* policies in terms of transfer post-test score for the responsive students. Results suggest that the WIG-Low and possibly the Ensemble feature selection approaches can facilitate the MDP inducing more effective policies than the Random policy, while the ECR-based feature selection approach cannot be as effective as the former two approaches.

### 3.7.3 Problem Solving vs. Worked Example under policies

Table 3.15(a) presents the mean and SD of PS and WE count decided by policies across Experiment 1-3, and Table 3.15(b) shows results of one-way ANOVA tests between Responsive and Unresponsive under each policy condition. One-way ANOVA tests show that the significant difference on PS Count only exists between *Ensemble-Imme-Unres* and *Ensemble-Imme-Resp* in that the former assigned the significantly more PS than the later, while there is no significant difference on both PS and WE counts between Responsive and Unresponsive group under other four policies.

Table 3.15(c) shows results of the Tukey HSD tests for each pairwise policy comparison under each group type {Responsive, Unresponsive, Total}. Particularly, *Ensemble-Imme* has the significantly different PS and WE counts comparing with the other four policies, among which there are some significant differences on WE Count instead of PS Count. For *Total* groups without splitting students into *Responsive* and *Unresponsive*, *WIG-det* and *Com-Random* assigned the significant more WE than both *Ensemble-Delay* and *MDP-ECR*. For *Unresponsive* groups, *WIG-det-Unres* had the significant more WE than *MDP-ECR-Unres*, and *Com-Random-Unres* had the significant more WE than both *Ensemble-Delay-Unres* and *MDP-ECR-Unres*. For *Responsive* groups, *WIG-det-Resp* assigned the significant more WE than both *Ensemble-Delay-Resp* and *MDP-ECR-Resp*.

As the summary, although the PS and WE counts reflect the difference of policies, it is not the key reason why *Ensemble-Imme* and *WIG-det* policies outperform *Random*, which requires further data analysis.

**Table 3.15** PS and WE Counts and Comparisons for each Policy across Experiment 1-3

(a): PS and WE Count for each group

| Policy         | PS Count   |            |            | WE Count   |            |            |
|----------------|------------|------------|------------|------------|------------|------------|
|                | Unres      | Resp       | Total      | Unres      | Resp       | Total      |
| WIG-det        | 6.22(1.21) | 6.04(1.23) | 5.81(0.99) | 6.22(0.94) | 6.25(0.79) | 6.23(0.85) |
| Ensemble-Imme  | 2.46(1.85) | 1.27(0.59) | 1.96(1.44) | 9.08(0.86) | 9.13(0.52) | 9.11(0.68) |
| Ensemble-Delay | 5.33(0.65) | 5.89(1.36) | 5.57(1.03) | 5.33(0.65) | 5.00(1.12) | 5.19(0.87) |
| Com-Random     | 5.37(1.79) | 5.97(1.68) | 5.65(1.76) | 6.23(1.19) | 5.89(1.05) | 6.07(1.13) |
| MDP-ECR        | 5.82(0.95) | 5.80(1.08) | 6.12(1.21) | 5.00(0.35) | 5.20(0.86) | 5.09(0.64) |

(b): Unresponsive vs. Responsive comparison results for each policy

| Policy         | One-way ANOVA Tests                  |                             |
|----------------|--------------------------------------|-----------------------------|
|                | PS Count                             | WE Count                    |
| WIG            | $F(1, 40) = 0.22, p = 0.64$          | $F(1, 40) = 0.01, p = 0.92$ |
| Ensemble-Imme  | $F(1, 26) = 5.6, p = \mathbf{0.025}$ | $F(1, 26) = 0.05, p = 0.83$ |
| Ensemble-Delay | $F(1, 19) = 1.54, p = 0.23$          | $F(1, 19) = 0.74, p = 0.4$  |
| Com-Random     | $F(1, 80) = 2.44, p = 0.12$          | $F(1, 80) = 1.81, p = 0.18$ |
| MDP-ECR        | $F(1, 30) = 0.004, p = 0.95$         | $F(1, 30) = 0.77, p = 0.39$ |

(c): Tukey multiple comparison results among policies

| Pairwise Policy Comparison |     |                | PS Count (p-value) |                 |                 | WE Count (p-value) |                 |                 |
|----------------------------|-----|----------------|--------------------|-----------------|-----------------|--------------------|-----------------|-----------------|
|                            |     |                | Unres              | Resp            | Total           | Unres              | Resp            | Total           |
| Ensemble-Imme              | vs. | WIG-det        | <b>&lt;1e-5</b>    | <b>&lt;1e-5</b> | <b>&lt;1e-5</b> | <b>&lt;1e-5</b>    | <b>&lt;1e-5</b> | <b>&lt;1e-5</b> |
| Ensemble-Imme              | vs. | Ensemble-Delay | <b>9.6e-4</b>      | <b>&lt;1e-5</b> | <b>&lt;1e-5</b> | <b>&lt;1e-5</b>    | <b>1e-05</b>    | <b>&lt;1e-5</b> |
| Ensemble-Imme              | vs. | Com-Random     | <b>7.6e-4</b>      | <b>&lt;1e-5</b> | <b>&lt;1e-5</b> | <b>&lt;1e-5</b>    | <b>&lt;1e-5</b> | <b>&lt;1e-5</b> |
| Ensemble-Imme              | vs. | MDP-ECR        | <b>1.6e-4</b>      | <b>&lt;1e-5</b> | <b>&lt;1e-5</b> | <b>&lt;1e-5</b>    | <b>&lt;1e-5</b> | <b>&lt;1e-5</b> |
| WIG-det                    | vs. | Ensemble-Delay | 0.15               | 1.0             | 0.67            | <b>0.049</b>       | 0.10            | <b>5.4e-4</b>   |
| WIG-det                    | vs. | Com-Random     | 0.37               | 1.0             | 0.91            | 1.0                | 1.0             | 1.0             |
| WIG-det                    | vs. | MDP-ECR        | 1.0                | 1.0             | 1.0             | <b>3.9e-4</b>      | <b>0.007</b>    | <b>&lt;1e-5</b> |
| Ensemble-Delay             | vs. | Com-Random     | 1.0                | 1.0             | 1.0             | <b>0.016</b>       | 0.49            | <b>0.004</b>    |
| Ensemble-Delay             | vs. | MDP-ECR        | 1.0                | 1.0             | 1.0             | 1.0                | 1.0             | 1.0             |
| Com-Random                 | vs. | MDP-ECR        | 1.0                | 1.0             | 1.0             | <b>&lt;1e-5</b>    | 0.18            | <b>&lt;1e-5</b> |

Bold value indicates the significant difference at  $p < 0.05$ .

### 3.8 Conclusions, Limitations, & Discussion

We conducted four experiments to investigate the effectiveness of reinforcement learning induced policies using the MDP framework. Overall, an aptitude-treatment interaction effect consistently exists among Experiments 1–3 and the post-hoc comparisons. Furthermore, our students were split based on their response time, and we found the Unresponsive groups have similar learning performance under different policies employed by the ITS, whereas Responsive groups are more sensitive to the induced policies in that those under an effective policy would perform significantly better than those under an ineffective policy.

When applying RL to induce policies, we explored the impact of reward function, state representation, and policy execution. For policy execution, no significant improvement was found for the stochastic policy execution due to a ceiling effect. For future studies, the ceiling effect may be eliminated if we assign harder questions to students during the transfer post-test and adjust the grading rubric for the post-test to provide more finely grained evaluation and continuous scores.

In many domains, RL is applied with an immediate reward function. For example, in an automatic call center system, the agent can receive an immediate reward for every question it asks because the impact of each question can be assessed instantaneously [Wil08]. Immediate rewards are often chosen for RL-based policy induction because it is easier to assign appropriate credit or blame when the feedback is tied to a single decision. The more that rewards or punishments are delayed, the harder it becomes to properly assign credit or blame. However, for an ITS, the most appropriate rewards to use are student learning gains, which are typically unavailable until the entire tutoring session is complete. This is due to the complex nature of the learning process, making it difficult to assess student learning moment by moment. More importantly, many instructional interventions that boost short-term performance may not be effective over the long-term; for example, an instructional intervention may reduce the time a student spends solving a problem, but may also lead to shallow learning of the material [Bak04]. We explored both immediate and delayed rewards in our policy induction and empirically evaluated the impact of the induced policies on student learning. Our results show that using immediate rewards can be more effective than using delayed rewards, probably because of the vanishing reward problem: the discount factor in the MDP framework makes the rewards in the early decisions become extremely small with respect to the delayed reward.

For state representation, we explored feature selection based on the MDP framework. Although many feature selection methods such as embedded incremental feature selection [Wri12], *LSPI* [Li09], and *Neighborhood Component Analysis* [Gol05] can be applied to RL, most of these methods are designed for *model-free* RL, and we focus on *model-based* RL due to the high cost of collecting training data on ITSs. While correlation-based feature selection methods have been widely used for supervised learning for selecting the most relevant state features to the output label [Hal99; YL03],

in this work we explored five correlation-based metrics with two options: one option is to select the next feature that is the **most correlated (High)** to the currently selected feature set whereas the other option is to select the **least correlated (Low)**. Choosing the most correlated feature may be effective since the feature is more likely to be related to decision making; however, it may not make much more of a contribution than the currently selected feature set. Alternatively, choosing the least correlated feature may raise the diversity of the feature set, enriching the state representation; however, this has the risk of selecting irrelevant or noisy features. Our results show that low correlation methods perform significantly better than high correlation methods, the RL-based approach from our previous work [Chi11], and the baseline random method in terms of the expected cumulative reward (ECR). In particular, low correlation methods improve over high correlation methods as much as 142.48%, with an average of 45.2% improvement in ECR. In general, we have: Low correlation-based > Ensemble > High correlation-based > ECR-based  $\gg$  Random (Sec. 3.4.6).

Empirical results from Experiments 2 and 3 show that by applying effective feature selection to MDP, the Responsive groups using an RL-induced policy can significantly outperform their peers using a random policy. Additionally, post-hoc comparison results (Sec. 3.7.2) show that the empirical effectiveness of policies can be ordered as: *WIG-det* > *MDP-ECR*, *Random* (Sec. 3.7.2). Therefore, our results suggest that a low correlation-based feature selection approach is more effective than other feature selection methods for RL.

There are several caveats in our experiments that provide enlightenment regarding future work. First of all, we retrospectively split students into Responsive vs. Unresponsive groups using response time because we do not fully understand why the differences between Responsive vs. Unresponsive groups exist. To answer such a question, we need to perform deep log analysis for our future work. Second, although we detect different performance among the different RL-induced policies, it is still not clear what makes them effective or why they are effective. Future work is needed to shed some light on understanding the induced policies and to compare the machine induced policies with existing learning theory. Third, we mainly compare the RL-induced policies with a Random policy in our experiments and it is not clear if the same results would hold if we compare them against a stronger baseline such as those used in previous research [MI11; McL14; Naj14; Sal10]. Finally, in this work, we selected a small set of features from 133 observable state features which severely limits the effectiveness of tabular MDP methods. Many of the relevant factors such as motivation, affect, and prior knowledge, cannot be observed directly nor are they described explicitly. On the other hand, Partially-observable MDPs (POMDPs) 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.

# PARTIALLY OBSERVABLE MARKOV DECISION PROCESS

## 4.1 Introduction

Pedagogical strategy induction in ITSs is complicated by the fact that the real state space is often large and continuous and there is no prior knowledge about the appropriate structure of state representations. This severely limits the effectiveness of traditional tabular MDP framework that contains a small predefined state representations which consist a set of features suggested by the learning literature or feature selection approaches. However, there are so many unobservable factors such as motivation, proficiency, affect, cannot be observed directly or described explicitly. On the other hand, POMDP is able to model unobservable factors by using the belief state space, comprising probability distributions over latent states. More specifically, POMDP assigns probability to each latent state based on the observation with a wide range of features at each time step [Pin06; Pin03]. Consequently, the advantages of POMDP over MDP are that the former can deal with a large set of features, and can model the unobservable factors. As a result, we believe that POMDP will be more effective than tabular MDP for ITSs. While there has been substantial prior research in this area, the process of applying RL to induce pedagogical policies is not yet well understood nor have we fully explored the reasons that the induced policies are effective or ineffective. In this chapter, we empirically evaluate the effectiveness of POMDP and tabular MDP frameworks in the context

of ITSs through two experiments.

Furthermore, we investigate the effectiveness of the deterministic and stochastic policy executions for both POMDP and MDP policies. In stochastic policy execution, there is a small probability that the tutor deviates from the policy and takes a randomly-selected action at each decision point. If current decision of a policy is sub-optimal, it is still possible for the tutor to take the optimal action in stochastic execution (Chapter 3, Sec 3.2.3). We argue that stochastic execution can be more effective than deterministic execution.

This chapter is modified from a paper published in [She18a]. The rest of this chapter is arranged as follows: Section 4.2 describes the POMDP framework and POMDP policy induction. Section 4.3 presents the overview of our two empirical studies and research questions. Section 4.4 and Section 4.5 reports experimental results for the two experiments respectively. Section 4.6 presents our post-hoc comparison results. Finally, we summarize our conclusions, limitations and in Section 4.7.

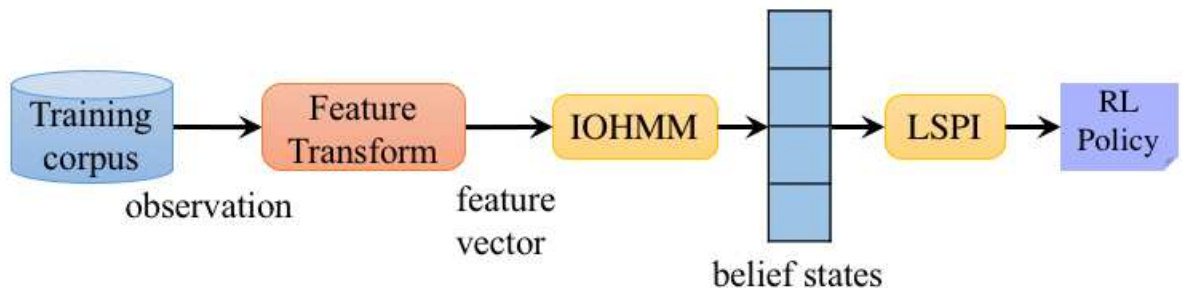
## 4.2 POMDP Framework

The POMDP framework is defined as a tuple  $\langle S, O, A, R, P_s, P_o, Prior, B \rangle$ . Here  $S$  represents the set of hidden states  $\{S_1, S_2, \dots, S_K\}$  with the set size  $K$ ; and  $O$  is the set of observations with a wide range of features.  $A$  and  $R$  denote the set of actions and the reward function respectively.  $P_s$  denotes the hidden state transition probability:  $P_s(s', s, a) = Pr(s'|s, a)$ , and  $P_o$  is the emission probability:  $P_o(o, s, a) = Pr(o|s, a)$ . *Prior* denotes the prior probability distribution of hidden states. In addition,  $B$  denotes the belief state space, where each element  $b_t(s) = Pr(s|o_{1:t}, a_{1:t})$  is the probability distribution of the hidden state  $s$  at each time step  $t$  after executing the action sequence  $a_{1:t}$  and obtaining the observation sequence  $o_{1:t}$ . We can estimate  $b_t(s)$  as

$$b_t(s) = \frac{1}{Z} \sum_{s'} b_{t-1}(s') P_s(s', s, a_t) P_o(o_t, s, a_t) \quad (4.1)$$

Where  $Z$  is the normalization item. The belief state at the first step is calculated by multiplying the prior probability and the emission probability. Therefore, the decision-making process in the POMDP framework can be viewed as a two-step iterative process, where the first step is to choose optimal action based upon the current belief state and the second step is to update the belief state based upon equation (4.1).

Figure 4.1 shows the primary process for inducing the POMDP policy that we use. It can be divided into 3 stages: 1) feature transformation, 2) belief state estimation, and 3) policy induction.



**Figure 4.1** The process of POMDP policy induction

### 4.2.1 Feature Transformation

There are two different types of feature transformation processes including *feature selection* and *feature extraction*. In the *feature selection* process, we apply the MDP-based feature selection approach (Section 3.2.3) [SC16a] to select a small set of features as the input of the POMDP framework. For *feature extraction*, we apply Factor Analysis for Mixed Data (FAMD) [Pag04] 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 on the scaled space to extract the important components.

### 4.2.2 Belief State Estimation

We utilize Input-Output Hidden Markov Models (IOHMM) to translate the FAMD-based observations into belief states [BF95]. In this context the input and output denote the action and observation. More specifically, the belief state at each time step is calculated by following function (4.1) via the forward-backward IOHMM algorithm. Parameters for the IOHMM are estimated through the standard Expectation-Maximization (EM) algorithm. We treat the last observation in each trajectory as the terminal state to evaluate the transition probability from hidden states to the terminal ones. When training the IOHMM, we initially assign all of the parameters to random values. In order to avoid local optima we also run the EM algorithm 10 times with random initial parameter settings and use the maximum likelihood obtained as the global optimal result.

### 4.2.3 POMDP Policy Induction

We induce the POMDP policy by the Q-learning based algorithm, which mainly contains 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:

$$Q_t(b, a) = \sum_s b_t(s) \cdot Q(s, a) \quad (4.2)$$

Thus,  $Q_t(b, a)$  is a linear combination of the  $Q(s, a)$  for each hidden state with its corresponding belief state  $b_t(s)$  calculated via function (4.1). When the process converges,  $\pi^*$  is induced by taking the optimal action  $a$  at time  $t$  associated with the highest  $Q_t(b, a)$ .

## 4.3 Experiments Overview & Research Questions

### 4.3.1 Induced Policies

Given the training data set *DT-Imme* (Section 3.3.2), we induce and employ five policies listed in Table 4.1. POMDP-det and FPOMDP-det are induced by POMDP with WIG-Low, a feature selection approach (Section 3.2.3) and FAMD approach (Section 4.2.1) respectively. Specifically, the comparison between POMDP-det and MDP-det facilitates us understand whether belief state alone can make POMDP induce an effective policy. Additionally, the comparison between FPOMDP-det with MDP policies can indicate that whether the belief state, generated from a wide range of features, can facilitate POMDP inducing an effective policy.

**Table 4.1** Implemented policies in two experiments for POMDP study

| Policy     | Framework | Feature Transformation                       | Policy Execution | Policy Induction |
|------------|-----------|--|------------------|------------------|
| MDP-det    | MDP       | Feature Selection by WIG-Low (Section 3.4.3) | Deterministic    | Value Iteration  |
| MDP-sto    | MDP       | Same Feature Selection                       | Stochastic       | Value Iteration  |
| POMDP-det  | POMDP     | Same Feature Selection                       | Deterministic    | Q Learning       |
| FPOMDP-det | POMDP     | FAMD   | Deterministic    | Q Learning       |
| FPOMDP-sto | POMDP     | FAMD   | Stochastic       | Q Learning       |

Random: random yet reasonable decision (baseline)

Note: MDP-det policy is the WIG-Low policy shown in Table 3.9

### 4.3.2 Research Questions

In this chapter, we have three research questions:

**Q1:** Is the POMDP policy more effective than the MDP policy given a same set of selected features?

**Q2:** Does the POMDP policy induced with a wide range of features outperform the MDP policy?

**Q3:** Can the stochastic policy execution further improve the effectiveness of either the POMDP or the MDP policy as compared to the deterministic policy?

To answer the above research questions, we run several statistical tests in the following two experiments shown in Table 4.2. Since no significant difference was found between the two Random groups in Experiments 1 and 2, we then added post-hoc comparisons among all of the policies across two experiments.

**Table 4.2** Overview of Experiments

| Experiment          | Implemented Policies                           | Research Question |    |    |
|---------------------|--|-------------------|----|----|
|                     |  | Q1                | Q2 | Q3 |
| Experiment 1        | <i>POMDP-det, MDP-det, Random</i>              | ✓                 |    |    |
| Experiment 2        | <i>FPOMDP-det, FPOMDP-sto, MDP-sto, Random</i> |                   | ✓  | ✓  |
| Post-hoc Comparison | All of implemented policies                    |                   | ✓  | ✓  |

Specifically, we run the one-way ANOVA tests to investigate whether there is significant difference among policies on students' learning outcomes including pre- and pro- test scores, NLG, and also run the Tukey HSD tests to identify whether there is significant difference on students' learning outcomes between any two conditions.

### 4.3.3 Empirical Evaluation

Different from the four experiments in MDP application (Section 3.6) using *transfer post-test score* to evaluate students' performance, we define the post-test score and learning gain to evaluate students' performance in the following two experiments.

