THE UNIVERSITY OF CHICAGO

LEARNING IN OPERATIONAL SETTINGS

A DISSERTATION SUBMITTED TO
THE FACULTY OF THE UNIVERSITY OF CHICAGO
BOOTH SCHOOL OF BUSINESS
IN CANDIDACY FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

BY
CAGLA KECELI

CHICAGO, ILLINOIS
AUGUST 2023

To my parents, family and friends

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

I am deeply grateful to my advisor and committee chair Dan Adelman, who encouraged me greatly throughout my academic journey. It goes without saying that this thesis would not exist, if it was not for his insightful advice, full support and dedication over the years. I am very thankful to Kiran Turaga, who collaborated with us during the majority of my Ph.D., and has contributed in many ways. I deeply thank Alba V. Olivares Nadal, who not only helped me to improve my work, but also provided a path for future avenues.

I thank the faculty at the University of Chicago, Booth School of Business, for their immense help in gaining the skillset necessary for research. I would like to thank my committee members, John Birge and Rad Niazadeh, for their helpful comments and support in regards to my research. I thank Kate Haas LaVigne for her help in the crucial step of data collection.

I am grateful to Josh Morris-Levenson, Hunter DD Witmer and Kanix Wang for their suggestions in constructing the ideas in this thesis. I would like to thank the faculty at Sabanci University and Johns Hopkins Carey Business School for their help that led me to pursue a Ph.D. in this field: Gunduz Ulusoy, Kemal Kilic, Demir Yener, Ozge Sahin, Reza Djavanshir, and Tinglong Dai.

I thank my friends and my cohort whose camaraderie I enjoyed throughout these years: Alex Zhao, Amir Alwan, Antonis Tsiflis, Cem Randa, Ebru Kasikaralar, Eren Ozbay, Gizem Yilmaz, Gorkem Unlu, Gulin Tuzcuoglu, Josh Shea, Lisa Hillas, Lisa Li, Monty Montgomery, Zuguang Gao. Thank you to Antonis, for making this journey feel much easier than it was. Thanks to the Ph.D. office for their support, and a special thanks to Malaina Brown for her kind guidance.

Biggest thanks to my parents and grandparents, who have always supported me.

The project in Chapter 3 was formally determined to be a quality improvement project, not human subjects research, and was therefore not overseen by the Institutional Review Board, per institutional policy.

ABSTRACT

This thesis consists of three papers in the field of learning, in operational settings. In broad terms, the first paper explores how to learn the underlying parameters by Thompson sampling, i.e., by sampling and updating the belief on the unknown parameter. Our results apply to broad settings, including the settings of the latter papers. Through an empirical study, the second paper studies how to estimate the mean and standard deviation of tasks when they consist of multiple subtasks, which corresponds to learning from data. Finally, the third paper sets up a learning-from-past-experience framework and investigates the long-run ramifications of making worker-task assignments free of workers' innate performance information.

The first paper (Chapter 2) is titled "Thompson Sampling for Infinite-Horizon Discounted Decision Processes".[1] We model a Markov decision process, parametrized by an unknown parameter, and study the asymptotic behavior of a sampling-based algorithm; Thompson sampling (TS). Showing that the standard notion of regret can grow (super-)linearly and it cannot capture the notion of learning in realistic settings with non-trivial state evolution, we decompose the traditional expected regret into three meaningful components. We argue that only one of the components is a fair metric to evaluate a policy, which we call the expected residual regret. It forgets the immutable consequences of past actions; instead, it allows the system to run during a learning period of n stages and starts tallying regret against the optimal policy from period n onwards. We study the performance guarantees of this new notion, in the context of implementing Thompson sampling. In particular, we show that this metric is upper bounded by a term that decays exponentially to 0, almost surely. We present conditions under which the posterior sampling error of Thompson sampling converges to 0 almost surely, i.e., complete learning. We characterize the probabilistic version of the expected residual regret and present conditions under which it converges to 0 almost surely.

[1] I would like to thank Daniel Adelman and Alba V. Olivares Nadal for their constructive advice.

The second paper (Chapter 3) is titled “Estimating the Mean and Variance of Heterogeneous Tasks”. The third paper (Chapter 4) is titled “Equitable Data-Driven Assignments of Workers to Tasks”.[2] These two chapters are tightly related to each other. In Chapter 4, we put forward a simple yet effective method of predicting completion times of tasks, assuming that the completion time is a function of worker-task familiarity. Our prediction algorithm requires standardizing task times and familiarities, i.e., requires estimating the mean and standard deviation of time and familiarity across tasks. Yet, in many settings it is not straightforward to estimate the mean and variance of a task that does not appear frequently. This constitutes the motivation for Chapter 3, which is an empirical study that views surgical encounters as “tasks” and the procedures within surgical encounters as “subtasks”. We estimate the mean and variance of surgical encounters by utilizing the procedure codes that uniquely correspond to each procedure within an encounter. To estimate the mean and variance of surgical encounters, we adapt two statistical methods into this novel setting. In the first approach, we adapt the random coefficients model. In the second approach, we adapt hierarchical clustering, thereby bundling surgeries of similar compositions into the same group. We compare both methods under three independent procedure coding schemes. We derive a novel goodness-of-fit measure to evaluate the quality of the variance estimation. Our results show that neither method nor coding scheme is universally superior across all service lines.

In Chapter 4, we develop a practical, equitable algorithm to predict task completion times, which obscures workers’ performance information, to account for realistic considerations. We compare the equitable (i.e., performance-blind) algorithm’s steady-state predictions, when used in a sequential assignment framework, against the policy that doesn’t obscure worker-specific performance, i.e., the performance-aware policy. In our setting, an equitable assignment is defined to treat any two individuals, who have the same familiarity

[2] The motivating ideas of both of these papers have originated from thought-provoking discussions with Kiran Turaga, Hunter DD Witmer, and Daniel Adelman.

with a particular task, as interchangeable without loss of optimality, regardless of their innate characteristics. In order to bound the performance discrepancy between the equitable policy and the performance-aware policy in steady-state, we characterize an alternative policy, called the egalitarian policy. This policy imposes that (i) No worker has a greater propensity to execute a task than any other worker, and (ii) No task has a greater propensity to be performed by a worker than any other task. We show that the egalitarian policy yields the worst-case solution under certain assumptions on the primitives, and under weaker assumptions, it serves as a reasonable benchmark on the penalty of adopting the equitable policy. We uncover that the steady-state, performance-aware policy is optimized by 1-1 matching.

CHAPTER 1

INTRODUCTION

The matching algorithms that researchers produce often assume that one can freely estimate or explore the cost that depends on the worker-task pair, which entails workers' innate information. Yet, in certain realistic settings, it is desired to keep innate information hidden, such as performance data. For example, one can think of unionized settings, where a study that estimates staff members' idiosyncrasies would be unwanted as it can cause favoritism, or even discrimination. We broadly classify worker-related information into two types: performance information and task familiarity information. In our context, familiarity, i.e., task familiarity, pertains to how often a worker has performed a task in the past.

We build an equitable framework of assigning workers to tasks such that individuals, i.e., workers, who have the same familiarity with a task are treated as interchangeable, regardless of their innate performance information. Thus, respecting workers' (innate) performance-privacy is one of the main factors in designing the prediction algorithm for making assignments. The other considerations are dealing with small sample sizes and capturing the universal effect of worker-task familiarity on completion times. This is not to say that the resulting assignments omit the performance-information of workers; the performance-related parameters still determine task completion times. Thus, the assignments are indirectly related to workers' innate performances through the realizations of the previous assignments, and the performance information is only obscured at the stage of the assignments decisions.

We model the impact of workers' task familiarity on the task duration, where task completion time depends on pair-level unknown base-level performance and unknown familiarity effect. In particular, Chapter 2 investigates how to learn the underlying parameters by Thompson sampling, i.e., by updating the belief on an unknown parameter. Our results apply to broad settings, including the worker-task assignment problem, i.e., where the system is driven by the base performance and familiarity effect of worker-task pairs. The prediction algorithm necessitates standardizing the familiarity and time variables; however, the estimation of mean and familiarity is not straightforward when tasks are multi-step objects with ample heterogeneity. Chapter 3 explores how to learn the mean and standard deviation of multi-step tasks empirically from data. Finally, Chapter 4 sets up a learning-by-doing framework ("gaining familiarity by doing") and investigates the long-run consequences of making worker-task assignments in a way that is free of workers' innate performance information.

Chapter 2. We model a discrete-time Markov decision process (MDP), parametrized by an unknown parameter θ (a single point in the finite parameter space), and study the asymptotic behavior of Thompson sampling. We illustrate that the standard notion of regret can grow (super-)linearly and fails to capture the notion of learning in realistic settings with non-trivial state evolution. We assume Borel state-control spaces, allowing the spaces to be infinite, while using the discounted-reward criterion. Thus, our work allows questions of regret and sampling algorithms to be addressed in broader settings than before.

We offer a novel decomposition of the standard notion of expected regret into three components, only one of which is suitable for all learning problems. The first component is the expected regret of the past, i.e., from period 0 up to period $n-1$ (finite-time regret). The second component is the expected regret that captures the infinite-horizon (future) consequences of being in a suboptimal state in period n. The third and final component is associated with the DM's ability to implement what is optimal moving forward. This notion is the only component that is "controllable" by the DM, once they have arrived at the period-n state. We propose the last component as a sensible notion of regret, which we call the expected residual regret. The expected residual regret forgets both the past and the future "sunk" portions of the standard notion of regret.

To the best of our knowledge, our work is the first to decompose the traditional notion of the infinite-horizon regret into interpretable components. We establish the relation between the expected residual regret and the notion of asymptotic discount optimality (ADO) from the adaptive learning literature. To our knowledge, the concept of ADO has not been linked to the regret literature before. We derive novel results on the performance guarantees of the expected residual regret. We show that it is upper bounded by an exponentially decaying term and that it exhibits complete learning under Thompson sampling. To our knowledge, complete learning has not been studied in Borel state-control spaces and under the discounted reward criterion.

Finally, we characterize a sample-path version of the expected residual regret, i.e., the (probabilistic) residual regret, which is a stronger notion as it pertains to individual sample paths. We show that this metric also converges to 0 under certain mild assumptions.

Chapter 3. The aforementioned prediction algorithm necessitates standardizing the familiarity and time variables; however, the estimation of mean and standard deviation is not straightforward when tasks are multi-step objects with ample heterogeneity. Estimating the variance of multi-step tasks is not only useful for our prediction algorithm, it is a valuable question in itself. For example, a task with high variance may signal a need for intervention, while the converse may indicate an opportunity to disseminate best practices. Our application setting is the healthcare industry. In particular, we estimate the mean and variance of surgical cases, which are performed in the operating room (OR). The potential benefits of identifying variance include better utilization of the OR's and the improvement of metrics such as overtime and patient satisfaction.

Our study utilizes data of surgical encounters performed at the University of Chicago Medical Center, Surgery Department. To observe how surgical cases are composed of codes and their heterogeneity, consider the following example. "Debridement Leg Plastic" is a relatively common procedure code and appears as the single procedure of a surgical case in 336 out of the full sample of 70K cases. Yet, often times, surgical cases comprise of multiple codes. The number of surgical cases in which both "Debridement Leg Plastic" and "Debridement Foot Plastic" have been performed is merely 5 out of the full sample.

Given the sheer amount of different types of procedures, a non-negligible portion of surgical encounters appear rarely. For example, 35% of all surgical cases, when encoded using the CPT terminology, appear only once in the data. To our knowledge, our work is the first to estimate the variance of surgical cases. We adapt two different statistical models into this novel setting and compare their performance, i.e., compare how well they estimate mean and variance. The methods we adapt are the random coefficients model and the hierarchical clustering model.

Chapter 4. In light of the considerations for developing the prediction algorithm, we pose the question: How to assign workers to tasks while preserving performance-privacy, dealing with small samples, and capturing the universal effect task familiarity on task time? To satisfy the considerations, we develop a practical prediction algorithm called the 5-Step algorithm or [ASAPI], which stands for Aggregate-Standardize-Aggregate-Predict-Invert. The 5-step algorithm predicts task completion times of workers. We use standardized variables to ensure that task familiarities and completion times are commensurate across heterogeneous tasks. Historical data is standardized at the task-level; this circumvents revealing worker-specific statistics, while ameliorating sample size issues. Its key step is to fit a single regression model to the standardized data, pooled across all tasks and workers, to estimate the organization-wide effect of familiarity on completion time. Through this estimate, the algorithm predicts the completion times of new data points, i.e., it predicts the completion time of each newly-arriving task by each available worker, by only using the familiarity information.

The prediction from the 5-step algorithm is then fed into an optimization model that assigns workers to tasks in each period. Implementing the period's assignments generates new data, which is then used to refine the organization-wide effect of familiarity on time. Iteratively running the prediction and optimization steps constitutes the Predict-Then Optimize (PTO) loop.

We formulate the steady-state version of the daily assignment problem that optimally allocates workers to tasks. The upper bound on the penalty of using the PTO loop (with the prediction coming from the equitable 5-step algorithm) is explored by characterizing the worst-case policy: the “egalitarian policy”. The egalitarian policy is a policy that ensures (i) No worker has a greater propensity to execute a task, and (ii) No task has a greater propensity to be performed by a worker. Under a simple assumption, we show that the egalitarian policy is the worst policy in steady-state, and find the steady-state discrepancy between the optimal (true) policy and the egalitarian policy.

CHAPTER 2

THOMPSON SAMPLING FOR INFINITE-HORIZON DISCOUNTED DECISION PROCESSES

2.1 Introduction

We consider a control problem in which a decision-maker (DM) interacts with an environment without knowing the value of a parameter encoded into this environment. This environment is modeled by a discrete-time Markov decision process (MDP). The DM applies a control and, in return, the system moves to the next state and generates a reward. Neither of the outputs are deterministic; they are contingent on the current state, control and value of the unknown parameter. Using the state transition and reward information, the DM picks a control which yields the subsequent transition and reward.

Our model is mainly shaped by the stochastic adaptive control literature and, in particular, complements Kim [2017]. Numerous prominent papers parametrize MDP's by an unknown parameter and estimate this parameter using different methods. We investigate the asymptotic performance of a popular parameter estimation algorithm, Thompson sampling (TS), first described by Thompson [1933]. Aiming to maximize the total reward, yet not knowing the underlying parameter that determines the rewards and state transitions, the DM faces a trade-off between choosing controls tied to unexplored parameter candidates and choosing controls which are likely to yield high rewards. This exploration vs. exploitation trade-off is often addressed in online decision-making problems by TS. After estimating the unknown parameter by TS, the DM selects a control while assuming the estimate is the true parameter.

Despite the logarithmic expected regret guarantees in bandit settings [Agrawal and Goyal, 2012], we document through examples that the expected regret no longer has logarithmic guarantees when the chain structure underneath the MDP is generic. In particular, we document how the expected regret can grow (super-)linearly. This means that the standard notion of expected regret used in the literature, i.e., the gap between the cumulative reward of the DM and the cumulative reward of an omniscient agent, is not fully informative when the underlying chain is general. The expected regret is unable to discern that learning is taking place, even though it is. These kinds of examples may be implicitly known to researchers who work in this area, yet they have not been documented in the literature to the best of our knowledge.

We provide the first theoretical results that capture a different notion of (expected) regret of the TS algorithm. In particular, we extend a result of Kim [2017] (on the convergence of the expected posterior error) to discounted-reward MDP's and to general state and control spaces. We assume Borel state-control spaces, allowing the spaces to be infinite. Thus, our work allows questions of regret and sampling algorithms to be addressed in broader settings in which they have not been addressed before. Extending the performance guarantees of the new notion of regret to MDP's with a general chain structure is an important potential avenue of study.

The literature on parametrized MDP's almost always considers the long-run average regret per period. In contrast, we study guarantees when the performance criterion is the expected infinite-horizon discounted reward. Since problems of economic significance are often most properly formulated with discounting, having the objective function formulated as an infinite-horizon discounted reward problem would allow the ideas of learning through sampling to be applied in these settings.

We develop a general canonical probability space for adaptive learning algorithms based on sampling, of which TS is a special case. Kim [2017] and Banjević and Kim [2019] construct a similar probability space for TS, but do not include the sampled parameters in the sample space. We are not aware of other work which incorporates this canonical formulation. Having the sampling algorithms posed in a coherent framework allows for it to be understood and studied rigorously. For example, the formulation makes evident that the underlying process is history-dependent, i.e., not Markovian. To make this clear, we will refer to the adaptive version of the process with an unknown parameter as θ-MDP instead of MDP.

We offer a novel decomposition of the standard notion of expected regret. Our notion of expected regret can be computed at any finite period n, from the vantage point of a DM in period 0. We identify three distinct components of the expected regret, only one of which is always suitable for learning problems. The first component is the expected regret associated with the past, i.e., from period 0 up to period n. This quantifies the difference between the expected reward received by the DM and the expected reward of the policy which knows the true parameter. The second component is the expected regret that emanates from the policy's state in period n, which is inextricably tied to the past decisions. It captures the expected infinite-horizon (future) consequences of being in a suboptimal state in period n. The third and final component captures the DM's ability to implement what is optimal moving forward, conditional on the period-n state. This component is the only component that is "controllable" by the DM, once they have arrived at the period-n state. It reflects how the best opportunities available to the DM from period n onward compare to how the system will evolve under the sampling policy. Our analysis focuses on this concept which we call *expected residual regret*. The expected residual regret forgets both the past and the future "sunk" portions of the total regret.

Decomposing the standard expected regret is a novel approach to view and quantify the performance of a policy. Through this decomposition, we connect the notion of expected regret to a concept called *asymptotic discount optimality* (ADO) [Schäl, 1987, Hernández-Lerma, 2012]. We demonstrate how the ADO concept relates to expected residual regret. To the best of our knowledge, this concept has not been used within the sampling context before and has not been connected to the notion of (expected) regret.

We show that the posterior belief on the true parameter converges to 1 when TS is implemented; this behavior is called *complete learning.* Complete learning is a stronger result than learning in expectation, i.e., expected posterior belief, because it gives assurance about the individual sample paths. To the best of our knowledge, most results are on the expected-valued version of posterior belief. The notion of complete learning has been studied frequently in bandit contexts, including the prominent work of Freedman [1963], yet to our knowledge it has not been applied to Borel state-control spaces and has not been analyzed under the discounted reward criterion. In the adaptive learning literature, there is an alternative to TS, known as the minimum contrast estimator, which is shown to achieve complete learning [Hernández-Lerma, 2012]. However, complete learning has not been connected to the concept of ADO, and there is no notion of probabilistic ADO.

Building on expected residual regret, we define its sample-path version, called *probabilistic residual regret* or *residual regret.* Similar to expected residual regret, it captures the DM's ability to implement what is optimal moving forward, but starting from the random state of the MC in period n. It quantifies the difference in optimal rewards and TS-driven rewards. Therefore, the probabilistic residual regret is a random quantity whose expected value is the expected residual regret. Conceptually, probabilistic residual regret is similar to complete learning as they are both concerned with individual sample paths.

The paper is organized as follows. In Section 2.2, we provide additional literature review. Then we motivate the need to modify the definition of expected regret since the standard notion of expected regret can grow linearly in non-trivial settings. In Section 2.3, we model the sampling algorithm and formulate the canonical probability space which involves the sampled parameters. In Section 2.4, we characterize TS by defining the posterior update and control selection mechanisms. In Section 2.5, we decompose the standard notion of expected regret into components and interpret each component, highlighting expected residual regret as the only "actionable" component. In Section 2.6, we provide an asymptotic analysis of the expected residual regret. In Section 2.7, we show that TS exhibits complete learning, i.e., the posterior sampling error converges to 0 almost surely. We also show that probabilistic residual regret converges to 0 almost surely.

2.2 Literature Review and Motivation

Here, we highlight papers of similar setup and provide an example where the standard expected regret fails to discern learning.

2.2.1 Literature Review

TS has proved not only successful for multi-armed bandit problems (MAB) (a degenerate case of MDP); but also for parametrized MDP's, i.e., θ-MDP's, which need not be Markovian. The work on θ-MDP's typically requires assumptions on the underlying chain to give performance guarantees on the expected regret. For example, Kim [2017] shows that TS achieves asymptotically optimal expected regret when the Markov chain under the optimal policy, which knows the true parameter, is ergodic.[1] Another example is Gopalan and Mannor [2015], which provides a probabilistic logarithmic upper bound on the expected regret of TS, assuming that the starting state is recurrent under the optimal policy generated by any of the possible parameters. Although Kearns and Singh [2002] and Leike et al. [2016] assume a more general chain setting, they study decision processes over finite state-control spaces. In contrast to the finite state-control spaces or the expected average-reward criterion of these works, we assume a general state-control space, under the discounted-reward criterion, similar to Schäl [1987]. We impose additional assumptions on the underlying chain to be able to extend the work of Kim [2017] into our more general setting.

Kim [2017] shows that the expected posterior sampling error decays exponentially. We extend this result to a broader framework. We also show that the probabilistic version of

[1] A Markov chain is *ergodic* if the transition matrix corresponding to every deterministic stationary policy consists of a single recurrent class [Puterman, 2014]

the posterior sampling error decays exponentially, which we believe has not been explored yet.

Our work mostly complements Kim [2017] and Banjević and Kim [2019]; there is a stronger connection with the former since we also assume a finite parameter space. The latter work adopts a continuous parameter space, similar to Hernández-Lerma [2012], which results in a setting that is harder to analyze.

The stream of work that analyzes the behavior of TS do not usually consider infinite-horizon, and are most often in the MAB setting, with history-independent samples to the best of our knowledge. Kalkanli and Ozgur [2020] analyze the asymptotic behavior of TS with history-dependence, but in the context of the MAB problem. Although we utilize different methodologies than theirs, our work carries this analysis into the under-explored context of the θ-MDP.

2.2.2 Motivation

TS has been shown to have good performance in the MAB setting. The MAB problem is equivalent to a θ-MDP with no state, or alternatively, a one-step θ-MDP. In each decision period t, the DM samples a parameter from the posterior distribution. By treating the sample as the true parameter, the DM chooses a control, i.e., plays one of the constantly-many arms, and immediately observes a reward. The reward of each arm is generated according to some fixed (unknown) distribution and the objective is to maximize the total expected reward. Arms' rewards are generated independently of each other. Let μ_i denote the (unknown) expected reward of arm i and $i(t)$ be the arm played in period t. The expected finite-time regret is the expected total difference between the optimal strategy of pulling the arm with the highest mean and the strategy followed by the DM, i.e.,

$$E[\text{Regret}(n)] := E\left[\sum_{t=1}^{n}(\mu^* - \mu_{i(t)})\right].$$

where $\mu^* \coloneqq \max_i \mu_i$. In the pioneering Lai and Robbins [1985], the (expected) regret of any bandit algorithm is lower bounded, in the limit, by

$$E[\text{Regret}(n)] \geq \left[\sum_{i=2}^{K} \frac{\Delta_i}{D(\mu_i || \mu^*)} + o(1) \right] \ln(n), \tag{2.1}$$

where D denotes the Kullback–Leibler divergence and $\Delta_i \coloneqq \mu^* - \mu_i$. The bound in (2.1) shows that the best achievable expected regret is of order $\ln(n)$. Complementing the logarithmic lower bound for any bandit algorithm, Agrawal and Goyal [2012] upper bounds the expected regret of the TS algorithm by

$$E[\text{Regret}(n)] \leq O\left(\left(\sum_{i=2}^{K} \frac{1}{\Delta_i^2} \right)^2 \ln(n) \right).$$

Since the order of the upper bound on the expected regret of TS matches the logarithmic lower bound for any algorithm in (2.1), the expected regret of TS grows logarithmically. We underline that these results are valid for the MAB setting. These results can also apply to the θ-MDP setting with a trivial state process, such that the stochastic process is driven by an iid state process[2]. In addition, they assume $\beta = 1$, i.e., no discounting. Our setting is fundamentally different, i.e., the states evolve based on the controls and are not identically distributed, hence these results do not apply in general. Example 2.2.1 illustrates a simple yet non-trivial state process.

Example 2.2.1 (Expected regret grows linearly)**.** Consider the three-state process shown in Figure 2.1. In period $t = 0$, the DM lies in x_0 and can choose either control A or B, with an immediate reward of 0. Control A leads to the state x_A, and from then onward only control A can be chosen; the DM is "stuck". Control B leads to the state x_B, and similarly, only control B can be chosen from then onward. The true parameter can either be

[2] The states evolve independently of the controls and have the same probability distribution.

A or B. We represent the one-step reward generating function by $R^A(\cdot)$ when A is the true parameter, and $R^B(\cdot)$ otherwise. We assume that the prior belief on the true parameter is not degenerate, i.e., not equal to 1.

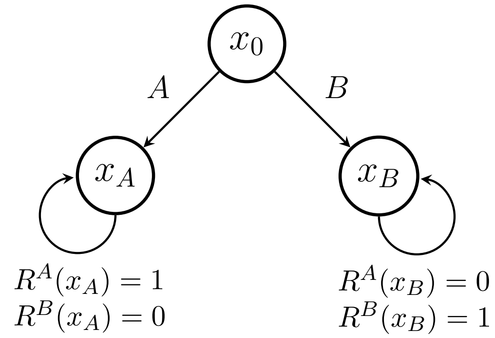


Figure 2.1: Constant reward depending only on the first control, picked at t=0.

In Example 2.2.1 if the first guess is right, the DM receives a reward of 1 forever and otherwise receives no reward at all. Thus, unlike in the setting of Agrawal and Goyal [2012], in broader settings the expected finite-horizon regret does not necessarily grow logarithmically.

Consider Example 2.2.1 with an alternative setup such that when the guess at $t = 0$ is correct, the reward generated in the corresponding state is equal to the number of periods the policy has spent in that state. If the DM makes a wrong guess they are stuck with 0 reward forever, while the oracle earns a sequence of increasing rewards. Here, the expected (undiscounted) regret grows super-linearly. While the standard expected regret grows linearly or super-linearly, nonetheless learning still happens. If the DM receives a reward of 1 in the next period, then they immediately learn whether A or B drives the reward process. This example shows the motivation to construct an alternative, more “lenient”, notion of expected regret that forgets the immutable consequences of past actions.

2.3 Model Setup

We study a discounted-reward stochastic control problem over an infinite horizon. The underlying MDP is indexed by some parameter θ. In Section 2.3.1, we formulate the MDP when the DM knows θ, building on the mathematical framework of Hernández-Lerma [2012]. In the subsequent sections, we assume the DM solves the θ-MDP using estimates of θ, i.e., without knowing the value of θ. We construct the probability space in Section 2.3.2. This lays the groundwork for TS.

2.3.1 Markov Decision Process for Known θ

Consider a discrete-time MDP, $(\mathcal{X}, \mathcal{U}, \{\mathcal{U}(x), x \in \mathcal{X}\}, f^{\theta}, q^{\theta})$. In our setting, the reward and state transition densities depend on the parameter $\theta \in \mathcal{P}$, which is a single point in the finite parameter space $\mathcal{P}$. Later, when θ is assumed to be unknown, we will represent the random sample drawn in period t by Θ_t and its realization by θ_t. We reserve upper-case letters to denote random variables, lower-case letters to realizations of random variables, script letters to spaces and sometimes sets, and bold upper-case letters to elements of σ-algebras to be defined. Let $\mathbb{R}$ denote the set of real numbers, and $\mathcal{B}(\cdot)$ the Borel σ-algebra of a topological space.

The tuple $(\mathcal{X}, \mathcal{U}, \{\mathcal{U}(x), x \in \mathcal{X}\}, f^{\theta}, q^{\theta})$ consists of:

1. State space $\mathcal{X}$, a Borel space. We denote the system state in period $t \in \{0, 1, 2, \dots\}$ by $X_t \in \mathcal{X}$.

2. Control space $\mathcal{U}$, a Borel space. The control applied in period t is $U_t \in \mathcal{U}$.

3. Set of admissible controls $\mathcal{U}(x)$, which is a compact subset of $\mathcal{U}$ for every $x \in \mathcal{X}$. Let $\mathcal{U} = \bigcup_{x \in \mathcal{X}} \mathcal{U}(x)$. The set of admissible state-control pairs,

$$\mathbb{K} := \{(x, u) \mid x \in \mathcal{X},\, u \in \mathcal{U}(x)\},$$

is assumed to be a measurable subset of the product space $\mathcal{X} \times \mathcal{U}$.

4. Given x_t and u_t in period t, the system generates random reward $R_t \in \mathscr{R}_c \subset \mathbb{R}_+$, where $\mathscr{R}_c$ is compact and measurable according to a conditional distribution $F^\theta(\cdot \mid x_t, u_t)$. The conditional distribution $F^\theta(\cdot \mid x, u)$ admits a measurable, continuous, one-step reward density[3] $f^\theta : \mathscr{R}_c \to \mathbb{R}_+$ with respect to a σ-finite measure λ on $(\mathscr{R}_c, \mathcal{B}(\mathscr{R}_c))$, such that

$$F^\theta(\mathbf{R} \mid x, u) \coloneqq \int_{\mathbf{R}} f^\theta(r \mid x_t = x, u_t = u)\, d\lambda(r), \quad \forall \mathbf{R} \in \mathcal{B}(\mathscr{R}_c), (x, u) \in \mathbb{K}.$$

The expected reward in period t is

$$r^\theta(x_t, u_t) \coloneqq \int_{\mathscr{R}_c} r f^\theta(r \mid x_t, u_t) d\lambda(r), \quad \forall\, (x_t, u_t) \in \mathbb{K}. \tag{2.2}$$

5. Given x_t and u_t in period t, the system transitions into random state X_{t+1} according to a conditional distribution $Q^\theta(\cdot \mid x_t, u_t)$. The conditional distribution $Q^\theta(\cdot \mid x, u)$ admits a one-step transition density[4] $q^\theta : \mathbb{K} \to \mathcal{X}$ with respect to a σ-finite measure η on $(\mathcal{X}, \mathcal{B}(\mathcal{X}))$ such that

$$Q^\theta(\mathbf{X} \mid x, u) \coloneqq \int_{\mathbf{X}} q^\theta(y \mid x_t = x, u_t = u) d\eta(y), \quad \forall \mathbf{X} \in \mathcal{B}(\mathcal{X}).$$

Similarly, given x_t and u_t in period t,

$$Q^\theta(\mathbf{X} \mid x, u) \coloneqq \text{Prob}(X_{t+1} \in \mathbf{X} \mid x_t = x, u_t = u), \quad \forall \mathbf{X} \in \mathcal{B}(\mathcal{X}).$$

Remark. Since $f^\theta(\cdot \mid x_t, u_t)$ is continuous on compact set $\mathscr{R}_c$, it attains its minimum and maximum in $\mathscr{R}_c$. Thus, there exists a constant $M \geq 0$ such that $|r^\theta(x_t, u_t)| \leq M$

[3] Radon-Nikodym derivative of F^θ with respect to λ.

[4] Radon-Nikodym derivative of Q^θ with respect to η

$\forall(x_t, u_t) \in \mathbb{K}$. To see why this holds, note that

$$\begin{aligned}|r^\theta(x_t, u_t)| \leq \int_{\mathscr{R}_c} |rf^\theta(r \mid x_t, u_t)| d\lambda(r) &\leq \int_{\mathscr{R}_c} \max(|rf^\theta(r \mid x_t, u_t)|) d\lambda(r) \\ &= \max_{r \in \mathscr{R}_c} |rf^\theta(r \mid x_t, u_t)| \lambda(\mathscr{R}_c) < \infty,\end{aligned}$$

where the first inequality follows by (2.2) and Jensen's inequality, and the equality holds since $\max(|rf^\theta(r \mid x_t, u_t)|)$ is constant. Then, $\max_{r \in \mathscr{R}_c} |rf^\theta(r \mid x_t, u_t)|$ is finite because $f^\theta(\cdot \mid x_t, u_t)$ is continuous on a compact set, and $\lambda(\mathscr{R}_c)$ is finite since $\mathscr{R}_c \subset \mathbb{R}_+$ is compact and λ is a σ-finite measure.