Recall that the last question of each level in DT is a PS without the tutor's help and it is functioned as a mini-post test for evaluating student's knowledge on concepts of that level (Section 3.3). We treat the mini-post test score as the *LevelScore* for each level. Since the bulk of the relevant content is covered in levels 3–6, the students scores on these four levels are used as our *post-test* to measure their post-training performance. More specifically, we calculated the score as *post-test*=

$\sum_{i=3}^6 LevelScore(i)/4$ . In addition to the pre- and post-test scores, we also evaluated students performance based on their *learning gain* ( $LG = post-test - pre-test$ ) and their *Normalized Learning Gain* ( $NLG = \frac{post-test - pre-test}{100 - pre-test}$ ) where 100 is the maximum post-test score.

## 4.4 Experiment 1: POMDP with Selected Features

### 4.4.1 Participants & Conditions

A total of 130 students enrolled in the Discrete Mathematics course at North Carolina State University in the Fall 2016 semester, and were randomly assigned to one of three conditions: MDP-det ( $N = 46$ ), POMDP-det ( $N = 42$ ) and Random ( $N = 42$ ). A total of 106 students completed the experiment: MDP-det ( $N = 40$ ), POMDP-det ( $N = 36$ ), Random ( $N = 30$ ). A  $\chi^2$  test showed no significant difference in the completion rate among the three conditions:  $\chi^2 = 0.147, p = 0.929$ .

**Table 4.3** Mean and SD of learning performance for each condition in Experiment 1

| Measure   | POMDP-det    | MDP-det      | Random       | ANOVA       |            |
|-----------|--------------|--------------|--------------|-------------|------------|
|           |              |              |              | $F(2, 103)$ | $p$ -value |
| pre-test  | 40.46(20.89) | 41.98(21.21) | 37.83(21.75) | 0.21        | 0.81       |
| post-test | 49.01(16.37) | 45.28(16.21) | 48.0(15.01)  | 0.56        | 0.57       |
| NLG       | 0.042(0.50)  | -0.094(0.56) | 0.066(0.51)  | 1.03        | 0.36       |

### 4.4.2 Results

**Learning Performance.** Table 4.3 show the students' learning performance under each condition in Experiment 1. One-way ANOVA tests showed no significant difference among the three conditions in terms of their pre-test scores. Much to our surprise, no significant difference was found among conditions on the post-test scores and NLG. Although Table 4.3 suggests that POMDP-det had slightly higher post-test and NLG scores than MDP-det, the differences are not significant. Note that NLG of all of the three conditions were not high and it suggests that that the POMDP and MDP policies may not be effective and performed closely to the Random policy.

**Behaviors.** Table 4.4 presents the total time (in hours) that students spent in DT, *PSCount* and *WECOUNT*, which are the number of PS and WE that were determined by the policies respectively. Note that there were extra 9 problems determined by hard-coded pre-defined rules rather than policies. One-way ANOVA tests showed no significant difference on either *Time* or *WECOUNT* among the three conditions. But there was significant difference on *PSCOUNT*. Particularly, the Tukey HSD

**Table 4.4** Mean and SD of Behavior Variables for each condition in Experiment 1

| Measure  | POMDP-det  | MDP-det    | Random     | ANOVA       |            |
|----------|------------|------------|------------|-------------|------------|
|          |            |            |            | $F(2, 103)$ | $p$ -value |
| Time     | 3.72(1.89) | 3.03(1.86) | 3.45(2.31) | 1.16        | 0.33       |
| PSCount  | 6.22(1.79) | 5.10(1.71) | 6.47(1.52) | 6.82        | 0.002 **   |
| WECCount | 5.91(1.20) | 6.30(0.91) | 5.80(1.27) | 1.97        | 0.15       |

· marginally significant at  $p < 0.1$ ; \* significant at  $p < 0.05$ ; \*\* significant at  $p < 0.005$ .

test suggests that POMDP-det solved significantly less PSs than both MDP-det ( $p = .012$ ) and Random ( $p = .003$ ).

### 4.4.3 Conclusion & Discussion

Consistent with prior research on applying MDP to ITSs, Experiment 1 results show that the MDP policy performed no better than Random. Much to our surprise, the POMDP-det policy did not outperform the MDP-det and Random policies. One of the possible explanations is that the limited state representation of POMDP restricts the effectiveness of the POMDP policy. Specifically, we strictly controlled the set of features used in MDP to be the same as the observation space of POMDP so that the primary difference between MDP and POMDP was that the later used latent states and the belief state space. As the MDP-det policy was not effective, results suggest that simply adding latent states and belief state space did not make a substantive difference. Therefore in Experiment 2, we opted to generate the belief state space with a large range of features and then to compare that expanded POMDP against MDP.

## 4.5 Experiment 2: POMDP with a wide range of features

### 4.5.1 Participants & Conditions

A total of 183 students, who enrolled in the Discrete Mathematics course at North Carolina State University in the Spring 2017 semester, were randomly assigned into four conditions: FPOMDP-sto ( $N = 47$ ), FPOMDP-det ( $N = 46$ ), MDP-sto ( $N = 46$ ), and Random ( $N = 44$ ).

### 4.5.2 Results

**Learning Performance.** Table 4.5 presents the students' learning performance in Experiment 2. One-way ANOVA tests indicate that while No significant difference is found among the four conditions on pre-test score, there is the significant difference among conditions on either post-test score

or NLG. Furthermore, the Tukey HSD tests show that FPOMDP-sto scored significantly higher post-test than Random:  $p = 0.008$ , and FPOMDP-det scored marginally significant higher than Random  $p = 0.09$ ; both FPOMDP-sto and FPOMDP-det achieved the significantly higher NLG than Random:  $p = 0.009$  and  $p = 0.011$  respectively. Finally, no significant difference is found on post-test score and NLG either between the two FPOMDP conditions and MDP-sto, or between MDP-sto and Random. In short, results indicate that both FPOMDP-sto and FPOMDP-det outperformed Random while the later two performed similarly. Thus we can't verify whether stochastic policy execution is more effective than the deterministic execution.

**Table 4.5** Mean and SD of learning performance for each condition in Experiment 2

| Measure   | FPOMDP-det         | FPOMDP-sto          | MDP-sto      | Random       | ANOVA       |            |
|-----------|--------------------|---------------------|--------------|--------------|-------------|------------|
|           |                    |                     |              |              | $F(3, 179)$ | $p$ -value |
| pre-test  | 35.03(20.55)       | 45.03(22.29)        | 41.86(23.20) | 45.04(23.99) | 1.78        | 0.15       |
| post-test | 55.09(15.03)       | <b>60.58(21.55)</b> | 52.18(21.07) | 48.23(18.01) | 3.04        | 0.03 *     |
| NLG       | <b>0.247(0.49)</b> | <b>0.253(0.56)</b>  | 0.042(0.71)  | -0.175(0.77) | 3.88        | 0.01 *     |

· marginally significant at  $p < 0.1$ ; \* significant at  $p < 0.05$ ; \*\* significant at  $p < 0.01$ .

**Table 4.6** Mean and SD of behavior variables for each condition in Experiment 2

| Measure  | FPOMDP-det | FPOMDP-sto | MDP-sto    | Random     | ANOVA       |            |
|----------|------------|------------|------------|------------|-------------|------------|
|          |            |            |            |            | $F(3, 179)$ | $p$ -value |
| Time     | 2.40(1.07) | 2.03(0.82) | 2.21(1.14) | 2.03(0.89) | 1.42        | 0.24       |
| PSCount  | 8.58(1.67) | 7.83(2.15) | 4.11(1.30) | 6.34(1.31) | 65.79       | 2e-16 ***  |
| WECCount | 5.06(0.49) | 5.51(0.91) | 7.74(1.31) | 5.97(0.93) | 69.69       | 2e-16 ***  |

· marginally significant at  $p < 0.1$ ; \* significant at  $p < 0.05$ ; \*\* significant at  $p < 0.01$ .

**Behavior.** Table 4.6 shows then mean and SD of the total time (in hours) that students spent, *PSCount* and *WECCount* which are the number of PSs and WEs determined by policies respectively. A one-way ANOVA test shows that there is no significant difference among the four conditions on *Time*, but there is a significant difference on either *PSCount* or *WECCount* among conditions. Specifically, the Tukey HSD tests show that MDP-sto solved the significantly less PSs and more WEs than other three conditions. Additionally, FPOMDP-sto solved significantly more PSs than Random ( $p < .000$ ) respectively. No significant difference either on *PSCount* or on *WECCount* be-

tween two FPOMDP policies, or between FPOMDP-sto and Random. Therefore, 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.

### 4.5.3 Conclusion & Discussion

While Experiment 1 shows that POMDP-det is no more effective than either MDP-det or Random, Experiment 2 indicates that by incorporating a large range of features into the POMDP framework through the FAMD approach, the FPOMDP-det and FPOMDP-sto policies outperform Random while the MDP-sto policy does not. Therefore, results support that POMDP with a wide range of features is able to induce more effective policy than the tabular MDP framework.

## 4.6 Post-hoc Comparison

In Experiments 1 & 2, 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 conduct a meta-comparison across the two experiments in the hope that this wider view will shed some light on our main results. Table 4.7 represents the learning performance for each group in the post-hoc comparison. Specifically, Table 4.7a presents the mean and standard deviation (SD) of learning performance for each group in post-hoc comparison and Table 4.7b shows the results of pairwise t tests.

### 4.6.1 Across Six Groups

**Pre-test Score.** All of the participants were enrolled in the study in the same way but in different semesters. Moreover, one-way ANOVA tests indicated that there was no significant difference between the two Random groups on their pre-test scores:  $F(1, 72) = 1.73, p = 0.19$ , post-test scores ( $F(1, 72) = .003, p = 0.95$ ), LG ( $F(1, 72) = 1.46, p = 0.23$ ), and NLG ( $F(1, 72) = 2.28, p = 0.14$ ). Thus, for the purposes of this analysis we combined the Experiment 1 and 2 Random groups into a single large Random group ( $N = 74$ ). Although, a One-way ANOVA test shows no significant differences among the six groups on the pre-test score:  $F(5, 283) = 1.58, p = 0.16$ , pairwise t tests in Table 4.7b indicate that FPOMDP-det scored the significant lower pre-test than FPOMDP-sto, POMDP-det, MDP-det and Random.

**Post-test Score.** A one-way ANCOVA test using pre-test score as covariate shows that there is a marginally significant difference on the post-test score among six groups:  $F(5, 283) = 2.01, p = 0.07$ . Then we compared the adjusted post-test score for each pair of groups, which is the post-test score adjusted by the pre-test score based on the ANCOVA test. More specifically, pairwise t tests in Table

**Table 4.7** learning Performance In Post-hoc comparison For POMDP Study

(a) Mean and SD of Pre-test, Post-test and NLG for each group

| Measure            | FPOMDP-sto          | FPOMDP-det         | POMDP-det    | MDP-sto      | MDP-det      | Random       |
|--------------------|---------------------|--------------------|--------------|--------------|--------------|--------------|
| pre-test           | 45.03(22.29)        | 35.03(20.55)       | 40.46(20.89) | 41.86(23.20) | 41.98(21.21) | 42.12(23.23) |
| post-test          | <b>60.58(21.55)</b> | 55.09(15.03)       | 49.01(16.37) | 52.18(21.07) | 45.28(16.21) | 48.14(16.75) |
| adjusted post-test | <b>58.85(17.72)</b> | 55.89(14.08)       | 54.28(15.42) | 52.18(18.02) | 50.54(15.71) | 50.92(4.61)  |
| NLG                | <b>0.253(0.56)</b>  | <b>0.247(0.49)</b> | 0.042(0.50)  | 0.042(0.71)  | -0.094(0.56) | -0.076(0.68) |

adjusted post-test score is the post-test score adjusted by the pre-test score based on the ANCOVA test

(b) Comparison results of Pairwise t tests

| Comparison                | Significant Difference ( $p$ -value) |                          |                      |
|---------------------------|--------------------------------------|--------------------------|----------------------|
|                           | Pre-test Score                       | adjusted Post-test Score | NLG                  |
| FPOMDP-sto VS. FPOMDP-det | $p = \mathbf{0.04}$                  | $p = 0.37$               | $p = 0.78$           |
| FPOMDP-sto VS. POMDP-det  | $p = 0.88$                           | $p = 0.19$               | $p = 0.14$           |
| FPOMDP-sto VS. MDP-sto    | $p = 0.52$                           | $p = \mathbf{0.04}$      | $p = 0.11$           |
| FPOMDP-sto VS. MDP-det    | $p = 0.66$                           | $p = \mathbf{0.01}$      | $p = \mathbf{0.005}$ |
| FPOMDP-sto VS. Random     | $p = 0.89$                           | $p = \mathbf{0.007}$     | $p = \mathbf{0.008}$ |
| FPOMDP-det VS. POMDP-det  | $p = \mathbf{0.04}$                  | $p = 0.65$               | $p = 0.23$           |
| FPOMDP-det VS. MDP-sto    | $p = 0.15$                           | $p = 0.26$               | $p = 0.20$           |
| FPOMDP-det VS. MDP-det    | $p = \mathbf{0.02}$                  | $p = 0.12$               | $p = \mathbf{0.01}$  |
| FPOMDP-det VS. Random     | $p = \mathbf{0.03}$                  | $p = 0.09$               | $p = \mathbf{0.02}$  |
| POMDP-det VS. MDP-sto     | $p = 0.46$                           | $p = 0.55$               | $p = 0.99$           |
| POMDP-det VS. MDP-det     | $p = 0.79$                           | $p = 0.31$               | $p = 0.23$           |
| POMDP-det VS. Random      | $p = 0.73$                           | $p = 0.29$               | $p = 0.41$           |
| MDP-sto VS. MDP-det       | $p = 0.29$                           | $p = 0.63$               | $p = 0.21$           |
| MDP-sto VS. Random        | $p = 0.61$                           | $p = 0.67$               | $p = 0.38$           |
| MDP-det VS. Random        | $p = 0.51$                           | $p = 0.91$               | $p = 0.58$           |

Bold value indicates the significant difference at  $p < 0.05$ .

4.7b indicate that FPOMDP-sto achieved significantly higher adjusted post-test score than MDP-sto, MDP-det and Random and FPOMDP-sto had marginally significantly higher adjusted post-test score than Random. Additionally, there is no significant difference among POMDP-det, MDP-sto, MDP-det and Random, although Table 4.7a shows that POMDP-det and MDP-sto achieved the higher post-test score than either MDP-det or Random.

**Normalized Learning Gain (NLG).** A one-way ANOVA test showed a significant difference among the six groups on NLG:  $F(5, 283) = 2.64, p = 0.02$ . Pairwise t tests in Table 4.7b show that both FPOMDP-sto and FPOMDP-det achieved significantly higher NLG than either MDP-det or Random. Finally, although no significant difference is found among MDP-sto, POMDP-det, MDP-det, and Random, Table 4.7a indicates that both MDP-sto and POMDP-det had higher NLG than either MDP-det or Random.

In conclusion, when combining all of the groups across Experiments 1 and 2, we had: FPOMDP-sto  $\approx$  FPOMDP-det  $>$  POMDP-det  $\approx$  MDP-sto  $\approx$  MDP-det  $\approx$  Random.

#### 4.6.2 POMDP vs. MDP Framework

Although FPOMDP-sto and FPOMDP-det have different policy execution, both of them are induced based on the full power of the POMDP framework which takes a large range of features into consideration. Thus, We combined FPOMDP-sto and FPOMDP-det to form the full power combined POMDP group. Similarly, we also integrated MDP-sto and MDP-det into the combined MDP group. Note that the POMDP-det group was not included in the combined POMDP group because it did not use the full power of the POMDP framework. This leaves us with a combined POMDP group ( $N = 93$ ) and combined MDP group ( $N = 86$ ).

Table 4.8 present the comparison results among POMDP, MDP and Random on the learning performance. One-way ANOVA tests indicate that while there is no significant difference on pre-test score among the three conditions, the significant difference is found among conditions either on post-test score or NLG. Similarly, a One-way ANCOVA test using pre-test as covariate also shows a significant difference on post-test among conditions:  $F(2, 249) = 8.85, p = .000$ . Particularly, the Tukey HSD tests showed that POMDP scored significantly higher post-test and NLG than MDP:  $p = 0.001, p = 0.01$  respectively; and POMDP also significantly outperformed Random on the post-test score:  $p = .000$  and the NLG:  $p = .009$ . Additionally, there is no significant difference between MDP and Random in terms of post-test score and NLG. In short, results indicate that POMDP is more suitable for pedagogical policy induction than MDP framework.

#### 4.6.3 Stochastic vs. Deterministic Policy Execution

Similarly, in order to test the impact of Stochastic vs. Deterministic policy execution, we combined FPOMDP-sto with MDP-sto to one combined Stochastic group ( $N = 93$ ) and also integrated FPOMDP-

**Table 4.8** Learning performance for each framework in post-hoc comparison

| Measure   | POMDP               | MDP           | Random       | ANOVA       |                 |
|-----------|---------------------|---------------|--------------|-------------|-----------------|
|           |                     |               |              | $F(2, 250)$ | $p$ -value      |
| pretest   | 40.08(21.92)        | 41.92(22.17)  | 42.12(23.23) | 0.21        | 0.80            |
| post-test | <b>57.87(18.71)</b> | 48.97(19.17)  | 48.14(16.75) | 7.58        | <b>0.00 ***</b> |
| NLG       | <b>0.251(0.528)</b> | -0.021(0.648) | -0.076(0.68) | 6.97        | <b>0.002 **</b> |

· marginally significant at  $p < 0.1$ ; \* significant at  $p < 0.05$ ; \*\* significant at  $p < 0.01$ .

det and MDP-det into one combined Deterministic group ( $N = 86$ ).

Table 4.9 shows the learning performance for the Stochastic, Deterministic and Random groups and the results of one-way ANOVA tests. Specifically, one-way ANOVA tests indicate that there is a significant difference on post-test score and a marginally significant difference on NLG, but no significant difference on pre-test score among the three groups. Similarly, A one-way ANCOVA test using pre-test as covariate also indicates that the significant difference on post-test scores is found among groups. Particularly, the Tukey HSD tests show that the Stochastic group scored the significantly higher post-test than Random:  $p = 0.01$ , and achieved marginally significantly higher the NLG than Random:  $p = 0.056$ . Additionally, a one-way ANCOVA test, using pre-test as covariate and {Stochastic, Deterministic} as the factor, indicates that the Stochastic group scored significantly higher post-test than the Deterministic group ( $F(1, 176) = 11.46$ ,  $p = .001$ ). No significant difference was found between the Deterministic and the Random groups. In short, it is apparent that Stochastic policy execution can be more effective than both the Deterministic and Random policy executions.

**Table 4.9** Learning performance for each policy execution in post-hoc comparison

| Measure   | Stochastic          | Deterministic | Random       | ANOVA       |                |
|-----------|---------------------|---------------|--------------|-------------|----------------|
|           |                     |               |              | $F(2, 250)$ | $p$ -value     |
| pretest   | 43.46(22.68)        | 38.26(21.03)  | 42.12(23.23) | 1.28        | 0.28           |
| post-test | <b>56.43(21.61)</b> | 50.53(16.26)  | 48.14(16.75) | 4.52        | <b>0.012 *</b> |
| NLG       | <b>0.14(0.65)</b>   | 0.08(0.55)    | -0.076(0.68) | 2.78        | 0.06 ·         |

· marginally significant at  $p < 0.1$ ; \* significant at  $p < 0.05$ ; \*\* significant at  $p < 0.01$ .

#### 4.6.4 Post-hoc Discussion

Post-hoc comparisons across Experiments 1 and 2 strongly confirm that the POMDP framework is more effective than the tabular MDP framework for the pedagogical policy induction. As discussed in Experiment 2, both the belief state space representation and the ability to deal with a large range of features contribute to the improved performance of POMDPs over tabular MDPs.

In addition, we showed that the stochastic policy execution outperformed the deterministic execution. This is likely due to the fact that when our induced policies are not optimal, the stochastic policy execution provides a chance for the agent to explore the state space and to obtain better results while the deterministic policy execution can only exploit it. Further research is needed to assess this in detail.

### 4.7 Conclusions, Limitations, & Discussion

In this chapter, we induce the MDP-det and the POMDP-det policies by the MDP and POMDP framework respectively given the same data with a small set of selected features. We also induce the FPOMD-det and FPOMDP-sto policies based on POMDP with a wide range of features. We compare the effectiveness of POMDP-based policies against the MDP and the Random policies in two experiments. Additionally, we explore the policy execution for both POMDP and MDP policies, and empirically compare the effectiveness of the stochastic policy execution against the deterministic execution.

Empirical results indicate that students following the POMDP-det policy performed similarly to their peers following the MDP and the Random policies. In other words, the POMDP policy induced with a small set of features is no more effective than the MDP and the Random policies, which suggests that the belief state alone may not facilitate POMDP inducing an effective policy. Furthermore, results of Experiment 2 and Post-hoc comparison show that both FPOMDP-det and FPOMDP-sto policies outperformed MDP and Random in terms of students' post-test score and learning gain. we conclude that the ability of dealing with a wide range of features and transferring it into the belief state space contribute the results.

However, it's unclear how much benefit that POMDP can obtain from the belief state given a high dimensional feature space. It's highly possible that the ability of handling the high dimensional feature space instead of the belief state space is the key reason why POMDP outperforms tabular MDP. Thus, we can't conclude that the POMDP framework beats the MDP framework for policy induction in the ITS domain. In order to have a fair comparison between POMDP and MDP, and investigate the benefit of belief state space in POMDP, we need to compare the POMDP framework with continuous state space MDP since both of them are able to deal with the wide range of continuous features.

# CONSTRAINED ACTION-BASED POMDP

## 5.1 Introduction

Different from prior work, in this chapter, we mainly focus on inducing the pedagogical strategy in a *constrained action-based* RL (CARL) scenario, which involves the additional action-based constraints such as a maximum number of times that an agent may take a specific action. For instance, the CARL scenario we investigated is Deep Thought (DT) [MB17], which contains a total of seven learning phases, called levels, covering different knowledge components, the action-based constraints and one type of tutorial decision: whether to provide students with a *Worked Example (WE)* or to ask them to engage in *Problem Solving (PS)*. Particularly, the action-based constraints in DT are presented as the last problem on each level must be done in PS, and prior to reaching that problem the students must complete at least one PS and one WE. DT did not allow students stay in the extreme situation where they always solve PS or WE in a level. In this scenario, the early decisions impose special constraints on the future actions. In other words, the available actions for an agent at any given situation are governed not only by the current state but also by prior decisions. Therefore, when deciding the next action, the agent should take the constraints into account.

Prior research on *constrained* RL has focused on inducing the optimal policy subject to constraints such as *safety* and *risk avoidance*. Systems that physically interact with humans, for example, need to satisfy the basic safety parameters or engage in risk avoidance [Ach17]. Similarly robots that seek to reach a target position as quickly as possible should also avoid dangerous places (say a

crater) that might render them irretrievable [Lee17]. Prior researchers [Alt99; Pou15; DD05; GF12] who have sought to address such *constrained* scenarios have typically specified an additional *cost* function which has a similar format to the reward and then imposing constraints on the values of the cost functions. However, such constraints are different from the action-based constraints on which our work is focused. So far as we know, no prior work has directly sought to address the action-based constraints in an interactive e-learning environment.

We propose a general framework called Constrained Action-based Partially Observable Markov Decision Processes (CAPOMDP) and apply this framework to transform the CARL problem into a normal RL problem by leveraging factored state representations to incorporate constraints into the state space itself. Furthermore, we explore two types of reward functions: learning gain and time. For the former, the goal is to maximize learning gain, while the goal for the latter is to reduce the amount of time spent on completing the entire tutor. Prior research use either learning gain or time but *not both*. We apply CAPOMDP to induce two types of policies including CAPOMDP<sub>LG</sub> and CAPOMDP<sub>Time</sub> using learning gain and time as reward respectively. Furthermore, we conduct two empirical experiments to evaluate effectiveness of CAPOMDP policies based upon three metrics: learning gain, time and learning efficiency, which is the ratio between the first two metrics. In Experiment 1, we compare CAPOMDP<sub>LG</sub> against a POMDP<sub>LG</sub> policy induced by the POMDP framework with LG as reward and a random policy, where the system *randomly* decides whether to present the next problem as WE or as PS. Because both PS and WE are always considered to be *reasonable* educational interventions in our learning context, we refer to such policy as a *random yet reasonable* policy or *random* policy in the following. In Experiment 2, we compare both CAPOMDP<sub>LG</sub> and CAPOMDP<sub>Time</sub> against a DQN<sub>LG</sub> policy induced by the Deep Q-Network approach using LG as reward and the random policy.

Empirical results suggest that there is an Aptitude Treatment Interaction (ATI) effect [CS77a], where the high incoming competence students are not sensitive to the policies in that they achieved the similar learning performance regardless of policies employed whereas the low incoming competence students are sensitive in that their learning performance is highly dependant on the effectiveness of policies. Furthermore, empirical results indicate that CAPOMDP<sub>LG</sub> can significantly improve the low incoming competence students' learning performance; CAPOMDP<sub>Time</sub> is able to significantly reduce the total training time that the low incoming competence students spent in the tutor. Both CAPOMDP<sub>LG</sub> and CAPOMDP<sub>Time</sub> outperform the baseline policies.

This chapter is modified from a paper published in [She18c]. The rest of this chapter is arranged as follows: Section 5.2 mathematically define the general reinforcement learning problem, the constrained RL problem and the constrained action-based RL problem. Section 5.3 describes the CAPOMDP framework and the CAPOMDP policy induction by the LSPI approach. Section 5.4 presents the overview of our two empirical studies and research questions. Section 5.6 and Section 5.7 reports experimental results for the two experiments respectively. Section 5.8 presents the post-

hoc comparison results and Section 5.9 shows the statistical analysis on students' behaviors and the impact of behaviors on learning performance. Finally, we summarize our conclusions, limitations and in Section 5.10.

## 5.2 Problem Statement

RL approaches aim to construct an optimal policy that maps a state into an action with the purpose of maximizing the expected cumulative reward (ECR). Within an ITS environment, Pedagogical strategy is treated as an optimal policy, state at a time step summarizes the current students' learning process and ITS learning context, action denotes the tutorial action such as PS and WE, and ECR can be specified as students' learning gain or the negative version of total time that the student spent on the tutor. Therefore, the pedagogical strategy induction task can be naturally transferred to a **general RL** problem, formalized as maximizing expected cumulative reward  $E_\pi$ :

$$\max_{\pi} E_{\pi} \left[ \sum_{t=0}^L \gamma^t R(s_t, a_t) \right] \quad (5.1)$$

where  $L$  denotes the length of a completed episode in the ITS context, and  $\gamma$  is the discount factor.  $R(s_t, a_t)$  denotes the reward function for a state-action pair  $(s_t, a_t)$  at time step  $t$ .

The **constrained RL** problem contains the constraints, and can be solved by the constrained MDP or POMDP framework [Alt99; WY07; Han17] to search the optimal policy that maximizes ECR while maintaining the expected cumulative cost (ECC) under the upper bound. The constrained RL problem can be formalized as follows:

$$\max_{\pi} E_{\pi} \left[ \sum_{t=0}^L \gamma^t R(s_t, a_t) \right] \left| \sum_{t=0}^L \gamma^t C(s_t, a_t) < \hat{c} \quad (5.2)$$

where  $C(s_t, a_t)$  denotes a cost function or a constant value for a state-action pair  $(s_t, a_t)$  at time step  $t$ , and  $\hat{c}$  represents the hard-coded constraint on expected cumulative cost (ECC).

By contrast, in the **constrained action-based RL (CARL)** problem, the agent makes decisions to maximize ECR while obeying specific action-based constraints. It's worthwhile to mention that action-based constraints are different from ECC. For example, the action-based constraints in our application limit the total number of times that PS and WE can be selected in each level. Therefore, rather than defining a cost function for each pair of state and action in constrained RL scenarios,

we formalize the CARL problem as:

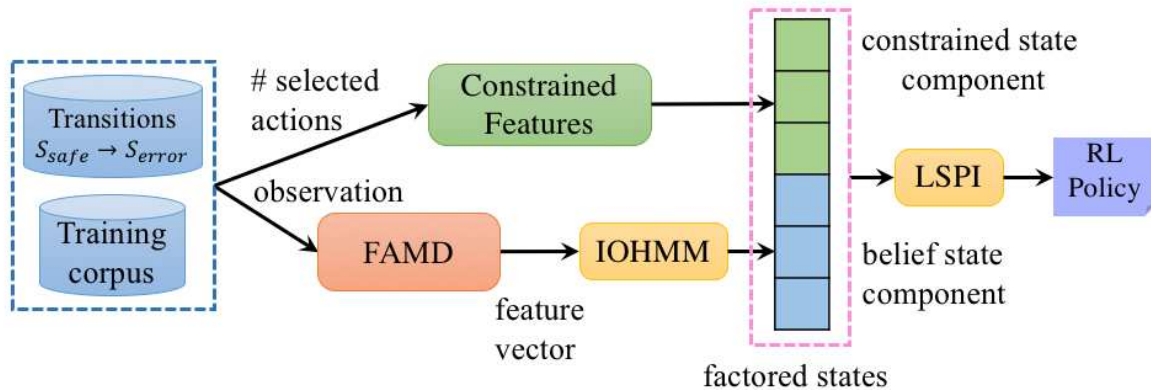
$$\max_{\pi} E_{\pi} \left[ \sum_{t=0}^L \gamma^t R(s_t, a_t) \right] \quad (5.3)$$

$$\text{s.t. } 0 < C_{\pi}(a) = \left[ \sum_{t=0}^L \mathbb{I}(a_t^{\pi} = a) \right] < \widehat{c}_a, \forall a \in A \quad (5.4)$$

where formula (5.4) describes the action-based constraints. More specifically,  $A$  is the set of all possible actions and  $L$  is the length of a trajectory;  $\widehat{c}_a$  and 0 denote the upper and lower bounds on the number of times that action  $a$  can be selected;  $a_t^{\pi}$  indicates that action  $a$  is selected by policy  $\pi$  at time step  $t$ , and  $\mathbb{I}(\cdot)$  is the indicator function in that it would return 1 if the expression in  $(\cdot)$  is true and 0 otherwise.  $C_{\pi}(a)$  denotes the total cost (5.4) of the action-based constraints and only depends on the actions. For our application, if the action-based constraints are active at time  $t$ , then state  $s_t$  is treated as a terminal state, where the agent cannot take any more actions.

### 5.3 Constrained Action-based POMDP (CAPOMDP)

As an extension of the POMDP framework, the CAPOMDP framework modifies the state representation and the reward function to incorporate action-based constraints and transfers the constrained action-based RL problem into a general RL problem. Figure 5.1 presents the general process of inducing the CAPOMDP policy. Compared with the POMDP policy induction process in Figure 4.1, the CAPOMDP procedure contains three same components including feature transformation using FAMD approach, belief state estimation using IOHMM and policy induction using LSPI with the factored states as inputs, and two additional components: 1) factored state representation; 2) strong negative reward for the transition from a safe state to an error state.



**Figure 5.1** The general process of the CAPOMDP policy induction

### 5.3.1 Factored State Representation

The factored state representation is constructed by concatenating the belief state with the constrained state. Specifically, the belief state is defined in the POMDP framework. The constrained state at time step  $t$  is defined as  $[c_1^t, c_2^t, \dots, c_{|A|}^t]$ , where each element is a constrained feature which counts the total number of times that the action was chosen up to the present time point. If an action  $a$  is selected at a particular time step, the value of the corresponding constrained feature  $c_a$  is incremented by 1. Thus, we can estimate  $c_a^t$  efficiently as:

$$c_a^t = \begin{cases} c_a^{t-1} + 1 & a^t = a \\ c_a^{t-1} & \text{else} \end{cases} \quad (5.5)$$

Consequently, the factored state at time  $t$  is represented as  $S^t = [b_1^t, b_2^t, \dots, b_K^t, c_1^t, c_2^t, \dots, c_{|A|}^t]$ . In other words, the factored state contains two independent components: the belief state and the constrained state. The former is used to model the learning process, while the latter only tracks the status of the actions and whether the selection of the action triggers the constraints. Therefore, the factored state transition can be decomposed into separate estimates of the transition for the belief state component via function (4.1) and the transition for the constrained state component via function (5.5).

Furthermore, we designate a factored state as the *safe* state if all of the elements in its constrained state component satisfy the constraint function (5.4), otherwise it's the *error* state. Additionally, error states are treated as one type of terminal state since they are disallowed in the system, while safe states permit actions to be taken which can transit to other states.

### 5.3.2 Reward Function

Since the basic ITS prohibits any appearance of an error state, we need to assign a strong negative reward for any transition from a safe state to an error state and treat error states as terminal states. We still retain the original reward for transitions between safe states in the training corpus since these transitions impose no additional cost. We therefore define the new constrained reward function as:

$$R_c(s_t, a_t) = \begin{cases} R(s_t, a_t) & s_t \in S_{safe}, s_{t+1} \in S_{safe} \\ -\hat{c} & s_t \in S_{safe}, s_{t+1} \in S_{error} \end{cases} \quad (5.6)$$

Where  $-\hat{c}$  indicates a strong negative value.  $R_c(s_t, a_t)$  represents the reward function with constraints, and  $R(s_t, a_t)$  denotes the real reward in the training corpus. However, our training corpus does not contain error states because the original system has hard-coded rules to avoid them. Thus, we are required to manually add transitions from the safe states to error states with strong negative

rewards in training dataset as shown in function (5.6).

### 5.3.3 Policy Induction

We implement Least Squares Policy Iteration (LSPI) [LP03] and treat the belief state of POMDP or the factored state in CAPOMDP as the input to induce the optimal policy, which consists of two steps: *policy evaluation* and *policy improvement*.

In the *policy evaluation* step, we approximate the Q-function  $Q(s, a)$ , the expected reward of taking action  $a$  at state  $s$ , using a linear model generalized as:

$$Q(s, a) = \sum_{i=0}^{|S|*|A|} w_i \phi_i(s, a) \quad (5.7)$$

Where  $\phi_i(s, a)$  indicates the basic element in state  $s$  associated with the action  $a$ , and  $s$  is a belief state representation in POMDP framework.  $|S|$  and  $|A|$  denote the size of the state set and action set respectively.  $w_i$  is the parameter of the linear model and it also involves a constant item (when  $i = 0$ ). Additionally, we have that the Q-function follows the Bellman equation:

$$Q^\pi = R + \gamma P \Pi_\pi Q^\pi \quad (5.8)$$

By integrating equation (5.7) and (5.8), Least Square Temporal Difference Q learning approach estimates the parameter  $w$  as:

$$\begin{cases} w = H^{-1} f \\ H = \sum_{(s,a,s') \in D} \phi(s, a) [\phi(s, a) - \gamma \phi(s', \pi(s'))]^T \\ f = \sum_{(s,a,s') \in D} \phi(s, a) R(s, a) \end{cases} \quad (5.9)$$

Where  $D$  is the training corpus, and  $\pi(s')$  denotes the action selected by current policy  $\pi$  given a state  $s'$ .  $H$  and  $f$  can be estimated from the training corpus.

In the *policy improvement* step,  $w$  is updated through the gradient decent approach toward to minimize the loss function, then LSPI checks whether  $w$  converges. If  $w$  does not converge, it goes back to the policy estimation step; otherwise, it terminates.

## 5.4 Experiment Setup

### 5.4.1 Procedure & Evaluation

Different from DT in Chapter 3 and 4, the new DT in this chapter organize the problems into seven 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 four 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 present it as PS or WE. The last problem of each level is same to all of students, and is required to be solved as a PS without help of the tutor, and thus functioned as a mini-test for evaluating students’ knowledge on the concepts of that level. In the **post-test** (level 7), all participants also receive the same set of PS problems and their performance in this level is evaluated as the post-test score. In addition, we defined the **Normalized Learning Gain (NLG)** as:

$$NLG = \frac{post - pre}{100 - pre} \quad (5.10)$$

Therefore, we evaluate students performance based on 1) pre- and post-test scores, 2) NLG, 3) time and 4) **Learning Efficiency** ( $LE \propto NLG/Time$ ). 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.

### 5.4.2 Training Corpus

The training corpus was collected from training 570 students in DT in the Fall 2014, 15, and 16 semesters of a Discrete Mathematics course. In these semesters, DT was programmed to make random decisions when selecting PS and WE. Note that, when collecting our original training data, DT had already implemented the action-based constraints requested by the class instructors. The average number of solved problems in the form of both PS and WE is 24.1 (SD=2.59). Furthermore, DT recorded the observation as a set of 133 state features, including 59 discrete and 74 continuous features, for representing the students’ behaviors and learning environment. DT calculated level score based on the last problem in each of training levels 2–6. For simplicity reasons, the range of level scores is normalized to [0, 100]. When inducing RL policies using learning gain as reward, we calculated the difference between the student’s current and prior level scores. If students quit the tutor during the training, we assigned a strong negative reward of -300 on the last problem he/she attempted. When inducing RL policies using time as reward, we used the *negative* log of time as the reward for inducing the policy: that is, when training on DT, the less time a student spent on completing the entire training portion, the better. There is a significant correlation between the *negative* log of times and students’ post-test scores:  $cor = 0.19$ ,  $p = .006$ .

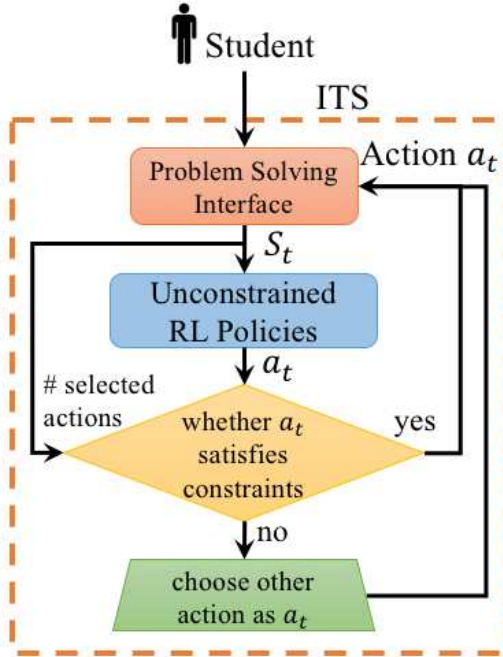


Figure 5.2 Unconstrained policy execution

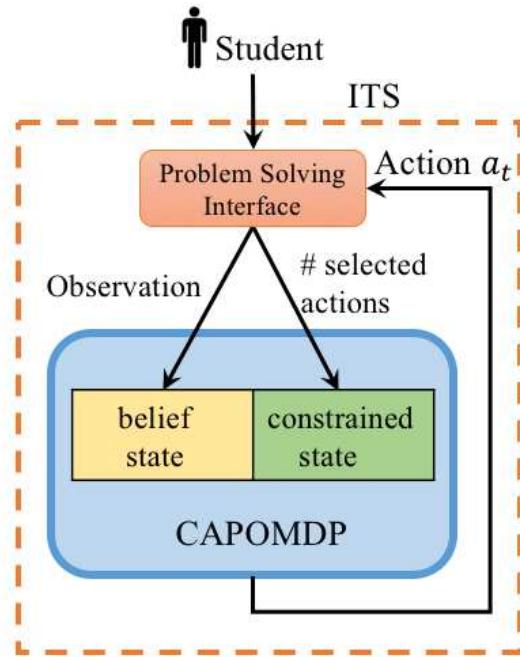


Figure 5.3 CAPOMDP policy execution

### 5.4.3 Policy Execution

Due to the hard-coded constraints, whenever DT makes a tutorial decision, the unconstrained policy execution process will first check whether the selected action violates the hard-coded action-based constraints shown in Figure 5.2. If the action is valid then it will be carried out; otherwise DT uses hard-coded rules to choose an alternative that satisfies the constraints. The unconstrained policies include POMDP, DQN and Random policies, and cannot be fully carried out.

By contrast, the CAPOMDP policy execution is shown in Figure 5.3. Since the action-based constraints are already incorporated into the policy, we expect that the induced CAPOMDP policies will be fully carried out and that the hard-coded rules will not be violated.

## 5.5 Experiment Overview & Research Questions

### 5.5.1 Induced Policies

In this chapter, we induce and employ five policies listed in Table 5.1. It's worthwhile to mention that We only have the deterministic policy execution in the CAPOMDP study, different from MDP and POMDP studies shown in Table 3.1 and Table 4.1 respectively.

Besides two types of CAPOMDP policies, we have three baselines:  $POMDP_{LG}$ ,  $DQN_{LG}$  and Ran-

dom. Specifically,  $\text{POMDP}_{\text{LG}}$  is different from  $\text{FPOMDP-det}$  shown in Table 4.1 (Section 4.3). First,  $\text{POMDP}_{\text{LG}}$  is induced based on a larger training data set comparing with  $\text{POMDP-det}$ . Second,  $\text{POMDP}_{\text{LG}}$  is generated by the LSPI approach, while  $\text{FPOMDP-det}$  is induced through Q-learning.