The DM aims to maximize the infinite-horizon expected total discounted reward. The *optimal value function* $\nu^\theta(x)$ represents the maximum such reward starting from state x, and it depends on the true (unknown) parameter θ. It is known from Hernández-Lerma and Lasserre [2012] that this problem is solved by the Bellman equation in (2.3).

$$\nu^\theta(x) := \sup_{u \in \mathcal{U}(x)} \left\{ r^\theta(x, u) + \beta \int_{y \in \mathcal{X}} \nu^\theta(y) dQ^\theta(dy \mid x, u) \right\}, \quad \forall x \in \mathcal{X}, \theta \in \mathcal{P}, \tag{2.3}$$

where $\beta \in [0, 1)$ is the discounting factor. If $\nu^\theta(x)$ is a solution to (2.3), let μ^θ denote the corresponding optimal policy for parameter θ. Under sufficient technical conditions, μ^θ is a stationary, deterministic, and Markovian policy. As showing these conditions holds in this setting are outside of our scope, we make the following assumption:

Assumption 0. For all $\theta \in \mathcal{P}$, there exists a unique solution to (2.3) such that the supremum is attained, and there exists an optimal policy μ^θ that is stationary, deterministic, and Markovian.

Remark. The condition in Assumption 0 is necessary for all of the main results of the paper. Therefore, we will not specifically refer to it in the Lemma, Proposition and Theorem statements.

In this section, we formulated an MDP parametrized by a known parameter θ. In Section 2.3.2, we define the optimal policy in the adaptive setting, i.e., where θ is an unknown (fixed) parameter.

2.3.2 Adaptive Learning with Sampling

From this point forward we take θ to be an unknown and fixed parameter. A control problem parametrized by an unknown parameter is called an *adaptive* control problem, we refer the reader to Hernández-Lerma [2012] for a comprehensive definition. As defined in Section 2.3.1, the reward density and transition density are parametrized by the (finitely-many) parameter values. Since we reserve θ for the true parameter, $f^{\gamma}(\cdot \mid x, u)$ and $q^{\gamma}(\cdot \mid x, u)$ represent the reward and transition density of an arbitrary parameter $\gamma \in \mathcal{P}$. Since θ is the underlying true parameter, $f^{\theta}(\cdot \mid x, u)$ and $q^{\theta}(\cdot \mid x, u)$ represent the reward and transition density which drive the actual (observed) process, i.e., the observed reward and the observed next state.

Next, we estimate the unknown parameter θ using a generic sampling algorithm.

Overview of Sampling Algorithm.

Had θ been known to the DM, the DM would have maximized the objective function by solving an optimization problem. Given the parameter uncertainty, the problem evolves into a learning problem, where the DM needs to choose suitable controls while gathering information on θ. To specify how the DM chooses controls in each period t, we first define the data available at the beginning of each period t, i.e., the history. We will use history and admissible history interchangeably. In contrast to the setting with known parameter θ, when θ is unknown the learning procedure makes use of the entire history, i.e., is not Markovian.

Figure 2.2 illustrates the random variables which drive the learning process, ordered in the sequence of occurrence. The DM observes the history information from period 0 up to period t, denoted by H_t. The history vector contains all information up to and including

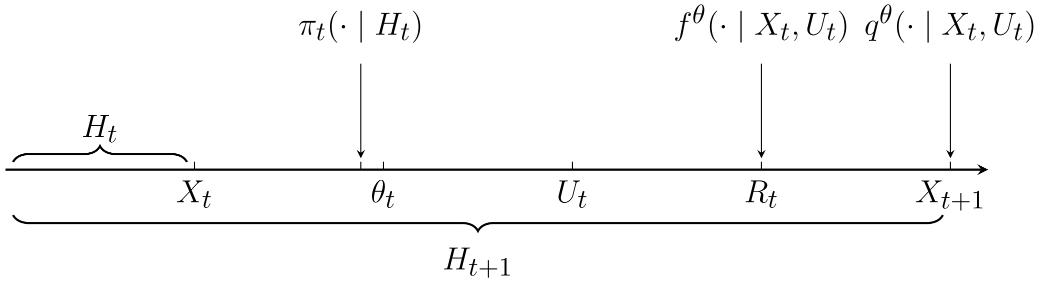


Figure 2.2: Evolution of the stochastic process, in the case when θ is not known.

X_t, i.e., the state in period t. Provided by the system designer, the function $\pi_t(\cdot \mid H_t)$ takes H_t as input and returns the distribution on the period-t sample Θ_t. Although $\pi_t(\cdot \mid H_t)$ is a deterministic function of H_t, it is random due to random H_t. Next, the DM draws a sample Θ_t from the parameter pool, with probability equal to $\pi_t(\Theta_t \mid H_t)$. The randomness of Θ_t arises from two sources: due to its dependence on random H_t through $\pi_t(\cdot \mid H_t)$, and the random nature of sampling. Given the history, the DM solves an optimization problem yielding the "optimal" control $U_t \in \mathcal{U}(X_t)$, assuming the true θ equals θ_t, the realized sample. The history includes only admissible controls, defined in Section 2.3.1. The state-control pair (X_t, U_t) gives rise to a random reward R_t, which depends on θ. R_t is drawn from the reward density $f^\theta(\cdot \mid X_t, U_t)$. Finally, given the transition density $q^\theta(\cdot \mid X_t, U_t)$ and the current state-control pair, the system transitions into a random next state X_{t+1}.

The collection of H_t, Θ_t, U_t, R_t and X_{t+1} constitutes H_{t+1}. The system designer provides $f^\gamma(\cdot \mid x, u)$ and $q^\gamma(\cdot \mid x, u)$ for all candidate values $\gamma \in \mathcal{P}$, which are utilized to update the belief vector, to be defined in Section 2.4. Nevertheless, the evolution of the process depends only on θ, i.e., nature acts according to $f^\theta(\cdot \mid x, u)$ and $q^\theta(\cdot \mid x, u)$. After observing the rewards and transitions in period t, the DM updates their belief vector $\pi_{t+1}(\cdot \mid H_{t+1})$ in period $t+1$. Conditional on the state-control pair, the reward and transitions are mutually independent.[5]

[5] Yet, there can be alternative settings where an exogenous random variable impacts both R_t and X_{t+1}, resulting in dependence between the reward and the next state.

Canonical Formulation.

The space of admissible histories up to period t is denoted by $\mathcal{H}_t$. For $t = 0$, we have $\mathcal{H}_0 \coloneqq \mathcal{X}$, i.e., $H_0 \coloneqq X_0$. For $t \geq 1$, the space of admissible histories has product form,

$$\mathcal{H}_t \coloneqq (\mathcal{X} \times \mathcal{P} \times \mathcal{U} \times \mathscr{R}_c)^t \times \mathcal{X} = \mathcal{H}_{t-1} \times \mathcal{P} \times \mathcal{U} \times \mathscr{R}_c \times \mathcal{X}.$$

The history spaces $\mathcal{H}_0$ and $\mathcal{H}_t$ $(t = 1, 2, \dots)$ are endowed with their product Borel σ-algebras $\mathcal{B}(\mathcal{X})$, $\mathcal{B}(\mathcal{P})$, $\mathcal{B}(\mathcal{U}(x))$ and $\mathcal{B}(\mathscr{R}_c)$. The history random variable follows the recursion

$$H_t \coloneqq (X_0, \Theta_0, U_0, R_0, \dots, X_{t-1}, \Theta_{t-1}, U_{t-1}, R_{t-1}, X_t), \tag{2.4}$$

with $(X_t, U_t) \in \mathbb{K}$, where $t \in \mathbb{N}_{\geq 2}$ denotes the period. The history vector's realization is denoted by

$$h_t \coloneqq (x_0, \theta_0, u_0, r_0, \dots, x_{t-1}, \theta_{t-1}, u_{t-1}, r_{t-1}, x_t),$$

similarly, with $(x_t, u_t) \in \mathbb{K}$. In addition, $\overline{\mathcal{H}}_0 \coloneqq \mathcal{H}_0 = \mathcal{X}$, and for $t \geq 1$, $\mathcal{H}_t$ is a subspace of

$$\overline{\mathcal{H}}_t \coloneqq (\mathcal{X} \times \mathcal{P} \times \mathcal{U} \times \mathscr{R}_c)^t \times \mathcal{X} = \mathcal{H}_{t-1} \times \mathcal{P} \times \mathcal{U} \times \mathscr{R}_c \times \mathcal{X}.$$

The infinite sequence of four-tuples $\overline{\mathcal{H}}_\infty$ is the sample space, denoted by Ω,

$$\Omega = \overline{\mathcal{H}}_\infty \coloneqq (\mathcal{X} \times \mathcal{P} \times \mathcal{U} \times \mathscr{R}_c)^\infty = \mathcal{X} \times \mathcal{P} \times \mathcal{U} \times \mathscr{R}_c \times \mathcal{X} \times \mathcal{P} \times \mathcal{U} \times \mathscr{R}_c \dots$$

Ω is the space of histories $\overline{H}_t = (X_0, \Theta_0, U_0, R_0, X_1, \Theta_1, U_1, R_1, \dots)$ with $X_t \in \mathcal{X}$, $\Theta_t \in \mathcal{P}$, $U_t \in \mathcal{U}$, $R_t \in \mathscr{R}_c$ for $t \geq 0$. The state, parameter, control and reward variables are defined as projections from Ω to sets $\mathcal{X}$, $\mathcal{P}$, $\mathcal{U}$ and $\mathscr{R}_c$, respectively.

A typical element of the sample space, $\omega \in \Omega$, is an infinite sequence of the form below:

$$\omega = (x_0, \theta_0, u_0, r_0, x_1, \theta_1, u_1, r_1, \dots) \text{ where } x_t \in \mathcal{X},\ \theta_t \in \mathcal{P},\ u_t \in \mathcal{U},\ r_t \in \mathscr{R}_c, \quad \forall\, t \geq 0.$$

Definition 2.3.1. The posterior distribution $(\pi_t(\cdot \mid H_t) : \mathcal{P} \mid \mathcal{H}_t \to \mathbb{R}_+)$ is a belief distribution over $\Theta_t \in \mathcal{P}$. It is a function of the random and time-dependent history H_t.

Definition 2.3.2. A randomized policy $\mu = \{\mu_t\}$ is a sequence of stochastic kernels μ_t on $\mathcal{U}$ given $\mathcal{H}_t$ and $\mathcal{P}$, satisfying

$$\mu_t(\mathcal{U}(x_t) \mid h_t, \theta_t) = 1 \text{ for all } h_t \in \mathcal{H}_t,\ \theta_t \in \mathcal{P}, \text{ and } t \geq 0.$$

where μ_t is an element of the set of admissible control policies, denoted by $\mathcal{M}$.

$\mathcal{B}(\Omega)$ is the corresponding product σ-algebra of Ω. The final element of the probability space is the probability measure $\mathbb{P}^{\mu,\theta}_{x_0} : \mathcal{B}(\Omega) \to [0, 1]$. It represents the probability measure when policy $\mu \in \mathcal{M}$ is used, the initial state is $X_0 = x_0$, and the true parameter is θ. The expectation operator with respect to $\mathbb{P}^{\mu,\theta}_{x_0}$ is $\mathbb{E}^{\mu,\theta}_{x_0}$. Whenever this expectation is taken, we take the random variables $(X_n, \Theta_n, U_n, R_n)_{\forall n}$ as generated by $\mathbb{P}^{\mu,\theta}_{x_0}$. We have a collection of sample paths that are parametrized by μ, θ and x_0. The operands of these operators are specified by the underlying state process induced by μ, θ and x_0. We emphasize that the space of admissible histories, $\mathcal{H}_t$, is contained in $\Omega = \overline{\mathcal{H}}_\infty$, and therefore, the (admissible) history random variable H_t is defined on $(\Omega, \mathcal{B}(\Omega), \mathbb{P}^{\mu,\theta}_{x_0})$.

A randomized, history-dependent policy μ induces a probability measure $\mathbb{P}^{\mu,\theta}_{x_0}$ on $(\Omega, \mathcal{B}(\Omega))$. By the Ionescu-Tulcea Theorem, proved in Proposition 7.28 of Shreve [1978], for any given policy $\mu = \{\mu_t\} \in \mathcal{M}$, any initial state $X_0 = x_0$ and true parameter $\theta \in \mathcal{P}$, there exists a unique probability measure $\mathbb{P}^{\mu,\theta}_{x_0}$ on $(\Omega, \mathcal{B}(\Omega))$, satisfying:

(a) $\mathbb{P}^{\mu,\theta}_{x_0}(\mathcal{H}_\infty) = 1$,

(b) $\mathbb{P}_{x_0}^{\mu,\theta}(X_0 = x_0) = 1$,

(c) $\mathbb{P}_{x_0}^{\mu,\theta}(\Theta_t = \theta_t \mid h_t) = \pi_t(\theta_t \mid h_t)$ for all $\theta_t \in \mathcal{P}$ given $h_t \in \mathcal{H}_t$ and $t \geq 0$,

(d) $\mathbb{P}_{x_0}^{\mu,\theta}(U_t \in \mathbf{U} \mid h_t, \theta_t) = \mu_t(\mathbf{U} \mid h_t, \theta_t)$ for all $\mathbf{U} \in \mathcal{B}(\mathcal{U})$ given $h_t \in \mathcal{H}_t$, $\theta_t \in \mathcal{P}$, and $t \geq 0$,

(e) $\mathbb{P}_{x_0}^{\mu,\theta}(R_t \in \mathbf{R} \mid h_t, \theta_t, u_t) = \int_{\mathbf{R}} f^\theta(r \mid x_t, u_t) d\lambda(r)$ for all $\mathbf{R} \in \mathcal{B}(\mathscr{R}_c)$ given $h_t \in \mathcal{H}_t$, $\theta_t \in \mathcal{P}$, $u_t \in \mathcal{U}(x_t)$, and $t \geq 0$. When conditioned on x_t and u_t, R_t is independent of θ_t and $h_t \setminus \{x_t\}$.

(f) $\mathbb{P}_{x_0}^{\mu,\theta}(X_{t+1} \in \mathbf{X} \mid h_t, \theta_t, u_t, r_t) = \int_{\mathbf{X}} q^\theta(y \mid x_t, u_t) d\eta(y)$ for all $\mathbf{X} \in \mathcal{B}(\mathcal{X})$ given $h_t \in \mathcal{H}_t$, $\theta_t \in \mathcal{P}$, $u_t \in \mathcal{U}(x_t)$, $r_t \in \mathscr{R}_c$, and $t \geq 0$. When conditioned on x_t and u_t, X_{t+1} is independent of θ_t, r_t and $h_t \setminus \{x_t\}$.

The probability measure $\mathbb{P}_{x_0}^{\mu,\theta}$ induced by the policy μ satisfies all (a)-(f), where

(a) is by the definition of probability measure.

(b) is by construction, it implies the initial state of the process is x_0 with probability 1.

(c) shows the history-dependent posterior distribution, from which the sample θ_t is generated.

(d) is the decision rule, i.e., the collection of policies. In each period t, the DM selects controls not only by the current state x_t, but by the entire history vector h_t. The decision rule also depends on the sample θ_t. Since h_t ends with x_t, we add θ_t as a condition.

(e) characterizes the random reward drawn from the distribution which knows the noisy version of θ. Given x_t and u_t, the random reward R_t, generated from density $f^\theta(\cdot \mid X_t, U_t)$, does not depend on the sample θ_t. Since θ_t does not give any additional information, it can be dropped. If not conditioned on x_t and u_t, then R_t depends on h_t, θ_t, and u_t.

(f) is the state transition law. Set $\mathbf{X}$ represents the states that are accessible from x_t. Given x_t and u_t, the random next state X_{t+1}, generated from density $q^{\theta}(\cdot \mid x_t, u_t)$, is independent of θ_t and r_t. Since θ_t and r_t are superfluous, they can be dropped. If not conditioned on x_t and u_t, then X_{t+1} depends on h_t, θ_t, u_t and r_t.

When the expectation is over one step instead of the entire process, we use a different notations, i.e., different than $\mathbb{E}_{x_0}^{\mu,\theta}$. If the expectation is taken with respect to the random reward, we denote it by $E_{f^\theta}[\cdot \mid x, u]$, where density f^θ is with respect to σ-finite measure λ. If the expectation is with respect to the random next state, the operator is $E_{q^\theta}[\cdot \mid x, u]$, where density q^θ is with respect to the σ-finite measure η. If it is with respect to both the random reward and the random next state, then we use $E_{f^\theta q^\theta}[\cdot \mid x, u]$. Whenever there is a $\cdot$ in these expectation operators, the $\cdot$ implies a random variable.

Objective Function (Performance Criteria).

Given the initial state x_0 and the discount factor $\beta \in (0, 1)$, the expected discounted reward over the infinite-horizon of implementing a policy μ from period $t = 0$ onward is

$$V_{x_0}^{\mu,\theta}(0) := \mathbb{E}_{x_0}^{\mu,\theta}\left[\sum_{t=0}^{\infty} \beta^t R_t\right], \quad \forall \mu \in \mathcal{M},\, x_0 \in \mathcal{X}. \tag{2.5}$$

When the rewards from period 0 up until $n-1$ are dismissed and the discounting starts from period n onward, we have

$$V_{x_0}^{\mu,\theta}(n) := \mathbb{E}_{x_0}^{\mu,\theta}\left[\sum_{t=n}^{\infty} \beta^{t-n} R_t\right]. \tag{2.6}$$

As in (2.5), the DM implements policy μ, starting from (known) state x_0 in period 0. However, the rewards in (2.6) are accumulated only from period $t = n$ onward, when the (random) system state is X_n. We will use the notation n when we fix a specific time period and t when we take summations.

For a given policy, the DM should know the value of θ to compute (2.5) and (2.6). Alternatively, the DM can statistically estimate these quantities if they have access to the reward density and transition density of the oracle to simulate the policy.

Recall from (2.3) that the optimal value function of the standard MDP problem is $\nu^{\theta}(x)$, with optimal policy μ^{θ}. In the adaptive setting, when the DM is assumed to know θ, the optimal policy is μ^{θ}. Hence, without loss of generality, we define $\nu^{\theta}(x_0)$ in (2.7) with respect to $\mathbb{E}_{x_0}^{\mu,\theta}$, i.e.,

$$\nu^{\theta}(x_0) \coloneqq \sup_{\mu \in \mathcal{M}} V_{x_0}^{\mu,\theta}(0) = \sup_{\mu \in \mathcal{M}} \mathbb{E}_{x_0}^{\mu,\theta}\left[\sum_{t=0}^{\infty} \beta^t R_t\right] = \mathbb{E}_{x_0}^{\mu^{\theta},\theta}\left[\sum_{t=0}^{\infty} \beta^t R_t\right], \quad \forall x_0 \in \mathcal{X}. \tag{2.7}$$

The process starts from state x_0 in period 0. We call μ^{θ} the θ-optimal policy. Under Assumption 0, this policy is stationary, deterministic, and Markovian.

2.4 Thompson Sampling

Given the formulation of the adaptive control problem, there is flexibility in how the evolution of the stochastic process can be "customized". The specification arises two-fold: by the posterior update and the control selection. Until now, we denoted a general admissible policy by $\mu = \{\mu_t\}$ and the belief update rule by $\pi_t(\cdot \mid h_t)$. From here onward, we adopt a particular policy, Thompson sampling (TS). We define the TS policy by specifying the decision rule μ and the function $\pi_t(\cdot \mid h_t)$ to update the belief vector. While the DM chooses μ, the system designer determines π_t. We use TS, TS algorithm, TS decision rule, and TS policy interchangeably throughout the paper. Also, we denote TS by τ, which we will formally define later in this section. The probability measure induced by TS and its corresponding expectation operator are denoted by $\mathbb{P}_{x_0}^{\tau,\theta}$ and $\mathbb{E}_{x_0}^{\tau,\theta}$, respectively.

2.4.1 Posterior Update

The TS algorithm generates an estimate θ_t in each period, by using the "synthetic" belief update function $\pi_t(\theta_t \mid H_t)$. Initially, the DM holds the prior belief $\pi_0(\theta_0 \mid h_0) > 0$ on the true parameter θ. That is, the unknown θ is modeled by the $|\mathcal{P}|$-valued random variable Θ_t, with initial prior distribution

$$\pi_0(\theta_0 \mid h_0) \coloneqq \mathbb{P}^{\tau,\theta}_{x_0}(\Theta_0 = \theta_0 \mid h_0), \quad \forall \theta_0 \in \mathcal{P}.$$

At the beginning of period t, the DM updates her belief over the parameter candidates, by computing the (random) posterior distribution

$$\pi_t(\theta_t \mid H_t) \coloneqq \mathbb{P}^{\tau,\theta}_{x_0}(\Theta_t = \theta_t \mid H_t), \quad \forall \theta_t \in \mathcal{P}. \tag{2.8}$$

The expected value of $\pi_t(\theta_t \mid H_t)$,

$$\pi_t(\theta_t) \coloneqq \mathbb{E}^{\tau,\theta}_{x_0}[\pi_t(\theta_t \mid H_t)],$$

is deterministic, and its dependence on (x_0, τ, θ) is implicit. We employ Bayes' Theorem to conduct the update,

$$\pi_t(\theta_t \mid H_t) \coloneqq \frac{\mathcal{L}^{\theta_t}_t(H_t)\pi_0(\theta_0 \mid h_0)}{\sum\limits_{\gamma \in \mathcal{P}} \mathcal{L}^{\gamma}_t(H_t)\pi_0(\gamma \mid h_0)}, \tag{2.9}$$

where $\mathcal{L}^{\gamma}_t(H_t) : \mathcal{H}_t \to \mathbb{R}$ is the (history-dependent) likelihood function. For any $\gamma \in \mathcal{P}$,

$$\mathcal{L}^{\gamma}_t(H_t) \coloneqq \prod_{s=1}^{t} f^{\gamma}(R_{s-1} \mid X_{s-1}, U_{s-1}) q^{\gamma}(X_s \mid X_{s-1}, U_{s-1}).$$

The joint density $f^{\gamma}(\cdot \mid x,u)q^{\gamma}(\cdot \mid x,u)$ specifies a joint probability measure on $[0,1] \times \mathcal{X}$,

$$\rho^{\gamma}_{x,u}(\mathbf{R},\mathbf{X}) \coloneqq \int_{\mathbf{R}} f^{\gamma}(r \mid x,u)d\lambda(r) \int_{\mathbf{X}} q^{\gamma}(y \mid x,u)d\eta(y),$$

for $\mathbf{R} \subseteq \mathcal{B}(\mathscr{R}_c)$, $\mathbf{X} \subseteq \mathcal{B}(\mathcal{X})$. Then, for any parameter value $\gamma \in \mathcal{P}$, the ratio of the Radon-Nikodym derivative is

$$\frac{d\rho^{\theta}_{x,u}}{d\rho^{\gamma}_{x,u}} = \frac{f^{\theta}(\cdot \mid x,u)q^{\theta}(\cdot \mid x,u)}{f^{\gamma}(\cdot \mid x,u)q^{\gamma}(\cdot \mid x,u)}.$$

Definition 2.4.1. The relative entropy of $\rho^{\theta}_{x,u}$ with respect to $\rho^{\gamma}_{x,u}$ is

$$\mathcal{K}(\rho^{\theta}_{x,u} \mid \rho^{\gamma}_{x,u}) \coloneqq E_{f^{\theta}q^{\theta}}\left[\log\left(\frac{d\rho^{\theta}_{x,u}}{d\rho^{\gamma}_{x,u}}\right)\right],$$

given $\rho^{\theta}_{x,u}$ is absolutely continuous with respect to $\rho^{\gamma}_{x,u}$.

Note that the expectation operator $E_{f^{\theta}q^{\theta}}[\cdot \mid x,u]$ was defined earlier in Section 2.3.2, such that we integrate over the random reward and next state, for one step only. Next, we specify the decision rule to characterize the TS policy.

2.4.2 Decision Rule

Definition 2.4.2. The Thompson sampling policy $\tau = \{\tau_t\}$ is a sequence of stochastic kernels τ_t on $\mathcal{U}$ given h_t and θ_t, satisfying

$$\tau_t(\cdot \mid h_t, \theta_t) \coloneqq \mu^{\theta_t}(\cdot \mid x_t).$$

Under Assumption 0, μ^{θ_t} is a stationary, deterministic, and Markovian policy. Since μ^{θ_t} only depends on x_t, (by definition) the τ policy depends on x_t, instead of h_t. In each period t, the TS decision rule samples θ_t and employs μ^{θ_t}, i.e., it picks the control that maximizes the expected infinite-horizon discounted reward by treating θ_t as the true value of the unknown parameter θ. The TS decision rule is deterministic, given θ_t. However, until the (random and history-dependent) sample Θ_t is drawn from the posterior distribution $\pi_t(\theta_t \mid H_t)$, it is a randomized decision rule.

Recall the evolution of the stochastic process from Section 2.3.2. Based on the state-control pair, the DM observes a noisy reward generated by $f^{\theta}(\cdot \mid x_t, u_t)$, and thus, cannot immediately identify the true value θ. Then, through the transition density $q^{\theta}(\cdot \mid x_t, u_t)$, the current state transitions into the next state. After observing the reward and the transition, the DM updates the posterior on every $\gamma \in \mathcal{P}$ using (2.9). Afterwards, a new parameter estimate θ_{t+1} is sampled from the updated distribution, leading to the next control u_{t+1}.

We underline that the history vector includes the sample, i.e., $H_t \coloneqq (X_0, \Theta_0, U_0, R_0, \dots)$ contains the period-t sample θ_t, which renders the TS policy well defined on the probability space $(\Omega, \mathcal{B}(\Omega), \mathbb{P}^{\tau,\theta}_{x_0})$, defined in Section 2.3.2.

Lemma 2.4.1 (Degenerate prior)**.** *TS is equivalent to the θ-optimal policy when the prior distribution is degenerate, i.e., $\pi_0(\theta \mid h_0) = 1$.*

We defer the proof of Lemma 2.4.1 to Section 2.9.1.

2.5 Decomposition of Expected Regret

The objective of this section is to decompose the standard regret into interpretable components and illustrate how the third component, i.e., residual regret, decays to 0 through a numerical example. Because we are in an infinite-horizon setting, we can consider expected regret in any given fixed period n in the future, as assessed in period 0. In what follows, we decompose this "expected regret process" into three different components, two of which that cannot be changed from period n onward and a third which can be influenced. The first component is the traditional notion of expected finite-horizon regret.

2.5.1 Standard Notion of Expected Regret

One would naturally quantify the expected regret of an admissible policy by taking the difference between the optimal value function and the TS policy's value function. By the definitions in (2.5) and (2.7), this difference is equal to

$$
\begin{aligned}
E[\text{Regret}^{\theta}(0,\infty)] &\coloneqq \nu^{\theta}(x_0) - V_{x_0}^{\tau,\theta}(0) \\
&= \mathbb{E}_{x_0}^{\mu^{\theta},\theta}\left[\sum_{t=0}^{\infty}\beta^t R_t\right] - \mathbb{E}_{x_0}^{\tau,\theta}\left[\sum_{t=0}^{\infty}\beta^t R_t\right], \qquad (2.10)
\end{aligned}
$$

where $(0,\infty)$ represents the starting and ending periods, inclusive. The expected infinite-horizon regret $E[\text{Regret}^{\theta}(0,\infty)]$ is a function of two different expectation operators, so the expectation "E" represents a label rather than a formal mathematical expression. We emphasize that even though R_t appears in both terms of (2.10), one of them is driven by the process generated by μ^{θ}, while the other one is driven by τ.

By construction, $E[\text{Regret}^{\theta}(0,\infty)]$ can be partitioned into two components; a finite component that tallies rewards up until some period $n-1$, and the remainder that goes into infinity, i.e.,

$$
E[\text{Regret}^{\theta}(0,\infty)] \coloneqq E[\text{Regret}^{\theta}(0,n-1)] + E[\text{Regret}^{\theta}(n,\infty)]. \qquad (2.11)
$$

When $\beta \in [0,1)$, the difference in (2.10) is the expected value of the regret felt by the DM discounted back to period 0, i.e., in period-0 "dollars". To express the regret of the θ-MDP in period-n dollars, we multiply (2.10) with β^{-n}, to obtain

$$
E[\text{Regret}_n^{\theta}(0,\infty)] \coloneqq (\nu^{\theta}(x_0) - V_{x_0}^{\tau,\theta}(0))\beta^{-n} \qquad (2.12)
$$

$$
= \left[\mathbb{E}_{x_0}^{\mu^{\theta},\theta}\left[\sum_{t=0}^{\infty}\beta^t R_t\right] - \mathbb{E}_{x_0}^{\tau,\theta}\left[\sum_{t=0}^{\infty}\beta^t R_t\right]\right]\beta^{-n} \qquad (2.13)
$$

$$= \left[\mathbb{E}_{x_0}^{\mu^\theta,\theta} \left[\sum_{t=0}^{n-1} \beta^t R_t \right] - \mathbb{E}_{x_0}^{\tau,\theta} \left[\sum_{t=0}^{n-1} \beta^t R_t \right] \right] \beta^{-n} \tag{2.14}$$

$$+ \left[\mathbb{E}_{x_0}^{\mu^\theta,\theta} \left[\sum_{t=n}^{\infty} \beta^t R_t \right] - \mathbb{E}_{x_0}^{\tau,\theta} \left[\sum_{t=n}^{\infty} \beta^t R_t \right] \right] \beta^{-n}. \tag{2.15}$$

The subscript n in $E[\text{Regret}_n^\theta(0,\infty)]$ corresponds to the period at which the money is evaluated. This shows that the decomposition (2.11) also applies to the regret expressed in any period-n dollars.

$$E[\text{Regret}_n^\theta(0,\infty)] := E[\text{Regret}_n^\theta(0,n-1)] + E[\text{Regret}_n^\theta(n,\infty)].$$

When $\beta = 1$, there is no difference between (2.10) and (2.13). We decompose (2.13) into two components, namely (2.14) and (2.15). In particular, (2.14) represents the expected finite-horizon regret, and by dividing it by β^n we convert it to period-n dollars. We denote it by $E[\text{Regret}_n^\theta(0,n-1)]$. Recall that μ^θ is an optimal policy for the infinite-horizon discounted reward maximization problem, with value $\nu^\theta(x_0)$. When $\beta = 1$, $E[\text{Regret}_n^\theta(0,n-1)]$ becomes the traditional expected regret over a finite horizon. We observe from Example 2.2.1 that the non-discounted version of $E[\text{Regret}_n^\theta(0,n-1)]$ violates the order of $\ln(n)$. The expected regret component (2.15) can be interpreted as the expected infinite-horizon regret, starting in period n, in period-n dollars. Distributing β^{-n} inside, (2.15) can alternatively be written as

$$\mathbb{E}_{x_0}^{\mu^\theta,\theta} \left[\sum_{t=n}^{\infty} \beta^{t-n} R_t \right] - \mathbb{E}_{x_0}^{\tau,\theta} \left[\sum_{t=n}^{\infty} \beta^{t-n} R_t \right]. \tag{2.16}$$

By the first thesis-v1.pdfremark in Section 2.3.1, the infinite geometric series property, and assuming $\beta \in [0,1)$, (2.16) is upper bounded by $\frac{2M}{1-\beta}$, which is a constant independent of n. When $\beta = 1$, (2.16) may grow to infinity in n.