**Table 5.1** Implemented policies in two experiments for CAPOMDP study

| Policy                         | Framework  | Reward | Feature Transformation | Policy Induction |
|--------------------------------|------------|--------|------------------------|------------------|
| $\text{CAPOMDP}_{\text{LG}}$   | CAPOMDP    | LG     | FAMD                   | LSPI             |
| $\text{CAPOMDP}_{\text{Time}}$ | CAPOMDP    | Time   | FAMD                   | LSPI             |
| $\text{POMDP}_{\text{LG}}$     | POMDP      | LG     | FAMD                   | LSPI             |
| $\text{DQN}_{\text{LG}}$       | model-free | LG     | Standardization        | Deep Q-Network   |

Random: random yet reasonable decision (baseline)

Note: LG denotes learning gain

Besides POMDP and CAPOMDP policies, we apply the Deep Q-Network (DQN) approach [Mni15] to construct a strong baseline policy. DQN uses a neural network to map a state  $s$  to Q-values  $Q(s, a)$  for each action  $a$ . The neural network consists of three Long short-term memory (LSTM) layers [HS97] with 1000 units each, followed by a fully connected layer with 2 output units, one for each action. We trained the network using the DQN algorithm on the training corpus with learning gain as the reward and without considering the action-based constraints. When implementing the DQN policy in the ITS, the selected action for each given state was associated with the highest Q-value, between  $Q(s, PS)$  and  $Q(s, WE)$ .

### 5.5.2 Research Questions

In this chapter, we investigate two primary research questions:

**Q1:** Can the  $\text{CAPOMDP}_{\text{LG}}$  policy outperform the  $\text{POMDP}_{\text{LG}}$  policies ?

**Q2:** Can the  $\text{CAPOMDP}_{\text{LG}}$  policy outperform the  $\text{DQN}_{\text{LG}}$  policies ?

**Q3:** IS the  $\text{CAPOMDP}_{\text{Time}}$  policy outperform three baselines ?

**Q4:** whether using learning gain or time as the reward makes the CAPOMDP framework induce a more effective pedagogical strategy?

To answer the above research questions, We conducted two empirical experiments shown in Table 5.2. Note that We applies four policies including:  $\text{CAPOMDP}_{\text{LG}}$ ,  $\text{CAPOMDP}_{\text{Time}}$ ,  $\text{DQN}_{\text{LG}}$ , and

Random in Experiment 2. However, due to administration errors, very few students were randomly assigned to CAPOMDP<sub>LG</sub> and we mainly focus on comparing the effectiveness of CAPOMDP<sub>Time</sub>, DQN<sub>LG</sub>, and Random in Experiment 2. Furthermore, we conduct a post-hoc comparison across two experiments. Since all students are drawn from the same target population, we combine two CAPOMDP<sub>LG</sub> groups in Experiment 1 and 2 and also integrate the two Random groups.

**Table 5.2** Overview of Experiments

| Experiment          | Implemented Policies                                 | Research Question |    |    |    |
|---------------------|--|-------------------|----|----|----|
|                     |  | Q1                | Q2 | Q3 | Q4 |
| Experiment 1        | CAPOMDP <sub>LG</sub> , POMDP <sub>LG</sub> , Random | ✓                 |    |    |    |
| Experiment 2        | CAPOMDP <sub>Time</sub> , DQN <sub>LG</sub> , Random |                   |    | ✓  |    |
| Post-hoc Comparison | All of implemented policies                          | ✓                 | ✓  | ✓  | ✓  |

Results of Experiment 1, 2 and post-hoc comparison indicate that the Aptitude Treatment Interaction (ATI) effect consistently exists. We defined High and Low groups based upon students' incoming competence, the pre-test score. More specifically, we did a single median split of pre-test scores for all groups across two experiments since all students experienced an identical procedure. As shown in the following sections, this split reasonably reflects the incoming competence of the students.

## 5.6 Experiment 1: Improving Learning Gain Using CAPOMDP

### 5.6.1 Participants & Conditions

190 students enrolled in the Discrete Mathematics course at NCSU in Fall 2017 and were randomly assigned into three conditions: CAPOMDP<sub>LG</sub>, POMDP<sub>LG</sub> and Random. They were further divided into the HighIC and LowIC groups using the median split on the pre-test scores as described above. Table 5.3 presents the group size of each condition. Students are evenly assigned into HighIC and LowIC groups under POMDP<sub>LG</sub> and Random conditions, while more students were assigned to CAPOMDP<sub>LG</sub>-HighIC than CAPOMDP<sub>LG</sub>-LowIC. A  $\chi^2$  test shows no significant difference in the distribution of HighIC vs. LowIC among three conditions:  $\chi^2(2, N = 190) = 2.44, p = 0.29$ .

**Table 5.3** Participants and Conditions in Experiment 1

| Incoming Competence | CAPOMDP <sub>LG</sub> | POMDP <sub>LG</sub> | Random | Total |
|---------------------|-----------------------|---------------------|--------|-------|
| High-IC             | 25                    | 36                  | 37     | 98    |
| Low-IC              | 15                    | 37                  | 40     | 92    |
| Total               | 40                    | 73                  | 77     | 190   |

## 5.6.2 Experiment 1 Results

Table 5.4a presents the mean and standard deviation (SD) for students' learning performance for each condition in Experiment 1, Table 5.4b shows the learning performance for each incoming-competence group, and Table 5.4c shows results of the ANOVA tests. Furthermore, Table 5.5 and 5.6 present students' learning performance of three High-IC and three Low-IC groups respectively.

### 5.6.2.1 Pre-test Score

One-way ANOVA tests show that no significant difference is found on the pre-test score among the three conditions {CAPOMDP<sub>LG</sub>, POMDP<sub>LG</sub>, Random}, but a significant difference exists between HighIC and LowIC groups. As expected, Table 5.4b and 5.4c show that the HighIC group scored significantly higher than the LowIC group on the pre-test score. Additionally, a two-way ANOVA test using condition and incoming competence as factors indicates no interaction effect between condition and incoming competence (Table 5.4c). One-way ANOVA tests also indicate that there is no significant difference either among three HighIC groups (Table 5.5) or among three LowIC groups (Table 5.6).

### 5.6.2.2 Post-test Score

A two-way ANOVA test using condition and incoming competence as two factors shows that there is a significant interaction effect on post-test score, and a one-way ANOVA tests show that there is no significant difference among the three conditions (Table 5.4a). There is a significant difference between HighIC and LowIC, in that HighIC scored significantly higher on the post-test than the LowIC group (Table 5.4b, 5.4c). Additionally, one-way ANOVA tests show that while there is no significant difference among three HighIC groups (Table 5.5), a significant difference was found among three LowIC groups (Table 5.6).

To avoid the impact of multiple significance tests, we report pairwise comparison results from the Tukey HSD test <sup>1</sup>. Specifically, the Tukey HSD test shows that CAPOMDP<sub>LG</sub>-LowIC scored signif-

<sup>1</sup>Post hoc comparison using the Tukey HSD test with Bonferroni correction of p-values

**Table 5.4** Learning Performance in Experiment 1**(a)** Learning Performance for Each Condition (Cond)

| Cond                  | Pre-test     | Post-test    | NLG          | Time       | LE           |
|-----------------------|--------------|--------------|--------------|------------|--------------|
| CAPOMDP <sub>LG</sub> | 42.71(21.47) | 50.66(16.86) | -0.046(0.75) | 2.71(1.25) | 0.031(0.96)  |
| POMDP <sub>LG</sub>   | 36.69(22.98) | 44.12(17.82) | -0.108(0.76) | 3.62(1.57) | -0.052(0.72) |
| Random                | 37.53(23.58) | 45.29(18.46) | -0.006(0.62) | 3.57(1.35) | -0.007(0.61) |

**(b)** Learning Performance for Each Incoming Competence Group (IC)

| IC      | Pre-test     | Post-test    | NLG         | Time       | LE          |
|---------|--------------|--------------|-------------|------------|-------------|
| High-IC | 57.95(10.98) | 50.27(17.68) | -0.41(0.81) | 2.98(1.31) | -0.07(0.41) |
| Low-IC  | 17.36(10.16) | 41.39(17.22) | 0.32(0.28)  | 3.87(1.47) | 0.29(0.33)  |

**(c)** Results of One-way and Two-way ANOVA tests

| Dependent Variable | One-way ANOVA<br>Factor: Cond |                 | One-way ANOVA<br>Factor: IC |                 | Two-way ANOVA<br>Factors: Cond×IC |                 |
|--------------------|-------------------------------|-----------------|-----------------------------|-----------------|-----------------------------------|-----------------|
|                    | <i>F</i> (2, 187)             | <i>p</i> -value | <i>F</i> (1, 188)           | <i>p</i> -value | <i>F</i> (2, 184)                 | <i>p</i> -value |
|                    | Pre-test                      | 0.96            | 0.38                        | 696.2           | 2e-16 ***                         | 0.57            |
| Post-test          | 1.82                          | 0.16            | 12.26                       | 0.0006 ***      | 4.05                              | 0.019*          |
| NLG                | 0.38                          | 0.68            | 68.96                       | 1.94e-14 ***    | 1.26                              | 0.29            |
| Time               | 6.19                          | 0.002 **        | 19.51                       | 1.69e-05 ***    | 1.43                              | 0.24            |
| LE                 | 0.17                          | 0.84            | 40.94                       | 1.22e-09 ***    | 0.024                             | 0.97            |

· marginally significant at  $p < 0.1$ ; \* significant at  $p < 0.05$ ; \*\* significant at  $p < 0.01$ ; \*\*\* significant at  $p < 0.001$

icantly higher than Random-LowIC ( $p = 0.01$ ) and marginally significantly higher than POMDP<sub>LG</sub>-LowIC ( $p = 0.08$ ), and the later two groups performed similarly on post-test score ( $p = 0.59$ ).

### 5.6.2.3 Normalized Learning Gain (NLG)

A two-way ANOVA test using condition and incoming competence as factors shows that there is no significant interaction effect on NLG. One-way ANOVA tests show that there is no significant difference among the three conditions (Table 5.4a, 5.4c). Additionally, Table 5.4b and 5.4c show that the HighIC group achieved significantly lower NLG than the LowIC group, as expected.

Furthermore, one-way ANOVA tests show that while the significant difference on NLG is not found among the three HighIC groups split by condition (Table 5.5), there is a significant difference identified among three LowIC groups (Table 5.6). Particularly, the Tukey HSD test indicates that CAPOMDP<sub>LG</sub>-LowIC achieved the significantly higher NLG than Random-LowIC ( $p = 0.03$ ). No

**Table 5.5** Learning Performance of HighIC Groups in Experiment 1

| Measure   | CAPOMDP <sub>LG</sub><br>-High-IC | POMDP <sub>LG</sub><br>-High-IC | Random<br>-High-IC | One-way ANOVA    |                 |
|-----------|-----------------------------------|---------------------------------|--------------------|------------------|-----------------|
|           |                                   |                                 |                    | <i>F</i> (2, 95) | <i>p</i> -value |
| Pre-test  | 56.78(11.36)                      | 57.30(11.21)                    | 59.37(10.64)       | 0.51             | 0.61            |
| Post-test | 49.73(18.27)                      | 47.13(16.29)                    | 53.70(18.43)       | 1.28             | 0.28            |
| NLG       | -0.36(0.79)                       | -0.55(0.86)                     | -0.31(0.75)        | 0.91             | 0.41            |
| Time      | 2.39(1.07)                        | 3.23(1.41)                      | 3.13(1.24)         | 3.65             | 0.029*          |
| LE        | -0.25(1.11)                       | -0.42(0.83)                     | -0.26(0.78)        | 0.38             | 0.68            |

· marginally significant at  $p < 0.1$ ; \* significant at  $p < 0.05$ ; \*\* significant at  $p < 0.01$ ; \*\*\* significant at  $p < 0.001$

**Table 5.6** Learning Performance of LowIC Groups in Experiment 1

| Measure   | CAPOMDP <sub>LG</sub><br>-Low-IC | POMDP <sub>LG</sub><br>-Low-IC | Random<br>-Low-IC | One-way ANOVA    |                 |
|-----------|----------------------------------|--------------------------------|-------------------|------------------|-----------------|
|           |                                  |                                |                   | <i>F</i> (2, 89) | <i>p</i> -value |
| Pre-test  | 19.26(10.98)                     | 16.63(9.79)                    | 17.33(10.37)      | 0.35             | 0.71            |
| Post-test | 52.22(14.69)                     | 41.19(18.96)                   | 37.53(14.90)      | 4.26             | 0.017 *         |
| NLG       | 0.48(0.21)                       | 0.32(0.32)                     | 0.27(0.25)        | 3.16             | 0.047 *         |
| Time      | 3.24(1.37)                       | 4.00(1.64)                     | 3.98(1.32)        | 1.65             | 0.19            |
| LE        | 0.51(0.32)                       | 0.31(0.32)                     | 0.23(0.24)        | 5.01             | 0.008 **        |

· marginally significant at  $p < 0.1$ ; \* significant at  $p < 0.05$ ; \*\* significant at  $p < 0.01$ ; \*\*\* significant at  $p < 0.001$

significant difference is found for other pairs of LowIC groups.

#### 5.6.2.4 Time

While a two-way ANOVA test using policy and incoming competence as factors indicates that there is no significant interaction effect (Table 5.4c), a one-way ANOVA test shows that a significant difference exists between HighIC and LowIC in that the former spent significantly less time than the latter (Table 5.4b, 5.4c). Similarly, one-way ANOVA tests show that there is a significant difference among three conditions (Table 5.4a, 5.4c). The Tukey HSD test and Table 5.4a indicates that CAPOMDP<sub>LG</sub> spent significantly less time than both POMDP<sub>LG</sub> ( $p = 0.003$ ) and Random ( $p = 0.005$ ) regardless of incoming competence.

When testing for differences on time spent, no significant difference is found among three LowIC groups (Table 5.6), but there is a significant difference among HighIC groups (Table 5.5). The Tukey HSD test indicates that CAPOMDP<sub>LG</sub>-HighIC spent significantly less time than Random-High-IC ( $p = 0.03$ ), and marginally significantly less time than POMDP<sub>LG</sub>-HighIC ( $p = 0.08$ ), and

the latter two groups spent similar time on the tutor.

### 5.6.2.5 Learning Efficiency (LE)

A two-way ANOVA test using policy and incoming competence as factors shows no significant interaction effect on learning efficiency (LE) ( $LE \propto NLG/Time$ ). A one-way ANOVA test also shows no significant difference among the three conditions (Table 5.4a,5.4c), but a significant difference is found between the groups in that the HighIC group achieved significantly lower learning efficiency (LE) than the LowIC group (Table 5.4b, 5.4c). This is interesting, since the HighIC group achieved lower normalized learning gains but also spent less time on the tutor.

Despite this, one-way ANOVA tests show that while no significant difference is found among the three HighIC groups (Table 5.5), a significant difference exists among the three LowIC groups (Table 5.6). Particularly, The Tukey HSD test shows that CAPOMDP<sub>LG</sub>-LowIC achieved significantly higher LE than Random-LowIC ( $p = 0.005$ ) and marginally significantly higher LE than POMDP<sub>LG</sub>-LowIC ( $p = 0.06$ ). No significant difference is found between POMDP<sub>LG</sub>-LowIC and Random-LowIC.

## 5.6.3 Experiment 1 Conclusion

To summarize, we find significant differences between HighIC and LowIC groups: the LowIC group has significantly higher normalized learning gains (NLG), spends significantly longer time in the tutor, and achieves significantly higher learning efficiency (LE) than the HighIC group. More importantly, Experiment 1 exhibits an ATI effect: while no significant difference on learning performance, except time, is found among three HighIC groups, significant differences are found among three LowIC groups on post-test score, normalized learning gains (NLG), and learning efficiency (LE). In short, it seems that students' learning performance is not impacted by the induced pedagogical strategies for the high incoming competence (HighIC) students. However, for the low incoming competence (LowIC) students, the CAPOMDP<sub>LG</sub> policy significantly benefits them more than DQN<sub>LG</sub> and Random policies on the post-test, normalized learning gains (NLG), and learning efficiency (LE).

## 5.7 Experiment 2: Reducing Total Time Using CAPOMDP

### 5.7.1 Participants & Conditions

139 students, enrolled in the Discrete Mathematics course at NCSU in Spring 2018, were randomly assigned into four conditions: CAPOMDP<sub>LG</sub>, CAPOMDP<sub>Time</sub>, DQN<sub>LG</sub> and Random. Furthermore, students were divided by incoming competence into the HighIC and LowIC groups using median split on the pre-test scores where the median value is the same as in Experiment 1. Combining con-

dition and incoming competence, we have a total of eight groups shown in Table 5.7. While it seems that HighIC vs. LowIC is imbalanced, a  $\chi^2$  test shows no significant difference in the distribution of HighIC vs. LowIC among the three conditions:  $\chi^2(2, N = 139) = 0.12, p = 0.94$ . Note that since only 12 students were assigned to CAPOMDP<sub>LG</sub> due to administration errors, the CAPOMDP<sub>LG</sub> condition is excluded in the statistical analysis for Experiment 2, but will be included in the Post-hoc comparison.

**Table 5.7** Participants and Conditions in Experiment 2

| Incoming Competence | CAPOMDP <sub>LG</sub> | CAPOMDP <sub>Time</sub> | DQN <sub>LG</sub> | Random | Total |
|---------------------|-----------------------|-------------------------|-------------------|--------|-------|
| HighIC              | 9                     | 34                      | 21                | 26     | 90    |
| LowIC               | 3                     | 18                      | 13                | 15     | 49    |
| Total               | 12                    | 52                      | 34                | 41     | 139   |

## 5.7.2 Experiment 2 Result

Table 5.8a presents the mean and standard deviation (SD) for students' learning performance for each condition in Experiment 2, Table 5.8b shows the learning performance for each incoming competence group, and Table 5.8c shows the results of the ANOVA tests. Furthermore, Table 5.9 and 5.10 present students' learning performance for the three HighIC groups and the three LowIC groups respectively.

### 5.7.2.1 Pre-test score

One-way ANOVA tests show that no significant difference is found on the pre-test score among the four conditions {CAPOMDP<sub>Time</sub>, DQN<sub>LG</sub>, Random}, but that a significant difference exists between the two incoming competence groups {HighIC, LowIC}. As expected, Table 5.8b and 5.8c show that the HighIC group scored significantly higher than their LowIC peers on the pre-test score. Additionally, a two-way ANOVA test using condition and incoming competence as factors indicates no interaction effect (Table 5.8c). One-way ANOVA tests also indicate that there is no significant difference either among three HighIC groups (Table 5.9) or among three LowIC groups (Table 5.10).

### 5.7.2.2 Post-test score

A two-way ANOVA test using condition and incoming competence as factors shows that there is no significant interaction effect on post-test score (Table 5.8c). One-way ANOVA tests show that

**Table 5.8** Learning Performance in Experiment 2**(a)** Learning Performance for Each Condition (Cond)

| Cond                    | Pre-test     | Post-test    | NLG         | Time       | LE           |
|-------------------------|--------------|--------------|-------------|------------|--------------|
| CAPOMDP <sub>Time</sub> | 44.95(19.28) | 48.40(15.94) | -0.12(0.68) | 3.08(1.47) | -0.004(0.71) |
| DQN <sub>LG</sub>       | 42.43(21.47) | 48.83(16.43) | 0.01(0.51)  | 3.52(1.51) | 0.11(0.59)   |
| Random                  | 44.06(20.14) | 53.75(17.51) | 0.10(0.58)  | 2.90(1.33) | 0.13(0.71)   |

**(b)** Learning Performance for Each Incoming Competence Group (HighIC and LowIC)

| IC     | Pre-test     | Post-test    | NLG         | Time       | LE          |
|--------|--------------|--------------|-------------|------------|-------------|
| HighIC | 56.96(11.17) | 53.63(16.79) | -0.21(0.67) | 2.87(1.47) | -0.06(0.79) |
| LowIC  | 21.16(8.11)  | 44.28(14.73) | 0.34(0.23)  | 3.61(1.29) | 0.31(0.28)  |

**(c)** Results of One-way and Two-way ANOVA tests

| Dependent Variable | One-way ANOVA     |                 | One-way ANOVA     |                 | Two-way ANOVA     |                 |
|--------------------|-------------------|-----------------|-------------------|-----------------|-------------------|-----------------|
|                    | Factor: Cond      |                 | Factor: IC        |                 | Factors: Cond×IC  |                 |
|                    | <i>F</i> (2, 124) | <i>p</i> -value | <i>F</i> (1, 125) | <i>p</i> -value | <i>F</i> (2, 121) | <i>p</i> -value |
| Pre-test           | 0.16              | 0.85            | 362.8             | 2e-16 ***       | 0.38              | 0.68            |
| Post-test          | 1.36              | 0.26            | 9.92              | 0.002 **        | 0.81              | 0.45            |
| NLG                | 1.61              | 0.20            | 28.83             | 3.71e-07 ***    | 0.88              | 0.417           |
| Time               | 1.80              | 0.17            | 8.1               | 0.005 **        | 4.15              | 0.018 *         |
| LE                 | 0.58              | 0.56            | 9.44              | 0.003 **        | 1.22              | 0.29            |

· marginally significant at  $p < 0.1$ ; \* significant at  $p < 0.05$ ; \*\* significant at  $p < 0.01$ ; \*\*\* significant at  $p < 0.001$

there is no significant difference among three conditions, but a significant difference exists between HighIC and LowIC in that the HighIC students scored the significantly higher on the post-test than the LowIC students (Table 5.8b, 5.8c). Additionally, one-way ANOVA tests show that there is no significant difference either among the three HighIC groups (Table 5.9) or among the three LowIC groups (Table 5.10).

### 5.7.2.3 Normalized Learning Gain (NLG)

Similarly, a two-way ANOVA test using condition and incoming competence as factors shows that there is no significant interaction effect on normalized learning gain (NLG) (Table 5.8c). One-way ANOVA tests show that there is no significant difference among the three conditions, but there is a significant difference between HighIC and LowIC (Table 5.8c) on NLG. As we expected, Table 5.8b shows that the HighIC group achieved the significantly lower NLG than their LowIC peers.

**Table 5.9** Learning Performance of HighIC Groups by condition in Experiment 2

| Measure   | CAPOMDP <sub>Time</sub><br>-HighIC | DQN <sub>LG</sub><br>-HighIC | Random<br>-HighIC | One-way ANOVA    |                 |
|-----------|------------------------------------|------------------------------|-------------------|------------------|-----------------|
|           |                                    |                              |                   | <i>F</i> (2, 78) | <i>p</i> -value |
| Pre-test  | 56.75(11.30)                       | 57.04(11.37)                 | 57.16(11.29)      | 0.01             | 0.99            |
| Post-test | 50.07(16.93)                       | 53.53(16.89)                 | 58.37(15.96)      | 1.84             | 0.16            |
| NLG       | -0.36(0.72)                        | -0.16(0.56)                  | -0.04(0.65)       | 1.79             | 0.17            |
| Time      | 3.12(1.66)                         | 3.13(1.58)                   | 2.34(0.91)        | 2.65             | 0.077           |
| LE        | -0.21(0.76)                        | 0.04(0.74)                   | 0.05(0.87)        | 1.05             | 0.35            |

· marginally significant at  $p < 0.1$ ; \* significant at  $p < 0.05$ ; \*\* significant at  $p < 0.01$ ; \*\*\* significant at  $p < 0.001$

**Table 5.10** Learning Performance of LowIC Groups by condition in Experiment 2

| Measure   | CAPOMDP <sub>Time</sub><br>-LowIC | DQN <sub>LG</sub><br>-LowIC | Random<br>-LowIC | One-way ANOVA    |                 |
|-----------|-----------------------------------|-----------------------------|------------------|------------------|-----------------|
|           |                                   |                             |                  | <i>F</i> (2, 43) | <i>p</i> -value |
| Pre-test  | 22.67(7.98)                       | 18.85(8.73)                 | 21.35(7.78)      | 0.84             | 0.44            |
| Post-test | 45.26(13.79)                      | 41.23(12.86)                | 45.75(17.67)     | 0.38             | 0.68            |
| NLG       | 0.34(0.22)                        | 0.31(0.22)                  | 0.36(0.28)       | 0.15             | 0.86            |
| Time      | 3.00(1.08)                        | 4.15(1.20)                  | 3.88(1.38)       | 3.90             | 0.027*          |
| LE        | 0.39(0.35)                        | 0.23(0.19)                  | 0.29(0.22)       | 1.31             | 0.28            |

· marginally significant at  $p < 0.1$ ; \* significant at  $p < 0.05$ ; \*\* significant at  $p < 0.01$ ; \*\*\* significant at  $p < 0.001$

Additionally, one-way ANOVA tests show that there is no significant difference either among the three HighIC groups (Table 5.9) or among the three LowIC groups (Table 5.10). Much to our surprise, while DQN<sub>LG</sub> is induced using learning gain as a reward, it did not outperform Random. Additionally, CAPOMDP<sub>Time</sub> is induced using time as reward, but result shows that it did not hurt students' normalized learning gains.

#### 5.7.2.4 Time

A two-way ANOVA test using policy and incoming competence as factors indicates that there is a significant interaction effect (Table 5.8c) for Time. While an one-way ANOVA test show that there is no significant difference on Time among conditions, Table 5.8a suggests that CAPOMDP<sub>Time</sub> and Random spent less time than DQN<sub>LG</sub>. As expected, a one-way ANOVA test shows that a significant difference exists between HighIC and LowIC in that the HighIC students spent significantly less time than the LowIC students (Table 5.8b, 5.8c). Additionally, one-way ANOVA tests show that there is no significant difference among the three HighIC groups (Table 5.9), but there is a significant

difference among LowIC groups (Table 5.6). Specifically, pairwise t-tests and Table 5.10 show that CAPOMDP<sub>Time</sub>-LowIC spent significantly less time than either Random-LowIC or DQN<sub>LG</sub>-LowIC:  $p = 0.04$  and  $p = 0.01$  respectively. Although Table 5.10 suggests that Random-LowIC spent less time than DQN<sub>LG</sub>-LowIC, the difference is not significant.

Based on these differences, we conclude that there is an Aptitude Treatment Interaction (ATI) effect on time. Particularly, low incoming competence (LowIC) students following the CAPOMDP<sub>Time</sub> policy spent less time than their peers following Random and DQN<sub>LG</sub> policies, while high incoming competence (HighIC) students spent similar total time and their performance was not affected by policies.

#### 5.7.2.5 Learning Efficiency (LE)

A two-way ANOVA test using policy and incoming competence as factors indicates that there is no significant interaction effect on learning efficiency (LE) (Table 5.8c). One-way ANOVA tests show that while there is no significant difference among conditions (Table 5.8a, 5.8c), a significant difference exists between HighIC and LowIC in that HighIC students achieved the significantly lower LE than LowIC students (Table 5.8c, 5.8c).

Furthermore, one-way ANOVA tests shows that there is no significant difference either among the three HighIC groups (Table 5.9) or among the three LowIC groups (Table 5.10). Although Table 5.10 suggests that CAPOMDP<sub>Time</sub>-LowIC achieved higher LE than DQN<sub>Time</sub>-LowIC and Random<sub>Time</sub>-LowIC, pairwise t tests only find a marginally significant difference between CAPOMDP<sub>Time</sub>-LowIC and DQN<sub>LG</sub>-LowIC ( $p = 0.08$ ).

### 5.7.3 Experiment 2 Conclusion

To summarize, in Experiment 2 we mainly focus on evaluating the effectiveness of the CAPOMDP<sub>Time</sub> policy against that of the DQN<sub>LG</sub> and the Random policies. Similar to Experiment 1, students are split into HighIC vs. LowIC based on their pre-test scores and the same patterns are found between the HighIC and the LowIC groups: the latter had a significantly higher learning gains (NLG), spent significantly longer time on DT, and achieved a significantly higher learning efficiency (LE) than their HighIC peers. More importantly, while Experiment 1 exhibits an ATI effect on learning performance (post-test score, NLG and LE), and Experiment 2 exhibits an ATI effect on time, a significant difference is only found among the three LowIC groups in that the CAPOMDP<sub>Time</sub>-LowIC group spent significantly less time than the two DQN<sub>LG</sub>-LowIC and Random-LowIC groups.

Furthermore, Experiment 2 shows that for the high incoming competence (HighIC) students, it seems that neither their learning performance nor time on task are impacted by the induced pedagogical strategies. On the other hand, for the low incoming competence (LowIC) students, the CAPOMDP<sub>Time</sub> policy seemingly did not hurt their learning performance (post-test, NLG and LE).

Much to our surprise,  $DQN_{LG}$  performed closely to the Random policy. One of the possible reasons is that action-based constraints restrict the empirical effectiveness of the  $DQN_{LG}$ . We will give more analysis results in Log Analysis Section.

## 5.8 Post-hoc Comparisons

### 5.8.1 Participants & Conditions

In both Experiment 1 (Exp1) and Experiment 2 (Exp2), students were drawn from the same target population and all of them were enrolled in the experiments with the same method but in different semesters. By assigning students to each condition randomly, it provides the most rigorous test of our hypotheses. In this section, we conduct a post-hoc comparison across two experiments in the hope that this wider view will shed some light on our main results. Especially while  $CAPOMDP_{LG}$  outperformed Random in Exp1, it is not sure whether the same results would hold for Exp2 since we only had a small number of students assigned to  $CAPOMDP_{LG}$  due to administration errors. Therefore, we combine the two Random groups into a single Random group, and integrate the two  $CAPOMDP_{LG}$  groups into a single  $CAPOMDP_{LG}$  group, and then compare their performance with  $CAPOMDP_{Time}$ ,  $DQN_{LG}$  and  $POMDP_{LG}$ .

**Table 5.11** Participants and Conditions in Post-hoc Comparison

| Incoming Competence | $CAPOMDP_{LG}$ | $CAPOMDP_{Time}$ | $DQN_{LG}$ | $POMDP_{LG}$ | Random | Total |
|---------------------|----------------|------------------|------------|--------------|--------|-------|
| HighIC              | 34             | 34               | 21         | 35           | 63     | 188   |
| LowIC               | 18             | 18               | 13         | 37           | 55     | 141   |
| Total               | 52             | 52               | 34         | 73           | 118    | 329   |

Before combining two Random groups in Exp1 and Exp2, we compare their learning performance. One-way ANOVA tests show that while there is no significant difference between two Random groups on pre-test score:  $F(1, 76) = 2.76, p = 0.1$ , NLG:  $F(1, 76) = 0.43, p = 0.52$ , and LE:  $F(1, 76) = 0.46, p = 0.49$ , there is a significant difference on time:  $F(1, 76) = 4.08, p = .046$ , in that Random in Exp1 spent significantly more time than Random in Exp2 ( $M = 2.90, SD = 1.32$ ), and there is a significant difference on post-test score:  $F(1, 76) = 5.10, p = .027$  in that Random in Exp2 ( $M = 59.25, SD = 20.08$ ) scored a significantly higher post-test than Random in Exp1 ( $M = 48.31, SD = 22.71$ ). In short, Random in Exp2 performed better in post-test and spent less time than Random in Exp1. Therefore, by combining two Random groups, we get a stronger baseline condition

**Table 5.12** Learning Performance in Post-Hoc Comparison**(a)** Learning Performance for Each Condition (Cond)

| Cond                    | Pre-test     | Post-test    | NLG         | Time       | LE           |
|-------------------------|--------------|--------------|-------------|------------|--------------|
| CAPOMDP <sub>Time</sub> | 44.95(19.28) | 48.40(15.94) | -0.12(0.68) | 3.08(1.47) | -0.004(0.71) |
| CAPOMDP <sub>LG</sub>   | 43.97(20.76) | 51.31(15.74) | -0.05(0.72) | 2.98(1.39) | -0.002(0.89) |
| DQN <sub>LG</sub>       | 42.43(21.47) | 48.83(16.43) | 0.01(0.51)  | 3.52(1.51) | 0.11(0.59)   |
| POMDP                   | 36.69(22.98) | 44.12(17.82) | -0.11(0.78) | 3.62(1.57) | -0.05(0.73)  |
| Random                  | 39.80(22.57) | 48.23(18.51) | 0.03(0.61)  | 3.34(1.37) | 0.04(0.65)   |

**(b)** Learning Performance for Each Incoming Competence Group (HighIC and LowIC)

| IC     | Pre-test     | Post-test    | NLG         | Time       | LE          |
|--------|--------------|--------------|-------------|------------|-------------|
| HighIC | 57.46(11.05) | 51.89(17.05) | -0.31(0.26) | 2.98(1.40) | -0.21(0.84) |
| LowIC  | 18.72(9.58)  | 42.56(16.43) | 0.33(0.74)  | 3.78(1.41) | 0.31(0.29)  |

**(c)** Results of One-way and Two-way ANOVA tests

| Dependent Variable | One-way ANOVA     |                 | One-way ANOVA     |                 | Two-way ANOVA      |                 |
|--------------------|-------------------|-----------------|-------------------|-----------------|--------------------|-----------------|
|                    | Factor: Cond      |                 | Factor: IC        |                 | Factors: Cond×Prof |                 |
|                    | <i>F</i> (4, 324) | <i>p</i> -value | <i>F</i> (1, 327) | <i>p</i> -value | <i>F</i> (4, 329)  | <i>p</i> -value |
| Pre-test           | 1.51              | 0.2             | 1107              | 2e-16 ***       | 0.94               | 0.44            |
| Post-test          | 1.42              | 0.23            | 24.86             | 1e-06 ***       | 2.82               | 0.025 *         |
| NLG                | 0.78              | 0.54            | 98.04             | 2e-16 ***       | 1.63               | 0.16            |
| Time               | 2.04              | 0.08            | 25.52             | 7.29e-07 ***    | 1.98               | 0.09            |
| LE                 | 0.39              | 0.82            | 48.15             | 2.13e-11 ***    | 1.67               | 0.16            |

· marginally significant at  $p < 0.1$ ; \* significant at  $p < 0.05$ ; \*\* significant at  $p < 0.01$ ; \*\*\* significant at  $p < 0.001$

than Random in Exp1 alone. All students were divided into HighIC and LowIC using the same median split described in Exp1 and Exp2. Combining policy and incoming competence, we have a total of 10 groups shown in Table 5.11. A  $\chi^2$  test shows no significant difference in the distribution of HighIC vs. LowIC among the three conditions:  $\chi^2(4, N = 329) = 5.68, p = 0.22$ .

### 5.8.2 Post-hoc Comparison Results

Table 5.12a presents the mean and standard deviation (SD) for students' learning performance for each condition in the post-hoc comparison. Table 5.12b shows the learning performance for each incoming competence group, and Table 5.12c shows results of the ANOVA tests. Furthermore, Table 5.13 and 5.14 present students' learning performance for the five HighIC and five LowIC groups

respectively.

**Table 5.13** Learning Performance of HighIC Groups in Post-Hoc Comparison

| Measure   | CAPOMDP <sub>LG</sub><br>-HighIC | CAPOMDP <sub>Time</sub><br>-HighIC | DQN <sub>LG</sub><br>-HighIC | POMDP <sub>LG</sub><br>-HighIC | Random<br>-HighIC | One-way ANOVA     |                 |
|-----------|----------------------------------|------------------------------------|------------------------------|--------------------------------|-------------------|-------------------|-----------------|
|           |                                  |                                    |                              |                                |                   | <i>F</i> (4, 183) | <i>p</i> -value |
| Pre-test  | 56.75(11.30)                     | 56.75(11.30)                       | 57.04(11.37)                 | 57.31(11.21)                   | 58.46(10.88)      | 0.20              | 0.93            |
| Post-test | 50.84(16.58)                     | 50.07(16.93)                       | 53.53(16.89)                 | 47.13(16.29)                   | 55.62(17.47)      | 1.66              | 0.16            |
| NLG       | -0.33(0.74)                      | -0.36(0.72)                        | -0.16(0.56)                  | -0.55(0.86)                    | -0.19(0.72)       | 1.58              | 0.18            |
| Time      | 2.82(1.40)                       | 3.12(1.66)                         | 3.13(1.58)                   | 3.23(1.41)                     | 2.80(1.17)        | 0.80              | 0.52            |
| LE        | -0.26(0.99)                      | -0.21(0.76)                        | 0.04(0.74)                   | -0.42(0.83)                    | -0.13(0.82)       | 1.20              | 0.31            |

· marginally significant at  $p < 0.1$ ; \* significant at  $p < 0.05$ ; \*\* significant at  $p < 0.01$ ; \*\*\* significant at  $p < 0.001$

**Table 5.14** Learning Performance of LowIC Groups in Post-Hoc Comparison

| Measure   | CAPOMDP <sub>LG</sub><br>-LowIC | CAPOMDP <sub>Time</sub><br>-LowIC | DQN <sub>LG</sub><br>-LowIC | POMDP <sub>LG</sub><br>-LowIC | Random<br>-LowIC | One-way ANOVA     |                 |
|-----------|---------------------------------|-----------------------------------|-----------------------------|-------------------------------|------------------|-------------------|-----------------|
|           |                                 |                                   |                             |                               |                  | <i>F</i> (4, 136) | <i>p</i> -value |
| Pre-test  | 19.82(10.05)                    | 22.67(7.98)                       | 18.85(8.73)                 | 16.63(9.78)                   | 18.43(9.83)      | 1.29              | 0.28            |
| Post-test | 52.21(14.42)                    | 45.26(13.79)                      | 41.23(12.86)                | 41.19(18.96)                  | 39.77(15.96)     | 2.23              | 0.07 ·          |
| NLG       | 0.47(0.21)                      | 0.34(0.22)                        | 0.31(0.22)                  | 0.32(0.31)                    | 0.29(0.25)       | 1.64              | 0.17            |
| Time      | 3.27(1.37)                      | 3.00(1.07)                        | 4.15(1.20)                  | 4.01(1.64)                    | 3.96(1.33)       | 2.72              | 0.032 *         |
| LE        | 0.48(0.30)                      | 0.38(0.36)                        | 0.23(0.19)                  | 0.31(0.32)                    | 0.23(0.26)       | 3.03              | 0.023 *         |

· marginally significant at  $p < 0.1$ ; \* significant at  $p < 0.05$ ; \*\* significant at  $p < 0.01$ ; \*\*\* significant at  $p < 0.001$

### 5.8.2.1 Pre-test score

One-way ANOVA tests show that no significant difference is found on the pre-test score among five conditions {CAPOMDP<sub>LG</sub>, CAPOMDP<sub>Time</sub>, DQN<sub>LG</sub>, POMDP<sub>LG</sub>, Random}, but the significant difference exists between two incoming competence {HighIC, LowIC} groups. As expected, Table 5.12b and 5.12c show that the HighIC group scored significantly higher than the LowIC group on the pre-test score. Additionally, a two-way ANOVA test using condition and incoming competence as factors indicates no interaction effect (Table 5.12c). One-way ANOVA tests also indicate that

there is no significant difference either among the five HighIC groups (Table 5.13) or among the five LowIC groups on pre-test score (Table 5.14).

### 5.8.2.2 Post-test score

A two-way ANOVA test using condition and incoming competence as factors shows that there is the significant interaction effect on post-test score (Table 5.12c). One-way ANOVA tests show that there is no significant difference among conditions but a significant difference between HighIC and LowIC in that the HighIC students scored the significantly higher on the post-test than the LowIC students (Table 5.12b, 5.12c).

Additionally, one-way ANOVA tests show that there is no significant difference among the five HighIC groups (Table 5.13), but there exists a marginal significant difference among the five LowIC groups (Table 5.14). Particularly, pairwise t tests indicate that CAPOMDP<sub>LG</sub>-LowIC scored significantly higher on the post-test than DQN<sub>LG</sub>-LowIC, POMDP<sub>LG</sub>-LowIC and Random-LowIC:  $p = 0.03$ ,  $p = 0.02$ , and  $p = 0.004$  respectively, and no significant difference is found among the latter three groups. Although Table 5.14 shows that CAPOMDP<sub>LG</sub>-LowIC has a higher post-test score than CAPOMDP<sub>Time</sub>-LowIC, the difference is not significant.

### 5.8.2.3 Normalized Learning Gain (NLG)

A two-way ANOVA test using condition and incoming competence as factors shows that there is no significant interaction effect on NLG (Table 5.12c). One-way ANOVA tests show that while there is no significant difference in NLG among conditions, a significant difference is found between HighIC and LowIC in that the HighIC students achieved significantly lower NLG than the LowIC students (Table 5.12b, 5.12c). A one-way ANOVA test shows that there is no significant difference among the five HighIC groups. Table 5.13 suggests that DQN<sub>LG</sub>-HighIC achieved the highest NLG among five HighIC groups, pairwise t tests indicate that POMDP<sub>LG</sub>-HighIC had the significantly lower NLG than DQN<sub>LG</sub>-HighIC and Random-HighIC:  $p = 0.044$  and  $p = 0.042$  respectively.

Although a one-way ANOVA test shows that no significant difference on NLG is found among the five LowIC groups (Table 5.14), pairwise t tests find a significant difference for the pairs of the LowIC groups. Particularly, pairwise t tests indicate that CAPOMDP<sub>LG</sub>-LowIC achieved significantly higher NLGs than DQN<sub>LG</sub>-LowIC ( $p = 0.04$ ), POMDP<sub>LG</sub>-LowIC ( $p = 0.03$ ) and Random-LowIC ( $p = 0.004$ ), and had the marginally significantly higher NLG than CAPOMDP<sub>Time</sub>-LowIC ( $p = 0.07$ ). No significant difference is found among CAPOMDP<sub>Time</sub>-LowIC, DQN<sub>LG</sub>-LowIC, POMDP<sub>LG</sub>-LowIC, or Random-LowIC.

Therefore, we conclude that the CAPOMDP<sub>LG</sub> policy outperforms other policies for low incoming competence (LowIC) students and is able to significantly improve their NLG, while the high incoming competence (HighIC) students are not impacted by the policies.

#### 5.8.2.4 Time

A two-way ANOVA test using condition and incoming competence as factors shows that there is a marginal significant interaction effect on time (Table 5.12c). One-way ANOVA tests show that while there is no significant difference among conditions, a significant difference is found between HighIC and LowIC in that the HighIC spent significantly less time than the LowIC (Table 5.12b, 5.12b).