We revisit Example 2.2.1, which was introduced in Section 2.2.2. The value of the τ policy $V_{x_0}^{\tau,\theta}(0)$ is equal to the probability that the initial sample is equal to the true parameter θ

times the infinite-horizon reward of having the first guess right. In the case of sampling the true parameter at $t = 0$, the DM earns a reward of 1 in all periods, except for the first period ($t = 0$). In this case, the total discounted reward accrued starting at $t = 1$ is $\frac{1}{1-\beta}$. By the definition of $V_{x_0}^{\tau,\theta}(0)$, we convert the total reward into period-0 dollars by multiplying $\frac{1}{1-\beta}$ with β. Therefore,

$$V_{x_0}^{\tau,\theta}(0) = \pi_0(\theta \mid h_0)\frac{\beta}{1-\beta}.$$

The value of the θ-optimal policy in period-0 dollars is

$$\nu^\theta(x_0) = \frac{\beta}{1-\beta}.$$

Hence, by (2.13) the expected infinite-horizon regret of the τ policy is

$$E[\text{Regret}_n^\theta(0,\infty)] := (\nu^\theta(x_0) - V_{x_0}^{\tau,\theta}(0))\beta^{-n} = \left((1-\pi_0(\theta \mid h_0))\frac{\beta}{1-\beta}\right)\beta^{-n}.$$

We now inspect the limiting behavior of this metric (as $n \to \infty$). When the discount factor $0 \leq \beta < 1$, we have

$$\lim_{n\to\infty} E[\text{Regret}_n^\theta(0,\infty)] = \lim_{n\to\infty}\left((1-\pi_0(\theta \mid h_0))\frac{\beta}{1-\beta}\right)\beta^{-n} = \infty. \tag{2.17}$$

The expected finite-horizon regret (2.14) of the τ policy is

$$\begin{aligned}
E[\text{Regret}_n^\theta(0,n-1)] &:= \left(\mathbb{E}_{x_0}^{\mu^\theta,\theta}\left[\sum_{t=0}^{n-1}\beta^t R_t\right] - \mathbb{E}_{x_0}^{\tau,\theta}\left[\sum_{t=0}^{n-1}\beta^t R_t\right]\right)\beta^{-n} \\
&= \left(\sum_{t=1}^{n-1}\beta^t - \pi_0(\theta \mid h_0)\sum_{t=1}^{n-1}\beta^t\right)\beta^{-n} \\
&= \left(\sum_{t=1}^{n-1}\beta^t(1-\pi_0(\theta \mid h_0))\right)\beta^{-n} \qquad (2.18) \\
&= \frac{\beta(1-\beta^{n-1})\beta^{-n}}{1-\beta}(1-\pi_0(\theta \mid h_0)) = \frac{\beta^{1-n}-1}{1-\beta}(1-\pi_0(\theta \mid h_0)).
\end{aligned}$$

Similar to (2.17), we inspect the limiting behavior of the expected finite-time regret, which yields

$$\lim_{n\to\infty} E[\text{Regret}_n^\theta(0, n-1)] = \lim_{n\to\infty} \frac{\beta^{1-n} - 1}{1-\beta}(1 - \pi_0(\theta \mid h_0)) = \infty.$$

When $\beta = 1$ (2.18) is equal to $(n-1)(1-\pi_0(\theta \mid h_0))$, thus the expected finite-time regret grows linearly, not logarithmically as in Agrawal and Goyal [2012].

2.5.2 Components of Expected Regret

We now propose an alternative decomposition of $E[\text{Regret}_n^\theta(0, \infty)]$. By construction, the second term in (2.16) is equal to the value function of the τ policy, i.e., $V_{x_0}^{\tau,\theta}(n)$. Moreover, the first term in (2.16) is equal to the expectation of the optimal value function with respect to the θ-optimal policy, i.e.,

$$\mathbb{E}_{x_0}^{\mu^\theta,\theta}[\nu^\theta(X_n)] := \mathbb{E}_{x_0}^{\mu^\theta,\theta}\left[\mathbb{E}_{X_n}^{\mu^\theta,\theta}\left[\sum_{t=0}^{\infty}\beta^t R_t'\right]\right] = \mathbb{E}_{x_0}^{\mu^\theta,\theta}\left[\sum_{t=n}^{\infty}\beta^{t-n} R_t\right]. \tag{2.19}$$

In (2.19), X_n is generated by running the μ^θ policy starting in period $t = 0$ from state x_0. Conditional on the random "starting" state X_n, $\{R_0', R_1', \dots\}$ is the random reward process generated by the optimal policy μ^θ. When unconditioned on X_n, $R_t' \sim R_{t+n}$ given starting state x_0. Adding and subtracting $\mathbb{E}_{x_0}^{\tau,\theta}[\nu^\theta(X_n)]$, and regrouping yields the following decomposition.

$$\begin{aligned}
&E[\text{Regret}_n^\theta(0, \infty)] \\
&:= (\nu^\theta(x_0) - V_{x_0}^{\tau,\theta}(0))\beta^{-n} \\
&= E[\text{Regret}_n^\theta(0, n-1)] && \text{(Expected finite-time regret)} \quad (2.20) \\
&\quad + \mathbb{E}_{x_0}^{\mu^\theta,\theta}[\nu^\theta(X_n)] - \mathbb{E}_{x_0}^{\tau,\theta}[\nu^\theta(X_n)] && \text{(Expected state regret)} \quad (2.21)
\end{aligned}$$

$$+\, \mathbb{E}_{x_0}^{\tau,\theta}[\nu^{\theta}(X_n)] - V_{x_0}^{\tau,\theta}(n) \qquad \text{(Expected residual regret).} \qquad (2.22)$$

Notice that (2.20) is the same quantity as (2.14). Next, we discuss (2.21) and (2.22).

Expected State Regret.

The expected state regret (2.21) captures the unavoidable future consequences of landing in a suboptimal state after implementing TS for n periods, which we formally define below.

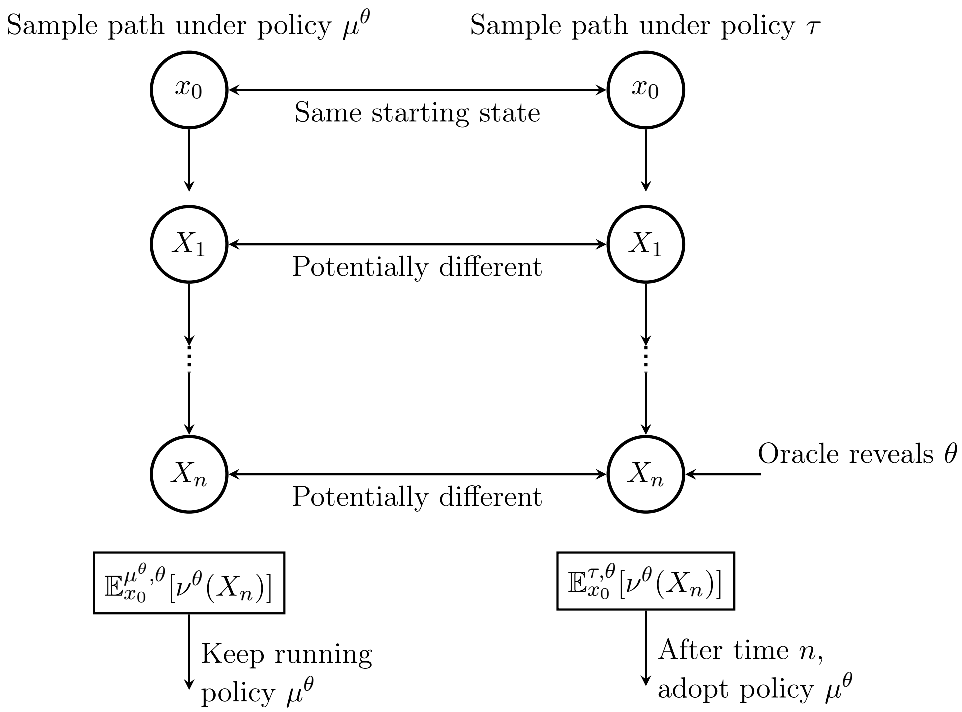
Definition 2.5.1. The expected state regret,

$$\mathcal{S}_{x_0}^{\tau,\theta}(n) \coloneqq \mathbb{E}_{x_0}^{\mu^{\theta},\theta}[\nu^{\theta}(X_n)] - \mathbb{E}_{x_0}^{\tau,\theta}[\nu^{\theta}(X_n)],$$

is the expected forward-looking regret under an optimal policy from period n onward. It quantifies the difference between landing in a random state X_n when the τ policy is started in period 0 versus a potentially different random state X_n, had an optimal policy been followed instead from the starting period.

Although it tallies the difference in rewards starting from period n into the infinite future, we only denote the starting period n inside the parenthesis.

The random state X_n in $\mathbb{E}_{x_0}^{\mu^{\theta},\theta}[\nu^{\theta}(X_n)]$ is induced by running the μ^{θ} policy for n periods, whereas the random X_n in $\mathbb{E}_{x_0}^{\tau,\theta}[\nu^{\theta}(X_n)]$ arises under the τ policy. Starting from the respective random states X_n, both sample paths follow the θ-optimal policy. Figure 2.3 illustrates two representative sample paths. If there is no state process or if the state process is iid, then these two terms would be equal. Since our problem involves a nontrivial state process, the DM who implements the τ policy may end up in a "bad" part of the state space, leading to an unavoidable, positive penalty. Recall that $\nu^{\theta}(\cdot)$ is the optimal value function of the DM who *knows* the true θ. After period n, the DM cannot do any better than μ^{θ}.



Sample path under policy μ^θ
Sample path under policy τ
x_0
x_0
Same starting state
X_1
X_1
Potentially different
$\vdots$
X_n
X_n
Potentially different
Oracle reveals θ
$\mathbb{E}_{x_0}^{\mu^\theta,\theta}[\nu^\theta(X_n)]$
$\mathbb{E}_{x_0}^{\tau,\theta}[\nu^\theta(X_n)]$
Keep running policy μ^θ
After time n, adopt policy μ^θ

Figure 2.3: Illustration of two different sample paths giving rise to the expected state regret.

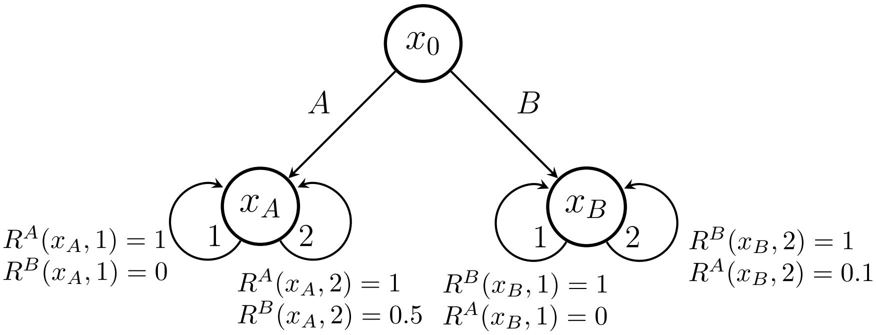


Figure 2.4: Deterministic reward depending on the control, where true parameter is B.

Example 2.5.1 (Absorption into an unfavorable set of states)**.** Consider Figure 2.4. Similar to Example 2.2.1, in period $n = 0$, the DM can choose either A or B with an immediate reward of 0. After being absorbed into one of x_A or x_B, the DM can pick either control 1 or 2 and receives a deterministic reward, as a function of the control and true parameter. Suppose the true parameter θ is B, which is unknown to the DM who performs TS.

If $\pi(A \mid x_0) \approx 1$, then the DM would initially sample A, and consequently would stay in x_A forever. The τ policy may then pick control 1 (generating a reward of 0) or pick control 2 (generating a reward of 0.5). Recall that the system designer provides the transition and reward densities, i.e., the reward structures $R^A(\cdot)$ and $R^B(\cdot)$ are both known to the DM. Since the rewards are deterministic and have different values, regardless of the control picked at $n = 1$, the τ policy will immediately learn that $\theta = B$. Hence, τ will always pick control 2 from $n = 2$ onward, incurring a reward of 0.5. The θ-optimal policy will pick parameter B in period $n = 0$, receiving a reward of 1 forever. Then, by (2.21), the undiscounted expected total regret of being in state x_A, as opposed to x_B, increases linearly, by $1 - 0.5 = 0.5$ in each period. In contrast, the final term (2.22) equals 0 when $n \geq 2$. Once landing in state x_A, after 1 period both TS and an optimal policy will choose the same control, i.e., control 2, forever. However, (2.22) does not always converge to 0 as rapidly. In the next section, we illustrate that when the rewards are not deterministic, the τ policy learns more slowly.

Expected Residual Regret.

Now, we will formally introduce the third component of the expected infinite-horizon expected regret, i.e., (2.22), and study its asymptotic behavior.

Definition 2.5.2. The expected residual regret, i.e., $\mathcal{R}^{\tau,\theta}_{x_0}(n)$, is the expected forward-looking regret from period n onward into the infinite future. This regret is between a policy which implements τ until it switches to the optimal policy μ^θ in period n as opposed to continuing with τ. Formally,

$$\mathcal{R}^{\tau,\theta}_{x_0}(n) := \mathbb{E}^{\tau,\theta}_{x_0}[\nu^\theta(X_n)] - V^{\tau,\theta}_{x_0}(n). \tag{2.23}$$

Similar to the expected state regret, we only denote the starting period n inside the parenthesis. Consider the first term of (2.23),

$$\mathbb{E}_{x_0}^{\tau,\theta}[\nu^{\theta}(X_n)] := \mathbb{E}_{x_0}^{\tau,\theta}\left[\sup_{\mu\in\mathcal{M}} V_{X_n}^{\mu,\theta}(0)\right] = \mathbb{E}_{x_0}^{\tau,\theta}\left[\sup_{\mu\in\mathcal{M}} \mathbb{E}_{X_n}^{\mu,\theta}\left[\sum_{t=0}^{\infty}\beta^t R_t\right]\right].$$

The θ-optimal policy starts from a random state X_n, which is driven by running the τ policy for n periods, starting at x_0. Recall Sections 2.3.2 and 2.4 for details. The expectation is taken over all paths leading to all possible X_n. Hence, $\mathbb{E}_{x_0}^{\tau,\theta}[\nu^{\theta}(X_n)]$ is a deterministic quantity. Consider the second term of (2.23), $V_{x_0}^{\tau,\theta}(n)$, which is equal to (2.6) by substituting the τ policy,

$$V_{x_0}^{\tau,\theta}(n) := \mathbb{E}_{x_0}^{\tau,\theta}\left[\sum_{t=n}^{\infty}\beta^{t-n} R_t\right].$$

The expectation operator is induced by τ, the starting state x_0 and the true parameter θ. Hence, similar to the first term, $V_{x_0}^{\tau,\theta}(n)$ does not forget the past; periods 0 to $(n-1)$ impact the state wherein the process finds itself in period n. Both $\mathbb{E}_{x_0}^{\tau,\theta}[\nu^{\theta}(X_n)]$ and $V_{x_0}^{\tau,\theta}(n)$ discard the rewards generated during the first n periods; however, they are not independent of the past, since the random state X_n is driven by the tuple (x_0, τ, θ). In summary, we decompose the standard regret into three components, i.e.

$$E[\text{Regret}^{\theta}(0,\infty)] = E[\text{Regret}_n^{\theta}(0,n-1)] + \mathcal{S}_{x_0}^{\tau,\theta}(n) + \mathcal{R}_{x_0}^{\tau,\theta}(n). \qquad (2.24)$$

Notice that in (2.24) $\mathcal{S}_{x_0}^{\tau,\theta}(n)$ and $\mathcal{R}_{x_0}^{\tau,\theta}(n)$ tally the difference in rewards starting from n into infinity, while $E[\text{Regret}^{\theta}(0,\infty)]$ and $E[\text{Regret}_n^{\theta}(0,n-1)]$ specify both the starting and ending periods inside the parenthesis. $E[\text{Regret}_n^{\theta}(0,n-1)]$ is the accumulation of the past losses, before period n. This component of the expected regret is sunk, in the sense that the DM cannot change it starting from period n. Similarly, the expected state regret $\mathcal{S}_{x_0}^{\tau,\theta}(n)$ is the accumulation of the future losses, as a result of irrevocably being in a given state in period n, thus is also sunk. $E[\text{Regret}_n^{\theta}(0,n-1)]$ and $\mathcal{S}_{x_0}^{\tau,\theta}(n)$ measure the past and future consequences, respectively, of adopting TS from period 0 up until period n. From the perspective of a DM who is already in period n, neither can be influenced. In contrast, we regard the expected residual regret $\mathcal{R}_{x_0}^{\tau,\theta}(n)$ as "controllable" because, starting in period n, a DM can choose a policy other than TS from that period onward. Therefore, amongst the three components, only expected residual regret represents the efficacy of continuing with τ into the future.

Next, we illustrate how the expected residual regret quantifies the effectiveness of future decisions through an example. If the DM picks the best control(s) from period n onward, no matter how unfavorable the state X_n is, the expected residual regret starting in that period is 0.

Illustration of Expected Residual Regret.

To illustrate how learning occurs in a setting with stochastic rewards, consider a single-state θ-MDP example. We show how the expected residual regret is driven down to 0 as a result of τ learning over time.

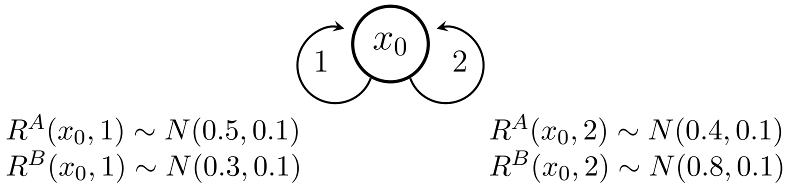

x_0
1
2
$R^A(x_0,1) \sim N(0.5, 0.1)$
$R^B(x_0,1) \sim N(0.3, 0.1)$
$R^A(x_0,2) \sim N(0.4, 0.1)$
$R^B(x_0,2) \sim N(0.8, 0.1)$

Figure 2.5: Stochastic rewards depending on the control, where the true parameter is B.

Example 2.5.2 (Expected residual regret converges to 0)**.** Consider the single-state θ-MDP in Figure 2.5. The underlying reward structure is illustrated on the arcs, which represent the controls. The rewards are stochastic. Based on the true parameter (which we assume is B) and the control picked at every step, the reward is generated from a normal distribution with known mean and variance. Not knowing the current state, the DM draws a sample from the posterior distribution in each period and obtains either $\Theta_t = A$ or $\Theta_t = B$. The initial prior belief on A is 0.5. In each step of the process, if the sample drawn is A, then the DM picks control 1. This is because control 1 has an expected reward of 0.5 when sampling A, while control 2 has an expected reward of 0.4. On the other hand, if the DM samples B, then they pick control 2 since the corresponding expected reward is higher.

Figure 2.6a illustrates the evolution of the expected belief on the "wrong" parameter $\mathbb{E}_{x_0}^{\tau,B}[\pi_n(A)]$, i.e., the expected posterior sampling error, averaged over 100 runs of TS policy. We observe that the DM learns the true parameter B not immediately, but after approximately 40 iterations. Beyond that point, the DM always picks control 2, in order to maximize the infinite-horizon expected total reward. Figure 2.6b illustrates the decline in the expected residual regret for the same example.

Consider a hybrid policy that switches from τ to μ^B after running τ for n periods. Since the true parameter is B, the reward distributions follow $N(0.3, 0.1)$ and $N(0.8, 0.1)$ when control 1 or 2 is selected, respectively, i.e., the realizations of $R^B(x_0, 1)$ and $R^B(x_0, 2)$ are the rewards associated with control 1 and 2, respectively. Therefore, the expected reward of control 2 is greater than the expected reward of control 1 and the μ^B policy always picks control 2, i.e.,

$$\int r f^B(x_0, 2)\, dr = 0.8 > \int r f^B(x_0, 1)\, dr = 0.3.$$

From period n onward, a hybrid policy gains total expected reward of

$$\begin{aligned}\mathbb{E}_{x_0}^{\tau,B}[\nu^B(X_n)] &= \mathbb{E}_{x_0}^{\tau,B}\left[\mathbb{E}_{X_n}^{\mu^B,B}\left[\sum_{t=0}^{\infty} \beta^t R_t\right]\right] \\ &= \sum_{t=0}^{\infty} \beta^t 0.8 = \frac{0.8}{1-\beta} = \nu^B(x_0).\end{aligned}$$

We emphasize that n corresponds to a fixed time period, while t is used when summing over the rewards. Consider the μ_1 policy that always picks control 1. Had the DM implemented such a policy, the expected residual regret would have been

$$\begin{aligned}\mathcal{R}_{x_0}^{\mu_1,B}(n) &\coloneqq \mathbb{E}_{x_0}^{\mu_1,B}[\nu^B(X_n)] - V_{x_0}^{\mu_1,B}(n) \\ &= \frac{0.8}{1-\beta} - \mathbb{E}_{x_0}^{\mu_1,B}\left[\sum_{t=n}^{\infty}\beta^{t-n}R_t\right] \\ &= \sum_{t=0}^{\infty}\beta^t 0.8 - \sum_{t=n}^{\infty}\beta^{t-n}0.3 = \sum_{t=0}^{\infty}\beta^t 0.5 = \frac{0.5}{1-\beta}.\end{aligned}$$

Thus, $\frac{0.5}{1-\beta}$ is an upper bound on the expected residual regret for all n. After some number of periods, the rate at which the policy τ picks control 1 becomes negligible. Once the policy no longer picks control 1, the DM has figured out the true parameter is B. If we call this period $\tilde{n}$, the expected residual regret becomes

$$\mathcal{R}_{x_0}^{\tau,B}(n) \coloneqq \mathbb{E}_{x_0}^{\tau,B}[\nu^B(X_n)] - V_{x_0}^{\tau,B}(n) = \sum_{t=0}^{\infty}\beta^t 0.8 - \sum_{t=n}^{\infty}\beta^{t-n}0.8 = 0, \quad \forall n \geq \tilde{n}.$$

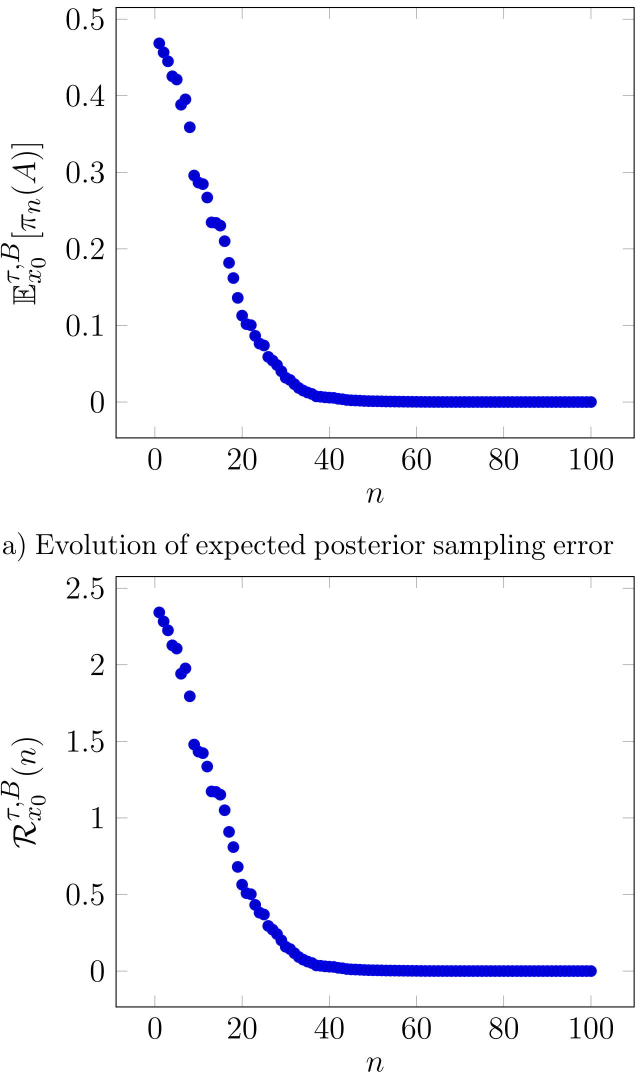
Starting from $n = 0$ and using Example 2.5.2 assumptions, the expected residual regret can be computed as

$$\mathcal{R}_{x_0}^{\tau,B}(n) = \sum_{t=0}^{\infty}\beta^t 0.8 - \left(\mathbb{E}_{x_0}^{\tau,B}[\pi_n(A)]\frac{0.3}{1-\beta} + \mathbb{E}_{x_0}^{\tau,B}[\pi_n(B)]\frac{0.8}{1-\beta}\right).$$

Figure 2.6b shows the evolution of $\mathcal{R}_{x_0}^{\tau,B}$ for $\beta = 0.9$. When a different $\beta \in [0, 1)$ is chosen, the trajectory remains the same, but the expected residual regret values are different for small n.

2.6 Analysis of Expected Residual Regret for Thompson Sampling

The goal of this section is to show that the expected residual regret of TS vanishes (i.e., converges to 0) in an exponential rate. To do so, we first relate the concept of ADO [Hernández-



(a) Evolution of expected posterior sampling error

(b) Evolution of expected residual regret

Figure 2.6: Evolution of expected posterior and expected residual regret in Example 2.5.2 when first sample is wrong

Lerma, 2012] to the expected residual regret, showing that they are equivalent in our setting. This allows us to combine the machinery of Hernández-Lerma [2012] with Kim [2017] to obtain our result.

2.6.1 ADO and Expected Residual Regret

To study adaptive control problems in the discounted case, Schäl [1987] introduced an asymptotic definition of optimality, asymptotic discount optimality. Hernández-Lerma [2012] describes the idea behind it as "to allow the system to run during a learning period of n stages" and defines an asymptotically discount optimal (ADO) policy as follows:

Definition 2.6.1 (ADO)**.** A policy μ is called *asymptotically discount optimal* (ADO) if,

$$\left|V_{x_0}^{\mu,\theta}(n) - \mathbb{E}_{x_0}^{\mu,\theta}[\nu^\theta(X_n)]\right| \to 0 \text{ as } n \to \infty, \quad \forall x_0 \in \mathcal{X},$$

where $\nu^\theta(X_n)$ is defined by (2.7) and $V_{x_0}^{\mu,\theta}(n)$ by (2.6).

We label an ADO policy as θ-ADO to emphasize the dependence on the underlying parameter θ. Traditionally, expected regret is formulated without the absolute value [Lai and Robbins, 1985]. It follows that the expected residual regret is equivalent to the θ-ADO expression, i.e.,

$$\mathcal{R}_{x_0}^{\tau,\theta}(n) := \mathbb{E}_{x_0}^{\tau,\theta}[\nu^\theta(X_n)] - V_{x_0}^{\tau,\theta}(n) = |V_{x_0}^{\tau,\theta}(n) - \mathbb{E}_{x_0}^{\mu,\theta}[\nu^\theta(X_n)]|. \tag{2.25}$$

Showing (2.25) will be instrumental in bounding the expected residual regret.

Lemma 2.6.1. *The absolute value in the θ-ADO expression can be omitted in our setting, i.e.,*

$$|V_{x_0}^{\tau,\theta}(n) - \mathbb{E}_{x_0}^{\tau,\theta}[\nu^\theta(X_n)]| = \mathbb{E}_{x_0}^{\tau,\theta}[\nu^\theta(X_n)] - V_{x_0}^{\tau,\theta}(n).$$

The statement in Lemma 2.6.1 is self-evidently true, but because the notation is cumbersome in this setting, we defer the proof to Section 2.9.1.

Rather than calling policies θ-ADO, we say that they have "vanishing expected residual regret".

Definition 2.6.2. A policy μ has vanishing expected residual regret if,

$$\lim_{n\to\infty} \mathcal{R}_{x_0}^{\mu,\theta}(n) := \lim_{n\to\infty} \left[\mathbb{E}_{x_0}^{\mu,\theta}[\nu^\theta(X_n)] - V_{x_0}^{\mu,\theta}(n)\right] = 0, \quad \forall x_0 \in \mathcal{X},$$

where θ is the true parameter and X_n is the random period-n state, which is obtained by running an admissible policy μ for n periods.

If TS has vanishing expected residual regret, then eventually its expected performance converges to that of an optimal policy.

2.6.2 Temporal Difference Error

To bound the expected residual regret, we first establish a connection to the temporal-difference error function [Sutton and Barto, 2018]. This will later allow us to provide a bound for the former by bounding the latter. This function, which is parametrized by θ, quantifies the discrepancy between the reward-to-go of choosing the optimal control instead of an arbitrary control in a given state.

Definition 2.6.3 (Hernández-Lerma, 2012)**.** We denote the temporal-difference error function by $\phi^\theta : \mathbb{K} \to \mathbb{R}$, where

$$\phi^\theta(x,u) := r^\theta(x,u) + \beta \int \nu^\theta(y) Q^\theta(dy \mid x,u) - \nu^\theta(x).$$

The first two terms of $\phi^\theta(x,u)$ constitute the reward-to-go of choosing (an arbitrary) control $u \in \mathcal{U}(x)$ in state x, while $\nu^\theta(x)$ is the reward-to-go of choosing the optimal control in x.

Lemma 2.6.2 (Temporal-difference error)**.** *For every initial state $x_0 \in \mathcal{X}$, a policy μ has vanishing expected residual regret, i.e.,*

$$\lim_{n\to\infty} \left[\mathbb{E}_{x_0}^{\mu,\theta}[\nu^\theta(X_n)] - V_{x_0}^{\mu,\theta}(n)\right] = 0, \quad \forall x_0 \in \mathcal{X},$$

if and only if $\phi^\theta(X_t, U_t) \to 0$ *in probability-*$\mathbb{P}_{x_0}^{\mu,\theta}$ *for every* $x_0 \in \mathcal{X}$.

We defer the proof of Lemma 2.6.2 to Section 2.9.1, adapted from Hernández-Lerma [2012] into our setting. The proof establishes a connection between the expected residual regret and the expected value of ϕ^θ. In particular, for a policy μ,

$$\mathcal{R}_{x_0}^{\mu,\theta}(n) = -\sum_{t=n}^{\infty} \beta^{t-n} \mathbb{E}_{x_0}^{\mu,\theta} \phi^\theta(X_t, U_t). \tag{2.26}$$

2.6.3 Expected Residual Regret Bounds

In this section, we provide convergence results for TS. We will bound the expected residual regret by bounding $\mathbb{E}_{x_0}^{\tau,\theta}[\phi^\theta(X_t, U_t) \mid \theta_t \neq \theta]$ and $\mathbb{E}_{x_0}^{\tau,\theta}[\phi^\theta(X_t, U_t)]$, respectively. The DM draws a sample in each period t, represented by the random variable Θ_t. Recall from Section 2.4.1, given the true parameter $\theta \in \mathcal{P}$,

$$\pi_t(\theta \mid H_t) = \mathbb{P}_{x_0}^{\tau,\theta}(\Theta_t = \theta \mid H_t) \tag{2.27}$$

is the probability that the sample Θ_t is equal to θ, conditional on H_t. Since the condition is not a known h_t, (2.27) is also a random variable. Similarly,

$$1 - \pi_t(\theta \mid H_t) = \mathbb{P}_{x_0}^{\tau,\theta}(\Theta_t \neq \theta \mid H_t) \tag{2.28}$$

is the (random) probability that the sample Θ_t is *not* equal to θ, given (random) H_t. Taking the expectation of both sides of (2.28),

$$\mathbb{E}_{x_0}^{\tau,\theta}[1 - \pi_t(\theta \mid H_t)] = \mathbb{E}_{x_0}^{\tau,\theta}[\mathbb{P}_{x_0}^{\tau,\theta}(\Theta_t \neq \theta \mid H_t)], \tag{2.29}$$

resolves the uncertainty of H_t. By the law of iterated expectations, (2.29) simplifies into

$$1 - \pi_t(\theta) = \mathbb{P}_{x_0}^{\tau,\theta}(\Theta_t \neq \theta), \tag{2.30}$$

i.e., the posterior probability that the sample is *not* equal to the true parameter θ. Then, by the law of total expectation, we partition the expectation of ϕ^θ, i.e.,

$$\begin{aligned}\mathbb{E}_{x_0}^{\tau,\theta}[\phi^\theta(X_t, U_t)] = \mathbb{P}_{x_0}^{\tau,\theta}(\Theta_t \neq \theta)\mathbb{E}_{x_0}^{\tau,\theta}[\phi^\theta(X_t, U_t) \mid \Theta_t \neq \theta] \\ +\mathbb{P}_{x_0}^{\tau,\theta}(\Theta_t = \theta)\mathbb{E}_{x_0}^{\tau,\theta}[\phi^\theta(X_t, U_t) \mid \Theta_t = \theta],\end{aligned}$$

where U_t is the control that maximizes the reward-to-go by treating the sampled estimate Θ_t as the true value of the unknown parameter θ. We can simply rewrite the probability terms to obtain,

$$\begin{aligned}\mathbb{E}_{x_0}^{\tau,\theta}[\phi^\theta(X_t, U_t)] = (1 - \pi_t(\theta))\mathbb{E}_{x_0}^{\tau,\theta}[\phi^\theta(X_t, U_t) \mid \Theta_t \neq \theta] \\ +\pi_t(\theta)\mathbb{E}_{x_0}^{\tau,\theta}[\phi^\theta(X_t, U_t) \mid \Theta_t = \theta].\end{aligned}$$

But then, when $\Theta_t = \theta$, by the decision rule $\tau_t(U_t \mid h_t, \theta)$, the optimal control (of state x_t) is taken, and the temporal difference error $\phi^\theta(X_t, U_t)$ becomes 0, by definition. Hence, when $\Theta_t = \theta$,

$$\mathbb{E}_{x_0}^{\tau,\theta}[\phi^\theta(X_t, U_t)] = (1 - \pi_t(\theta))\mathbb{E}_{x_0}^{\tau,\theta}[\phi^\theta(X_t, U_t) \mid \Theta_t \neq \theta]. \tag{2.31}$$

Note that, although we express the τ policy as a sequence of stochastic kernels, it is a deterministic policy by Assumption 0, when conditioned on θ_t. Before presenting the main result of this section, we first bound the expectation of $\phi^\theta(X_t, U_t)$ in Lemma 2.6.3, then extend a result from Kim [2017] into our setting by Lemma 2.6.4.