A one-way ANOVA test shows that there is no significant difference among five HighIC incoming competence (HighIC) groups. Although Table 5.13 suggests that CAPOMDP<sub>LG</sub>-HighIC and Random-HighIC spent less time than CAPOMDP<sub>Time</sub>-High, DQN<sub>Time</sub>-HighIC and POMDP<sub>LG</sub>-HighIC, the difference is not significant.

Furthermore, a one-way ANOVA test shows that the significant difference is found among the five LowIC groups (Table 5.14). Specifically, pairwise t tests indicate that CAPOMDP<sub>Time</sub>-LowIC spent significantly less time than DQN<sub>LG</sub>-LowIC ( $p = 0.01$ ), POMDP<sub>LG</sub>-LowIC ( $p = 0.01$ ) and Random-LowIC ( $p = 0.004$ ), and no significant difference is found among the latter three LowIC groups. Although Table 5.14 suggests that CAPOMDP<sub>LG</sub>-LowIC spent less time than the latter three groups, the difference is not significant.

Thus, we conclude that total time for the high incoming competence (HighIC) groups is not impacted by the policies, while the CAPOMDP<sub>Time</sub> policy outperformed other policies and can significantly reduce the total time that low incoming competence (LowIC) students spend in the tutor.

#### 5.8.2.5 Learning Efficiency (LE)

A two-way ANOVA test using condition and incoming competence as factors shows that there is no significant interaction effect on LE (Table 5.12c). One-way ANOVA tests show that while there is no significant difference among conditions (Table 5.12a), a significant difference is found between HighIC and LowIC in that the HighIC students achieved significantly lower learning efficiency (LE) than the LowIC students (Table 5.12b, 5.12c).

A one-way ANOVA test shows that there is no significant difference among the five HighIC groups on learning efficiency (LE). Table 5.13 suggests that DQN<sub>LG</sub>-HighIC achieved the highest LE compared with other HighIC groups, and pairwise t tests show that DQN<sub>LG</sub>-HighIC had the significantly higher LE than POMDP<sub>LG</sub>-HighIC  $p = 0.03$ .

Furthermore, a one-way ANOVA test shows a significant difference among the five LowIC groups (Table 5.14). Specifically, pairwise t tests indicates that CAPOMDP<sub>LG</sub>-LowIC achieved significantly higher LE than either DQN<sub>LG</sub>-LowIC ( $p = 0.007$ ) or Random-LowIC ( $p = 0.005$ ). No significant difference is found among CAPOMDP<sub>Time</sub>-LowIC, POMDP<sub>LG</sub>-LowIC, DQN<sub>LG</sub>-LowIC and Random-LowIC, although Table 5.14 suggests that CAPOMDP<sub>Time</sub>-LowIC had the higher LE than the latter three LowIC groups.

### 5.8.3 Conclusion of Post-hoc Comparisons

We compare the effectiveness of five policies, including  $CAPOMDP_{LG}$ ,  $CAPOMDP_{Time}$ ,  $POMDP_{LG}$ ,  $DQN_{LG}$  and Random. Students are split into high and low incoming competence (HighIC and LowIC) groups based on their pre-test scores and the same patterns are found between HighIC and LowIC groups: the LowIC students had a significantly higher learning gains (NLG) and efficiency (LE), and spent significantly longer time in the tutor than students with high incoming competence. Furthermore, we find the Aptitude Treatment Interaction effect on students' learning performance exists, and identify different patterns for high and low incoming competence students.

For *high incoming competence* (HighIC) students, their learning outcomes are generally not sensitive to the policies since no significant difference is found among them following different policies. Because of their high pre-test scores, the HighIC students have limited learning gains and efficiency that policies can improve. However, one exception is that  $POMDP_{LG}$  hurts the HighIC students' learning performance since the  $POMDP_{LG}$ -HighIC group had the lowest NLG and LE among five HighIC groups.

For *low incoming competence* (LowIC) students, their performance is sensitive to the policies. Specifically, the  $CAPOMDP_{LG}$  policy is able to improve their post-test score, learning gain, and learning efficiency. The  $CAPOMDP_{Time}$  policy can also reduce the time that the LowIC students spent in the tutor. Much to our surprise,  $POMDP_{LG}$  and  $DQN_{LG}$  didn't have a positive impact on the LowIC students' learning outcomes, performing similarly to the Random policy.

## 5.9 Log Analysis

### 5.9.1 Impact of Action-based Constraints on Learning Performance

Recall that the unconstrained policies including  $POMDP_{LG}$  and  $DQN_{LG}$  performed similarly to the Random policy, and cannot improve students' learning performance. What's worse,  $POMDP_{LG}$  even hurt the high incoming competence (HighIC) students' learning. One of possible reasons is that action-based constraints restrict the effectiveness of unconstrained policies. Both  $CAPOMDP_{LG}$  and  $CAPOMDP_{Time}$  can be fully carried out, while  $POMDP_{LG}$ ,  $DQN_{LG}$  and Random can only be partially carried out. We hypothesize that the policies with the high carry-out ratio outperforms than policies with low ratios in terms of students' learning performance.

The  $POMDP_{LG}$  Policy is carried out 73.9% ( $SD = 9.71\%$ ) of the time. Students under the  $POMDP_{LG}$  condition are divided into the High ( $N = 44$ ) and Low ( $N = 29$ ) Carry-out groups by splitting on carry-out percentage with an artificially-set value 0.75, and their learning performances are shown in Table 5.15. Note that, while the means indicate that post-test scores were higher than mean (non-isomorphic) pre-test scores, a large number of students in both groups had negative NLGs, bringing the overall mean down. Table 5.15a indicates that the High and Low Carry-out groups have

**Table 5.15** Learning Performance of High vs. Low Carry-out**(a)** High vs. Low Carry-out under the POMDP<sub>LG</sub> condition

| Measure    | High         | Low          | One-way ANOVA |            |
|------------|--------------|--------------|---------------|------------|
|            |              |              | $F(1, 71)$    | $p$ -value |
| Pre-test   | 40.07(21.62) | 31.55(24.39) | 2.45          | 0.12       |
| Post-test  | 46.71(16.29) | 40.19(19.55) | 2.39          | 0.13       |
| NLG        | -0.10(0.81)  | -0.12(0.75)  | 0.01          | 0.94       |
| Time(hour) | 3.34(1.54)   | 4.05(1.53)   | 3.74          | 0.06       |
| LE         | -0.01(0.74)  | -0.12(0.71)  | 0.39          | 0.53       |

**(b)** High vs. Low Carry-out for the High incoming competence students

| Measure   | High         | Low         | One-way ANOVA |            |
|-----------|--------------|-------------|---------------|------------|
|           |              |             | $F(1, 34)$    | $p$ -value |
| Pre-test  | 55.02(11.41) | 64.15(7.47) | 4.99          | 0.03 *     |
| Post-test | 48.44(18.15) | 43.19(8.17) | 0.69          | 0.41       |
| NLG       | -0.39(0.89)  | -1.01(0.57) | 3.64          | 0.06       |
| Time      | 3.13(1.43)   | 3.54(1.38)  | 0.56          | 0.45       |
| LE        | -0.26(0.82)  | -0.91(0.72) | 4.46          | 0.04 *     |

**(c)** High vs. Low Carry-out for the Low incoming competence students

| Measure   | High         | Low          | One-way ANOVA |            |
|-----------|--------------|--------------|---------------|------------|
|           |              |              | $F(1, 35)$    | $p$ -value |
| Pre-test  | 16.34(8.31)  | 16.88(11.09) | 0.03          | 0.87       |
| Post-test | 43.97(12.86) | 38.84(23.01) | 0.67          | 0.42       |
| NLG       | 0.36(0.23)   | 0.28(0.37)   | 0.58          | 0.45       |
| Time      | 3.67(1.68)   | 4.28(1.58)   | 1.29          | 0.26       |
| LE        | 0.38(0.33)   | 0.24(0.31)   | 2.04          | 0.16       |

the similar learning outcomes. A one-way ANOVA test shows that the High Carry-out group spent marginally significantly less time than the Low Carry-out group, but there were no differences between groups on pre-test, post-test, normalized learning gain (NLG), or learning efficiency (LE).

Table 5.15b presents the learning performance of High and Low carry-out groups for the high incoming competence (HighIC) students. Note that, the post-test scores in this group are lower than the pre-test scores. This is because the tests are not isomorphic; the pre-test is performance on level 1 while the post-test is performance on level 7, which is much more difficult. Much to our surprise, the HighIC, Low Carry-out group had significantly higher Pre-test scores than their HighIC, High Carry-out peers, which indicates that the action-based constraints occurred more in the POMDP<sub>LG</sub> execution where students have higher Pre-test scores.

Additionally, one-way ANOVA tests show that for the HighIC students, the High carry-out group achieved significantly higher learning efficiency (LE) and marginally significantly higher normalized learning gains (NLG) than the Low carry-out group. Although results indicate that the HighIC students with high carry-out in the POMDP<sub>LG</sub> condition performed better than the HighIC students with Low Carry-out, such differences may be caused by the different incoming competence (pre-

test score). In order to obtain a robust result, we run the one-way ANCOVA tests using Carry-out ratio and pre-test score as a factor and a covariate, respectively, on NLG and LE. One-way ANCOVA tests indicate that High carry-out group achieved a marginally significantly higher NLG than their Low Carry-out peers:  $F(1, 70) = 3.34, p = 0.07$ , and had significantly higher LE than their Low Carry-out peers:  $F(1, 70) = 5.08, p = 0.03$ . Therefore, we conclude that POMDP<sub>LG</sub> with High Carry-out execution had the positive impact on high incoming competence (HighIC) students' learning performance.

Table 5.15c shows the learning performance of High and Low Carry-out groups for the low incoming competence (LowIC) students. Although there is no significant difference between LowIC students in High and Low Carry-out groups, results suggest that the High Carry-out group performed better in terms of Post-test score, NLG, and LE, and spent less Time than their Low Carry-out group.

On the contrary, the DQN<sub>LG</sub> and Random Policies are carried out 56.59% ( $SD = 6.98\%$ ) and 52.96% ( $SD = 8.68\%$ ) of the time respectively. However, when splitting students into High and Low carry-out groups, we did not find any significant difference on students' learning performance between High and Low Carry-out under either DQN<sub>LG</sub> or Random condition. One of possible reasons is that both DQN<sub>LG</sub> and Random are ineffective policies, and action-based constraints can interrupt the policy execution but can't change the performance of ineffective policies.

## 5.9.2 Relation between Behaviors to Learning Outcomes

In this section, we mainly focus on the relation between behavior variables with learning outcomes in order to identify which type of behavior benefits students' learning performance. Specifically, We define and list the examined behaviors, and their descriptive statistics for the whole population, in Table 5.16. Note that we only consider the PS for practice problems in level 2-6, excluding the last problem of each level. We also exclude all problems on level 1 and 7 where the set of problems is same for all of the students. Remember that in the system, constraints dictate that there must be one WE and one PS before the final problem on a level, and that the final must be PS. We consider the last problem to be fixed, so we do not examine that problem as a PS in the sequence analyses here. In the remainder of this section, we examine only "practice problems" before the last problem on a level, and each level from 2-6 has either 2 or 3 practice problems.

### 5.9.2.1 Whole Population

Table 5.17a presents the results of Pearson correlation test for each pair of behavior variables and learning outcomes for the whole population. Specifically, *Interactions* has a significant negative correlation with Post-test score, NLG, and LE, and the a significant, positive correlation with Time. This finding suggests that students with the lower *Interactions* achieved the better learning per-

**Table 5.16** Definition, Mean and standard deviation (SD) of Behavior Variables

| Variable           | Definition   | Mean(SD)     |
|--------------------|--|--------------|
| <b>Interaction</b> | Number of actions that students take during training phase | 2351(1214)   |
| <b>HintCount</b>   | Number of hints that students required                     | 11.09(18.01) |
| <b>PSCount</b>     | Number of problems solved (PS) by students                 | 11.8(1.76)   |
| <b>WECount</b>     | Number of worked examples (WE) that students viewed        | 6.57(1.35)   |
| <b>PS-PS</b>       | Times PS is followed by another PS                         | 5.75(4.71)   |
| <b>PS-WE</b>       | Times PS is followed by WE                                 | 2.36(1.57)   |
| <b>WE-PS</b>       | Times WE is followed by PS                                 | 2.72(1.65)   |
| <b>WE-WE</b>       | Times WE is followed by another WE                         | 1.25(1.26)   |

Note: PS in the above variables are in level 2-6 without the last problem of each level

formance. Similarly, students with lower *HintCount* have higher Post-test score and lower Time. Although *WECount* has no significant correlation with learning outcomes, *PSCount* has a significant positive correlation with Time, which indicates that students, who did more problem solving, would spend more time in the tutor as expected. The PS-WE and the WE-WE pairs are not significantly correlated with learning outcomes.

As expected, the PS-PS pair has a significant positive correlation with Time. Due to constraints, there are only two patterns for PS-PS practice problem pairs: WE-PS-PS and PS-PS-WE. Pearson's correlation tests find that WE-PS-PS has a significant positive correlation with Time  $r = 0.12, p = 0.03$ , but that PS-PS-WE does not. This indicates that, when students have a PS-PS pattern, assigning one worked example WE to students at the beginning of a level increases their total time in the tutor, but assigning a WE after two consecutive PS problems does not have an impact on total time.

Much to our surprise, the number of WE-PS pairs has a significant negative correlation with Post-test score, and a significant positive correlation with Time. This indicates that students who solved more WE-PS pairs would take longer in the tutor, and have a lower post-test score. Our constraints meant that there are three patterns: WE-PS-WE, WE-PS-PS in a level with three practice problems, and one pattern: WE-PS in a level with two practice problems. WE-PS-PS is significantly correlated with Time as mentioned above. Additionally, Pearson's correlation tests also show that WE-PS-WE has a marginally significant positive correlation with Time:  $r = 0.10, p = 0.06$ ; WE-PS-WE also has a significant negative correlation with Post-test Score:  $r = -0.12, p = 0.03$ . Therefore, when controlling for *WECount*, alternating WE and PS in this tutor may not be the best policy overall.

**Table 5.17** Correlation Tests Between Behavior Variables with Learning Outcomes**(a)** Pearson Correlation Tests For The Whole Population

| Variable     | Post-test Score |                 | NLG      |                 | Time     |                 | LE       |                 |
|--------------|-----------------|-----------------|----------|-----------------|----------|-----------------|----------|-----------------|
|              | <i>r</i>        | <i>p</i> -value | <i>r</i> | <i>p</i> -value | <i>r</i> | <i>p</i> -value | <i>r</i> | <i>p</i> -value |
| Interactions | -0.56           | 2.2e-16 ***     | -0.17    | 0.002 **        | 0.78     | 2.2e-16***      | -0.16    | 0.004**         |
| HintCount    | -0.21           | 0.000***        | -0.01    | 0.81            | 0.39     | 2.0e-13 ***     | -0.006   | 0.90            |
| PSCount      | -0.02           | 0.68            | 0.02     | 0.67            | 0.21     | 0.0001 ***      | -0.004   | 0.94            |
| WECount      | -0.08           | 0.15            | -0.05    | 0.34            | -0.02    | 0.68            | -0.04    | 0.47            |
| PS-PS        | 0.01            | 0.84            | 0.03     | 0.50            | 0.19     | 0.0005 ***      | 0.02     | 0.74            |
| PS-WE        | 0.02            | 0.67            | 0.02     | 0.69            | -0.07    | 0.17            | 0.02     | 0.74            |
| WE-PS        | -0.12           | 0.02 *          | -0.03    | 0.47            | 0.19     | 0.0002 ***      | -0.06    | 0.24            |
| WE-WE        | 0.001           | 0.86            | -0.02    | 0.62            | -0.08    | 0.13            | -0.004   | 0.94            |

**(b)** Pearson Correlation Tests For The High Incoming Competence Students

| Variable     | Post-test Score |                 | NLG      |                 | Time     |                 | LE       |                 |
|--------------|-----------------|-----------------|----------|-----------------|----------|-----------------|----------|-----------------|
|              | <i>r</i>        | <i>p</i> -value | <i>r</i> | <i>p</i> -value | <i>r</i> | <i>p</i> -value | <i>r</i> | <i>p</i> -value |
| Interactions | -0.58           | 2.2e-16 ***     | -0.38    | 5.84e-08 **     | 0.79     | 2.2e-16***      | -0.25    | 0.000 ***       |
| HintCount    | -0.28           | 0.000***        | -0.14    | 0.05 ·          | 0.48     | 2.14e-12 ***    | -0.06    | 0.42            |
| PSCount      | -0.01           | 0.88            | 0.006    | 0.93            | 0.22     | 0.003 **        | 0.01     | 0.86            |
| WECount      | -0.07           | 0.31            | -0.11    | 0.12            | -0.03    | 0.65            | -0.09    | 0.21            |
| PS-PS        | 0.05            | 0.47            | 0.09     | 0.23            | 0.26     | 0.0002 ***      | 0.08     | 0.28            |
| PS-WE        | 0.03            | 0.67            | 0.01     | 0.88            | -0.12    | 0.10            | 0.007    | 0.92            |
| WE-PS        | -0.14           | 0.04 *          | -0.16    | 0.02 *          | 0.19     | 0.008 **        | -0.15    | 0.04 *          |
| WE-WE        | -0.004          | 0.95            | -0.001   | 0.98            | -0.07    | 0.32            | 0.01     | 0.84            |

**(c)** Pearson Correlation Tests For The Low Incoming Competence Students

| Variable     | Post-test Score |                 | NLG      |                 | Time     |                 | LE       |                 |
|--------------|-----------------|-----------------|----------|-----------------|----------|-----------------|----------|-----------------|
|              | <i>r</i>        | <i>p</i> -value | <i>r</i> | <i>p</i> -value | <i>r</i> | <i>p</i> -value | <i>r</i> | <i>p</i> -value |
| Interactions | -0.46           | 8.36e-09 ***    | -0.33    | 6.39e-05 ***    | 0.73     | 2.2e-16***      | -0.52    | 2.8e-11 ***     |
| HintCount    | -0.05           | 0.51            | 0.04     | 0.66            | 0.23     | 0.005 ***       | -0.11    | 0.17            |
| PSCount      | -0.03           | 0.74            | 0.05     | 0.53            | 0.21     | 0.014 *         | -0.12    | 0.16            |
| WECount      | -0.05           | 0.55            | -0.08    | 0.37            | -0.05    | 0.53            | -0.004   | 0.95            |
| PS-PS        | -0.06           | 0.44            | -0.03    | 0.69            | 0.12     | 0.14            | -0.14    | 0.08 ·          |
| PS-WE        | 0.03            | 0.76            | 0.03     | 0.69            | -0.03    | 0.72            | 0.04     | 0.65            |
| WE-PS        | -0.05           | 0.58            | 0.01     | 0.88            | 0.15     | 0.06 ·          | -0.06    | 0.48            |
| WE-WE        | 0.003           | 0.97            | -0.03    | 0.68            | -0.07    | 0.38            | 0.005    | 0.94            |

· marginally significant at  $p < 0.1$ ; \* significant at  $p < 0.05$ ; \*\* significant at  $p < 0.01$ ; \*\*\* significant at  $p < 0.001$

### 5.9.2.2 High Incoming Competence (HighIC) Students

Table 5.17b shows the results of Pearson correlation tests for each pair of behavior variables and learning outcomes for the HighIC students. As described above, similar patterns of the variables including *Interactions*, *HintCount*, *PSCount* and *WECount* are identified for the high incoming competence HighIC students. Much to our surprise, the number of WE-PS pairs has a significant negative correlation with NLG and LE, and a significant positive correlation with Time for HighIC students. This indicates that WE-PS pairs have a negative impact on HighIC students' learning outcomes. Again, since *WECount* had no correlations for the HighIC group, this suggests that an interleaved WE-PS pattern may not be ideal for the HighIC group.

### 5.9.2.3 Low Incoming Competence (LowIC) Students

Table 5.17c present the results of Pearson correlation tests for each pair of behavior variables and learning outcomes for the *low incoming competence* (LowIC) students. Similar patterns of the variables including *Interactions*, *PSCount* and *WECount* are identified for the LowIC students. Different from the above results, *HintCount* is not significantly (negatively) correlated with Post-test score for the low incoming competence students, which indicates that LowIC students with high *HintCount* are still able to achieve high post-test scores, since they can learn from hints. Furthermore, PS-PS pairs has a marginally significant negative correlation with LE, while there is no significant correlation between other pairs of PS and WE with the other learning outcomes. Similarly, we investigate the impact of both WE-PS-PS and PS-PS-WE practice problem patterns on learning outcomes. Pearson's correlation tests indicate that only PS-PS-WE has a significant negative correlation with LE:  $r = -0.18, p = 0.03$ . This shows that assigning a late worked example (WE) is not good for the LowIC group.