Lemma 2.6.3 (Lower bound on expected ϕ^θ)**.** *The expected value of $\phi^\theta(X_t, U_t)$, conditional* *on $\Theta_t \neq \theta$, is lower bounded by a non-positive constant, i..e,*

$$\mathbb{E}_{x_0}^{\tau,\theta}[\phi^\theta(X_t, U_t) \mid \Theta_t \neq \theta] \geq -2M\left(\frac{1+\beta}{1-\beta}\right), \tag{2.32}$$

where $M \geq 0$ is the upper bound on the absolute value of the expected reward, per Remark .

Proof of Lemma 2.6.3. In this proof, we slightly modify the U_t notation to indicate its dependence on the sample. Let $U_t^{\Theta_t}$ denote the control picked when sample Θ_t is drawn from $\mathcal{P}$, and let U_t^θ represent the control picked when knowing θ. Otherwise, we cannot distinguish the U_t controls.

$$\begin{aligned}\mathbb{E}_{x_0}^{\tau,\theta}[\phi^\theta(X_t, U_t^{\Theta_t}) \mid \Theta_t \neq \theta] =& \mathbb{E}_{x_0}^{\tau,\theta}[R_t(X_t, U_t^{\Theta_t}) + \beta \int \nu^\theta(x_{t+1}) q^\theta(dx_{t+1} \mid X_t, U_t^{\Theta_t})\, d\eta \mid \Theta_t \neq \theta] \\ & - \mathbb{E}_{x_0}^{\tau,\theta}[\nu^\theta(X_t) \mid \Theta_t \neq \theta],\end{aligned}$$

which is equal to

$$\begin{aligned}&\mathbb{E}_{x_0}^{\tau,\theta}\left[R_t(X_t, U_t^{\Theta_t}) + \beta \int \nu^\theta(x_{t+1}) q^\theta(dx_{t+1} \mid X_t, U_t^{\Theta_t})\, d\eta \mid \Theta_t \neq \theta\right] \\ &\quad - \mathbb{E}_{x_0}^{\tau,\theta}\left[R_t(X_t, U_t^{\theta}) + \beta \int \nu^\theta(x_{t+1}) q^\theta(dx_{t+1} \mid X_t, U_t^{\theta})\, d\eta \mid \Theta_t \neq \theta\right].\end{aligned}$$

We can rearrange the above expression to obtain

$$\mathbb{E}_{x_0}^{\tau,\theta}[R_t(X_t, U_t^{\Theta_t}) \mid \Theta_t \neq \theta] - \mathbb{E}_{x_0}^{\tau,\theta}[R_t(X_t, U_t^{\theta}) \mid \Theta_t \neq \theta] \tag{2.33}$$

$$+ \mathbb{E}_{x_0}^{\tau,\theta}\left[\beta \int \nu^\theta(x_{t+1}) q^\theta(dx_{t+1} \mid X_t, U_t^{\Theta_t})\, d\eta \mid \Theta_t \neq \theta\right] \tag{2.34}$$

$$- \mathbb{E}_{x_0}^{\tau,\theta}\left[\beta \int \nu^\theta(x_{t+1}) q^\theta(dx_{t+1} \mid X_t, U_t^{\theta})\, d\eta \mid \Theta_t \neq \theta\right]. \tag{2.35}$$

Then, we introduce the function $g^{\theta} : \mathcal{X} \times \mathcal{P} \to \mathbb{R}_+$

$$g^{\theta}(x, \theta) \coloneqq \int \nu^{\theta}(y) q^{\theta}(dy \mid x, U_t^{\theta}) \, d\eta, \tag{2.36}$$

such that (2.34) is

$$\beta \mathbb{E}_{x_0}^{\tau,\theta}[g^{\theta}(X_t, \Theta_t) \mid \Theta_t \neq \theta]$$

and (2.35) is

$$-\beta \mathbb{E}_{x_0}^{\tau,\theta}[g^{\theta}(X_t, \theta) \mid \Theta_t \neq \theta].$$

By Remark and the infinite series property, we have

$$\left| \nu^{\theta}(x) \right| \leq \frac{M}{1-\beta}, \quad \forall x \in \mathcal{X}.$$

By (2.36),

$$\left| g^{\theta}(x, \theta) \right| \leq \frac{M}{1-\beta} \int q^{\theta}(dy \mid x, U_t^{\theta}) \, d\eta \leq \frac{M}{1-\beta}.$$

Taking the conditional expectation yields the same upper and lower bounds, i.e.,

$$\left| \mathbb{E}_{x_0}^{\tau,\theta}[g^{\theta}(x, \theta) \mid \Theta_t \neq \theta] \right| \leq \frac{M}{1-\beta}.$$

We lower bound (2.34)

$$\beta \mathbb{E}_{x_0}^{\tau,\theta}[g^{\theta}(X_t, \Theta_t) \mid \Theta_t \neq \theta] \geq \frac{-\beta M}{1-\beta},$$

and upper bound (2.35)

$$\beta \mathbb{E}_{x_0}^{\tau,\theta}[g^{\theta}(X_t, \theta) \mid \Theta_t \neq \theta] \leq \frac{\beta M}{1-\beta},$$

to get a lower bound on their difference,

$$\beta\mathbb{E}_{x_0}^{\tau,\theta}[g^\theta(X_t,\Theta_t) \mid \Theta_t \neq \theta] - \beta\mathbb{E}_{x_0}^{\tau,\theta}[g^\theta(X_t,\theta) \mid \Theta_t \neq \theta] \geq \frac{-2\beta M}{1-\beta}.$$

The difference between the first two reward terms, i.e., (2.33), can be no smaller than $-2M$, thus

$$\mathbb{E}_{x_0}^{\tau,\theta}[\phi^\theta(X_t,U_t^{\Theta_t}) \mid \Theta_t \neq \theta] \geq -2M + \frac{2\beta M}{1-\beta} = -2M\left(\frac{1+\beta}{1-\beta}\right).$$

□

Lemma 4 of Kim [2017] bounds the expected probability that θ is not selected in period t. We will now extend this result to the discounted infinite-horizon framework. However, for us to be able to generalize the finite-dimensional state and control spaces that Kim [2017] deals with into Borel (possibly infinite) spaces, we need to make the following assumption.

Assumption 1. We assume

$$\inf_{x\in\mathcal{X},u\in\mathcal{U},r\in\mathscr{R}_c} f^\gamma(r \mid x,u) > 0, \quad \forall\gamma\in\mathcal{P} \tag{2.37}$$

and

$$\inf_{x\in\mathcal{X},u\in\mathcal{U},y\in\mathcal{X}} q^\gamma(y \mid x,u) > 0, \quad \forall\gamma\in\mathcal{P}. \tag{2.38}$$

For any $x \in \mathcal{X}$, $u \in \mathcal{U}$, and any distinct parameter value, $\gamma \neq \theta \in \mathcal{P}$, there exists a positive constant $\epsilon(x,u,\theta,\gamma) > 0$ such that

$$\inf_{x\in\mathcal{X},u\in\mathcal{U}} \mathcal{K}(\rho_{x,u}^\theta \mid \rho_{x,u}^\gamma) > \epsilon(x,u,\theta,\gamma). \tag{2.39}$$

We underline that under finite state and control spaces, Assumption 1 simplifies into the prerequisites of Kim [2017]. As Kim [2017] explains, (2.39) in Assumption 1 ensures that given $\theta \neq \gamma$, the probability measures $\rho^{\theta}_{x,u}$ and $\rho^{\gamma}_{x,u}$ are distinguishable as measured by the relative entropy. In Section 2.9.2, we illustrate the implications of Assumption 1 through Example 2.5.2.

The following lemma extends a result from Kim [2017] to our setting.

Lemma 2.6.4 (Extension of Lemma 4 of Kim [2017])**.** *Under Assumption 1, Lemma 4 of Kim [2017] extends to our setting. That is to say, implementing TS and starting from any* $x_0 \in \mathcal{X}$*, there exists constants* a_θ*,* $b_\theta > 0$ *such that*

$$\mathbb{E}^{\tau,\theta}_{x_0}\left[1 - \pi_t(\theta \mid H_t)\right] \leq a_\theta e^{-b_\theta t}, \tag{2.40}$$

where a_θ *and* b_θ *are defined as in Kim [2017].*

Because Lemma 2.6.4 is an adaptation of Lemma 4 of Kim [2017] into our setting, we defer it to Section 2.9.1. By the law of iterated expectations, (2.40) simplifies into

$$1 - \pi_t(\theta) \leq a_\theta e^{-b_\theta t}. \tag{2.41}$$

We are now ready to introduce the main result of this section.

Proposition 1 (Upper bound on $\mathcal{R}^{\tau,\theta}_{x_0}(n)$)**.** When Assumption 1 holds, the expected residual regret converges to 0 exponentially fast.

$$\mathcal{R}^{\tau,\theta}_{x_0}(n) := \mathbb{E}^{\tau,\theta}_{x_0}[\nu^\theta(X_n)] - V^{\tau,\theta}_{x_0}(n) \;\leq\; \frac{2M(1+\beta)a_\theta e^{-b_\theta n}}{(1-\beta)^2},$$

where a_θ and b_θ are positive constants, defined in Lemma 4 of Kim [2017].

Proof of Proposition 1. Equations (2.31), (2.32) and (2.41) together yield

$$-\mathbb{E}_{x_0}^{\tau,\theta}[\phi^{\theta}(X_t,U_t)] \leq \frac{2M(1+\beta)a_{\theta}e^{-b_{\theta}t}}{1-\beta}. \tag{2.42}$$

By the proof of Lemma 2.6.2, $\phi^{\theta}(X_t,U_t)$ and $\mathcal{R}_{x_0}^{\tau,\theta}(n)$ are related, see (2.26). Then, by (2.42), the right-hand side of (2.26) is upper bounded,

$$-\sum_{t=n}^{\infty}\beta^{t-n}\mathbb{E}_{x}^{\tau,\theta}[\phi^{\theta}(X_t,U_t)] \leq \sum_{t=n}^{\infty}\beta^{t-n}\frac{2M(1+\beta)a_{\theta}e^{-b_{\theta}t}}{1-\beta}.$$

Since $b_{\theta} \geq 0$, we have $e^{-b_{\theta}t} \leq e^{-b_{\theta}n}$ for all $t \geq n$. Thus, the above equation becomes

$$-\sum_{t=n}^{\infty}\beta^{t-n}\mathbb{E}_{x}^{\tau,\theta}[\phi^{\theta}(X_t,U_t)] \leq \sum_{t=n}^{\infty}\beta^{t-n}\frac{2M(1+\beta)a_{\theta}e^{-b_{\theta}n}}{1-\beta}. \tag{2.43}$$

By (2.26) and applying the infinite geometric series formula to (2.43), we obtain

$$\begin{aligned}-\sum_{t=n}^{\infty}\beta^{t-n}\mathbb{E}_{x_0}^{\tau,\theta}[\phi^{\theta}(X_t,U_t)] &= \mathbb{E}_{x_0}^{\tau,\theta}[\nu^{\theta}(X_n)] - V_{x_0}^{\tau,\theta}(n) \\ &= \mathcal{R}_{x_0}^{\tau,\theta}(n) \leq \frac{2M(1+\beta)a_{\theta}e^{-b_{\theta}n}}{(1-\beta)^2},\end{aligned}$$

thus showing that the upper bound on the expected residual regret decays exponentially. □

2.7 Complete Learning and Probabilistic Residual Regret

The goal of this section is to define a probabilistic version of the expected residual regret (the expected value of the probabilistic version is the expected residual regret itself, i.e., $\mathcal{R}_{x_0}^{\tau,\theta}(n)$, and show the conditions under which it converges $\mathbb{P}_{x_0}^{\tau,\theta}$-almost surely to 0 in this framework. We show that the posterior distribution of TS converges $\mathbb{P}_{x_0}^{\tau,\theta}$-almost surely to a point mass at θ. This behavior is called *complete learning.*

2.7.1 Complete Learning

Kim [2017] has shown that the expected posterior sampling error of TS converges to 0 exponentially fast (see Lemma 2.6.4). Our result builds on this result by using the Dominated Convergence Theorem. Formally,

Assumption 2 (Existence of the limit of $\pi_t(\theta \mid H_t)$)**.** Suppose that $\lim_{t\to\infty} \pi_t(\theta \mid H_t)$ exists $\mathbb{P}^{\tau,\theta}_{x_0}$-a.s.

By imposing this assumption, we avoid the nonexistence of the limit due to oscillation. Under Assumption 1, the expected posterior sampling error of TS vanishes. We will show that this, i.e., $\lim_{t\to\infty} \mathbb{E}^{\tau,\theta}_{x_0}[\pi_t(\theta \mid H_t)] = 1$, and Assumption 2 together imply

$$\lim_{t\to\infty} \pi_t(\theta \mid H_t) = 1, \quad \mathbb{P}^{\tau,\theta}_{x_0}\text{-almost surely.}$$

The next lemma is the final piece to show the occurrence of complete learning.

Lemma 2.7.1. *Suppose that TS does not exhibit complete learning, i.e.,* $\lim_{t\to\infty} \pi_t(\theta \mid H_t) < 1$*,* $\mathbb{P}^{\tau,\theta}_{x_0}$*-a.s. Then, there exists an* $n \geq 1$ *s.t.*

$$\mathbb{P}^{\tau,\theta}_{x_0}\left(\lim_{t\to\infty} \pi_t(\theta \mid H_t) < 1 - \frac{1}{n}\right) > 0, \quad \mathbb{P}^{\tau,\theta}_{x_0}\text{-almost surely.}$$

The proof of Lemma 2.7.1 is in Section 2.9.1.

Theorem 2.7.2 (TS learns θ as $t \to \infty$)**.** *Suppose that Assumptions 1 and 2 hold. Then,*

$$\lim_{t\to\infty} \pi_t(\theta \mid H_t) = 1, \quad \mathbb{P}^{\tau,\theta}_{x_0}\text{-almost surely.}$$

Proof of Theorem 2.7.2. By Lemma 2.6.4, Assumption 2, and the Dominated Convergence

Theorem, it follows that

$$1 = \lim_{t\to\infty} \mathbb{E}_{x_0}^{\tau,\theta}[\pi_t(\theta \mid H_t)] = \mathbb{E}_{x_0}^{\tau,\theta}\left[\lim_{t\to\infty} \pi_t(\theta \mid H_t)\right], \quad \mathbb{P}_{x_0}^{\tau,\theta}\text{-a.s.}$$

Then,

$$\begin{aligned}
1 &= \mathbb{E}_{x_0}^{\tau,\theta}\left[\lim_{t\to\infty} \pi_t(\theta \mid H_t)\right] \\
&= \mathbb{E}_{x_0}^{\tau,\theta}\left[\lim_{t\to\infty} \pi_t(\theta \mid H_t)\middle| \left\{\lim_{t\to\infty} \pi_t(\theta \mid H_t) < 1 - \frac{1}{n}\right\}\right] \mathbb{P}_{x_0}^{\tau,\theta}\left(\lim_{t\to\infty} \pi_t(\theta \mid H_t) < 1 - \frac{1}{n}\right) \\
&\quad + \mathbb{E}_{x_0}^{\tau,\theta}\left[\lim_{t\to\infty} \pi_t(\theta \mid H_t)\middle| \left\{\lim_{t\to\infty} \pi_t(\theta \mid H_t) \geq 1 - \frac{1}{n}\right\}\right] \mathbb{P}_{x_0}^{\tau,\theta}\left(\lim_{t\to\infty} \pi_t(\theta \mid H_t) \geq 1 - \frac{1}{n}\right) \\
&\leq (1 - 1/n)\mathbb{P}_{x_0}^{\tau,\theta}\left(\lim_{t\to\infty} \pi_t(\theta \mid H_t) < 1 - \frac{1}{n}\right) + 1 - \mathbb{P}_{x_0}^{\tau,\theta}\left(\lim_{t\to\infty} \pi_t(\theta \mid H_t) < 1 - \frac{1}{n}\right) \\
&= 1 - \frac{1}{n}\mathbb{P}_{x_0}^{\tau,\theta}\left(\lim_{t\to\infty} \pi_t(\theta \mid H_t) < 1 - \frac{1}{n}\right), \quad \mathbb{P}_{x_0}^{\tau,\theta}\text{-a.s.}
\end{aligned}$$

However, by Lemma 2.7.1, $\lim_{t\to\infty} \pi_t(\theta \mid H_t) < 1$ implies

$$\begin{aligned}
1 &= \mathbb{E}_{x_0}^{\tau,\theta}\left[\lim_{t\to\infty} \pi_t(\theta \mid H_t)\right] \\
&\leq 1 - \frac{1}{n}\mathbb{P}_{x_0}^{\tau,\theta}\left(\lim_{t\to\infty} \pi_t(\theta \mid H_t) < 1 - \frac{1}{n}\right) < 1, \quad \mathbb{P}_{x_0}^{\tau,\theta}\text{-almost surely,}
\end{aligned}$$

leading to contradiction. For all $n \geq 1$, it must be that

$$\mathbb{P}_{x_0}^{\tau,\theta}\left(\lim_{t\to\infty} \pi_t(\theta \mid H_t) < 1 - \frac{1}{n}\right) = 0, \quad \mathbb{P}_{x_0}^{\tau,\theta}\text{-almost surely.}$$

Thus, we must have $\lim_{t\to\infty} \pi_t(\theta \mid H_t) = 1$, $\mathbb{P}_{x_0}^{\tau,\theta}$-almost surely. This completes the proof. □

2.7.2 Vanishing Probabilistic Residual Regret

We define the residual regret as the "probabilistic" version of the expected residual regret, which was defined in Section 2.5.2. In the remainder of this section, we use the terms probabilistic residual regret and residual regret interchangeably. The residual regret almost mimics the definition in (2.23); however, it is a random expectation, and is obtained by conditioning on the random history H_n, generated by running TS from period 0 to n.

Definition 2.7.1. The residual regret $\mathbb{R}_{x_0}^{\tau,\theta}(n)$ is a random expectation that represents the forward-looking regret from period n onward into the infinite future. This regret is between a policy which implements τ until it switches to the optimal policy μ^θ in period n as opposed to continuing with τ. Formally,

$$\begin{aligned}
\mathbb{R}_{x_0}^{\tau,\theta}(n) &:= \nu^\theta(X_n) - \mathbb{E}_{x_0}^{\tau,\theta}\left[\sum_{t=n}^{\infty}\beta^{t-n}R_t \mid H_n\right] && (2.44)\\
&= \sup_{\mu\in\mathcal{M}} \mathbb{E}_{x_0}^{\mu,\theta}\left[\sum_{t=n}^{\infty}\beta^{t-n}R_t \mid H_n\right] - \mathbb{E}_{x_0}^{\tau,\theta}\left[\sum_{t=n}^{\infty}\beta^{t-n}R_t \mid H_n\right] && (2.45)\\
&= \mathbb{E}_{x_0}^{\mu^\theta,\theta}\left[\sum_{t=n}^{\infty}\beta^{t-n}R_t \mid H_n\right] - \mathbb{E}_{x_0}^{\tau,\theta}\left[\sum_{t=n}^{\infty}\beta^{t-n}R_t \mid H_n\right], && (2.46)
\end{aligned}$$

where the conditioning on the random history vector is denoted explicitly.

The residual regret is constructed in a way such that its expected value is the expected residual regret itself, i.e.,

$$\mathcal{R}_{x_0}^{\tau,\theta}(n) = \mathbb{E}_{x_0}^{\tau,\theta}[\mathbb{R}_{x_0}^{\tau,\theta}(n)]. \qquad (2.47)$$

Recall the interpretation of the first term of (2.46); the stochastic process is driven by a "hybrid" policy that follows the τ policy for the first n periods, and then adopts the optimal policy μ^θ in period n after the oracle reveals θ. It is a random quantity due to the random starting state X_n in period n, induced by TS. The second term is the random expected infinite sum obtained by implementing TS, starting from random state X_n, and omitting the rewards from period 0 to $n-1$.

Next, we show that the residual regret converges to 0 in the limit $\mathbb{P}_{x_0}^{\tau,\theta}$-a.s., assuming it exists.

Assumption 3 (Existence of the limit of $\mathbb{R}_{x_0}^{\tau,\theta}(n)$)**.** Suppose that $\lim_{n\to\infty} \mathbb{R}_{x_0}^{\tau,\theta}(n)$ exists $\mathbb{P}_{x_0}^{\tau,\theta}$-a.s.

From (2.44) observe that the residual regret is the difference of two terms. Consider the first term $\nu^\theta(X_n)$. Under some technical conditions, if the limit of the state process $\{X_n\}$ exists, then the limit of $\nu^\theta(X_n)$ exists. However, since we do not impose any restrictions on the underlying chain structure and the underlying stochastic process is history-dependent, we cannot guarantee that $\lim_{n\to\infty} X_n$ exists, and thus, there could be cases where $\lim_{n\to\infty} \nu^\theta(X_n)$ does not converge[6]. Therefore, we directly assume that the limit of the residual regret exists. Similar to the expected residual regret, the following corollary shows that under certain conditions the probabilistic residual regret also vanishes in the limit.

Corollary 2.7.2.1. *When the conditions of Proposition 1 and Assumption 3 hold, the residual regret of TS vanishes, i.e.,*

$$\lim_{n\to\infty} \mathbb{R}_{x_0}^{\tau,\theta}(n) = 0, \quad \mathbb{P}_{x_0}^{\tau,\theta}\text{-almost surely.}$$

Proof of Corollary 2.7.2.1. Recall that the probabilistic residual regret converges $\mathbb{P}_{x_0}^{\tau,\theta}$-a.s. to 0 if there exists a null set $\mathbf{A} \in \mathcal{B}(\Omega)$ with $\mathbb{P}_{x_0}^{\tau,\theta}(\mathbf{A}) = 0$ such that the statement holds if $\omega \notin \mathbf{A}$. Given Assumption 3, it suffices to show that $\forall\ \epsilon > 0$, $\nexists\ \mathbf{A} \in \mathcal{B}(\Omega)$ with $\mathbb{P}_{x_0}^{\tau,\theta}(\mathbf{A}) = \epsilon > 0$ such that $\lim_{n\to\infty} \mathbb{R}_{x_0}^{\tau,\theta}(n) \neq 0\ \forall\, \omega \in \mathbf{A}$. We prove this by contradiction.

Suppose that $\exists\ \mathbf{A} \in \mathcal{B}(\Omega)$ with $\mathbb{P}_{x_0}^{\tau,\theta}(\mathbf{A}) = \epsilon' > 0$ for some $\epsilon' > 0$ such that $\lim_{n\to\infty} \mathbb{R}_{x_0}^{\tau,\theta}(n) \neq 0\ \forall\, \omega \in \mathbf{A}$. Since $\mathbb{R}_{x_0}^{\tau,\theta}(n) > 0$ by construction, this limit is strictly positive, i.e., $\lim_{n\to\infty} \mathbb{R}_{x_0}^{\tau,\theta}(n) > 0\ \forall\, \omega \in \mathbf{A}$.

Then, for any $\delta > 0$, we define $\mathbf{A}_\delta$ as the largest measurable subset of $\mathbf{A}$ such that $\mathbf{A}_\delta \subset \{\omega \in \mathcal{B}(\Omega) : \lim_{n\to\infty} \mathbb{R}_{x_0}^{\tau,\theta}(n) > \delta\}$. According to Assumption 3 and the supposition in

[6] We do not require aperiodicity; the chain could be periodic, leading to oscillating rewards.

the previous paragraph, $\exists\ \delta'$ such that $\mathbb{P}_{x_0}^{\tau,\theta}(\mathbf{A}_{\delta'}) = \epsilon' > 0$. Then, we can write

$$\lim_{n\to\infty} \mathcal{R}_{x_0}^{\tau,\theta}(n) = \lim_{n\to\infty} \mathbb{E}_{x_0}^{\tau,\theta}[\mathbb{R}_{x_0}^{\tau,\theta}(n)] \tag{2.48}$$

$$= \lim_{n\to\infty} \left\{ \int_{\mathbf{A}_{\delta'}} \mathbb{R}_{x_0}^{\tau,\theta}(n)\, d\mathbb{P}_{x_0}^{\tau,\theta}(\omega) + \int_{\mathcal{B}(\Omega)\backslash \mathbf{A}_{\delta'}} \mathbb{R}_{x_0}^{\tau,\theta}(n)\, d\mathbb{P}_{x_0}^{\tau,\theta}(\omega) \right\} \tag{2.49}$$

$$= \int_{\mathbf{A}_{\delta'}} \lim_{n\to\infty} \mathbb{R}_{x_0}^{\tau,\theta}(n)\, d\mathbb{P}_{x_0}^{\tau,\theta}(\omega) + \int_{\mathcal{B}(\Omega)\backslash \mathbf{A}_{\delta'}} \lim_{n\to\infty} \mathbb{R}_{x_0}^{\tau,\theta}(n)\, d\mathbb{P}_{x_0}^{\tau,\theta}(\omega) \tag{2.50}$$

$$> \delta'\epsilon' > 0, \tag{2.51}$$

which contradicts Proposition 1. By construction, the first term in (2.49) converges to a number strictly greater than 0, and the second term converges to some non-negative number. Because $\mathbb{R}_{x_0}^{\tau,\theta}(n)$ exists by Assumption 3 and is finite[7] , by the Dominated Convergence Theorem, we can express (2.49) as (2.50), But then, (2.51) contradicts with Proposition 1. This implies that our supposition cannot hold, and thus, the residual regret converges $\mathbb{P}_{x_0}^{\tau,\theta}$-a.s. to 0. □

2.8 Concluding Remarks and Discussion

Fixing a finite period n, we decompose the expected infinite-horizon regret into three components. Two of the components are not actionable by the DM. The first component is the expected regret of the past. The second is the regret expected to be accrued in the future due to the random state the θ-MDP is in period n, which is a consequence of implementing a possibly suboptimal policy and cannot be revoked by the DM. The third component, which we called the expected residual regret, captures what a rational DM would consider as expected regret. It determines whether the DM will do the best moving forward, given their state in period n. Hence, we evaluate TS by only this third component, i.e., the expected

[7] Since the rewards are finite-valued, in any period, the difference in rewards of the first and second terms is bounded by a constant. By the infinite geometric series property, and assuming $\beta \in [0, 1)$, $\mathbb{R}_{x_0}^{\tau,\theta}(n)$ is finite.

residual regret.

We reiterate that our results hold for chain settings where $q^{\gamma}(\cdot \mid x, u)$ is strictly positive $\forall \gamma \in \mathcal{P}$, which rules out chains with absorbing states. Yet, by construction, the expected residual regret is a viable concept independent of the underlying chain structure; it is applicable to any structure. Hence, deriving the performance of the residual regret in broader settings is a potential direction of further research.

In our setting, we show that the expected residual regret of TS decays exponentially to 0 in the worst case (Proposition 1). We explore the conditions under which the posterior sampling error converges to 0 with almost sure probability (Theorem 2.7.2). At the crux of these theorems are certain sufficient conditions, which we present in Assumption 2, which rule out oscillating functions that would prevent complete learning. Finally, we show that a probabilistic version of the expected residual regret vanishes (Corollary 2.7.2.1), similar to its expected-valued version. At the crux of this corollary is the sufficiency condition that we present in Assumption 3. A future direction of research is to study the implications of Assumptions 2 and 3 in specific problem contexts. In addition to conditions on the underlying chain structure, complete learning may be a necessary but not sufficient condition for Assumption 3 to hold.

To the best of our knowledge, the notion of an ADO policy from the adaptive learning literature had not been connected to TS. By leveraging connections between these settings, we offer a novel concept of learning. Under some mild assumptions, we provide guarantees of learning in both expected and probabilistic senses in settings with general state and control spaces, using the infinite-horizon discounted reward criterion. An interesting extension of the setup in this paper is to analyze the performance of the novel concept of learning (i.e., expected residual regret) under broader settings, e.g., under general chain structures.

2.9 Appendix

2.9.1 Proofs of Lemmas

Proof of Lemma 2.4.1. When the Bayesian update is conducted with a degenerate distribution, it returns a degenerate distribution. We show this by induction. We have $\pi_0(\theta \mid h_0) = 1$ and assume $\pi_n(\theta \mid H_n) = 1$.

$$\begin{aligned}\pi_{n+1}(\theta \mid H_{n+1}) :=& \frac{\mathcal{L}^\theta(H_{n+1})\pi_0(\theta \mid h_0)}{\sum_{\gamma\in\mathcal{P}} \mathcal{L}^\gamma(H_{n+1})\pi_0(\gamma \mid h_0)} \\ =& \frac{\mathcal{L}^\theta(H_{n+1})\pi_0(\theta \mid h_0)}{\mathcal{L}^\theta(H_{n+1})\pi_0(\theta \mid h_0) + \sum_{\gamma\neq\theta\in\mathcal{P}} \mathcal{L}^\gamma(H_{n+1})\pi_0(\gamma \mid h_0)} = \frac{\mathcal{L}^\theta(H_{n+1})}{\mathcal{L}^\theta(H_{n+1})} = 1.\end{aligned}$$

Since the posterior distribution is degenerate, TS always samples the true parameter θ from the parameter space. Therefore, the DM who runs the τ policy ends up implementing the θ-optimal policy. □

Proof of Lemma 2.6.1. Consider the θ-ADO statement, $|V^{\mu,\theta}_{x_0}(n) - \mathbb{E}^{\mu,\theta}_{x_0}[\nu^\theta(X_n)]|$. It can be rewritten as

$$\begin{aligned}&|V^{\mu,\theta}_{x_0}(n) - \mathbb{E}^{\mu,\theta}_{x_0}[\nu^\theta(X_n)]| \qquad (2.52)\\ &= \left|\mathbb{E}^{\tau,\theta}_{x_0} \sum_{t=n}^{\infty} \beta^{t-n} R_t - \mathbb{E}^{\tau,\theta}_{x_0}[\nu^\theta(X_n)]\right| \\ &= \left|\mathbb{E}^{\tau,\theta}_{x_0}\left[\mathbb{E}^{\tau,\theta}_{x_0}\left(\sum_{t=n}^{\infty} \beta^{t-n} R_t \mid X_n\right) - \mathbb{E}^{\tau,\theta}_{x_0}[\nu^\theta(X_n) \mid X_n]\right]\right| \\ &= \left|\mathbb{E}^{\tau,\theta}_{x_0}\left[\mathbb{E}^{\tau,\theta}_{x_0}\left(\sum_{t=n}^{\infty} \beta^{t-n} R_t \mid X_n\right) - \nu^\theta(X_n)\right]\right| \\ &= \left|\mathbb{E}^{\tau,\theta}_{x_0}\left[\mathbb{E}^{\tau,\theta}_{x_0}\left(\sum_{t=n}^{\infty} \beta^{t-n} R_t(X_t,U_t) \mid X_n\right) - \sup_{\mu\in\mathcal{M}} \mathbb{E}^{\mu,\theta}_{X_n}\left[\sum_{t=0}^{\infty} \beta^t R_t(X_t,U_t)\right]\right]\right|. \qquad (2.53)\end{aligned}$$

The second equality is by the law of iterated expectations, and the fourth by substituting in the definition of the optimal value function. Evidently, the first term of (2.53) is upper bounded by the second term. Thus, (2.53) is equal to

$$\mathbb{E}_{x_0}^{\tau,\theta}\left[\sup_{\mu\in\mathcal{M}}\mathbb{E}_{X_n}^{\mu,\theta}\left[\sum_{t=0}^{\infty}\beta^t R_t(X_t,U_t)\right]-\mathbb{E}_{x_0}^{\tau,\theta}\left(\sum_{t=n}^{\infty}\beta^{t-n}R_t(X_t,U_t)\mid X_n\right)\right],$$

for any initial state $x_0\in\mathcal{X}$. Rewriting the optimal value function in closed form, we obtain

$$\mathbb{E}_{x_0}^{\tau,\theta}\left[\nu^\theta(X_n)-\mathbb{E}_{x_0}^{\tau,\theta}\left(\sum_{t=n}^{\infty}\beta^{t-n}R_t(X_t,U_t)\mid X_n\right)\right],$$

which is equal to

$$\mathbb{E}_{x_0}^{\tau,\theta}[\nu^\theta(X_n)]-\mathbb{E}_{x_0}^{\tau,\theta}\left[\sum_{t=n}^{\infty}\beta^{t-n}R_t(X_t,U_t)\right]=\mathbb{E}_{x_0}^{\tau,\theta}[\nu^\theta(X_n)]-V_{x_0}^{\tau,\theta}(n).$$

□

Proof of Lemma 2.6.2. This proof is an adaptation of Theorem 3.6 of Hernández-Lerma [2012]. By the definition in Section 2.6.1,

$$\phi^\theta(X_t,U_t)=\mathbb{E}_{x_0}^{\mu,\theta}[R_t(X_t,U_t)+\beta\nu^\theta(X_{t+1})-\nu^\theta(X_t)\mid H_t,U_t],$$

for any initial state $x_0\in\mathcal{X}$, admissible policy $\mu\in\mathcal{M}$, true parameter value θ, $t\geq 0$. For any $t\geq 1$, history in period t is $h_t=(x_0,\theta_0,a_0,r_0,x_1,\theta_1,a_1,r_1,\ldots,x_t)$. Multiplying by β^{t-n} yields

$$\beta^{t-n}\phi^\theta(X_t,U_t)=\mathbb{E}_{x_0}^{\mu,\theta}[\beta^{t-n}R_t(X_t,U_t)+\beta^{t-n+1}\nu^\theta(X_{t+1})-\beta^{t-n}\nu^\theta(X_t)\mid H_t,U_t].$$

Taking the expectation of both sides and by the law of total expectation, we obtain

$$\mathbb{E}_{x_0}^{\mu,\theta}[\beta^{t-n}\phi^\theta(X_t,U_t)]=\mathbb{E}_{x_0}^{\mu,\theta}[\beta^{t-n}R_t(X_t,U_t)+\beta^{t-n+1}\nu^\theta(X_{t+1})-\beta^{t-n}\nu^\theta(X_t)].$$

Summing over all $t \geq n$ gives

$$\sum_{t=n}^{\infty} \beta^{t-n} \mathbb{E}_{x_0}^{\mu,\theta}[\phi^\theta(X_t, U_t)] = \sum_{t=n}^{\infty} \mathbb{E}_{x_0}^{\mu}[\beta^{t-n} R_t(X_t, U_t)] \\ + \mathbb{E}_{x_0}^{\mu,\theta}\left[\sum_{t=n}^{\infty} (\beta^{t-n+1}\nu^\theta(X_{t+1}) - \beta^{t-n}\nu^\theta(X_t))\right].$$

The above simplifies into

$$\sum_{t=n}^{\infty} \beta^{t-n} \mathbb{E}_{x_0}^{\mu,\theta}[\phi^\theta(X_t, U_t)] = V_{x_0}^{\mu,\theta}(n) - \mathbb{E}_{x_0}^{\mu,\theta}[\nu^\theta(X_n)].$$

Recalling the result of Lemma 2.6.1 as,

$$-\sum_{t=n}^{\infty} \beta^{t-n} \mathbb{E}_{x_0}^{\mu,\theta}[\phi^\theta(X_t, U_t)] = \mathbb{E}_{x_0}^{\mu,\theta}[\nu^\theta(X_n)] - V_{x_0}^{\mu,\theta}(n).$$

In the limit, μ has vanishing expected residual regret (is θ-ADO); that is, for every $x_0 \in \mathcal{X}$,

$$\lim_{n\to\infty} (\mathbb{E}_{x_0}^{\mu,\theta}[\nu^\theta(X_n)] - V_{x_0}^{\mu,\theta}(n)) = 0,$$

if and only if, for every $x_0 \in \mathcal{X}$,

$$\lim_{n\to\infty} -\sum_{t=n}^{\infty} \beta^{t-n} \mathbb{E}_{x_0}^{\mu,\theta}[\phi^\theta(X_t, U_t)] = 0.$$

This is equivalent to, for every $x_0 \in \mathcal{X}$,

$$\lim_{t\to\infty} -\mathbb{E}_{x_0}^{\mu,\theta}[\phi^\theta(X_t, U_t)] = 0.$$

Hence, a policy μ has vanishing expected residual regret if and only if, for every $x_0 \in \mathcal{X}$,

$$\lim_{t\to\infty} \mathbb{E}_{x_0}^{\mu,\theta}[\phi^\theta(X_t, U_t)] = 0.$$

By Theorem 4.1.4 of Chung [2001], if $\mathbb{E}_{x_0}^{\mu,\theta}[\phi^\theta(X_t, U_t)]$ converges to 0, then $\phi^\theta(X_t, U_t)$ converges to 0 in probability-$\mathbb{P}_{x_0}^{\mu,\theta}$, for every $x_0 \in \mathcal{X}$, proving the forward direction of Lemma 2.6.2. It remains to show the reverse direction to complete the proof. By the same theorem of Chung [2001], for a uniformly bounded sequence $\{\phi^\theta\}$, convergence in probability and $\mathcal{L}^p$ (in expectation) are equivalent. By the first remark in Section 2.3.1, it follows that $\{\phi^\theta\}$ is bounded by some finite number. Hence, $\phi^\theta(X_t, U_t) \to 0$ in probability-$\mathbb{P}_{x_0}^{\mu,\theta}$ implies $\mathbb{E}_{x_0}^{\mu,\theta}[\phi^\theta(X_t, U_t)] \to 0$ as $t \to 0$. □

Proof of Lemma 2.6.4. We walk the reader through the proof of Kim [2017] while using our notation. The proof initially defines the stochastic process, for any $\gamma \neq \theta$,

$$Z_t^\gamma = \sum_{s=0}^{t} \log \Lambda_s^\gamma,$$

where, using our notation,

$$\begin{aligned}
\Lambda_0^\gamma &= 1 \\
\Lambda_s^\gamma &= \frac{f^\theta(R_{s-1} \mid X_{s-1}, U_{s-1}) q^\theta(X_s \mid X_{s-1}, U_{s-1})}{f^\gamma(R_{s-1} \mid X_{s-1}, U_{s-1}) q^\gamma(X_s \mid X_{s-1}, U_{s-1})},
\end{aligned}$$

for $0 < s \leq t$. Next, Kim [2017] defines filtration $(\mathcal{H}_t : t \geq 0)$ by $\mathcal{H}_t = \sigma(H_t)$, where H_t is by the definition in (2.4). Then, it follows that the stochastic process Z_t^γ is a submartingale with respect to $\mathcal{H}_t$ under probability measure $\mathbb{P}_{x_0}^{\tau,\theta}$. It is crucial that Z_t^γ is a submartingale. Kim [2017] decomposes it into an $\mathcal{H}_t$ martingale under $\mathbb{P}_{x_0}^{\tau,\theta}$,

$$M_t^\gamma := \sum_{s=0}^{t} (\log \Lambda_s^\gamma - \mathbb{E}_{x_0}^{\tau,\theta}[\log \Lambda_s^\gamma \mid \mathcal{H}_{s-1}]),$$

and a predictable process,

$$A_t^\gamma := \sum_{s=0}^{t} \mathbb{E}_{x_0}^{\tau,\theta}[\log \Lambda_s^\gamma \mid \mathcal{H}_{s-1}].$$

We explain how the proof extends to our setting through Assumption 1. The argument is twofold. The first requirement is that the increments of M_t^γ are bounded above and below by some d, i.e.,

$$|\log \Lambda_s^\gamma - \mathbb{E}_{x_0}^{\tau,\theta}[\log \Lambda_s^\gamma \mid \mathcal{H}_{s-1}]| \leq d, \tag{2.54}$$

for some $d > 0$. To see that (2.54) holds in our setting, note that by (2.37) and (2.38), i.e., $f^\gamma(r \mid x, u)$ and $q^\gamma(y \mid x, u)$ are bounded away from 0, we have that $|\log f^\gamma(r \mid x, u)| < \infty$ and $|\log q^\gamma(y \mid x, u)| < \infty$. This satisfies (2.54), which is needed for Azuma's inequality, a crucial step of Kim [2017]'s proof, to hold. The second requirement of the adaptation of the proof is

$$\sum_{s=0}^{t} \mathbb{E}_{x_0}^{\tau,\theta}[\log \Lambda_s^\gamma \mid \mathcal{H}_{s-1}] \geq \epsilon t, \tag{2.55}$$

i.e., (2.55) is an increasing predictable process. By (2.39), it follows that

$$\mathbb{E}_{x_0}^{\tau,\theta}[\log \Lambda_s^\gamma \mid \mathcal{H}_{s-1}] > \epsilon(x, u, \theta, \gamma) := \epsilon, \quad \forall s \leq t, \tag{2.56}$$

i.e., each increment of A_t^γ is strictly positive, which satisfies (2.55). The reader is referred to Kim [2017] for the details of why (2.56) holds. □

Proof of Lemma 2.7.1. By the definition of almost-sure convergence, we have

$$\mathbb{P}_{x_0}^{\tau,\theta}\left(\lim_{t\to\infty} \pi_t(\theta \mid H_t) < 1\right) = 1.$$

Therefore,

$$1 = \mathbb{P}_{x_0}^{\tau,\theta}\left(\lim_{t\to\infty} \pi_t(\theta \mid H_t) < 1\right) = \mathbb{P}_{x_0}^{\tau,\theta}\left(\bigcup_{n=1}^{\infty} \left\{\lim_{t\to\infty} \pi_t(\theta \mid H_t) < 1 - \frac{1}{n}\right\}\right)$$

$$\leq \sum_{n=1}^{\infty} \mathbb{P}_{x_0}^{\tau,\theta} \left(\lim_{t\to\infty} \pi_t(\theta \mid H_t) < 1 - \frac{1}{n} \right).$$

where the inequality is by Boole's inequality. Hence, Lemma 2.7.1 is verified. □

2.9.2 Analysis of Assumption 1

Recall Assumption 1. Since Example 2.5.2 has a finite state space, (2.39) boils down to: For any $x \in \mathcal{X}$, $u \in \mathcal{U}$, and any two distinct parameter value, $\gamma \neq \theta \in \mathcal{P}$, there exists a positive constant $\epsilon(x, u, \theta, \gamma) > 0$ such that

$$\mathbb{K}(\nu_\theta^{x,u} \mid \nu_\gamma^{x,u}) \geq \epsilon(x, u, \theta, \gamma).$$

The above condition holds if and only if (by definition)

$$E_{f^\theta q^\theta}\left[\log\left(\frac{d\nu_\theta^{x,u}}{d\nu_\gamma^{x,u}}\right)\right] = E_{f^\theta q^\theta}\left[\log\left(\frac{f^\theta(\cdot \mid x,u) q^\theta(\cdot \mid x,u)}{f^\gamma(\cdot \mid x,u) q^\gamma(\cdot \mid x,u)}\right)\right] \geq \epsilon(x, u, \theta, \gamma),$$

where $E_{f^\theta q^\theta}[\cdot \mid x, u]$ is defined in Section 2.3.2. For illustration purposes, we make the same assumption we had in Example 2.5.2; the θ-MDP has only one state. This implies the transition kernel, $q^\theta(\cdot \mid x, u)$, is deterministic and we simplify the relative entropy expression. Due to the simpler version of the relative entropy, we utilize the expectation operator $E_{f^\theta}[\cdot \mid x, u]$, also defined in Section 2.3.2. It suffices to find a constant $\epsilon(x, u, \theta, \gamma) > 0$, for any x_A (in this case $\mathcal{X} = \{x_0\}$ where $x_0 = x_A$), any $u \in \mathcal{U}$ (control 1 or 2), $\theta = B$ and $\gamma = A$, such that

$$E_{f^\theta}\left[\log\left(\frac{f^\theta(\cdot \mid x,u)}{f^\gamma(\cdot \mid x,u)}\right)\right] \geq \epsilon(x, u, \theta, \gamma).$$

We have,

$$E_{f^\theta}\left[\log\left(\frac{\frac{1}{\sqrt{2\pi 0.1}}e^{-\frac{(r-\mu^\theta)^2}{2(0.1)}}\Big|x,u}{\frac{1}{\sqrt{2\pi 0.1}}e^{-\frac{(r-\mu^\gamma)^2}{2(0.1)}}\Big|x,u}\right)\right] = E_{f^\theta}\left[\log\left(\frac{e^{-(r-\mu^\theta)^2}|x,u}{e^{-(r-\mu^\gamma)^2}|x,u}\right)\right]$$

$$= E_{f^\theta}\left[\log\left(e^{-(r-\mu^\theta)^2+(r-\mu^\gamma)^2}|x,u\right)\right],$$

which can be simplified as

$$E_{f^\theta}[-(r-\mu^\theta)^2+(r-\mu^\gamma)^2 \mid x,u].$$

Case 1: When the state is x_A and control 2 is picked by the TS policy, $r \sim N(0.8, 0.1)$, and

$$\begin{aligned}
&E_{f^\theta}[-(r-0.8)^2+(r-0.4)^2 \mid x_A,2]\\
&= -Var[r \mid x_A,2] + E_{f^\theta}[(r-0.4)^2 \mid x_A,2]\\
&= -0.1 + E_{f^\theta}[r^2 - 0.8r + 0.16 \mid x_A,2]\\
&= -0.1 + E_{f^\theta}[r^2 \mid x_A,2] - 0.8E_{f^\theta}[r \mid x_A,2] + 0.16\\
&= -0.1 + Var[r \mid x_A,2] + (E_{f^\theta}[r \mid x_A,2])^2 - 0.8E_{f^\theta}[r \mid x_A,2] + 0.16\\
&= -0.1 + 0.1 + 0.64 - 0.64 + 0.16 = 0.16.
\end{aligned}$$

Case 2: When the state is x_A and control 1 is picked by the TS policy, $r \sim N(0.3, 0.1)$, and

$$\begin{aligned}
&E_{f^\theta}[-(r-0.3)^2+(r-0.5)^2 \mid x_A,1]\\
&= -Var[r \mid x_A,1] + E_{f^\theta}[(r-0.5)^2 \mid x_A,1]\\
&= -0.1 + E_{f^\theta}[r^2 - r + 0.25 \mid x_A,1]\\
&= -0.1 + E_{f^\theta}[r^2 \mid x_A,1] - E_{f^\theta}[r \mid x_A,1] + 0.25\\
&= -0.1 + Var[r \mid x_A,1] + (E_{f^\theta}[r \mid x_A,1])^2 - E_{f^\theta}[r \mid x_A,1] + 0.25
\end{aligned}$$

$$= -0.1 + 0.1 + 0.09 - 0.3 + 0.25 = 0.04.$$

As long as $0 < \epsilon(x, u, \theta, \gamma) \leq 0.04$, Assumption 1 holds for Example 2.5.2.

CHAPTER 3

ESTIMATING THE MEAN AND VARIANCE OF HETEROGENEOUS TASKS

3.1 Motivation and Literature Review

Estimating the mean and variance of processes can generate useful insights. For example, a process with high variance may signal a need for intervention, while the reverse scenario may indicate an opportunity to explore and disseminate best practices. Such estimations are especially relevant in the service industry where processes are distinct tasks being performed multiple times by agents.

In this work, our area of focus is the healthcare industry. We aim to estimate the mean and variance of surgical cases (surgical case and surgery will be used interchangeably). Surgical cases are well-defined tasks which are performed in a dedicated part of the hospital, i.e., in the operating room (OR).

Consider a group of surgeons who independently perform surgeries that are of similar nature, and they all take similar (mean) time to complete their cases. If one surgeon's case times have lower variance time than the rest of the group, investigating the root causes of the discrepancy may bring forth opportunities to improve the variance of other surgeons' case times. In the context of the OR, the potential benefits of identifying variance include better utilization of the rooms and the improvement of metrics such as overtime, patient satisfaction and safety; Cheng et al. [2018] found that prolonged operative time is associated with an increased risk of surgery complications, advocating for decreased operative times to be a universal goal for surgeons, hospitals, and policy makers.

A critical feature of the setting that we study is that tasks are multi-step objects, i.e., tasks consist of one or more sub-tasks. This fits well with treating surgical cases as tasks since they are composed of one or more *procedures*, as in Figure 3.1.

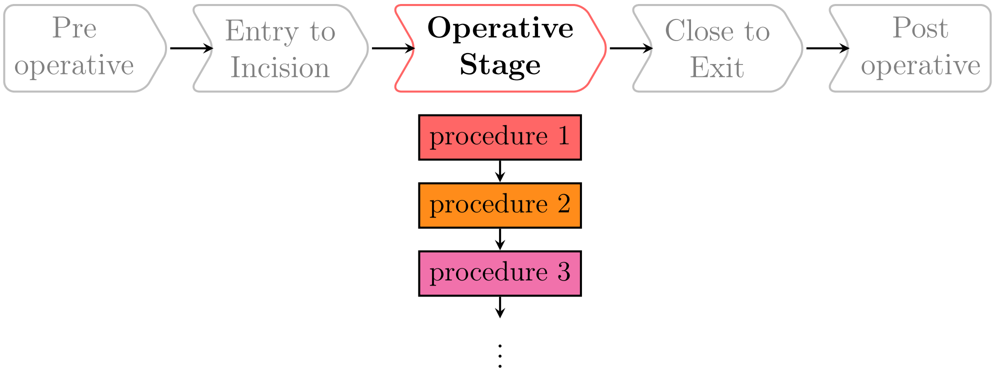


Figure 3.1: The surgical actions take place during the "Operative Stage", i.e., the surgery. It begins with first incision and ends with the closing of the patient. We have timestamps for first incision time and close time.

For efficient planning, it is crucial that one can predict the task time of a previously unseen task (or rarely seen). For this, it is necessary to estimate a "universal" slope coefficient (e.g., impact of task familiarity on time) using tasks that are similar in nature. However, running a single regression on the full sample would be naive and likely unsuccessful as task times are not necessarily commensurate across different task times. Instead, a reasonable coefficient, and thus a reasonable prediction, can be obtained by running a standardized regression. To serve as an input to the standardized regression, each (observed) task time can be standardized as follows,

$$\text{Standardized time} = \frac{\text{Observed time} - \text{Mean time of task type}}{\text{Standard deviation of task type}}. \tag{3.1}$$

Through standardization, tasks are brought into the same scale. Thus, the impact of task familiarity becomes generalizable across tasks, i.e., the impact is estimated in a way that is impervious to task types.

Mean estimation techniques for small sample sizes have been studied; however, to the best of our knowledge, there does not exist well-established variance estimation techniques for small samples. Our study extends the work of Li et al. [2009], which estimates the mean duration of surgical cases using the Current Procedural Terminology (CPT) coding scheme. In this study, we extend their results along three dimensions,

1. We use three independent procedure coding schemes and conduct a comparative study.
   - The internal coding scheme of University of Chicago Medical Center (UCM)
   - International Classification of Diseases (ICD-10). We use a truncated version of this to avoid the excessive thinning of data. In the remainder of the paper, they will be referred to as "ICD-10-trunc".
   - Current Procedural Terminology (CPT)
2. We estimate the variance of surgical cases based on the cases' composition of procedures.
3. In addition to a supervised model, we generate estimations using an unsupervised method.

To our knowledge, our work is the first to estimate the variance of surgical cases. We adapt two distinct statistical models to this novel setting and compare their performance in regards to their mean and variance predictions. Using the supervised and unsupervised methods, we compare the success of the results against each other. The methods we adapt are the random coefficients model and the hierarchical clustering model. In the context of the latter, we employ ANOVA for categorical variables, adapting the methodology from Light and Margolin [1971] into our setting.

3.2 Data

The full sample data in this study consists of 68,655 surgical cases performed between January 1, 2017 and July 21, 2022 at the University of Chicago Medical Center, Surgery Department. This sample excludes cases with operative times longer than 10 hours and less than 10 minutes, as these are outlier cases. We include twelve service lines in our analysis, which make up 96 percent of the raw data. We run our models separately for each service line, similar to the approach of Li et al. [2009]. Two of the reasons they list are relevant to us;

- Although the same procedures can be shared between multiple services, the codes are mostly different across service lines.

- Studies suggest that the service line itself is a relevant factor for predicting case durations [Strum et al., 2003].

Unlike UCM and CPT codes, ICD-10 codes have “meaningful” characters. We make use of this structure. Consider a surgical case where the (pre-surgery) diagnosis is “Full thickness burn with skin graft or inhalation injury with CC/MCC (Major complication or comorbidity). One of the three ICD-10 codes of this case is 0HRJXK3, shown in Figure 3.2: *Replacement of Left Upper Leg Skin with Nonautologous Tissue Substitute, Full Thickness, External*. The initial character is “0” for all procedure codes. While the second character gives the body system, the body part is represented jointly by the second and fourth characters, i.e., the fourth character is not meaningful by itself. The rest of the characters are meaningful independent of other characters. We determine the main factors that impact the duration of a surgical case to be “body part”, “operation” and “approach”, thereby omitting “device” and “qualifier” in order to avoid the thinning of data. Since the first character is ubiquitously 0, and the final two characters are omitted, we employ the truncated version of ICD-10 codes; the full code is replaced with its 4-character substring: HRJX.

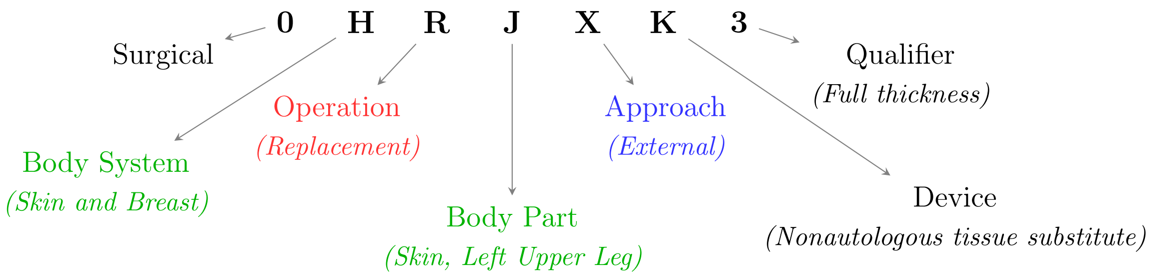


Figure 3.2: ICD-10 code structure

The challenge around estimating surgery variance is that surgeries often consist of multiple procedures. To illustrate this, consider the following example. Under the UCM scheme, "Debridement Leg Plastic" is a common procedure code and appears as the sole procedure in 336 single-procedure surgeries. Yet, the number of surgical cases in which both "Debridement Leg Plastic" and "Debridement Foot Plastic" were performed is merely 5 out of the full sample. Table 3.1 shows the heterogeneity of surgical cases; under UCM, the truncated version of ICD-10 (i.e., ICD-10-trunc) and CPT schemes. By Table 3.1, we note that about 1/3 of all surgical cases, when encoded by either the truncated ICD-10's or CPT's, appear only once in the data.

Table 3.1: How many times do surgeries appear in the data?

		Surgery Occurrence Times			
Coding System		**1**	**2 − 5**	**6 − 10**	**≥ 10**
UCM	University of Chicago Medical Center	11%	8%	5%	76%
ICD-10-trunc	International Classification of Diseases 10th Revision (truncated)	34%	14%	7%	45%
CPT	Current Procedural Terminology	35%	8%	6%	51%

Furthermore, nearly 1/3 of the full sample of surgeries have two or more procedures when encoded by the UCM scheme, and about 2/3 of cases have multiple procedures when encoded by CPT and ICD-10-trunc.

3.3 First Method: Random Coefficients Approach

The operative time of surgery i is equal to the sum of the (random) times of the procedures being performed.

$$Y_i = \sum_{k=0}^{K} X_{ik}\beta_{ik} = \beta_{i0} + \sum_{k=1}^{K} X_{ik}\beta_{ik}, \tag{3.2}$$

where $X_{i0} = 1$ and $X_{ik} \in \{0,1\}$ for $k \geq 1$, and $\beta_{ik} \in \mathbb{R}$ are the *random coefficients.* We let K denote the number of distinct procedures across all surgeries and $\mathcal{K} := \{1, 2, \ldots, K\}$ the set of procedures. We denote the vector of procedures by $\boldsymbol{X}_i := (X_{i0}, X_{i1}, \ldots, X_{iK})$. An element of $\boldsymbol{X}_i$ is set to 1 if the corresponding procedure is performed in surgery i, and set to 0 otherwise. $\boldsymbol{X}_i$ is observed, and it has length $(K+1)$. We define the matrix of surgeries, i.e.,

$$\mathbb{X} := \begin{bmatrix} \boldsymbol{X}_1' \\ \boldsymbol{X}_2' \\ \vdots \\ \boldsymbol{X}_N' \end{bmatrix}_{N\times(K+1)}$$

where N is the number of surgeries (i.e., observations) and $\boldsymbol{X}_i'$ represents the transpose of $\boldsymbol{X}_i$. The set $\mathcal{X}_i := X_{i0} \cup \{X_{ik} : k \in \mathcal{K}\}$ represents the procedures, and $\mathcal{X}_i^c := \{X_{ij}X_{ik} : j \neq k \in \mathcal{K}\}$ represents the set of cross-terms. Let $\tilde{\boldsymbol{X}}_i$ represent the vector that consists of all elements of $\mathcal{X}_i$ and $\mathcal{X}_i^c$; i.e., the vector of the procedures and pairwise interactions. Similar to $\boldsymbol{X}_i$, an element of $\tilde{\boldsymbol{X}}_i$ is set to 1 if the corresponding procedure is performed in surgery i. In $\tilde{\boldsymbol{X}}_i$, for the cross-term of two distinct procedures k and j to be nonzero, it has to be that both are performed in surgery i, i.e., $X_{ik} = X_{ij} = 1$. $\tilde{\boldsymbol{X}}_i$ is observed and has length $(K+1) + \frac{(K-1)K}{2}$ for all i. We also define the matrix of surgeries including cross terms, i.e.,

$$\tilde{\mathbb{X}} := \begin{bmatrix} \tilde{\boldsymbol{X}}_1' \\ \tilde{\boldsymbol{X}}_2' \\ \vdots \\ \tilde{\boldsymbol{X}}_N' \end{bmatrix}_{N\times\left\{(K+1)+\frac{(K-1)K}{2}\right\}}$$

where $\tilde{\boldsymbol{X}}_i'$ is the transpose of $\tilde{\boldsymbol{X}}_i$.

The vector $\boldsymbol{\beta}_i := (\beta_{i0}, \beta_{i1}, \ldots, \beta_{iK})$ represents the duration of each procedure for surgery i. It is unobserved and has length $(K+1)$. One can interpret β_{i0} as the (random) duration to access to the body organ(s) and to close the patient[1]. Note that we reserve bold letters for vectors.

Assumption 4a (identification). $\boldsymbol{X}_i \perp\!\!\!\perp \boldsymbol{\beta}_i$.

Assumption 4b (identification). $\boldsymbol{X}_i \perp\!\!\!\perp \boldsymbol{\beta}_i$ and $\boldsymbol{X}_i \perp\!\!\!\perp \beta_{ij}\beta_{ik}$, $\forall j \neq k \in \mathcal{K}$.

Assumption 4a states that the duration of the procedures are independent of combinations of procedures. That is, independent of which procedures are performed as part of the same surgery, the random duration of a procedure β_{ik} is drawn from the same distribution across surgeries. This does *not* imply that every procedure has the same distribution of duration. Assumption 4b states that the pairwise product of procedure durations are independent of the combinations of procedures.

Assumption 5. (uncorrelated) $Cov(\beta_{i0}, \beta_{ij}) = 0$, $\forall j \in \mathcal{K}$.

Assumption 5 states that each procedure's duration is uncorrelated with the intercept.

Assumption 6a. The matrix $\mathbb{E}[\boldsymbol{X}_i \boldsymbol{X}_i']$ is full rank.

We choose a design matrix $\mathbb{X}$ with full column rank. For this, we adopt the design matrix construction approach of Li et al. [2009]; that is,

> *In order to avoid singularity, CPT codes that always appear together should be treated as a whole as if they formed a new CPT code. ... The purpose of grouping is to establish the set of single CPT codes whose execution times can be estimated, the set of two-code CPT combinations whose combined time can be estimated, the set of three-code CPT combinations whose combined time can be estimated, and so on. A full-rank design matrix can then be constructed based on the grouping results.*

[1] Our data contains timestamps for when the first incision is made and the completion of the patient's closing.

Constructing a full ranked design matrix guarantees the solvability of the least-squares estimation. We extend this approach to UCM and ICD-10-trunc coding schemes.

Assumption 6b. The matrix $\mathbb{E}[\tilde{\boldsymbol{X}}_i\tilde{\boldsymbol{X}}_i']$ is full rank.

Results on identification of $\mathbb{E}[\boldsymbol{\beta}_i]$ and $Var[\beta_{ij}]$ follow by these assumptions.

Lemma 3.3.1 (Mean identification)**.** *Under Assumptions 4a and 6a,* $\mathbb{E}[\boldsymbol{\beta}_i]$ *is identified by linear regression.*

Proof. Proof of Lemma 3.3.1.

$$\begin{aligned}\mathbb{E}[\boldsymbol{X}_i\boldsymbol{X}_i']^{-1}\mathbb{E}[\boldsymbol{X}_iY_i] &= \mathbb{E}[\boldsymbol{X}_i\boldsymbol{X}_i']^{-1}\mathbb{E}[\boldsymbol{X}_i\boldsymbol{X}_i'\beta_i]\\ &= \mathbb{E}[\boldsymbol{X}_i\boldsymbol{X}_i']^{-1}\mathbb{E}[\mathbb{E}[\boldsymbol{X}_i\boldsymbol{X}_i'\boldsymbol{\beta}_i \mid \boldsymbol{X}_i]]\\ &= \mathbb{E}[\boldsymbol{X}_i\boldsymbol{X}_i']^{-1}\mathbb{E}[\boldsymbol{X}_i\boldsymbol{X}_i'\mathbb{E}[\boldsymbol{\beta}_i \mid \boldsymbol{X}_i]]\\ &= \mathbb{E}[\boldsymbol{X}_i\boldsymbol{X}_i']^{-1}\mathbb{E}[\boldsymbol{X}_i\boldsymbol{X}_i']\mathbb{E}[\boldsymbol{\beta}_i]\\ &= \mathbb{E}[\boldsymbol{\beta}_i],\end{aligned}$$

where the second equality is by the law of iterated expectations, and the fourth equality follows by Assumption 4a. □

Lemma 3.3.2 (Variance identification)**.** *Under Assumptions 4b, 5, and 6b,* $Var[\beta_{ik}]$ *is identified for* $k \in \mathcal{K}$*.*

Proof. Proof of Lemma 3.3.2.