## 5.9.3 Impact of Pedagogical Strategy on Behavior

In this section, we mainly identify whether the pedagogical strategy has significant impact on students' behaviors. Table 5.18a, 5.18b and 5.18c present the mean and standard deviation (SD) for each behavior variable for each condition for the whole population, the high and low incoming competence students, respectively.

### 5.9.3.1 Whole Population

In Table 5.18a, one-way ANOVA tests show that there is no significant difference on *Interaction* and *HintCount* among five conditions. Although CAPOMDP<sub>LG</sub> achieved the lowest *Interactions* and *HintCount* and DQN<sub>LG</sub> had the highest *Interactions* and *HintCount* among conditions, the differences are not significant. One-way ANOVA tests indicate that there is a significant difference among

conditions on *PSCount* *WECount*. Specifically, pairwise t tests indicate that  $DQN_{LG}$  solved significantly more problems (PSs) than  $CAPOMDP_{Time}$ ,  $POMDP_{LG}$ , and Random:  $p = 3.6e-07$ ,  $p = 0.001$ ,  $p = 0.02$  respectively. In the other hand,  $CAPOMDP_{Time}$  received significantly more worked examples (WEs) than  $CAPOMDP_{LG}$ ,  $DQN_{LG}$ ,  $POMDP_{LG}$ , and Random:  $p = 2e-16$ ,  $p = 1.5e-05$ ,  $p = 8.5e-06$ ,  $p = 1.3e-14$  respectively.

Furthermore, one-way ANOVA tests show that there are significant differences between ordered pairs of PS and WE, except the PS-PS pair. Particularly, pairwise t tests indicate that  $DQN_{LG}$  had significantly higher PS-PS than  $CAPOMDP_{LG}$  and  $POMDP_{LG}$ :  $p = 0.04$  and  $p = 0.03$  respectively;  $CAPOMDP_{LG}$  had significantly more PS-WE pairs than  $CAPOMDP_{Time}$ ,  $DQN_{LG}$  and  $POMDP_{LG}$ :  $p = 2e-16$ ,  $p = 5.4e-06$ ,  $p = 1.5e-05$  respectively, and  $CAPOMDP_{Time}$  had the lowest number of PS-WE pairs among the five conditions.

$POMDP_{LG}$  had significantly more WE-PS pairs than all of other four conditions. Recall that Table 5.12a shows that  $POMDP_{LG}$  achieved the lowest post-test score among five conditions, but the difference is not significant and Table 5.18a indicates that the number of WE-PS pairs had a significant, negative correlation with Post-test score. Combined, these results suggest that students in the  $POMDP_{LG}$  condition had more WE-PS pairs and this may be why they achieved lower Post-test scores.

### 5.9.3.2 High Incoming Competence (HighIC) Students

In Table 5.18b, one-way ANOVA tests indicate that significant differences were found on the number of *Interactions* among the five conditions for HighIC students. Pairwise t tests show that Random-HighIC had significantly lower *Interactions* than  $CAPOMDP_{Time}$ -HighIC ( $p = 0.04$ ) and marginally significantly lower than  $DQN_{LG}$ -HighIC ( $p = 0.07$ ). No significant difference is found among conditions on *HintCount*.

Moreover, one-way ANOVA tests show a significant difference among conditions on *PSCount* and *WECount*. Pairwise t tests indicate that  $DQN_{LG}$ -HighIC had significantly more problems solved (PSs) than other conditions;  $CAPOMDP_{Time}$ -HighIC had significantly fewer PSs, and significantly higher WEs than other conditions. In addition, there are significant differences among HighIC conditions on the pairs of PS and WE. Specifically, pairwise t tests indicate that  $DQN_{LG}$ -HighIC had significantly more PS-PS pairs than  $CAPOMDP_{LG}$ -HighIC,  $POMDP_{LG}$ -HighIC and Random-HighIC:  $p = 0.001$ ,  $p = 1.7e-05$  and  $p = 0.005$  respectively. This makes sense since  $DQN_{LG}$ -HighIC provided more problem solving making it more likely to have more PS-PS pairs.  $CAPOMDP_{LG}$ -HighIC and Random-HighIC had significantly higher PS-WE than the other three conditions.  $CAPOMDP_{Time}$ -HighIC had significantly higher WE-WE than others, which makes sense given that it provided more worked examples as described above.

Furthermore,  $POMDP_{LG}$ -HighIC had significantly higher WE-PS than other HighIC groups. Re-

**Table 5.18** Behavior Variables by Condition

(a) Behavior Variables by Condition for The Whole Population

| Variable     | CAPOMDP <sub>LG</sub> | CAPOMDP <sub>Time</sub> | DQN <sub>LG</sub> | POMDP <sub>LG</sub> | Random       | One-way ANOVA |             |
|--------------|-----------------------|-------------------------|-------------------|---------------------|--------------|---------------|-------------|
|              |                       |                         |                   |                     |              | F(4, 324)     | p-value     |
| Interactions | <b>2005(971)</b>      | 2399(1305)              | 2711(1438)        | 2491(1212)          | 2293(1179)   | 2.158         | 0.0736 ·    |
| HintCount    | <b>9.38(14.03)</b>    | 9.90(25.06)             | 13.64(24.92)      | 10.89(13.90)        | 11.74(15.96) | 0.38          | 0.82        |
| PSCount      | 6.42(1.59)            | 4.52(1.72)              | <b>6.85(2.39)</b> | 5.30(0.68)          | 6.01(1.32)   | 19.05         | 4.3e-14 *** |
| WECount      | 5.73(0.91)            | <b>7.78(0.88)</b>       | 6.06(1.86)        | 7.05(0.88)          | 6.41(0.98)   | 31.56         | 2e-16 ***   |
| PS-PS        | 0.71(0.96)            | 1.94(0.85)              | <b>2.15(1.52)</b> | 0.37(0.62)          | 1.11(0.96)   | 33.24         | 2e-16 ***   |
| PS-WE        | <b>3.27(1.77)</b>     | 0.27(0.49)              | 2.32(1.17)        | 1.87(0.82)          | 2.98(1.12)   | 64.87         | 2e-16 ***   |
| WE-PS        | 2.44(1.18)            | 2.31(0.96)              | 1.71(0.79)        | <b>4.93(0.25)</b>   | 2.16(1.15)   | 118.9         | 2e-16 ***   |
| WE-WE        | 0.73(0.91)            | <b>2.76(0.87)</b>       | 1.71(1.29)        | 0.19(0.51)          | 1.17(0.94)   | 70.88         | 2e-16 ***   |

(b) Behavior Variables by Condition for High Incoming Competence (HighIC) Students

| Variable     | CAPOMDP <sub>LG</sub><br>-HighIC | CAPOMDP <sub>Time</sub><br>-HighIC | DQN <sub>LG</sub><br>-HighIC | POMDP <sub>LG</sub><br>-HighIC | Random<br>-HighIC | One-way ANOVA |             |
|--------------|----------------------------------|------------------------------------|------------------------------|--------------------------------|-------------------|---------------|-------------|
|              |                                  |                                    |                              |                                |                   | F(4, 183)     | p-value     |
| Interactions | 1996(957)                        | 2448(1541)                         | 2519(1599)                   | 2102(837)                      | <b>1815(990)</b>  | 2.45          | 0.04 *      |
| HintCount    | 9.61(15.03)                      | 8.93(27.28)                        | 8.81(18.43)                  | <b>7.47(8.31)</b>              | 9.26(16.54)       | 0.08          | 0.98        |
| PSCount      | 6.44(1.26)                       | 4.53(1.54)                         | <b>6.95(2.49)</b>            | 5.22(0.64)                     | 5.90(1.42)        | 12.58         | 4.6e-09 *** |
| WECount      | 5.76(0.92)                       | <b>7.74(0.79)</b>                  | 6.19(1.99)                   | 6.69(0.71)                     | 6.39(1.10)        | 15.27         | 8.5e-11 *** |
| PS-PS        | 0.79(1.01)                       | 1.97(0.76)                         | <b>2.14(1.53)</b>            | 0.28(0.56)                     | 1.06(0.95)        | 20.87         | 3.4e-14 *** |
| PS-WE        | <b>3.18(1.84)</b>                | 0.29(0.52)                         | 2.52(1.17)                   | 1.67(0.72)                     | 3.08(1.08)        | 41.28         | 2e-16 ***   |
| WE-PS        | 2.47(0.83)                       | 2.26(0.83)                         | 1.67(0.58)                   | <b>4.95(0.24)</b>              | 1.95(1.07)        | 90.62         | 2e-16 ***   |
| WE-WE        | 0.76(0.92)                       | <b>2.74(0.79)</b>                  | 1.71(1.45)                   | 0.05(0.23)                     | 1.21(1.05)        | 39.44         | 2e-16 ***   |

(c) Behavior Variables by Condition for Low Incoming Competence (LowIC) Students

| Variable     | CAPOMDP <sub>LG</sub><br>-LowIC | CAPOMDP <sub>Time</sub><br>-LowIC | DQN <sub>LG</sub><br>-LowIC | POMDP <sub>LG</sub><br>-LowIC | Random<br>-LowIC | One-way ANOVA |             |
|--------------|---------------------------------|-----------------------------------|-----------------------------|-------------------------------|------------------|---------------|-------------|
|              |                                 |                                   |                             |                               |                  | F(4, 136)     | p-value     |
| Interactions | <b>2023(1024)</b>               | 2312(750)                         | 3022(1121)                  | 2880(1404)                    | 2840(1148)       | 2.68          | 0.03 *      |
| HintCount    | <b>8.94(12.31)</b>              | 11.61(21.18)                      | 21.46(32.16)                | 14.30(17.28)                  | 14.56(14.93)     | 0.98          | 0.42        |
| PSCount      | 6.39(2.12)                      | 4.50(2.06)                        | <b>6.69(2.29)</b>           | 5.38(0.72)                    | 6.12(1.20)       | 6.55          | 7.4e-05 *** |
| WECount      | 5.66(0.91)                      | <b>7.89(1.02)</b>                 | 5.84(1.67)                  | 7.41(0.90)                    | 6.43(0.83)       | 17.84         | 8.6e-12 *** |
| PS-PS        | 0.56(0.86)                      | 1.89(1.02)                        | <b>2.15(1.57)</b>           | 0.46(0.65)                    | 1.16(0.98)       | 12.44         | 1.2e-08 *** |
| PS-WE        | <b>3.44(1.65)</b>               | 0.22(0.47)                        | 2.00(1.15)                  | 2.08(0.86)                    | 2.87(1.17)       | 23.43         | 1.6e-14 *** |
| WE-PS        | 2.38(1.68)                      | 2.38(1.19)                        | 1.76(1.09)                  | <b>4.92(0.27)</b>             | 2.40(1.21)       | 38.1          | 2e-16 ***   |
| WE-WE        | 0.67(0.91)                      | <b>2.89(1.02)</b>                 | 1.69(1.03)                  | 0.32(0.67)                    | 1.13(0.79)       | 31.46         | 2e-16 ***   |

· marginally significant at  $p < 0.1$ ; \* significant at  $p < 0.05$ ; \*\* significant at  $p < 0.01$ ; \*\*\* significant at  $p < 0.001$

call that Table 5.13 indicates POMDP<sub>LG</sub>-HighIC had significantly lower NLG and LE than DQN<sub>LG</sub>-HighIC, and achieved significantly lower NLG than Random-HighIC. Table 5.17b indicates that the number of WE-PS pairs has a marginally significant, negative correlation with NLG and LE. Therefore, results indicate that having more WE-PS pairs, has a negative impact on HighIC students' learning outcomes – suggesting that the tutor should provide only one WE or two WEs in a row, rather than alternating WE and PS for HighIC students. Remember that having more WE-PS-WE sequences are significantly related to higher Time and lower Post-test scores for the whole population.

### 5.9.3.3 Low Incoming Competence (LowIC) Students

In Table 5.18c, patterns similar to those described above are identified among the LowIC students on *Interactions*, *PSCount*, *WECount*. Although DQN<sub>LG</sub>-LowIC had the highest *HintCount* among five LowIC conditions, the differences between DQN<sub>LG</sub>-LowIC and other LowIC conditions are not significant. Similarly, there are significant differences among LowIC conditions on the pairs of PS and WE.

Specifically, pairwise t tests indicate that CAPOMDP<sub>LG</sub>-LowIC had significantly more PS-WE pairs than other four LowIC conditions. Recall that Table 5.14 indicates that CAPOMDP<sub>LG</sub>-LowIC achieved significantly higher NLG and LE than either DQN<sub>LG</sub>-LowIC or Random-LowIC. However, since CAPOMDP<sub>LG</sub>-LowIC and Random-LowIC have similar numbers of PS-WE pairs, the number of PS-WE pairs alone does not explain the performance difference for CAPOMDP<sub>LG</sub>-LowIC.

Furthermore, we investigate the PS-WE-PS patterns for each of the LowIC groups since there are relatively high PS-WE and WE-PS pairs for CAPOMDP<sub>LG</sub>-LowIC. A one-way ANOVA test indicates a significant difference in PS-WE-PS patterns among LowIC conditions:  $F(4, 136) = 15.84, p = 1.14e-10$ . Particularly, pairwise t tests show that CAPOMDP<sub>LG</sub>-LowIC ( $M = 1.50, SD = 1.34$ ) had significantly more PS-WE-PS triplets than CAPOMDP<sub>Time</sub>-LowIC ( $M = 0.17, SD = 0.38$ ), DQN<sub>LG</sub>-LowIC ( $M = 0.38, SD = 0.51$ ) and POMDP<sub>LG</sub>-LowIC ( $M = 0.00, SD = 0.00$ ) and Random-LowIC ( $M = 0.44, SD = 0.69$ ): Although Pearson's correlation tests show that PS-WE-PS has a positive correlation with NLG and LE for LowIC students, the correlations are not significant. This result suggests that PS-WE-PS might be helpful for LowIC students' learning outcomes.

In addition, CAPOMDP<sub>Time</sub>-LowIC had significantly more WE-WE pairs than the other four LowIC groups. Recall that Table 5.12c shows that CAPOMDP<sub>Time</sub>-LowIC spent significantly less Time than DQN<sub>LG</sub>-IC, POMDP<sub>LG</sub>-LowIC and Random-LowIC, which may be because of the higher number of WE-WE pairs for CAPOMDP<sub>Time</sub>-Low.

#### 5.9.4 Summarization of Log Analysis

We identify that action-based constraints have a negative impact on the empirical effectiveness of unconstrained RL policy execution, especially for the HighIC students under POMDP<sub>LG</sub> condition. Specifically, HighIC students under the POMDP<sub>LG</sub> condition, with High Carry-out outperformed their HighIC peers in the same condition with Low Carry-out. Additionally, the effectiveness of DQN<sub>LG</sub> and Random are also impacted by the constraints, but not significantly.

Furthermore, we detect the relation between behavior variables and the learning outcomes. In general, we find that *Interactions* has a significant, negative correlation with Post-test score, NLG, and LE, and a significant, positive correlation with Time. *PSCount* has a significant, positive correlation with Time. *HintCount* has a significant, negative correlation with Post-test score and NLG for the high incoming competence (HighIC) students. On the other hand, *HintCount* is not significantly correlated with learning outcomes for students with low incoming competence (LowIC), suggesting that they can learn from hints. Much to our surprise, we found significant negative correlations between learning outcomes and the PS and WE pairs, with PS-PS, WE-PS, and WE-PS-WE patterns increasing time in the tutor, and WE-PS pairs having a negative impact on the Post-test for all students. Additionally, the results indicate that having more WE-PS pairs has a negative impact on high incoming competence (HighIC) students' learning outcomes. While not significant, PS-WE-PS correlated with higher NLG and LE for LowIC students. These differences highlight that sequencing of worked examples and problem solving may need to be different based on students' incoming competence.

We also compare behavior variables among different conditions and find significant differences. Specifically, for the HighIC students, we find that POMDP<sub>LG</sub>-HighIC had significantly more WE-PS pairs than other HighIC groups. Combining that with the results in Table 5.13, that POMDP<sub>LG</sub>-HighIC had the lowest NLG and LE among the HighIC conditions, we conclude that the WE-PS pairs have a negative impact on HighIC students' learning performance. For the LowIC students, we find the CAPOMDP<sub>LG</sub>-LowIC had the lowest *Interactions* and the highest number of the PS-WE pairs among five LowIC groups. However, these differences cannot explain why CAPOMDP<sub>LG</sub>-LowIC outperformed other LowIC groups in terms of NLG and LE. Moreover, CAPOMDP<sub>Time</sub>-LowIC had the highest *WECOUNT* and WE-WE pairs among five LowIC groups, resulting in lower Time for LowIC students to complete the tutor.

### 5.10 Conclusion & Discussion

We proposed the CAPOMDP framework to deal with the action-based constraints in an ITS called Deep Thought, and induced two types of CAPOMDP policies: CAPOMDP<sub>LG</sub> using learning gain (LG) as reward, and CAPOMDP<sub>Time</sub> using time as reward. Then we conducted two empirical stud-

ies and compared their effectiveness against three baselines: POMDP with learning gain reward ( $POMDP_{LG}$ ) and Deep Q-Network with learning gains (LG) reward ( $DQN_{LG}$ ), and the Random policy. Students are further divided into HighIC and LowIC groups according to their incoming competence through a median split on pre-test score. Results show that there is a Aptitude Treatment Interaction effect in our studies, where different patterns are identified for the High and Low incoming competence students.

High incoming competence (HighIC) students have limited normalized learning gains because of their high pre-test scores. Their learning outcomes, including Post-test scores, NLG, LE and Time are not impacted by the  $CAPOMDP_{LG}$ ,  $CAPOMDP_{Time}$ ,  $DQN_{LG}$ , or Random policies. However,  $POMDP_{LG}$  hurt the HighIC students' NLG and LE. We find that  $POMDP_{LG}$  assigned more WE-PS pairs than other policies, and conclude that the WE-PS pairs have a negative impact on HighIC students' learning outcomes.

For the low incoming competence (LowIC) students,  $CAPOMDP_{LG}$  is able to improve their learning performance, including Post-test scores, Learning Gain, and Learning Efficiency, and outperforms three baseline policies.  $CAPOMDP_{Time}$  can reduce the total Time that LowIC students spent in the tutor, and outperformed three baselines in terms of Time. Although results identify that there are significant differences among five conditions on the pairs of PS and WE, they can't explain why  $CAPOMDP_{LG}$  performed better than baselines in NLG and LE for LowIC students. Having more WEs and WE-WE pairs explained why  $CAPOMDP_{Time}$ -LowIC outperformed other conditions on Time for LowIC students. Much to our surprise,  $POMDP_{LG}$  and  $DQN_{LG}$  perform close to Random. The statistical analyses verify that the action-based constraints restrict the effectiveness of  $POMDP_{LG}$  since results indicate that students under the  $POMDP_{LG}$  with a High carry-out ratio outperformed their peers under the  $POMDP_{LG}$  with a Low carry-out ratio.

# CONCLUSIONS AND FUTURE WORK

## 6.1 Conclusions

We apply the Reinforcement Learning (RL) approaches to induce pedagogical strategies in the offline manner with the purpose of dealing with a particular pedagogical decision in the Deep Thought: Problem solving vs. worked example. We explore three different RL frameworks including MDP, POMDP and CAPOMDP, and summarize the conclusion and findings of each framework as follows.

**MDP.** Considering that the tabular MDP cannot handle large and continuous state spaces, we propose various correlation-based feature selection approaches to select a set of discretized features and to construct the effective state space of MDP. In addition, we use either immediate or delayed reward to induce the MDP policies and identify that the immediate reward can facilitate MDP inducing more effective policy than the delayed reward. However, we could not find any difference between the effectiveness of stochastic policy execution with that of the deterministic execution. Specifically, empirical results verify that both the low correlation-based or ensemble feature selection approaches have positive impact on the effectiveness of the MDP policies. In addition, the immediate reward can facilitate the MDP framework inducing a more effective policy than delayed reward. Specifically, the aptitude treatment interaction (ATI) effect consistently exists in these studies, where one particular type of students is more sensitive to the pedagogical strategy in that the MDP policies. The more responsive group using MDP policies induced by either ensemble or low

correlation-based feature selection using immediate reward, can significantly improve their transfer post-test score comparing with Random, while other students are not sensitive in that they can perform well regardless of policies.

**POMDP.** Compared with tabular MDP, POMDP is able to model the uncertainty of student states, to capture students' knowledge level through belief state, and to deal with high dimensional feature space through FAMD, a feature extraction approach. We first induce a POMDP and a MDP policy using the same data with a set of selected features and compare their effectiveness against Random in Experiment 1. The results indicate that there is no significant difference on students' learning performance among three policies, which shows that belief state alone is not enough to help POMDP induce an effective policy. Furthermore, we induce a POMDP policy using the full power of POMDP with a wide range of features as input. The empirical results from the Experiment 2 and Post-hoc comparison indicate the POMDP policy outperforms the MDP and Random policies in terms of students' transfer post-test score. This suggests that the belief state generated based on the wide range of features facilitates POMDP inducing an effective policy.