The conditional variance of the duration of surgery i can be written as

$$\begin{aligned}Var[Y_i \mid \tilde{\boldsymbol{X}}_i] &= Var\left[\sum_{k=0}^{K} X_{ik}\beta_{ik} \mid \tilde{\boldsymbol{X}}_i\right]\\ &= \sum_{k=0}^{K} Var\left[X_{ik}\beta_{ik} \mid \tilde{\boldsymbol{X}}_i\right] + 2\sum_{k=0}^{K-1}\sum_{j>k} Cov(X_{ik}\beta_{ik}, X_{ij}\beta_{ij} \mid \tilde{\boldsymbol{X}}_i)\end{aligned}$$

$$
\begin{aligned}
&= \sum_{k=0}^{K} Var\left[X_{ik}\beta_{ik} \mid \tilde{\boldsymbol{X}}_i\right] + 2\sum_{k=1}^{K-1}\sum_{j>k}^{K} Cov(X_{ik}\beta_{ik}, X_{ij}\beta_{ij} \mid \tilde{\boldsymbol{X}}_i) \\
&= \sum_{k=0}^{K} X_{ik}^2 Var\left[\beta_{ik} \mid \tilde{\boldsymbol{X}}_i\right] + 2\sum_{k=1}^{K-1}\sum_{j>k}^{K} X_{ik}X_{ij}Cov(\beta_{ik}, \beta_{ij} \mid \tilde{\boldsymbol{X}}_i) \\
&= Var\left[\beta_{i0}\right] + \sum_{k=1}^{K} X_{ik}Var\left[\beta_{ik}\right] + 2\sum_{k=1}^{K-1}\sum_{j>k}^{K} X_{ik}X_{ij}Cov(\beta_{ik}, \beta_{ij}), \qquad (3.3)
\end{aligned}
$$

where the third equality follows from Assumption 5 (i.e., no covariances between β_{i0} and β_{ik} for all $k \in \mathcal{K}$). The final equality follows by $X_{ik}^2 = X_{ik}$ since $X_{ik} \in \{0,1\}$ and $X_{i0} = 1$, and by Assumption 4b. Note that (3.3) is linear in $Var[\beta_{ik}]$ for $k = 0, \ldots, K$.

Remark. Revisiting (3.3), note that if $Cov(\beta_{i0}, \beta_{ij})$ is allowed to be nonzero $\forall j \in \mathcal{K}$, then

$$
\begin{aligned}
Var[Y_i \mid \tilde{\boldsymbol{X}}_i] &= Var\left[\sum_{k=0}^{K} X_{ik}\beta_{ik} \mid \tilde{\boldsymbol{X}}_i\right] \\
&= \sum_{k=0}^{K} Var\left[X_{ik}\beta_{ik} \mid \tilde{\boldsymbol{X}}_i\right] + 2\sum_{k=0}^{K-1}\sum_{j>k}^{K} Cov(X_{ik}\beta_{ik}, X_{ij}\beta_{ij} \mid \tilde{\boldsymbol{X}}_i) \\
&= Var[\beta_{i0} \mid \tilde{\boldsymbol{X}}_i] + \sum_{k=1}^{K} X_{ik}(Var[\beta_{ik} \mid \tilde{\boldsymbol{X}}_i] + 2Cov(\beta_{i0}, \beta_{ik} \mid \tilde{\boldsymbol{X}}_i)) \\
&\quad + 2\sum_{k=1}^{K-1}\sum_{j>k}^{K} Cov(X_{ik}\beta_{ik}, X_{ij}\beta_{ij} \mid \tilde{\boldsymbol{X}}_i) \\
&= Var[\beta_{i0}] + \sum_{k=1}^{K} X_{ik}(Var[\beta_{ik}] + 2Cov(\beta_{i0}, \beta_{ik})) \\
&\quad + 2\sum_{k=1}^{K-1}\sum_{j>k}^{K} X_{ik}X_{ij}Cov(\beta_{ik}, \beta_{ij}). \qquad (3.4)
\end{aligned}
$$

In (3.4), the coefficient on X_{ik} is $(Var[\beta_{ik}] + 2Cov(\beta_{i0}, \beta_{ik}))$. Linear regression cannot separately $Var[\beta_{ik}]$ and $Cov(\beta_{i0}, \beta_{ik})$. So we take $Cov(\beta_{i0}, \beta_{ik})$ to be 0 for all $k \in \mathcal{K}$, i.e., Assumption 5. If one need not estimate $Var[\beta_{ik}]$, then Assumption 5 can be removed, i.e., only consider Assumptions 4b and 6b.

Next, define

$$\boldsymbol{V} \coloneqq \begin{bmatrix} Var[Y_1 \mid \tilde{\boldsymbol{X}}_1 = \tilde{\boldsymbol{x}}_1] \\ Var[Y_2 \mid \tilde{\boldsymbol{X}}_2 = \tilde{\boldsymbol{x}}_2] \\ \vdots \\ Var[Y_N \mid \tilde{\boldsymbol{X}}_N = \tilde{\boldsymbol{x}}_N] \end{bmatrix}_{N\times 1}$$

Finally, define the vector of the random coefficients' variances, which includes covariance terms,

$$\boldsymbol{V}_\beta \coloneqq \begin{bmatrix} Var[\beta_{i0}] \\ Var[\beta_{i1}] \\ \vdots \\ Var[\beta_{iK}] \\ Cov(\beta_{i1}, \beta_{i2}) \\ \vdots \\ Cov(\beta_{i,K-1}, \beta_{i,K}) \end{bmatrix}_{\left\{(K+1)+\frac{(K-1)K}{2}\right\}\times 1}$$

In practice, many of the cross terms are set to zero. Yet, the column dimension of $\tilde{\mathbb{X}}$ (and the row dimension of $\boldsymbol{V}_\beta$) is fixed and is equal to $(K+1) + \frac{(K-1)K}{2}$. Under Assumption 5, we have

$$\boldsymbol{V} = \tilde{\mathbb{X}}\boldsymbol{V}_\beta, \tag{3.5}$$

and each element of $\boldsymbol{V}$ in (3.5) can be written as,

$$Var[Y_i \mid \tilde{\boldsymbol{X}}_i] = \tilde{\boldsymbol{X}}_i'\boldsymbol{V}_\beta,$$

which implies,

$$\mathbb{E}[(Y_i - \mathbb{E}[Y_i \mid \tilde{\boldsymbol{X}}_i])^2 \mid \tilde{\boldsymbol{X}}_i] = \tilde{\boldsymbol{X}}_i' \boldsymbol{V}_\beta = \sum_{k=0}^{K} X_{ik} Var[\beta_{ik}] \\ + 2 \sum_{k=1}^{K-1} \sum_{j>k}^{K} X_{ik} X_{ij} Cov(\beta_{ik}, \beta_{ij}). \tag{3.6}$$

Then, $\boldsymbol{V}_\beta$ is identified by regressing $\boldsymbol{V}$ on $\tilde{\mathbb{X}}$.

$$[\tilde{\mathbb{X}}'\tilde{\mathbb{X}}]^{-1}\tilde{\mathbb{X}}'\boldsymbol{V} = \boldsymbol{V}_\beta. \tag{3.7}$$

□

We now present the two-step estimation algorithm.

Estimation.

I. Generate the following variable using the data. This step generates the expression inside the expectation in (3.6).

$$\hat{V}_i := (Y_i - \mathbb{E}[Y_i \mid \tilde{\boldsymbol{X}}_i])^2 = (Y_i - \hat{Y}_i)^2,$$

where the mean surgery durations are estimated by running the regression $\hat{Y}_i := \mathbb{E}[Y_i \mid \boldsymbol{X}_i]$.

II. Then, regress $\boldsymbol{V}$ on $\tilde{\mathbb{X}}$. That is, carry out the regression in (3.7).

Remark (Dealing with Heteroskedasticity)**.** By Figure 3.3, we infer that the model suffers from heteroskedasticity. We note that service lines other than vascular surgery are also subject to heteroskedasticity. To remedy this, we obtain heteroskedasticity-robust standard errors that are asymptotically valid in the presence of any kind of heteroskedasticity [Wooldridge, 2010].

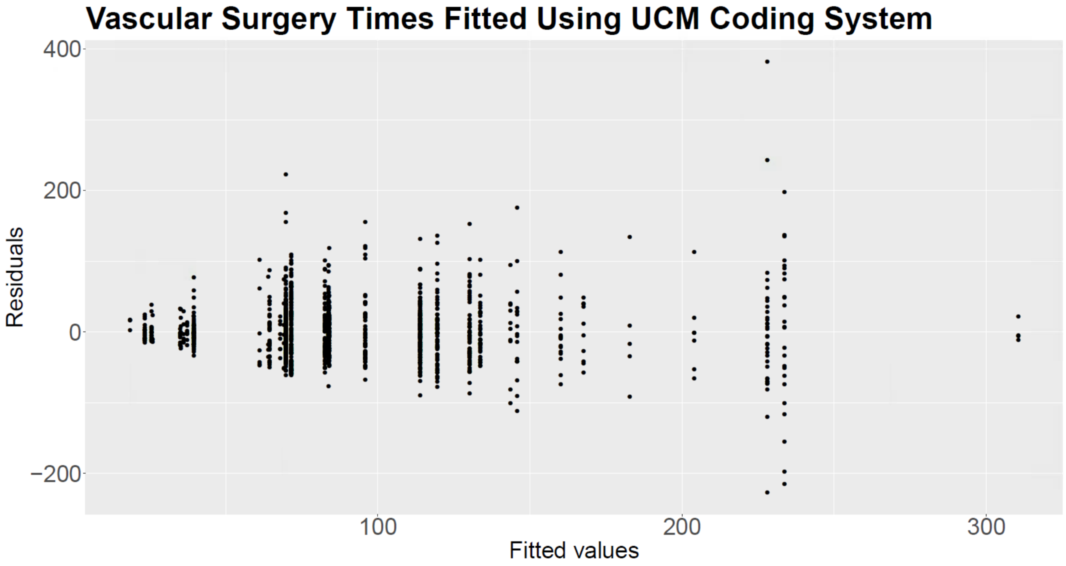


Figure 3.3: Heteroskedasticity is present; variances of residuals increase as fitted values increase.

Furthermore, the second regression (i.e., step II of the estimation method) combines data with an estimate to generate the dependent variable. This implies that the standard errors are not reliable. To remedy this, we apply bootstrapping which generates reliable standard errors.

Before ending this section, we describe the step-by-step data handling. Initially, we filter the full sample to obtain the common sample of UCM, ICD-10-trunc, CPT coding schemes, where we ensure that each code combination occurs at least 8 times in data to then assess the goodness of fit. Then looping over each service line, do:

1. Identify the distinctive codes and code combinations to create a full rank matrix [Li et al., 2009].

2. Partition data into test and training sets; 1/3 of data is used for test and 2/3 for training.

3. Run the estimation algorithm on the training set, i.e., fit the mean surgery durations, compute the variance estimator, fit the variances.

4. Using the test set, predict the case time and case variance for the remaining cases in the

sample.

5. Using the test set, compute performance statistics for the mean and variance estimations, in units of hours.

3.4 Second Method: Hierarchical Clustering Approach

In the unsupervised approach, we bundle distinct surgeries (i.e., distinct code combinations) into groups in a way that minimizes the within-cluster variation and maximizes the between-cluster variation. Our aim is to bundle surgeries that share codes together. Once the clusters are formed, we treat each cluster as if they constitute a single surgery type, and compute the mean and variance of all of the surgeries that belong to that cluster.[2] These statistics can then be used to standardize the surgery times as in (3.1).

In the hierarchical clustering framework, the dendrogram is a tree-based diagram that connects distinct surgeries. We present an example in Figure 3.4, where each leaf is a distinct surgery in the gynecology service line. The vertical level at which leaves fuse corresponds to the distance between them. The higher the fusion, the greater the distance.

Distance Metric. To plot the dendrogram, one needs to compute the pairwise distances between all distinct surgeries, i.e., code combinations. We formulate the distance between two surgeries as the ratio of the number of distinct codes divided by the total number of distinct codes, across the two surgeries [Bezem and Keijzer, 1996]. Dividing by the total number of distinct codes brings surgeries that share many codes closer, as desired.

Definition 3.4.1 (Modified Hamming Distance)**.**

$$d(\text{Surgery 1, Surgery 2}) := \frac{|\text{Symmetric difference}|}{|\text{Union}|}.$$

[2] There will be individual surgeries with the same code combination in the same cluster, but with different durations.

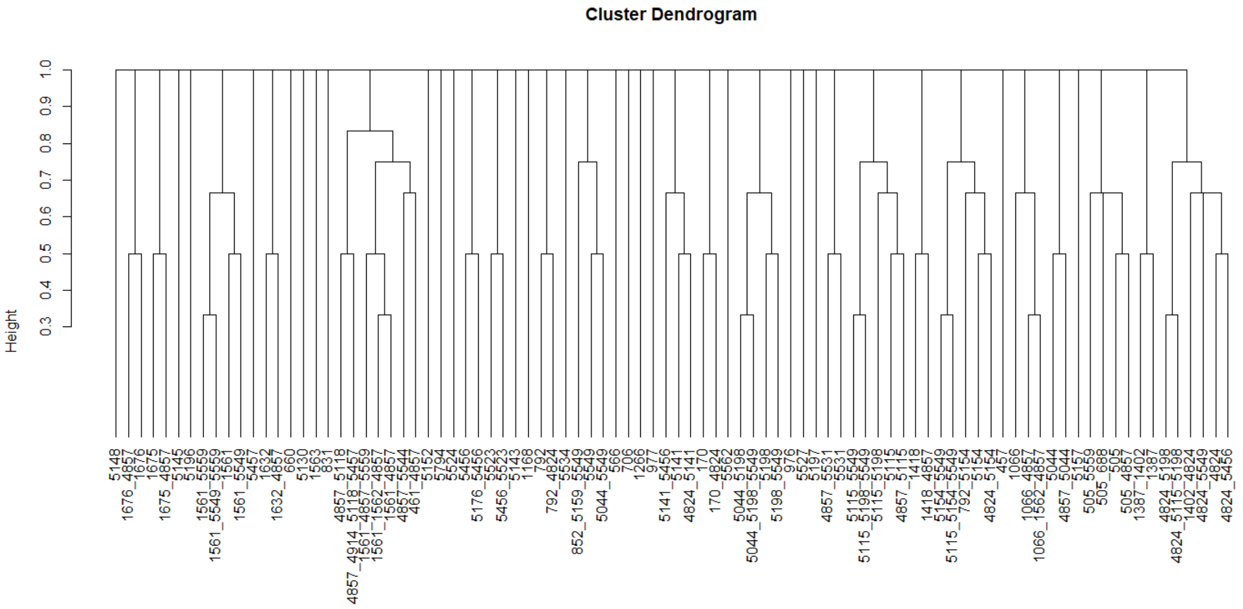


Figure 3.4: Dendrogram of Gynecology service based on truncated ICD-10 codes.

Consider the left-most leaf "5148" in Figure 3.4. Since no other leaf contains this code, its distance to the other leaves is 1. Now, consider the second and third left-most leaves, "1676_4857" and "1676". By Definition 3.4.1, the distance between them is 1/2, corresponding to the height at which they fuse.

The dissimilarity between two groups of observations is determined by various types of linkages, and thus, the fusion of branches depends on the linkage type. Complete and average linkage are the most commonly preferred linkage types as they yield more balanced dendrograms. Complete linkage computes all pairwise dissimilarities between the observations in group A and group B, and picks the largest dissimilarity, while average linkage picks the average. Our results are robust to both linkage types, and in the interest of brevity, we will share the results for average linkage.

Clustering Algorithm. Once the dendrogram is obtained, the next step is to determine the height at which to cut the dendrogram. The height uniquely maps onto the number of clusters and their compositions. Ultimately, clusters should have small within sum of squares (WSS) and large between sum of squares (BSS). However, a naive implementation of this would lead to each observation being its own cluster; WSS shrinks to its smallest value and BSS grows to its largest value when leaves are their own clusters. We opt for the Calinski-Harabasz (CH) index [Caliński and Harabasz, 1974], presented in the following definition. The CH index provides a practical solution to this problem; it scales BSS and WSS by their degrees of freedom, placing them on a similar scale.

Definition 3.4.2 (Calinski-Harabasz (CH) index)**.** The CH index is calculated for each possible h, i.e.,

$$CH(h) = \frac{\frac{BSS(h)}{G(h)-1}}{\frac{WSS(h)}{n-G(h)}},$$

where n is number of observations, G is number of clusters, h is the cutoff. Then, find the cutoff level h such that CH index is maximized, i.e.,

$$h^* = \text{argmax}_{h \in \{0,\dots,1-\epsilon\}}\ CH(h).$$

Note that the upper bound on h is $1-\epsilon$ because the CH index is *not* defined for $h = 1$, i.e., for $G(h) = 1$.

Model. Adopting the approach of Light and Margolin [1971], we let n_{ij} denote the number of occurrences of procedure code $i \in \mathcal{K} = \{1, \dots, K\}$ in cluster $j \in \mathcal{G} = \{1, \dots, G\}$. To see how a code can appear in a cluster multiple times, consider this example: If the dendrogram in Figure 3.4 is cut between $0.5 < h \leq 0.6$, then the code "1676" would appear twice in the cluster that is made up of "1676_4857" and "1676". Then, let $n_{+j} = \sum_{i \in \mathcal{K}} n_{ij}$ represent the number of occurrences of any code in cluster j. Similarly, let $n_{i+} = \sum_{j \in \mathcal{G}} n_{ij}$ represent the number of occurrences of code i, across all of the clusters. Then, the total number of occurrences of all codes across all of the clusters is

$$n = \sum_{j \in \mathcal{G}} n_{+j} = \sum_{i \in \mathcal{K}} n_{i+} = \sum_{i \in \mathcal{K}} \sum_{j \in \mathcal{G}} n_{ij}.$$

This information can be summarized via a $K \times G$ contingency table, which has the form of a matrix. We refer the interested reader to Light and Margolin [1971] to review the structure of the contingency table. Figure 3.2 shows its extension to our setting through an example.

Example 3.4.1. We consider a sample of ten distinct surgeries, where the surgeries are either single or multi-procedure. These surgeries consist of fifteen distinct procedures, i.e., codes, where each code corresponds to a row. As discussed, the height of the cut determines the number of clusters and their compositions. Figure 3.2 shows the allocation of codes to clusters when the dendrogram is cut at a height which uniquely maps into five clusters. Each cell in the matrix corresponds to n_{ij}. The sum of the elements in column j is n_{+j}, and the sum of the elements in row i is n_{i+}. For example, code 4982 is the most prevalent in this sample, and appears in all of the clusters.

Table 3.2: Contingency table with 15 distinct codes across 10 distinct surgeries.

	Clusters				
Distinct codes	**1**	**2**	**3**	**4**	**5**
1036	1	0	0	0	0
1387	0	0	1	0	0
1402	0	0	1	0	0
1471	0	0	1	0	0
326	1	0	0	0	0
484	0	0	0	0	1
4937	0	0	0	2	0
4944	0	0	0	1	0
$\boxed{4982}$	5	1	1	2	1
4983	1	0	0	0	1
5681	0	0	0	0	1
5767	0	1	0	0	0
5828	0	0	0	1	0
680	0	1	0	0	0
935	1	0	0	0	0

We now introduce two distinct ways of finding the optimal height h^*.

3.4.1 Clustering Approach 1: Clustering by Codes

A measure of variation for categorical data has been developed by Gini [1912] and advanced by Light and Margolin [1971]. The latter derives three components of variation: total sum of squares (TSS), total within-group sum of squares (WSS), and between-group sum of squares (BSS), i.e.,

$$\begin{aligned} TSS &= \frac{n}{2} - \frac{1}{2n}\sum_{i\in\mathcal{K}} n_{i+}^2 \\ WSS(h) &= \frac{n}{2} - \frac{1}{2}\sum_{j\in\mathcal{G}(h)} \frac{1}{n_{+j}}\sum_{i\in\mathcal{K}} n_{ij}^2 \\ BSS(h) &= \frac{1}{2}\left[\sum_{j\in\mathcal{G}(h)} \frac{1}{n_{+j}}\sum_{i\in\mathcal{K}} n_{ij}^2\right] - \frac{1}{2n}\sum_{i\in\mathcal{K}} n_{i+}^2. \end{aligned} \tag{3.8}$$

For each distinct value of h, we compute $WSS(h)$ and $BSS(h)$ to then find h^*. The intuition for (3.8) is obtained by revisiting Figure 3.2. As the number of clusters increases, i.e., as h decreases, the contingency table becomes more sparse. That is, the column sum n_{+j} decreases, which leads to a lower $WSS(h)$ and higher $BSS(h)$ by (3.8). In contrast, if the number of clusters increases, the row sum n_{i+} remains the same; yet because the codes are more spread out, $\sum_{i\in\mathcal{K}} n_{i+}^2$ decreases. This implies higher $BSS(h)$.[3]

Notice that surgery times are omitted when choosing h^* under the "Clustering by Codes" approach. Consequently, the value of h^*, and thus, the clusters are generalizable to any hospital with a similar pool of surgeries.

3.4.2 Clustering Approach 2: Clustering by Surgery Times

In this version, we compute $BSS(h)$ and $WSS(h)$ via (3.8), directly by using the surgery times, i.e., by using the traditional variance formulas for Euclidean distances instead of

[3] If one naively maximizes $\frac{BSS(h)}{WSS(h)}$ over h, then $h^* = 0$ ($k^* = K$), i.e., each code becomes its own cluster.

variance for categorical data. We use the same distance metric (modified hamming distance) to construct the dendrogram and the same clustering algorithm (CH index) to "cut" the dendrogram. We obtain very similar results to the first clustering approach, and thus, in the interest of brevity, we share the results for the second approach.

3.5 Comparative Analysis

The performance of the predicted surgical case times, as opposed to the observed times, is typically quantified by the *Mean Squared Error* (MSE),

$$\text{MSE} := \frac{1}{|\mathcal{O}|} \sum_{i \in \mathcal{O}} (Y_i - \hat{Y}_i)^2,$$

where we let $\mathcal{O}$ denote the set of observations, i.e., surgical cases, in a service line, and Y_i and $\hat{Y}_i$ represent the observed and the predicted time of observation i, respectively.

While MSE is a well-known metric to assess the quality of the predicted time, to the best of our knowledge there does not exist a metric to assess the quality of the predicted variances. Thus, we construct a novel metric, which we call the *Mean Squared Error in Variance* (MSEV). Using the same structure as MSE, we take the average squared difference between the predicted variance (of operative time) and its "observed" value.

Now, let $T(x)$ be a function that takes the index of a surgery as input and returns its type, i.e., its code combination,

$$\text{MSEV} := \frac{1}{|\mathcal{O}|} \sum_{i \in \mathcal{O}} \left((Y_i - \bar{Y}_i)^2 - \frac{1}{n_{type}} \sum_{\substack{j \in \mathcal{O}: \\ T(j) = T(i)}} (Y_j - \bar{Y}_{T(j)})^2 \right)^2 ,$$

where $n_{type} := |j \in \mathcal{O} : T(j) = T(i)|$ is the number of distinct code combinations, and $\bar{Y}_j$ represents the average duration of type $T(j)$.

We underline that our analysis is partitioned by service lines, i.e., MSE and MSEV are calculated separately for each service line.

Results. The goodness-of-fit comparisons are summarized in Tables 3.3-3.6. We compare the MSE and MSEV results along two dimensions, i.e., across (I) random coefficients (RC) and hierarchical clustering (HC) methods, and (II) UCM, ICD-trunc and CPT schemes.

Table 3.3 shows how the three coding schemes compare against each other, after eliminating infrequent surgery types that occur less than 8 times with respect to all three schemes. Eliminating infrequent types is necessary because MSEV uses the average surgery duration, i.e., average duration of the surgeries that consist of the same code combination. There remains 12,684 surgical cases, which amounts to about 18% of the full sample. Because the common sample excludes about 82% of the data, we run the same analysis for each pair, i.e., UCM and ICD-trunc, UCM and CPT, ICD-trunc and CPT, presented in Tables 3.4-3.6. This allows us to keep a relatively larger portion of the data; 21,163, 24,457 and 13,319 surgical cases, respectively. We underline that the outcomes are not to be compared across tables as their samples are different. Fixing a service line, there are six different configurations in Table 3.3 and four different configurations in Tables 3.4-3.6. In each table, we highlight by gray the smallest MSE and MSEV of predictions, i.e., the best-performing configuration(s) (there can be ties), across the two methods and coding schemes, for each service line.

The first main insight that we derive is that UCM codes are generally marginally better than the other two schemes, see Table 3.3 (strictly superior MSE in 3/12 services and strictly superior MSEV in 4/12 services). Yet, this statement does not hold for all services, e.g., colorectal and neurosurgery service lines have lower MSEV under ICD-trunc, while transplant service line has lower MSEV under CPT. The second main insight is that hierarchical clustering performs slightly better than random coefficients, under all schemes. Yet, we emphasize that the differences are not large enough to conclude that HC is strictly better. Furthermore, under some services, random coefficients is better or similar-performing than hierarchical clustering, e.g., neurosurgery and urology under ICD-trunc.

In Tables 3.3-3.6, we observe that the cardiac service line consistently has relatively large MSE and MSEV in comparison to the other services. This can be attributed to the long operative times of the surgeries in this line. Cardiac cases are longer surgeries which indicate that they are more complex and uncertain; thus, the uncertainty impacts the MSE and MSEV.

Table 3.3: Three-way comparison of UCM, ICD-trunc and CPT

Service	**Metric**	**UCM**		**ICD-trunc**		**CPT**	
		RC	**HC**	**RC**	**HC**	**RC**	**HC**
Cardiac	MSE	1.71	1.09	2.28	1.11	1.69	1.09
	MSE in Variance	6.15	5.56	6.25	6.00	6.49	5.93
Colorectal	MSE	0.34	0.33	0.41	0.33	0.43	0.36
	MSE in Variance	0.56	0.54	0.45	0.37	0.50	0.49
General	MSE	0.43	0.41	0.47	0.46	0.49	0.47
	MSE in Variance	1.34	1.30	1.72	1.48	1.84	1.80
Gynecology	MSE	0.75	0.73	0.84	0.82	0.77	0.77
	MSE in Variance	4.23	4.12	4.81	4.66	4.81	4.68
Neurosurgery	MSE	0.52	0.43	0.43	0.46	0.46	0.43
	MSE in Variance	0.90	0.81	0.66	0.70	0.81	0.78
Orthopaedic	MSE	0.31	0.30	0.34	0.31	0.33	0.32
	MSE in Variance	0.51	0.46	0.49	0.46	0.52	0.49
Otolaryngology	MSE	0.31	0.28	0.31	0.28	0.30	0.28
	MSE in Variance	0.93	0.87	1.07	0.98	1.12	1.10
Plastic	MSE	0.63	0.63	0.71	0.70	0.63	0.63
	MSE in Variance	1.01	1.01	1.47	1.47	1.01	1.01
Thoracic	MSE	0.14	0.13	0.12	0.07	0.13	0.12
	MSE in Variance	0.11	0.07	0.11	0.06	0.03	0.03
Transplant	MSE	0.32	0.31	0.34	0.28	0.53	0.30
	MSE in Variance	0.21	0.19	0.17	0.16	0.13	0.14
Urology	MSE	0.31	0.30	0.30	0.30	0.31	0.30
	MSE in Variance	0.44	0.44	0.41	0.41	0.42	0.41
Vascular	MSE	0.45	0.43	0.50	0.43	0.50	0.44
	MSE in Variance	0.96	0.73	0.50	0.47	0.98	0.75

Table 3.4 shows that UCM is typically better than ICD-trunc, yet the discrepancies are minimal. In addition, under both coding schemes, hierarchical clustering is slightly better than random coefficients. The same trend holds in Table 3.5, i.e., the UCM and hierarchical clustering configuration is generally better than the other three. Finally, in Table 3.6, we observe that the ICD-trunc and hierarchical clustering configuration performs relatively better than the other three, with the CPT and hierarchical clustering configuration as the second best option.

Table 3.4: Pairwise comparison of UCM and ICD-trunc

Service	**Metric**	**UCM**		**ICD-trunc**	
		RC	**HC**	**RC**	**HC**
Cardiac	MSE	2.11	2.00	2.06	1.52
	MSE in Variance	8.10	7.03	9.76	7.42
Colorectal	MSE	0.49	0.44	0.48	0.45
	MSE in Variance	1.82	1.55	2.10	2.05
General	MSE	0.42	0.42	0.49	0.48
	MSE in Variance	1.08	1.04	1.30	1.24
Gynecology	MSE	0.65	0.62	0.68	0.72
	MSE in Variance	2.19	2.16	2.45	2.39
Neurosurgery	MSE	0.71	0.67	0.64	0.64
	MSE in Variance	5.84	6.02	1.87	1.76
Orthopaedic	MSE	0.29	0.28	0.32	0.30
	MSE in Variance	0.74	0.72	0.84	0.81
Otolaryngology	MSE	0.35	0.33	0.40	0.34
	MSE in Variance	0.56	0.51	0.85	0.78
Plastic	MSE	0.68	0.62	0.77	0.66
	MSE in Variance	1.20	1.06	1.01	0.99
Thoracic	MSE	0.19	0.17	0.23	0.21
	MSE in Variance	0.26	0.24	0.28	0.26
Transplant	MSE	0.60	0.55	0.75	0.69
	MSE in Variance	1.17	1.15	2.61	2.62
Urology	MSE	0.29	0.28	0.29	0.28
	MSE in Variance	0.44	0.43	0.42	0.41
Vascular	MSE	0.49	0.48	0.51	0.49
	MSE in Variance	1.69	1.64	1.40	1.31

Table 3.5: Pairwise comparison of UCM and CPT

Service	**Metric**	**UCM**		**CPT**	
		RC	**HC**	**RC**	**HC**
Cardiac	MSE	1.51	1.49	1.67	1.57
	MSE in Variance	5.40	4.80	4.82	4.55
Colorectal	MSE	0.79	0.77	0.85	0.84
	MSE in Variance	3.02	2.92	3.28	3.20
General	MSE	0.93	0.89	1.09	1.07
	MSE in Variance	6.58	6.48	13.14	13.05
Gynecology	MSE	1.21	1.18	1.23	1.24
	MSE in Variance	7.90	7.67	9.27	9.26
Neurosurgery	MSE	0.71	0.65	0.68	0.70
	MSE in Variance	1.91	1.83	1.86	1.83
Orthopaedic	MSE	0.61	0.61	0.62	0.62
	MSE in Variance	2.86	2.81	2.66	2.56
Otolaryngology	MSE	0.37	0.34	0.36	0.35
	MSE in Variance	0.78	0.75	0.74	0.70
Plastic	MSE	0.82	0.87	0.84	0.80
	MSE in Variance	4.25	4.18	4.67	4.61
Thoracic	MSE	0.81	0.69	0.81	0.76
	MSE in Variance	4.70	4.45	4.67	4.41
Transplant	MSE	0.60	0.60	0.66	0.65
	MSE in Variance	1.11	0.98	1.22	1.09
Urology	MSE	0.69	0.66	0.72	0.67
	MSE in Variance	3.31	3.19	3.65	3.50
Vascular	MSE	1.27	1.20	1.31	1.21
	MSE in Variance	7.99	7.87	8.36	7.92

Table 3.6: Pairwise comparison of ICD-trunc and CPT

Service	**Metric**	**ICD-trunc**		**CPT**	
		RC	**HC**	**RC**	**HC**
Cardiac	MSE	1.14	1.12	1.08	1.09
	MSE in Variance	1.54	1.73	2.91	2.93
Colorectal	MSE	0.17	0.14	0.17	0.16
	MSE in Variance	0.14	0.12	0.13	0.12
General	MSE	0.44	0.44	0.46	0.45
	MSE in Variance	0.92	0.90	0.89	0.85
Gynecology	MSE	0.74	0.74	0.66	0.67
	MSE in Variance	3.32	3.23	2.57	2.56
Neurosurgery	MSE	0.53	0.54	0.55	0.51
	MSE in Variance	0.77	0.79	0.81	0.77
Orthopaedic	MSE	0.36	0.34	0.36	0.35
	MSE in Variance	0.68	0.67	0.78	0.78
Otolaryngology	MSE	0.22	0.20	0.22	0.21
	MSE in Variance	0.21	0.14	0.19	0.16
Plastic	MSE	0.71	0.69	0.70	0.69
	MSE in Variance	1.50	1.49	1.52	1.52
Thoracic	MSE	0.21	0.12	0.17	0.12
	MSE in Variance	0.22	0.10	0.13	0.07
Transplant	MSE	0.27	0.24	0.28	0.24
	MSE in Variance	0.11	0.10	0.18	0.11
Urology	MSE	0.33	0.30	0.31	0.30
	MSE in Variance	0.33	0.33	0.35	0.34
Vascular	MSE	0.52	0.44	0.57	0.47
	MSE in Variance	0.54	0.52	0.36	0.40

3.6 Concluding Remarks and Discussion

In this work, we adapt two well-known statistical techniques to the surgery setting, to estimate the unknown mean and variance of surgical procedures. Although the estimation of mean times of procedures have been widely studied, estimating the variance of procedures is an under-explored question. The main challenge of estimating the variances is that surgeries are a collection multiple procedures, for which the operative times are unobserved. Each surgical procedure is associated with a code, and we make use of this structure. Both of our proposed methods can easily be implemented on any data set where the surgeries consist of codes, there is no limit on the number of codes.

In the first approach, by modeling the operative time as the sum of the (random) procedure times and the (random) intercept, we show that the mean and variance of procedures are identified under certain mild assumptions. We develop a two-step estimation algorithm to compute the variance of procedures, and thus, the variance of surgical cases. In the second approach, we group surgeries according to their composition of codes. Under the HC framework, the number of total clusters is not known a priori. Using a reasonable dissimilarity metric and linkage type, we construct dendrograms under the three different coding schemes. Then, by the CH index method, we obtain the clusters.

Next, to compare the performances, we develop a metric to assess the goodness of fit of the variance estimations, which resembles the structure of MSE. We call this novel metric MSE in variance (MSEV). Through these metrics, our first result is that the UCM coding scheme is slightly better than the other two; however, in most of the service lines the differences in MSE and MSEV are very small, so this result should be taken with a grain of salt and necessitates further investigation with larger samples. The second main result is that hierarchical clustering generally outperforms the random coefficients method. However, we emphasize that the differences in performance are again fairly small.

As a limitation of the study, we reiterate that the analyses require using a common sample. This results in the elimination of the bulk of the data; about 82% of the full sample is eliminated in the three-way comparison, with the maximum number of procedures being four. To remedy this, we conduct pairwise analyses (again, subject to the removal of infrequently occurring surgeries). The pairwise analysis between UCM and CPT retains 36% of the full sample. This is an improvement, but it lacks information on ICD-trunc, and it lacks surgeries with more than four codes (due to removing infrequent surgeries). Furthermore, services such as cardiac, plastic and thoracic have less than 100 data points left. The other two pairwise comparisons have even smaller portion of the data remaining. To account for these, future work can extend the analyses to larger samples for a more robust comparison of performances. In the context of hierarchical clustering, a future direction of research is to develop alternative distance metrics and to use different clustering algorithms.

CHAPTER 4

EQUITABLE DATA-DRIVEN ASSIGNMENTS OF WORKERS TO TASKS

4.1 Introduction

It is prevalent in worker assignment problems that one can estimate the cost or reward of assigning a worker to a task. The algorithms that researchers produce presume that one can freely estimate the cost that depends on the worker-task pair, which entails a worker's innate abilities. Yet, sometimes the decision-maker is constrained in the information that they can use; and in certain settings, it is necessary to keep workers' performance information hidden. For example, a study that estimates workers' performances via a learning curve could be infeasible in unionized settings as it could promote favoritism, or even discrimination. This idea constitutes the main motivation of the approach we propose in this paper.

We broadly classify worker-related information into two types: performance and task familiarity information. In contrast to performance, which is idiosyncratic (or innate), task familiarity is not related to a worker's innate characteristics and identity; it is a result of how often tasks have "utilized" a worker historically. We develop an equitable framework to make assignments over time, where we define equity in the following manner: An equitable assignment is one that treats any two individuals –who have the same familiarity with a particular task– as interchangeable without loss of optimality, regardless of their innate characteristics. In the remainder of the paper, our usage of the terms equity and equitable will be based on this definition. We bound the overall performance loss from adopting equitable assignments, as opposed to optimal assignments that minimize total task times in steady-state.

To our knowledge, there is as of yet no framework that considers retaining workers' "performance-privacy" when making assignment decisions that inevitably impact workers.

In contrast to the literature, we consider an environment where making inferences that may reveal workers' innate abilities is not accepted.

Similar to prominent works, our paper assumes that learning happens at the worker level [Reagans et al., 2005]. However, we purposefully suppress worker performance at the stage of making the assignments to study its effects. In contrast, the vast amount of work that studies the phenomenon of learning by doing often makes inferences on workers' innate abilities by observed performance. For instance, Arlotto et al. [2014] studies optimal hiring and retention policies that are driven by worker capability. Nembhard [2001] and Staruch and Staruch [2021] study the assignment of workers to tasks based on individual learning characteristics. Conducting an empirically-based simulation study, Nembhard [2001] shows that there are potential opportunities to reallocate workers to improve firm-wide productivity by using performance information. In our analytical study, we tackle this issue from the opposite viewpoint, i.e., we investigate the steady-state consequences of suppressing worker performance when making worker-task assignments in a sequential fashion.

We study the general problem of assigning a finite set of individuals to a finite set of tasks, where each worker is assigned to at most one task in a series of discrete time periods. It can alternatively be posed as matching individuals with other individuals (e.g., ride-sharing or the online labor markets). Our framework includes a "prediction algorithm", which predicts the completion time of each arriving task by each arriving worker. The main considerations when designing this prediction algorithm are as follows:

Respecting performance-privacy. The algorithm must not estimate the underlying worker-task level performance parameters that govern task completion times. In addition, the algorithm must not compute performance statistics, e.g., mean and variance, of worker-specific task completion times.

Dealing with small sample sizes. A firm that does not pursue making equitable assignment decisions would aim to estimate the performance of every worker-task pair. This is not practical as data on certain worker-task pairs can be rare or nonexistent, and thus acting on only available data may lead to both suboptimal performance and inequity. An equitable approach may be more robust to small sample sizes.

Capturing firm-wide effect. In many practical settings, certain tasks are not performed many times. To ameliorate the small sample size issue, the algorithm should employ a "reasonable" firm-wide effect of familiarity on completion time. Furthermore, Pisano et al. [2001] provides empirical evidence on the inter-firm differences in learning rates, i.e., across firms within the same industry. Implementing our algorithm in different organizations will generate firm-specific rates of learning.

We develop our prediction algorithm in light of the above considerations, presented in Section 4.3. We label it as the 5-Step algorithm or [ASAPI], which stands for Aggregate-Standardize-Aggregate-Predict-Invert. Being heterogeneous entities, tasks of different types tend to take different lengths of time. Consequently, completion times are not commensurate across different task types. To address this issue, [ASAPI] employs a regression using standardized variables by standardizing historical data at the task-level. This circumvents revealing worker-specific statistics, while also addressing data thinness, if any.[1] Its key step is to fit a single regression model to the standardized data, which is then pooled across all tasks and workers, to estimate the universal effect of familiarity on time. Through this estimate, [ASAPI] predicts the times of new data points. The prediction is fed into an optimization model that assigns workers to tasks in each period. Implementing the period's assignments generates new data, which is used to refine the universal effect. Iteratively running the prediction and optimization steps constitutes the Predict-Then-Optimize (PTO) loop, shown

[1] If a given task has been performed by worker j only, this does not constitute a performance-privacy issue since worker j cannot be compared against any other worker $j' \neq j$ in performing this task.

in Figure 4.1.

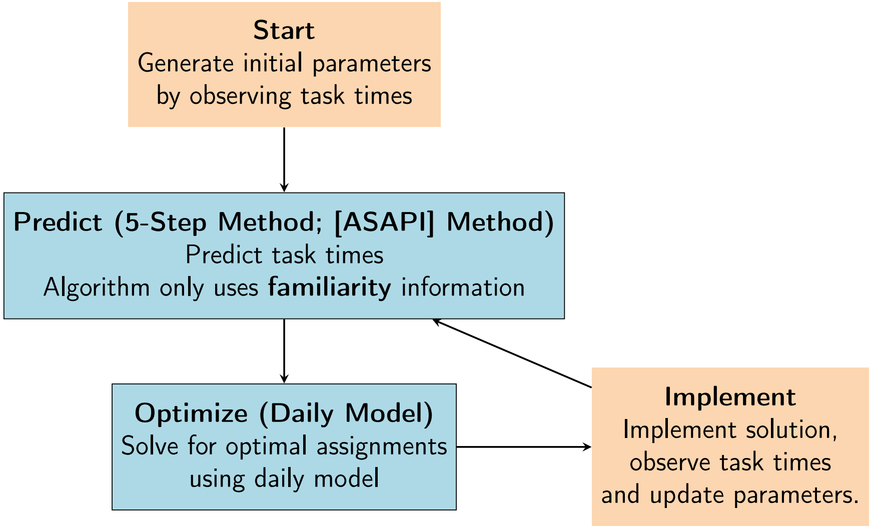

Start
Generate initial parameters
by observing task times
Predict (5-Step Method; [ASAPI] Method)
Predict task times
Algorithm only uses **familiarity** information
Optimize (Daily Model)
Solve for optimal assignments
using daily model
Implement
Implement solution,
observe task times
and update parameters.

Figure 4.1: Process diagram. The Start phase initiates the loop.

Running the prediction algorithm requires the analyst to have full access to the data. We assume that the analyst will run the analysis in good faith as an "honest broker", i.e., not publish worker-centric performance information, and not make assignment decisions based (directly) on workers' performances.

Industries with well-defined tasks performed by individuals include –but are not limited to– final assembly, healthcare, quality inspection, telemarketing, textile manufacture [Nembhard, 2001], furniture plants [Staruch and Staruch, 2021], experimental problem solving tasks [Littlepage et al., 1997], mining [Goodman and Leyden, 1991], and software development [Banker and Slaughter, 2000]. The healthcare setting is rich in worker-task pairings, where individuals perform tasks by themselves or in collaboration with others. The considerations in this paper are influenced by the operating room environment [Witmer et al., 2022]; in particular, by the real-life challenges of assigning nurses to surgeries.[2]

The rest of the paper is organized as follows: In Section 4.1.1 we review additional work

[2] Because our algorithm assigns nurses to surgeries based on familiarity and increased familiarity reduces completion times, we assume that our algorithm, i.e., obscuring nurse performance, does not have a detrimental effect on patient safety.

that our paper complements and builds on. In Section 4.2, we characterize the way in which task familiarity evolves and develop the PTO loop. In Section 4.3.1, we introduce the equitable [ASAPI] algorithm, and in Section 4.3.2, we characterize the finite-time predictions from the performance-aware and performance-blind (i.e., equitable) models. In Section 4.4, we formulate the steady-state versions of the models and show that the steady-state equitable model is fundamentally defective because it cannot drive the system into the true optimal solution. In Section 4.5, we explore upper bounds on the penalty of making equitable assignment decisions under certain assumptions, and we also derive the performance of an alternative policy called the egalitarian policy. Finally, we discuss potential future avenues in Section 4.6.

4.1.1 Additional Literature

Motivated by the evidence in support of the worker-specific learning curve, i.e., workers' task familiarity and performance are inversely related, we build a model where increased familiarity leads to shorter completion times. Factors such as absenteeism [Goodman and Leyden, 1991] or knowledge-intensiveness [Avgerinos and Gokpinar, 2017] give rise to fluid worker-task pairings, which cause the level of familiarity to vary over time. Our model accounts for the fluidity of worker-task pairs.

It has been posited that organizations have different abilities to benefit from their firm-wide experience [Jarmin, 1993]. Our methodology is motivated by the notion of capturing the firm-wide effect of familiarity on task completion time. Similarly, Pisano et al. [2001] conducts an empirical study with sixteen hospitals, where the dependent variable is surgical case completion time. The results provide evidence that firm-specific learning rates can differ significantly across independent organizations in the same industry, in support of our considerations.

Ultimately, we study an iterative matching algorithm. In each iteration, the algorithm adjusts the learning parameters used to predict the completion times, which then determine the optimal allocations across all workers and tasks. Although our problem has a sequential nature, we do not allow using estimates of the underlying parameters that drive the system (i.e., respecting performance-privacy). Thus, navigating the exploration-exploitation trade-off of the multi-armed bandit problem is not compatible with our proposed method, unlike the works of Johari et al. [2021] and Kalvit and Zeevi [2022] that exploit this trade-off. As a substitute of the underlying parameters that drive the realization, we use the "universal" (firm-wide) effect when predicting the outcomes.

4.1.2 Notation

We use capital letters to denote random variables. Let X represent an arbitrary random variable. Then, $X(n)$ denotes data collected up to period n. The Greek letters $\hat{\gamma}$, $\hat{\Gamma}$ and $\hat{\beta}$ denote linear regression estimates, and γ, Γ and β represent the true underlying parameters. Apart from the regression estimates and $\hat{\mu}(X(n))$ and $\hat{\sigma}(X(n))$, which represent empirical estimators of data, we use the "hat" symbol to denote predictions. We reserve bold letters to spaces and subscripts to pairs, workers, and tasks.

4.2 Model

Let $i \in \boldsymbol{I} = \{1, \ldots, I\}$ denote tasks, where each task is assumed to be of a different nature, and let $j \in \boldsymbol{J} = \{1, \ldots, J\}$ denote workers. We allow for only a subset of tasks and a subset of workers to be available in each period n. The random set of tasks that are to be executed in period n is denoted by $\boldsymbol{I}(n) \subseteq \boldsymbol{I}$, and the random set of workers that are available in period n is $\boldsymbol{J}(n) \subseteq \boldsymbol{J}$. The period-$n$ task and worker arrivals are assumed to be sampled from a general distribution $G(\boldsymbol{I}, \boldsymbol{J})$. The sets of available tasks and workers are known at the beginning of each period. We make two straightforward assumptions. First, the total number of tasks in any given period does not exceed the total number of workers who are available in that period, i.e., $|\boldsymbol{I}(n)| \leq |\boldsymbol{J}(n)|\ \forall n = \{0, 1, 2, \dots\}$. This ensures feasibility since workers execute tasks concurrently, and tasks are not shared among workers. Second, each worker and task can arrive at most once in a given period.

We assume that the system has been operating for a long time, i.e., there exists time and familiarity data on (most of) the worker-task pairs going back into negative periods. We emphasize that even though one can formulate the optimal allocations as a dynamic problem that looks into the infinite future, we opt for a myopic optimization problem because it is simple to implement and reinforces existing familiarities. Considering the dynamic version of this problem is future work.

4.2.1 Data Generating Process

Definition 4.2.1 (Encounter)**.** When worker j performs task i in period n, this constitutes a unique instance, which we call an *encounter*. An encounter is uniquely defined by the tuple (i, j, n), and its completion time, denoted by $T_{ij}(n)$, is random. With $X_{ij}(n) = \{0, 1\}$ denoting the assignment decision, the set of encounters across all tasks and workers up to, and including, period $N - 1$ is denoted by

$$\mathcal{E}(N) = \{(i, j, n) : X_{ij}(n) = 1,\ \forall n < N\}.$$

$\mathcal{E}(N)$ encapsulates the information that is in the first three columns and first $N-1$ rows of Figure 4.1. $F_{ij}(n) : \{0 \leq F_{ij}(n) \leq 1\}$ represents the *familiarity* of worker j with task i, accumulated up to period n. We assume that learning manifests itself as a reduction in completion time $T_{ij}(n)$, and $T_{ij}(n)$ is governed by a familiarity effect $\forall (i, j)$, which depends on the period-n task familiarity of workers $F_{ij}(n)$, i.e.,

$$T_{ij}(n) = \gamma^0_{ij} + \gamma^1_{ij} F_{ij}(n) + \epsilon_{ij}(n), \tag{4.1}$$

where $\epsilon_{ij}(n)$'s are mean-0 normal random variables $\forall(i,j)$.[3]

By (4.1), an encounter's completion time depends on the worker-task pair, not just the task itself. This stems from the intuition that not every individual has identical baseline performance when they undertake a task for the first time, and individuals' efficiencies evolve differently when they execute the task repeatedly. The familiarity effect in (4.1) captures the worker-task-centric performances; the parameters γ_{ij}^0 and γ_{ij}^1 reflect the worker-task pair's level of synergy.[4] We assume that worker j's proficiency in task i develops as the pair accumulates joint experience, hence γ_{ij}^1's are assumed to be strictly negative $\forall(i,j)$. Since γ_{ij}^0's reflect the baseline time of task completion, i.e., when a worker has no prior experience, they are strictly positive $\forall(i,j)$. To ensure that completion times are strictly positive on average, we assume that $\gamma_{ij}^0 > |\gamma_{ij}^1|\ \forall(i,j)$.

We allow some of the familiarity to be forgotten over time and model familiarity as evolving via exponential smoothing. Task familiarity accumulated up to period n decays at rate $\alpha \in (0,1)$, and the familiarity gained in n, which is either 0 or 1, is discounted at rate $(1-\alpha)$,

$$F_{ij}(n+1) = \alpha F_{ij}(n) + (1-\alpha)\mathbf{1}_{\{X_{ij}(n)=1\}}, \quad \forall(i,j) \in (\boldsymbol{I} \times \boldsymbol{J}). \tag{4.2}$$

Table 4.1 provides an excerpt of (dummy) data showing the information needed to calculate the familiarity of each worker with each task in period 101. In settings with a nontrivial number of workers and tasks in the system, there are certain immediate consequences: Not all (i,j) pairs appear in the data or some (i,j) pairs appear rarely. The analyst's minimization problem is sketched in the following section.

[3] Although possible, we do not consider negative times since it is a low probability event.

[4] We assume that γ_{ij}^0 and γ_{ij}^1 are fixed, i.e., they do not change depending on the worker-task assignment policy of the organization, e.g., an equitable policy.

Table 4.1: Table populated with dummy data. Time is measured in minutes. There are $I = 8$ tasks and $J = 7$ workers.

Period (n)	Task (i)	Worker (j)	Time
-100	Task 1	Worker 5	$T_{15}(-100) = 97$
-100	Task 3	Worker 2	$T_{32}(-100) = 66$
-100	Task 7	Worker 3	$T_{73}(-100) = 112$
⋮	⋮	⋮	⋮
1	Task 4	Worker 1	$T_{41}(1) = 88$
1	Task 2	Worker 3	$T_{23}(1) = 70$
1	Task 8	Worker 7	$T_{87}(1) = 34$
1	Task 6	Worker 2	$T_{62}(1) = 110$
2	Task 1	Worker 7	$T_{17}(2) = 79$
2	Task 7	Worker 3	$T_{73}(2) = 100$
⋮	⋮	⋮	⋮
100	Task 3	Worker 2	$T_{32}(100) = 63$
100	Task 2	Worker 6	$T_{26}(100) = 77$
100	Task 5	Worker 4	$T_{54}(100) = 90$

4.2.2 Single-Period Assignment Problem

In each period n, the analyst aims to assign workers to tasks in a way that minimizes the total time across the assignments. Decisions are represented by the binary variable $X_{ij}(n)$, which is equal to 1 if task i is assigned to worker j in period n, and 0 otherwise. The objective function coefficients are the predicted completion times for each (i, j) in n, which are represented by $\hat{T}_{ij}(n)$. The single-period (period-n) assignment model is

$$\min_{X_{ij}(n)} \quad \sum_{i \in \boldsymbol{I}(n)} \sum_{j \in \boldsymbol{J}(n)} \hat{T}_{ij}(n) \cdot X_{ij}(n) \tag{4.3a}$$

$$\text{s.t.} \quad \sum_{j \in \boldsymbol{J}(n)} X_{ij}(n) = 1, \qquad \forall i \in \boldsymbol{I}(n), \tag{4.3b}$$

$$\sum_{i \in \boldsymbol{I}(n)} X_{ij}(n) \leq 1, \qquad \forall j \in \boldsymbol{J}(n), \tag{4.3c}$$

$$X_{ij}(n) \in \{0, 1\}, \quad \forall (i, j) : i \in \boldsymbol{I}(n), j \in \boldsymbol{J}(n). \tag{4.3d}$$

The constraint (4.3b) ensures that each task is assigned to exactly one worker, and (4.3c) guarantees that each worker gets at most one task assigned. Since we assume $|\boldsymbol{I}(n)| \leq |\boldsymbol{J}(n)|$, $(|\boldsymbol{J}(n)| - |\boldsymbol{I}(n)|)$-many workers are idle whenever $|\boldsymbol{I}(n)| < |\boldsymbol{J}(n)|$. Finally, (4.3d) represents the set of binary constraints. We consider (4.3a)–(4.3d) as a performance-blind model as long as the prediction $\hat{T}_{ij}(n)$ obscures performance information.

Even though we consider the period-n prediction here, in the remainder of the paper we will suppose that the analyst lies at the beginning of period $n+1$, i.e., making predictions for period $n+1$, to reflect the predictive aspect of the problem. In Section 4.4.2, we extend the single-period model into the *steady-state* assignment model, which represents the long-run average version of this model.

4.2.3 Evolution of the Stochastic Process

In this section, we set up the stochastic process and define the history vector with the sequence of events.

Definition 4.2.2 (Historical data)**.** The space of admissible history vectors (i.e., information) up to period n is denoted by $\mathcal{H}(n)$. For some period in negative history, i.e., $\bar{n} < 0$, $\mathcal{H}(\bar{n}) \coloneqq \boldsymbol{F}$, where $\boldsymbol{F} \in [0, 1]$ is the space of the familiarity variable. For $n \geq \bar{n}$,

$$\mathcal{H}(n+1) \coloneqq (\boldsymbol{F} \times \boldsymbol{I} \times \boldsymbol{J} \times \boldsymbol{X} \times \boldsymbol{T})^n \times \boldsymbol{F} = \mathcal{H}(n) \times \boldsymbol{I} \times \boldsymbol{J} \times \boldsymbol{X} \times \boldsymbol{T} \times \boldsymbol{F}.$$

The history vector is composed of the same recursion, i.e.,

$$H(n+1) \coloneqq (F(\bar{n}), I(\bar{n}), J(\bar{n}), X(\bar{n}), T(\bar{n}), \ldots, X(n), T(n), F(n+1)).$$

Figure 4.2 depicts the sequence of events of the process, i.e., it shows the buildup of the history vector.

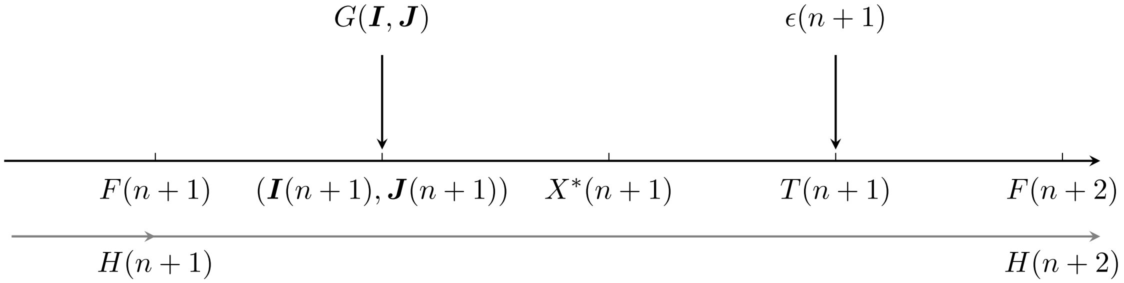

$G(\boldsymbol{I}, \boldsymbol{J})$
$\epsilon(n+1)$
$F(n+1)$
$(\boldsymbol{I}(n+1), \boldsymbol{J}(n+1))$
$X^*(n+1)$
$T(n+1)$
$F(n+2)$
$H(n+1)$
$H(n+2)$

Figure 4.2: History evolution of the PTO process.

The process in Figure 4.2 initially starts with the computation of the task familiarity, followed by sampling task and worker arrivals $(\boldsymbol{I}(n+1), \boldsymbol{J}(n+1))$ from the distribution $G(\boldsymbol{I}, \boldsymbol{J})$. This information, along with previous periods' time and familiarity levels, is used to obtain $X^*(n+1)$ by solving the problem in (4.3a)-(4.3d). The solution of the optimization is implemented, i.e., pairs that minimize the total predicted task time in period n are assigned together. This generates the real task times in period $n+1$, by sampling $\epsilon_{ij}(n+1)$ in (4.1).

For a fixed pair (i, j), let $N_{ij}(n) := |\{n' \in \{0, 1, \cdots, n\} : X_{ij}(n') = 1\}|$ represent the number of times worker j has performed task i up to period n. Then, the number of times that task i has been performed up to period n across all workers is

$$N_i(n) := \sum_{j \in \boldsymbol{J}} N_{ij}(n).$$

The number of times that worker j has performed a task up to n is denoted by

$$N_j(n) := \sum_{i \in \boldsymbol{I}} N_{ij}(n).$$

Finally, the organizational experience accumulated up to period n is represented by

$$N(n) \coloneqq \sum_{i \in \boldsymbol{I}} N_i(n).$$

In order to eliminate situations that would prevent the existence of the limit of task familiarity and the (binary) assignment decisions, we assume the following:

Assumption 7. The period-n familiarity $F_{ij}(n)$ and assignment decision $\mathbf{1}_{\{X_{ij}(n)=1\}}$ are stationary, and $\mathbb{E}[\lim_{n\to\infty} F_{ij}(n)]$ and $\mathbb{E}[\lim_{n\to\infty} \mathbf{1}_{\{X_{ij}(n)=1\}}]$ exist $\forall (i,j)$.

We let Π_{ij} represent the limiting (i.e., steady-state) probability of assigning task i to worker j, i.e.,

$$\Pi_{ij} = \mathbb{E}[\lim_{n\to\infty} \mathbf{1}_{\{X_{ij}(n)=1\}}] = \mathbb{P}(\lim_{n\to\infty} X_{ij}(n) = 1) = \mathbb{P}(X_{ij}(\infty) = 1).$$

Lemma 4.2.1 (Exponential Smoothing)**.** *Under Assumption 7 and familiarity evolution (4.2),* $\forall (i,j)$ *we have that* $\mathbb{E}[F_{ij}(\infty)] = \Pi_{ij}$*, where* Π_{ij} *is the limiting probability of assigning task* i *to worker* j*, i.e.,* $\mathbb{P}(X_{ij}(\infty) = 1)$ $\forall (i,j)$.

The proof of Lemma 4.2.1 is deferred to Section 4.7.2.

Assumption 8. Recall that Π_{ij} is the limiting probability of assigning i to j in a period. We assume,

$$\lim_{n\to\infty} \frac{N_{ij}(n)}{n} = \lim_{n\to\infty} \frac{1}{n} \sum_{n'=1}^{n} \mathbf{1}_{\{X_{ij}(n')=1\}} = \Pi_{ij}, \quad \forall (i,j).$$

4.2.4 The Predict-Then-Optimize (PTO) Loop

We are now ready to present the PTO loop. Algorithm 1 articulates the events that comprise each iteration of the loop, illustrated earlier in Figure 4.1. Recall that there exists a historical build-up of familiarity data going back into negative periods, which serves as an input into

Algorithm 1. Each iteration of the algorithm starts with the task familiarity $F_{ij}(n)$ for all (i, j) in period n, such that $n \geq 0$, which is updated each period through (4.2). To emphasize the predictive nature of the algorithm, suppose that the analyst has completed the first n iterations and lies at the beginning of period $n + 1$, i.e., running iteration $n + 1$ of the algorithm.

The period-$n + 1$ arrivals of tasks and workers are sampled from the general distribution $G(\boldsymbol{I}, \boldsymbol{J})$. Using the realized task times of the previous periods, predictions are made at the beginning of $n+1$ for period $n+1$, $\{\hat{T}_{ij}(n+1),\ \forall(i, j) \in \boldsymbol{I}(n+1) \times \boldsymbol{J}(n+1)\}$. The predictions are a function of the historical data up to $n + 1$, i.e., $H(n + 1)$, which includes $F_{ij}(n + 1)$ $\forall(i, j)$.

Having the set of feasible pairs $(\boldsymbol{I}(n + 1), \boldsymbol{J}(n + 1))$ in place, the optimization problem in (4.3a)-(4.3d) is solved for an optimal solution $X^*(n + 1)$. The solution $X^*(n + 1)$ is implemented in period $n + 1$, resulting in the following: random completion times $T_{ij}(n + 1)$ are sampled from (4.1) for the assigned pairs. In practice, the analyst observes the realized time; in the context of this study, we assume that the oracle samples it. At this point, the next period's problem begins by updating the familiarity, i.e., by computing $F_{ij}(n + 2)$ for all (i, j) using (4.2). This process is non-Markovian since the predictions $\hat{T}_{ij}(n+1)$'s depend on historical data, by virtue of the optimization.

4.3 Prediction Model

We now develop a model to predict task times in period $n + 1$, i.e., $\hat{T}_{ij}(n + 1)$, $\forall(i, j) \in \boldsymbol{I}(n + 1) \times \boldsymbol{J}(n + 1)$, using (historical) data up to period n. Before diving into the mechanics of the method, we define estimators for means and standard deviations of the following,

Pair-specific variables. $T_{ij}(n)$ and $F_{ij}(n)$ were introduced earlier in Section 4.2.1.

```
Input: ${F_{ij}(0), F_{ij}(-1), F_{ij}(-2), \dots\} \forall(i,j)$, $\alpha$, $G(\boldsymbol{I}, \boldsymbol{J})$, $\mathcal{E}(1)$,
       $\{T_{ij}(n) \forall(i,j,n) \in \mathcal{E}(n+1)\}_{\forall n \in \{\bar{n}, \bar{n}+1 \dots, 0\}, \bar{n}<0}$
for $n \geq 0$ do
    forall $(i,j) \in \boldsymbol{I} \times \boldsymbol{J}$ do
        $F_{ij}(n+1) \leftarrow \alpha F_{ij}(n) + (1-\alpha)\mathbf{1}_{\{X^*_{ij}(n)=1\}}$
    Sample $(\boldsymbol{I}(n+1), \boldsymbol{J}(n+1))$ from $G(\boldsymbol{I}, \boldsymbol{J})$
    forall $(i,j) \in \boldsymbol{I}(n+1) \times \boldsymbol{J}(n+1)$ do
        Predict $\hat{T}_{ij}(n+1)$
    Solve (4.3a)-(4.3d) using
      $\boldsymbol{I}(n+1), \boldsymbol{J}(n+1), \{\hat{T}_{ij}(n+1) : (i,j) \in \boldsymbol{I}(n+1) \times \boldsymbol{J}(n+1)\}$ to obtain
      $\{X^*_{ij}(n+1) : (i,j) \in \boldsymbol{I} \times \boldsymbol{J}\})$
    $\mathcal{E}(n+2) \leftarrow \mathcal{E}(n+1) \cup \{(i,j,n+1) : X^*_{ij}(n+1) = 1\}$
    forall $(i,j) \in \{(i,j,n+1) : X^*_{ij}(n+1) = 1\}$ do
        The oracle samples $\epsilon_{ij}(n+1)$
        $T_{ij}(n+1) \leftarrow \gamma^0_{ij} + \gamma^1_{ij} F_{ij}(n+1) + \epsilon_{ij}(n+1)$
    $n \leftarrow n+1$
```

Algorithm 1: The Predict-Then-Optimize (PTO) Loop

Task-specific variables. $T_i(n)$ and $F_i(n)$, where

$$T_i(n) \coloneqq \sum_{\{j \in \boldsymbol{J} : X_{ij}(n')=1\}} T_{ij}(n) \quad \text{and} \quad F_i(n) \coloneqq \sum_{\{j \in \boldsymbol{J} : X_{ij}(n')=1\}} F_{ij}(n).$$

Because at least two observations are needed to compute the pair and task-specific standard deviation of variables, the following estimators can be defined $\{\forall(i,j) \in \boldsymbol{I} \times \boldsymbol{J} : N_{ij}(n) \geq 2, N_i(n) \geq 2\}$:

(i) the empirical mean and empirical standard deviation of time and task familiarity of pair-specific data, i.e., $\{\hat{\mu}(T_{ij}(n)), \hat{\sigma}(T_{ij}(n)), \hat{\mu}(F_{ij}(n)), \hat{\sigma}(F_{ij}(n))\}$

(ii) the empirical mean and empirical standard deviation of time and task familiarity of task-specific data, i.e., $\{\hat{\mu}(T_i(n)), \hat{\sigma}(T_i(n)), \hat{\mu}(F_i(n)), \hat{\sigma}(F_i(n))\}$.

We provide their definitions in Section 4.7.1.