**CAPOMDP.** CAPOMDP is proposed to induce the pedagogical strategy subject to the action-based constraints in the ITS. Specifically, the  $CAPOMDP_{LG}$  and  $CAPOMDP_{Time}$  policies are induced based on the CAPOMDP framework using learning gain and time as rewards respectively. Empirical results also find a aptitude treatment interaction effect. Specifically, low incoming competence (Low-IC) students are sensitive to the effectiveness of policies in that the Low-IC students following  $CAPOMDP_{LG}$  achieved the significantly higher learning performance than Low-IC following three baseline policies including POMDP, Deep RL, and Random and the Low-IC students following  $CAPOMDP_{Time}$  spent significantly less time than Low-IC students following baseline policies. High incoming competence (High-IC) students are not sensitive to the policies since they can performed similarly and can always learn regardless of policies. One exception is that the POMDP policy hurts the High-IC students' learning performance. Results from log analysis suggest that WE-PS pairs have a negative impact on High-IC students' learning outcomes and indicate that alternating WE and PS in this tutor may not be the best policy overall.

## 6.2 Limitation

Although the series of experiments are carefully prepared, I am still aware of the limitations and shortcomings, listed as follows.

**Students' learning performance metrics** are not consistent across the studies. Particularly, the transfer post-test score is used for evaluation in the MDP study (Chapter 3). Because of the ceiling effect of transfer post-test score, the post-test score calculated based on the performance in levels 2 – 6 is applied in the POMDP study (Chapter 4). Similarly, the post-test score calculated based on the performance in level 7 is used for evaluation in the CAPOMDP study (Chapter 5). Since we modified

Deep Thought in each study, it is hard to use a unified evaluation standard and it is challenging to make consistent conclusions. It is possible that the implemented policies have impacts on different aspects of student learning that we may not be able to detect.

**POMDP vs. MDP.** In POMDP study, we compare the effectiveness of POMDP against the tabular MDP. Although results indicate that the POMDP policies is more effective than the tabular MDP policy, it's highly possible that the ability of handling the high dimensional feature space instead of the belief state space contributes to this result. Therefore, we cannot conclude that the POMDP framework beats the MDP framework for policy induction in the ITS domain. In order to have a fair comparison between POMDP and MDP, and to investigate the benefit of belief state space in POMDP, we need to compare the POMDP framework with continuous state space MDP since both of them are able to deal with the wide range of continuous features.

**Aptitude Treatment Interaction effect** is consistently identified in our studies. However, it is challenging to modify the experiment design or induce the advanced pedagogical strategies to deal with ATI effect. First of all, there is no unified standard to identify the responsive or unresponsive groups. In this work, *the average step time* and *pre-test score* are used to split students into responsive and unresponsive group in the MDP and CAPOMDP studies respectively. Although statistical tests such as ANOVA, and pairwise t tests are used in the post processing phase for comparing the performance among conditions, these methods cannot be used to generate robust and convincing rules for clearly defining which type of students are responsive to the effectiveness of policies. It is possible that we could take advantage of supervised learning to construct a model to classify or to cluster students into responsive or unresponsive groups with high precision and recall.

Consequently, once we can define responsive or unresponsive groups, we can induce various policies for each type of group using different rewards considering we have different expectations for groups. For instance, we may expect that the unresponsive group can always perform well within a limited time, and that the learning performance of the responsive group can be significantly improved with the help of an effective pedagogical strategy.

**Tracking students' learning process.** Although the CAPOMDP framework has shown convincing results, it is hard to explain the CAPOMDP policy, which models students' unobservable factors through belief state. Prior research work [Raf16] has shown that belief states can be interpreted and used for presenting students' latent knowledge. Future work could not only track students' learning process using interpreted belief state but also explain the policy of how to map the interpreted state into action.

### 6.3 Future Work

In order to deal with the aptitude treatment interaction effect, we could firstly apply the classification or clustering techniques to identify students' into the responsive and unresponsive groups

given the students' log file collected in the pre-test phase, where the responsive group is sensitive to the pedagogical strategy while unresponsive group's performance can not be impacted by the policy. Second, we could implement a sequential data mining [Bat12] approach to identify different behavior patterns in either responsive or unresponsive groups and investigate the relationship between behavior patterns and learning performance. Finally, we hypothesize that the identified behavior patterns could help the Reinforcement Learning framework to construct an effective and explainable state space, and further help us induce effective policies.

## BIBLIOGRAPHY

- [Ach17] Achiam, J. et al. “Constrained policy optimization”. *arXiv preprint arXiv:1705.10528* (2017).
- [Alt99] Altman, E. *Constrained Markov decision processes*. Vol. 7. CRC Press, 1999.
- [Bac09] Bach, F. R. “Exploring large feature spaces with hierarchical multiple kernel learning”. *NIPS*. 2009, pp. 105–112.
- [Bak04] Baker, R. S. et al. “Off-task behavior in the cognitive tutor classroom: when students game the system”. *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM. 2004, pp. 383–390.
- [Bat12] Batal, I. et al. “Mining recent temporal patterns for event detection in multivariate time series data”. *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 2012, pp. 280–288.
- [Bec00] Beck, J. et al. “ADVISOR: A machine learning architecture for intelligent tutor construction”. *AAAI/IAAI 2000*.552-557 (2000), pp. 1–2.
- [BF95] Bengio, Y. & Frasconi, P. “An input output HMM architecture”. *Advances in neural information processing systems*. 1995, pp. 427–434.
- [Bro89] Brown, J. S. et al. “Situated cognition and the culture of learning”. *Educational researcher* **18.1** (1989), pp. 32–42.
- [CS14] Chandrashekar, G. & Sahin, F. “A survey on feature selection methods”. *Computers & Electrical Engineering* **40.1** (2014), pp. 16–28.
- [CV10] Chi, M. & VanLehn, K. “Meta-cognitive strategy instruction in intelligent tutoring systems: how, when, and why”. *Journal of Educational Technology & Society* **13.1** (2010), p. 25.
- [Chi11] Chi, M. et al. “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** (2011), pp. 137–180.
- [Cle16] Clement, B. et al. “A Comparison of Automatic Teaching Strategies for Heterogeneous Student Populations”. *EDM 16-9th International Conference on Educational Data Mining*. 2016.
- [CS77a] Cronbach, L. J. & Snow, R. E. *Aptitudes and instructional methods: A handbook for research on interactions*. New York: Irvington, 1977.
- [CS77b] Cronbach, L. J. & Snow, R. E. *Aptitudes and instructional methods: A handbook for research on interactions*. Irvington, 1977.

- [D'M10] D'Mello, S. et al. "A time for emoting: When affect-sensitivity is and isn't effective at promoting deep learning". *International conference on intelligent tutoring systems*. Springer, 2010, pp. 245–254.
- [D'M11] D'Mello, S. K. et al. "A motivationally supportive affect-sensitive autotutor". *New perspectives on affect and learning technologies*. Springer, 2011, pp. 113–126.
- [DD05] Dolgov, D. & Durfee, E. "Stationary deterministic policies for constrained MDPs with multiple rewards, costs, and discount factors". *Ann Arbor* **1001** (2005), p. 48109.
- [Dor15] Doroudi, S. et al. "Towards Understanding How to Leverage Sense-Making, Induction and Refinement, and Fluency to Improve Robust Learning." *International Educational Data Mining Society* (2015).
- [GF12] Garcia, J. & Fernández, F. "Safe exploration of state and action spaces in reinforcement learning". *Journal of Artificial Intelligence Research* **45** (2012), pp. 515–564.
- [Gol05] Goldberger, J. et al. "Neighbourhood components analysis". *Advances in Neural Information Processing Systems*. 2005, pp. 513–520.
- [GEB] González-Espada, W. J. & Bullock, D. W. "Innovative applications of classroom response systems: Investigating students' item response times in relation to final course grade, gender, general point average, and high school ACT scores". *Electronic Journal for the Integration of Technology in Education* **6** (), pp. 97–108.
- [Hal99] Hall, M. A. "Correlation-based feature selection for machine learning". PhD thesis. The University of Waikato, 1999.
- [Han17] Hanheide, M. et al. "Robot task planning and explanation in open and uncertain worlds". *Artificial Intelligence* **247** (2017), pp. 119–150.
- [HS97] Hochreiter, S. & Schmidhuber, J. "Long short-term memory". *Neural computation* **9.8** (1997), pp. 1735–1780.
- [Igl03] Iglesias, A. et al. "An experience applying reinforcement learning in a web-based adaptive and intelligent educational system" (2003).
- [Igl09a] Iglesias, A. et al. "Learning teaching strategies in an adaptive and intelligent educational system through reinforcement learning". *Applied Intelligence* **31.1** (2009), pp. 89–106.
- [Igl09b] Iglesias, A. et al. "Reinforcement learning of pedagogical policies in adaptive and intelligent educational systems". *Knowledge-Based Systems* **22.4** (2009), pp. 266–270.
- [Jaa95] Jaakkola, T. et al. "Reinforcement learning algorithm for partially observable Markov decision problems". *Advances in neural information processing systems*. 1995, pp. 345–352.

- [Kal03] Kalyuga, S. et al. “The expertise reversal effect”. *Educational psychologist* **38.1** (2003), pp. 23–31.
- [Kel06] Keller, P. W. et al. “Automatic basis function construction for approximate dynamic programming and reinforcement learning”. *Proceedings of the 23rd international conference on Machine learning*. ACM. 2006, pp. 449–456.
- [Ken83] Kent, J. T. “Information gain and a general measure of correlation”. *Biometrika* **70.1** (1983), pp. 163–173.
- [Kim11] Kim, D. et al. “Point-based value iteration for constrained POMDPs”. *IJCAI*. 2011, pp. 1968–1974.
- [KA07] Koedinger, K. R. & Aleven, V. “Exploring the assistance dilemma in experiments with cognitive tutors”. *Educational Psychology Review* **19.3** (2007), pp. 239–264.
- [Koe97] Koedinger, K. R. et al. “Intelligent tutoring goes to school in the big city”. *International Journal of Artificial Intelligence in Education (IJAIED)* (1997).
- [Koe13] Koedinger, K. R. et al. “New potentials for data-driven intelligent tutoring system development and optimization”. *AI Magazine* **34.3** (2013), pp. 27–41.
- [KS98] Koenig, S. & Simmons, R. “Xavier: A robot navigation architecture based on partially observable markov decision process models”. *Artificial Intelligence Based Mobile Robotics: Case Studies of Successful Robot Systems* (1998), pp. 91–122.
- [KN09] Kolter, J. Z. & Ng, A. Y. “Regularization and feature selection in least-squares temporal difference learning”. *Proceedings of the 26th annual international conference on machine learning*. ACM. 2009, pp. 521–528.
- [Kop15] Koprinska, I. et al. “Correlation and instance based feature selection for electricity load forecasting”. *Knowledge-Based Systems* **82** (2015), pp. 29–40.
- [LP03] Lagoudakis, M. G. & Parr, R. “Least-squares policy iteration”. *Journal of machine learning research* **4**.Dec (2003), pp. 1107–1149.
- [LL06] Lee, C. & Lee, G. G. “Information gain and divergence-based feature selection for machine learning-based text categorization”. *Information processing & management* **42.1** (2006), pp. 155–165.
- [Lee17] Lee, J. et al. “Constrained Bayesian Reinforcement Learning via Approximate Linear Programming”. *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*. 2017.
- [Lev00] Levin, E. et al. “A stochastic model of human-machine interaction for learning dialog strategies”. *IEEE Transactions on speech and audio processing* **8.1** (2000), pp. 11–23.

- [Li09] Li, L. et al. “Reinforcement learning for dialog management using least-squares Policy iteration and fast feature selection.” *INTERSPEECH*. 2009, pp. 2475–2478.
- [Lit94] Littman, M. L. “Markov games as a framework for multi-agent reinforcement learning”. *Machine Learning Proceedings 1994*. Elsevier, 1994, pp. 157–163.
- [Man14] Mandel, T. et al. “Offline policy evaluation across representations with applications to educational games”. *Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems*. 2014, pp. 1077–1084.
- [MA04] Martin, K. N. & Arroyo, I. “AgentX: Using reinforcement learning to improve the effectiveness of intelligent tutoring systems”. *International Conference on Intelligent Tutoring Systems*. 2004, pp. 564–572.
- [MI11] McLaren, B. M. & Isotani, S. “When is it best to learn with all worked examples?” *Artificial Intelligence in Education*. Springer. 2011, pp. 222–229.
- [McL08] McLaren, B. M. et al. “When and how often should worked examples be given to students? New results and a summary of the current state of research”. *Proceedings of the 30th annual conference of the cognitive science society*. 2008, pp. 2176–2181.
- [McL14] McLaren, B. M. et al. “Exploring the Assistance Dilemma: Comparing Instructional Support in Examples and Problems”. *Intelligent Tutoring Systems*. Springer. 2014, pp. 354–361.
- [Mni15] Mnih, V. et al. “Human-level control through deep reinforcement learning”. *Nature* **518**.7540 (2015), p. 529.
- [MB17] Mostafavi, B. & Barnes, T. “Evolution of an intelligent deductive logic tutor using data-driven elements”. *International Journal of Artificial Intelligence in Education* **27**.1 (2017), pp. 5–36.
- [Mos15] Mostafavi, B. et al. “Data-Driven Proficiency Profiling.” *International Educational Data Mining Society* (2015).
- [Naj14] Najar, A. S. et al. “Adaptive Support versus Alternating Worked Examples and Tutored Problems: Which Leads to Better Learning?” *User Modeling, Adaptation, and Personalization*. Springer, 2014, pp. 171–182.
- [Nar15] Narasimhan, K. et al. “Language understanding for text-based games using deep reinforcement learning”. *arXiv preprint arXiv:1506.08941* (2015).
- [Pag04] Pagès, J. “Analyse factorielle de données mixtes”. *Revue de statistique appliquée* **52**.4 (2004), pp. 93–111.
- [PS02] Peshkin, L. & Shelton, C. R. “Learning from scarce experience”. *arXiv preprint cs/0204043* (2002).

- [Pin03] Pineau, J. et al. “Point-based value iteration: An anytime algorithm for POMDPs”. *IJCAI*. Vol. 3. 2003, pp. 1025–1032.
- [Pin06] Pineau, J. et al. “Anytime point-based approximations for large POMDPs”. *Journal of Artificial Intelligence Research* **27** (2006), pp. 335–380.
- [Pou15] Poupart, P. et al. “Approximate Linear Programming for Constrained Partially Observable Markov Decision Processes”. *AAAI*. 2015, pp. 3342–3348.
- [Raf16] Rafferty, A. N. et al. “Faster teaching via pomdp planning”. *Cognitive science* **40.6** (2016), pp. 1290–1332.
- [RH09] Razzaq, L. M. & Heffernan, N. T. “To Tutor or Not to Tutor: That is the Question.” *AIED*. 2009, pp. 457–464.
- [Ren02] Renkl, A. et al. “From example study to problem solving: Smooth transitions help learning”. *The Journal of Experimental Education* **70.4** (2002), pp. 293–315.
- [RL15] Rowe, J. P. & Lester, J. C. “Improving student problem solving in narrative-centered learning environments: A modular reinforcement learning framework”. *International Conference on Artificial Intelligence in Education*. Springer. 2015, pp. 419–428.
- [Roy00] Roy, N. et al. “Spoken dialogue management using probabilistic reasoning”. *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*. Association for Computational Linguistics. 2000, pp. 93–100.
- [Sal10] Salden, R. J. et al. “The expertise reversal effect and worked examples in tutored problem solving”. *Instructional Science* **38.3** (2010), pp. 289–307.
- [San10] Sanner, S. “Relational dynamic influence diagram language (rddl): Language description”. *Unpublished ms. Australian National University* (2010), p. 32.
- [SS02] Schnipke, D. L. & Scrams, D. J. “Exploring issues of examinee behavior: Insights gained from response-time analyses”. *Computer-based testing: Building the foundation for future assessments* (2002), pp. 237–266.
- [SC16a] Shen, S. & Chi, M. “Aim Low: Correlation-based Feature Selection for Model-based Reinforcement Learning.” *EDM*. 2016, pp. 507–512.
- [SC16b] Shen, S. & Chi, M. “Reinforcement Learning: the Sooner the Better, or the Later the Better?” *Proceedings of the 2016 Conference on User Modeling Adaptation and Personalization*. ACM. 2016, pp. 37–44.
- [She18a] Shen, S. et al. “Empirically Evaluating the Effectiveness of POMDP vs. MDP Towards the Pedagogical Strategies Induction”. *International Conference on Artificial Intelligence in Education*. Springer. 2018, pp. 327–331.

- [She18b] Shen, S. et al. “Exploring Induced Pedagogical Strategies Through a Markov Decision Process Framework: Lessons Learned”. *JEDM/ Journal of Educational Data Mining* **10.3** (2018), pp. 27–68.
- [She18c] Shen, S. et al. “Improving Learning & Reducing Time: A Constrained Action-Based Reinforcement Learning Approach”. *Proceedings of the 26th Conference on User Modeling, Adaptation and Personalization*. ACM. 2018, pp. 43–51.
- [Sin02] Singh, S. et al. “Optimizing dialogue management with reinforcement learning: Experiments with the NJFun system”. *Journal of Artificial Intelligence Research* **16** (2002), pp. 105–133.
- [Sin00] Singh, S. P. et al. “Reinforcement learning for spoken dialogue systems”. *Advances in Neural Information Processing Systems*. 2000, pp. 956–962.
- [Sno91] Snow, R. E. “Aptitude-treatment interaction as a framework for research on individual differences in psychotherapy.” *Journal of consulting and clinical psychology* **59.2** (1991), p. 205.
- [Sta13] Stamper, J. et al. “Experimental evaluation of automatic hint generation for a logic tutor”. *International Journal of Artificial Intelligence in Education* **22.1-2** (2013), pp. 3–17.
- [SB98a] Sutton, R. S. & Barto, A. G. *Introduction to reinforcement learning*. Vol. 135. MIT press Cambridge, 1998.
- [SB98b] Sutton, R. S. & Barto, A. G. *Reinforcement learning: An introduction*. MIT press, 1998.
- [TL08] Tetreault, J. R. & Litman, D. J. “A reinforcement learning approach to evaluating state representations in spoken dialogue systems”. *Speech Communication* **50.8** (2008), pp. 683–696.
- [Tho86] Thomas, R. D.L.V. S. et al. *Response Times: Their Role in Inferring Elementary Mental Organization: Their Role in Inferring Elementary Mental Organization*. Oxford University Press, USA, 1986.
- [VG11] Van Gog, T. et al. “Effects of worked examples, example-problem, and problem-example pairs on novices’ learning”. *Contemporary Educational Psychology* **36.3** (2011), pp. 212–218.
- [Van06] Vanlehn, K. “The behavior of tutoring systems”. *International journal of artificial intelligence in education* **16.3** (2006), pp. 227–265.
- [Van07] VanLehn, K. et al. “When are tutorial dialogues more effective than reading?” *Cognitive science* **31.1** (2007), pp. 3–62.
- [Vyg78] Vygotsky, L. “Interaction between learning and development”. *Readings on the development of children* **23.3** (1978), pp. 34–41.

- [Wan17] Wang, P. et al. “Interactive Narrative Personalization with Deep Reinforcement Learning”. *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*. 2017.
- [WM17] Whitehill, J. & Movellan, J. “Approximately optimal teaching of approximately optimal learners”. *IEEE Transactions on Learning Technologies* (2017).
- [Wil08] Williams, J. D. “The best of both worlds: unifying conventional dialog systems and POMDPs.” *INTERSPEECH*. 2008, pp. 1173–1176.
- [WY07] Williams, J. D. & Young, S. “Partially observable Markov decision processes for spoken dialog systems”. *Computer Speech & Language* **21.2** (2007), pp. 393–422.
- [Wri12] Wright, R. et al. *Embedded incremental feature selection for reinforcement learning*. Tech. rep. DTIC Document, 2012.
- [YP97] Yang, Y. & Pedersen, J. O. “A comparative study on feature selection in text categorization”. *Icml*. Vol. 97. 1997, pp. 412–420.
- [YL03] Yu, L. & Liu, H. “Feature selection for high-dimensional data: A fast correlation-based filter solution”. *ICML*. Vol. 3. 2003, pp. 856–863.
- [Zha01] Zhang, B. et al. “Spoken dialogue management as planning and acting under uncertainty.” *INTERSPEECH*. 2001, pp. 2169–2172.
- [Zib07] Zibran, M. F. “Chi-Squared test of independence”. *Department of Computer Science, University of Calgary, Alberta, Canada* (2007).