An OLS regression model can fit (4.1) using historical (training) data,

$$T_{ij}(F_{ij}(n); \boldsymbol{\gamma}_{ij}) \sim \gamma_{ij}^0 + \gamma_{ij}^1 F_{ij}(n), \quad \forall (i,j,n) \in \mathcal{E}(n+1).$$

Using the training data $\mathcal{E}(n+1)$ to produce estimates $\hat{\boldsymbol{\gamma}}_{ij} := (\hat{\gamma}_{ij}^0, \hat{\gamma}_{ij}^1)$ for the underlying parameters $\boldsymbol{\gamma}_{ij} := (\gamma_{ij}^0, \gamma_{ij}^1)$, the analyst can predict the task times in period-$(n+1)$, i.e., $T_{ij}(n+1)$, based on $F_{ij}(n+1)$, i.e.,

$$\hat{T}_{ij}(F_{ij}(n+1); \hat{\boldsymbol{\gamma}}_{ij}) = \hat{\gamma}_{ij}^0 + \hat{\gamma}_{ij}^1 F_{ij}(n+1), \quad \forall (i,j) \in \boldsymbol{I}(n+1) \times \boldsymbol{J}(n+1). \tag{4.4}$$

In practice, the analyst may be in a setting with a shortage of data on certain (i,j) pairs. We underline that the prediction model in (4.4) can only be used to predict the task times for pairs with an historical accumulation of data. In the next section, we formalize the prediction step (i.e., 5-Step algorithm; [ASAPI] algorithm) of the PTO process.

4.3.1 5-Step [ASAPI] Algorithm

We now provide the steps of the prediction model in period $n+1$. By the iterative nature of Algorithm 1, we have the estimators $\{\hat{\mu}(T_{ij}(n)), \hat{\sigma}(T_{ij}(n)), \hat{\mu}(F_{ij}(n)), \hat{\sigma}(F_{ij}(n))\}$ and $\{\hat{\mu}(F_i(n)), \hat{\sigma}(F_i(n)), \hat{\mu}(T_i(n)), \hat{\sigma}(T_i(n))\}$ readily available at the beginning of period $n+1$. In addition, through (4.2), $F_{ij}(n+1)$ is known $\forall (i,j) \in \boldsymbol{I} \times \boldsymbol{J}$. This implies that $\hat{\mu}(F_{ij}(n+1))$, $\hat{\sigma}(F_{ij}(n+1))$, $\hat{\mu}(F_i(n+1))$ and $\hat{\sigma}(F_i(n+1))$ are also available.

Step 1. [A] Aggregate the pair-specific subsets of the full data over workers, i.e., dropping the j index. The analyst can (in theory) fit I-many separate models,

$$T_{ij}(n) = \gamma_i^0 + \gamma_i^1 F_{ij}(n) + \epsilon_i(n), \quad \forall (i,j,n) \in \mathcal{E}(n+1), \tag{4.5}$$

where $\epsilon_i(n)$'s denote mean-0 normal random variables. In the remainder of the paper, we will use $\sim$, which represents a regression with a mean-0 random error term that depends on the specific regression model. Step 1 leaves us with data free of pair-specific performance information and transforms the problem into a single-worker, multi-task type of problem. At this stage, using $\mathcal{E}(n+1)$ to produce estimates $\hat{\boldsymbol{\gamma}}_i := (\hat{\gamma}_i^0, \hat{\gamma}_i^1)$ for the underlying parameters $\boldsymbol{\gamma}_i := (\gamma_i^0, \gamma_i^1)$, the analyst can predict period-$(n+1)$ task times, i.e., $T_{ij}(n+1)$, based on the period-$(n+1)$ familiarity,

$$\hat{T}_i(F_{ij}(n+1); \hat{\boldsymbol{\gamma}}_i) = \hat{\gamma}_i^0 + \hat{\gamma}_i^1 F_{ij}(n+1), \quad \forall (i,j) \in \boldsymbol{I}(n+1) \times \boldsymbol{J}(n+1). \tag{4.6}$$

This model allows the impact of familiarity to vary by task, but not worker. We label the predictions coming from (4.6) as the [AP] (Aggregate-Predict) model. They do not meet the criteria of a firm-wide familiarity effect, because $\hat{\gamma}_i^1$ depends on i. Although Step 1 ameliorates the small sample size issue by pooling observations into a larger sample, if a certain i appears rarely, this will lead to estimation issues. Figure 4.3 illustrates Step 1 by a toy problem with two tasks and three workers. The top plot distinguishes observations by (i,j), while the bottom plot distinguishes by i only.

Step 2. [S] Standardize completion time $T_{ij}(n)$ and familiarity level $F_{ij}(n)$ for each encounter (i,j,n) by using task-level means and standard deviations, i.e., endogenously.

Definition 4.3.1. For any encounter $(i,j,n) \in \mathcal{E}(n+1)$, the standardized familiarity is defined as

$$Z_{ij}^F(n) := \frac{F_{ij}(n) - \hat{\mu}(F_i(n))}{\hat{\sigma}(F_i(n))}. \tag{4.7}$$

Also, standardized time is defined as

$$Z_{ij}^T(n) := \frac{T_{ij}(n) - \hat{\mu}(T_i(n))}{\hat{\sigma}(T_i(n))}. \tag{4.8}$$

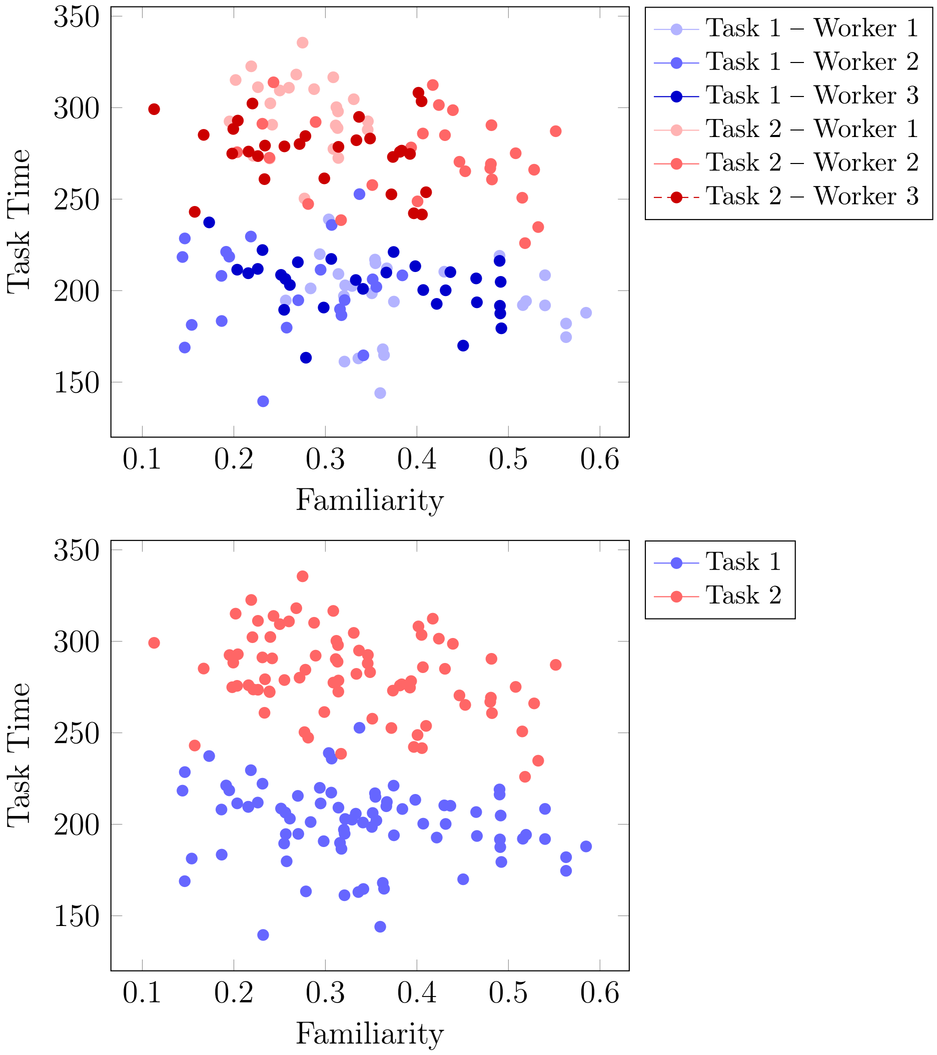


Figure 4.3: Implementation of Step 1 (Aggregate) of the [ASAPI] algorithm.

In contrast to (4.7) and (4.8), standardization can be with respect to workers (forbidden), pairs (forbidden), or with respect to the entire data (ineffective). By construction, a positive (negative) standardized variable implies that encounter (i, j, n) is above (below) the mean value of the historical observations. At this point in the process, the analyst can fit I-many standardized models, i.e., the model in (4.5) becomes

$$Z_{ij}^{T}(n) \sim \beta_i^0 + \beta_i^1 Z_{ij}^{F}(n), \quad \forall (i, j, n) \in \mathcal{E}(n+1). \tag{4.9}$$

Using $\mathcal{E}(n+1)$ to produce estimates $\hat{\boldsymbol{\beta}}_i := (\hat{\beta}_i^0, \hat{\beta}_i^1)$ for the underlying (standardized) parameters $\boldsymbol{\beta}_i := (\beta_i^0, \beta_i^1)$, one can predict standardized task times $Z_{ij}^T(n+1)$ on the basis of $\tilde{Z}_{ij}^F(n+1)$, i.e., with

$$\hat{Z}_i^T(\tilde{Z}_{ij}^F(n+1); \hat{\boldsymbol{\beta}}_i) = \hat{\beta}_i^0 + \hat{\beta}_i^1 \tilde{Z}_{ij}^F(n+1), \quad \forall (i,j) \in \boldsymbol{I}(n+1) \times \boldsymbol{J}(n+1), \tag{4.10}$$

where the standardized familiarity in period $n+1$ is computed using historical data up to n, i.e.,

$$\tilde{Z}_{ij}^F(n+1) := \frac{F_{ij}(n+1) - \hat{\mu}(F_i(n))}{\hat{\sigma}(F_i(n))}, \quad \forall (i,j). \tag{4.11}$$

Note that $\tilde{Z}_{ij}^F(n+1)$ in (4.11) is defined with one period lag, i.e., uses estimates up to, and including, period n, instead of $n+1$. This is to ensure the validity of our subsequent derivations and is an artifact of making predictions for the next period, i.e., period $n+1$. Furthermore, $\forall (i,j) \in \boldsymbol{I}(n+1) \times \boldsymbol{J}(n+1)$, we have

$$\begin{aligned} \frac{\hat{T}_i(F_{ij}(n+1); \hat{\boldsymbol{\gamma}}_i) - \hat{\mu}(T_i(n))}{\hat{\sigma}(T_i(n))} &= \hat{\beta}_i^1 \frac{F_{ij}(n+1) - \hat{\mu}(F_i(n))}{\hat{\sigma}(F_i(n))} \\ \Longrightarrow \hat{T}_i(F_{ij}(n+1); \hat{\boldsymbol{\gamma}}_i) &= \hat{\mu}(T_i(n)) + \hat{\gamma}_i^1 (F_{ij}(n+1) - \hat{\mu}(F_i(n))), \end{aligned} \tag{4.12}$$

which follows by virtue of linear regression, i.e., follows by the scaling between the slope coefficients of (4.6) and (4.10), i.e.,

$$\hat{\beta}_i^1 = \frac{\hat{\sigma}(F_i(n))}{\hat{\sigma}(T_i(n))} \hat{\gamma}_i^1, \quad \forall i. \tag{4.13}$$

Step 3. [A] Aggregate the task-specific subsets of the data over tasks, thereby dropping the i index. In (4.9), every task had its own regression; now there is only one regression, i.e., one standardized regression on the full sample.

$$Z_{ij}^T(n) \sim \beta^0 + \beta^1 Z_{ij}^F(n), \quad \forall (i,j,n) \in \mathcal{E}(n+1). \tag{4.14}$$

Step 2 by itself does not have an impact on the prediction; it is useful in conjunction Step 3. The transformations of historical data in (4.7) and (4.8) ensure that the completion times and familiarity levels are comparable across tasks. The intercept $\hat{\beta}_i^0$ is 0 by construction $\forall i$. The interpretation of $\hat{\beta}_i^1$ is straightforward: one standard deviation improvement in familiarity with task i (of any j) results in $\hat{\beta}_i^1$ standard deviation shorter completion time of task i. This assumes that the model is identified, i.e., the estimated coefficients $\hat{\gamma}_{ij}^1$ are negative $\forall (i,j)$, in which case $\hat{\beta}_i^1$ is negative $\forall i$, by construction. To see this, consider (4.13) and the relationship between $\hat{\gamma}_i^1$ and $\hat{\gamma}_{ij}^1$, presented in Lemma 4.3.1.

Lemma 4.3.1. *In any period n, the analyst can estimate $\hat{\gamma}_i^1$ and $\hat{\beta}^1$,*

$$\begin{aligned}\hat{\gamma}_i^1 = \sum_{j \in \mathcal{J}(n)} \Bigg(& \frac{\hat{\sigma}^2(F_{ij}(n))(N_{ij}(n)-1)}{\hat{\sigma}^2(F_i(n))(N_i(n)-1)} \hat{\gamma}_{ij}^1 \\ & + \frac{(\hat{\mu}(T_{ij}(n)) - \hat{\mu}(T_i(n)))(N_{ij}(n)-1)\hat{\mu}(F_{ij}(n))}{\hat{\sigma}^2(F_i(n))(N_i(n)-1)} \Bigg),\end{aligned} \tag{4.15}$$

where $\mathcal{J}(n) := \{j \in \boldsymbol{J} : N_{ij}(n) \geq 2\}$, $\forall i \in \mathcal{I}(n) := \{i \in \boldsymbol{I} : N_i(n) \geq 2\}^5$,

$$\hat{\beta}^1 = \sum_{i \in \mathcal{I}(n)} \frac{N_i(n) - 1}{\sum\limits_{i' \in \mathcal{I}(n)} N_{i'}(n) - 1} \hat{\beta}_i^1. \tag{4.16}$$

The proof of Lemma 4.3.1 is deferred to Section 4.7.2.

Step 4. [P] Predict the standardized time of period $n+1$, i.e., $Z_{ij}^T(n+1)$, based on $\tilde{Z}_{ij}^F(n+1)$. This is possible by using $\mathcal{E}(n+1)$ to produce estimates[6] $\hat{\boldsymbol{\beta}} := (\hat{\beta^0}, \hat{\beta}^1)$ for the

[5] There should be at least two observations so that the completion time and familiarity estimators are defined.

[6] We let $\hat{\beta}$ be the result of running the regression in (4.14).

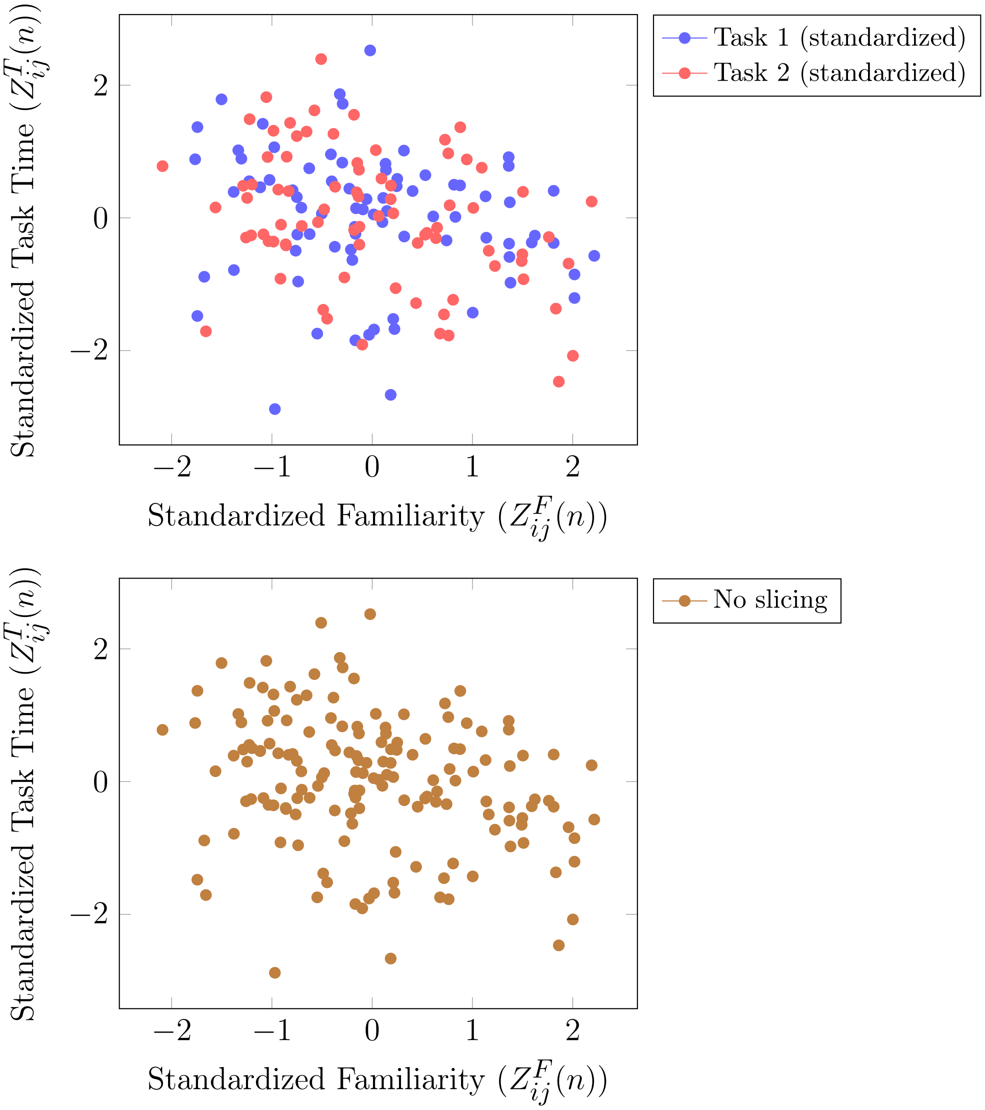


Figure 4.4: Top plot: Step 2 (Standardize) transforms Figure 4.3 into standardized scale. Bottom plot: Step 3 (Aggregate).

underlying parameters $\boldsymbol{\beta} \coloneqq (\beta^0, \beta^1)$, i.e., with

$$\hat{Z}^T(\tilde{Z}^F_{ij}(n+1); \hat{\boldsymbol{\beta}}) = \hat{\beta}^0 + \hat{\beta}^1 \tilde{Z}^F_{ij}(n+1), \quad \forall (i,j) \in \boldsymbol{I}(n+1) \times \boldsymbol{J}(n+1). \tag{4.17}$$

The slope coefficient $\hat{\beta}^1$ in (4.17) represents the firm-wide effect of standardized familiarity on standardized time. The intercept $\hat{\beta}^0$ is again 0 by construction. Steps 2 and 3 jointly meet the requirement of firm-wide effect $\hat{\beta}^1$; i.e., information becomes shareable across tasks.

Continuing the toy example from Figure 4.3, we illustrate these steps in Figure 4.4. The top plot shows Step 2, which distinguishes the observations by i, while the bottom plot shows Step 3; it contains all observations without distinguishing by tasks.

Before introducing the fifth and final step of [ASAPI], we formally define the prediction model.

Definition 4.3.2 (Prediction model)**.** Using historical data up to $n+1$ (which ends with $F(n+1)$) the analyst can obtain the predicted task times for period $n+1$, i.e., predict $T_{ij}(n+1)$ $\forall(i,j) \in \boldsymbol{I}(n+1) \times \boldsymbol{J}(n+1)$ with

$$\begin{aligned}\hat{T}_i(F_{ij}(n+1); \hat{\boldsymbol{\Gamma}}_i) &= \hat{\mu}(T_i(n)) + \hat{\sigma}(T_i(n))\hat{Z}^T(\tilde{Z}^F_{ij}(n+1); \hat{\boldsymbol{\beta}}) \\ &= \hat{\mu}(T_i(n)) + \hat{\sigma}(T_i(n))(\hat{\beta}^0 + \hat{\beta}^1 \tilde{Z}^F_{ij}(n+1)) \\ &= \hat{\mu}(T_i(n)) + \hat{\sigma}(T_i(n))\left(\hat{\beta}^1 \frac{F_{ij}(n+1) - \hat{\mu}(F_i(n))}{\hat{\sigma}(F_i(n))}\right), \qquad (4.18)\end{aligned}$$

where $\hat{\beta}^1$ is estimated via (4.14). The parameter vector $\hat{\boldsymbol{\Gamma}}_i \coloneqq (\hat{\Gamma}^0_i, \hat{\Gamma}^1_i)$ is defined by grouping the terms of (4.18),

$$\begin{aligned}\hat{T}_i(F_{ij}(n+1); \hat{\boldsymbol{\Gamma}}_i) &= \underbrace{\hat{\mu}(T_i(n)) - \hat{\beta}^1 \frac{\hat{\sigma}(T_i(n))\hat{\mu}(F_i(n))}{\hat{\sigma}(F_i(n))}}_{\text{intercept}} + \underbrace{\frac{\hat{\sigma}(T_i(n))\hat{\beta}^1}{\hat{\sigma}(F_i(n))}}_{\text{slope}} F_{ij}(n+1) \qquad (4.19) \\ &\coloneqq \hat{\Gamma}^0_i(\hat{\beta}^1) + \hat{\Gamma}^1_i(\hat{\beta}^1) F_{ij}(n+1), \qquad (4.20)\end{aligned}$$

$\forall(i,j) \in \boldsymbol{I}(n+1) \times \boldsymbol{J}(n+1)$, where $\hat{\Gamma}^0_i$ and $\hat{\Gamma}^1_i$ are functions of $\hat{\beta}_1$ and the underlying (identified) parameters are $\boldsymbol{\Gamma}_i \coloneqq (\Gamma^0_i, \Gamma^1_i)$.

Step 5. [I] Invert the predicted standardized time of period $n+1$, represented by $\hat{Z}^T(\tilde{Z}^F_{ij}(n+1); \hat{\boldsymbol{\beta}})$, into the predicted time $\hat{T}_i(F_{ij}(n+1); \hat{\boldsymbol{\Gamma}}_i)$ using historical data, where $\hat{\boldsymbol{\Gamma}}_i$ is defined in Definition 4.3.2. This concludes the [ASAPI] algorithm, visualized in Figure 4.5.

The 5-Step prediction model (4.19) stands in contrast to the task-level model, which, by

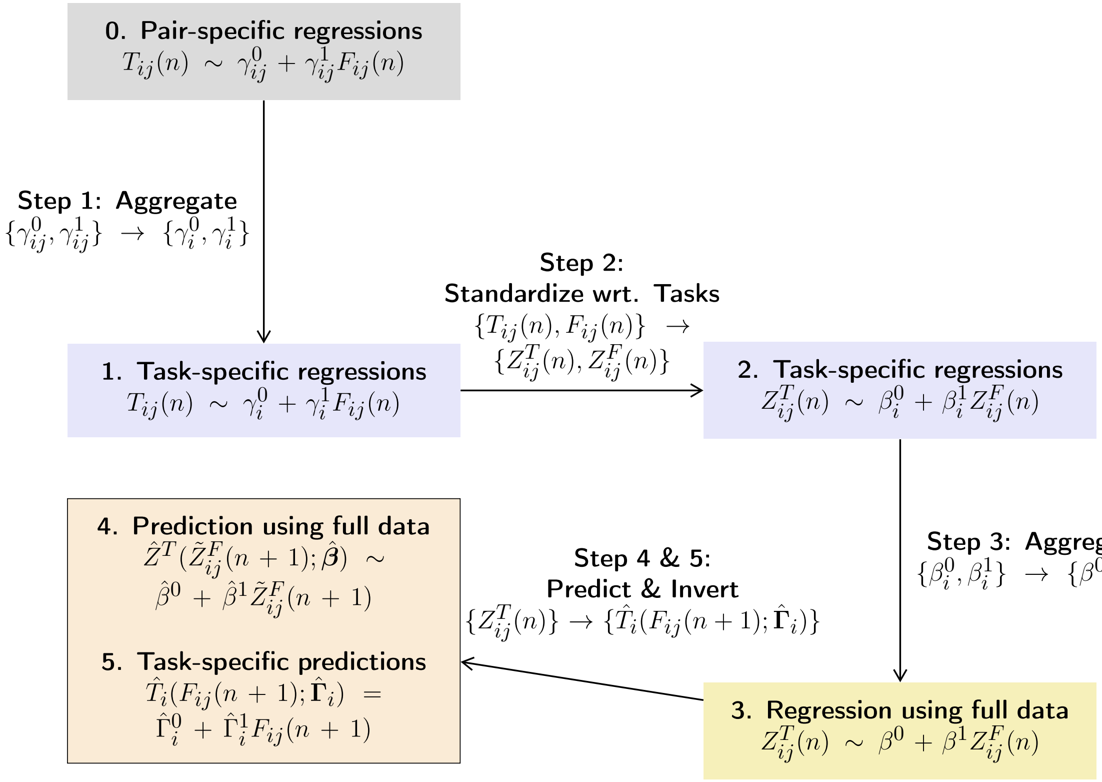

0. Pair-specific regressions
$T_{ij}(n) \sim \gamma^0_{ij} + \gamma^1_{ij} F_{ij}(n)$
Step 1: Aggregate
$\{\gamma^0_{ij}, \gamma^1_{ij}\} \to \{\gamma^0_i, \gamma^1_i\}$
Step 2:
Standardize wrt. Tasks
$\{T_{ij}(n), F_{ij}(n)\} \to \{Z^T_{ij}(n), Z^F_{ij}(n)\}$
1. Task-specific regressions
$T_{ij}(n) \sim \gamma^0_i + \gamma^1_i F_{ij}(n)$
2. Task-specific regressions
$Z^T_{ij}(n) \sim \beta^0_i + \beta^1_i Z^F_{ij}(n)$
4. Prediction using full data
$\hat{Z}^T(\tilde{Z}^F_{ij}(n+1); \hat{\boldsymbol{\beta}}) \sim \hat{\beta}^0 + \hat{\beta}^1 \tilde{Z}^F_{ij}(n+1)$
5. Task-specific predictions
$\hat{T}_i(F_{ij}(n+1); \hat{\boldsymbol{\Gamma}}_i) = \hat{\Gamma}^0_i + \hat{\Gamma}^1_i F_{ij}(n+1)$
Step 4 & 5:
Predict & Invert
$\{Z^T_{ij}(n)\} \to \{\hat{T}_i(F_{ij}(n+1); \hat{\boldsymbol{\Gamma}}_i)\}$
Step 3: Aggregate
$\{\beta^0_i, \beta^1_i\} \to \{\beta^0, \beta^1\}$
3. Regression using full data
$Z^T_{ij}(n) \sim \beta^0 + \beta^1 Z^F_{ij}(n)$

Figure 4.5: Illustration of the [ASAPI] algorithm. For brevity, we suppress that each regression model is fit $\forall(i,j,n) \in \mathcal{E}(n+1)$. Boxes with no borders contain regression models. The box with borders contains prediction models.

(4.12), has the form

$$\begin{aligned}\hat{T}_i(F_{ij}(n+1); \hat{\boldsymbol{\gamma}}_i) &= \hat{\mu}(T_i(n)) - \hat{\gamma}^1_i \hat{\mu}(F_i(n)) + \hat{\gamma}^1_i F_{ij}(n+1) \\ &= \underbrace{\hat{\mu}(T_i(n)) - \hat{\beta}^1_i \frac{\sigma(T_i(n))\hat{\mu}(F_i(n))}{\hat{\sigma}(F_i(n))}}_{\hat{\gamma}^0_i \text{ (intercept)}} + \underbrace{\frac{\hat{\sigma}(T_i(n))\hat{\beta}^1_i}{\hat{\sigma}(F_i(n))}}_{\hat{\gamma}^1_i \text{ (slope)}} F_{ij}(n+1),\end{aligned} \tag{4.21}$$

$\forall(i,j) \in \boldsymbol{I}(n+1) \times \boldsymbol{J}(n+1)$, where the second term is by (4.13). The functional form of (4.19) and (4.21) are the same except for the familiarity effect; (4.19) captures the universal effect, while (4.21) involves the task-specific effect. Consequently, the different intercepts and slopes of (4.19) and (4.21) ascertain that the [ASAPI] and [AP] models yield different predictions. For a given pair (i,j), the predicted times are equal only if the standardized coefficient of task i, i.e., $\hat{\beta}_i^1$, is equal to the firm-wide (standardized) coefficient $\hat{\beta}^1$.

Before concluding this section, recall that running the task-level regressions in (4.5) to obtain the predictions in (4.6), i.e., the leading [AP] of [ASAPI], does not depend on the firm-wide familiarity effect. A naive way to obtain a firm-wide effect is to pool the observations over (i, j), and thereby running a single regression on the full data, i.e.,

$$T_{ij}(n) \sim \gamma^0 + \gamma^1 F_{ij}(n), \quad \forall (i, j, n) \in \mathcal{E}(n+1). \tag{4.22}$$

Figure 4.6 shows that this approach performs poorly. The red and blue lines correspond to the estimations on past data, i.e., $T_{ij}(n) = \hat{\sigma}(T_i(n)) Z_{ij}^T(n) + \hat{\mu}(T_i(n))$, estimated separately for tasks 1 and 2 of the toy example. In contrast, the green line represents the estimations from the hypothetical regression in (4.22), estimated using the full data, which yields estimates that are visibly less precise than the [ASAPI] estimates.

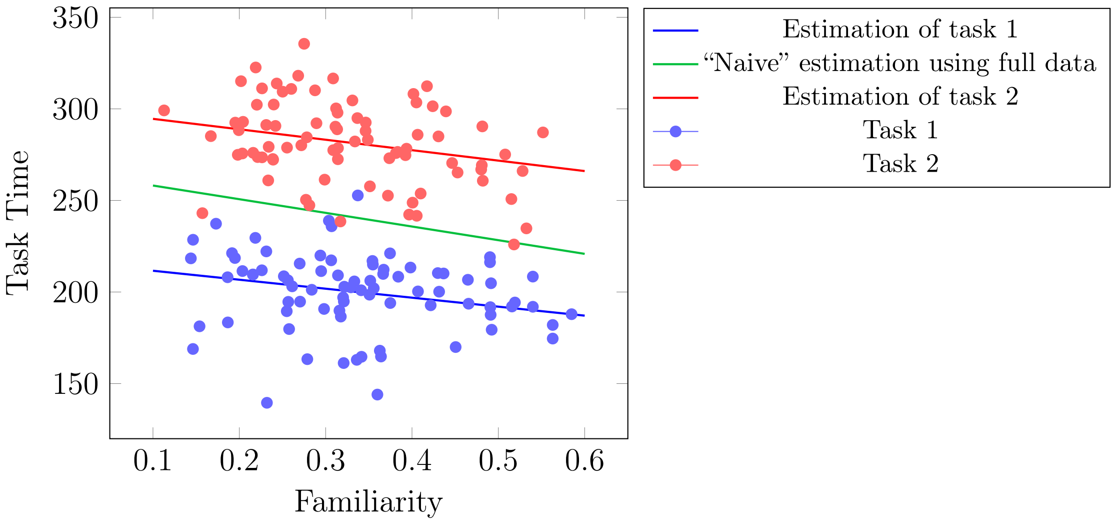


Figure 4.6: Estimations from [ASAPI] algorithm (red and blue lines) versus a hypothetical non-standardized regression (green line)

Building on this section, in the next section we characterize the predicted times of three different models; one that does not retain workers' performance-privacy (i.e., pair-specific model) and two that obscure it (i.e., [AP] and [ASAPI]). We derive expressions for finite-time predictions in Section 4.3.2. Ultimately, in Section 4.4, we characterize the expected discrepancy in steady-state due to using the latter two models.

4.3.2 Prediction Models

The predictions (4.20) and (4.21) are simple to compare in terms of $\hat{\beta}_i^1$ and $\hat{\beta}^1$, but we aim to compare them with respect to $\hat{\gamma}_i^1$ because $\hat{\gamma}_i^1$'s have a direct relation with the familiarity effect $\hat{\gamma}_{ij}^1$ of the true model. We rewrite the predicted times of the models in Section 4.3.1 to obtain similar expressions, in terms of $\hat{\gamma}_{ij}^1$ and $\hat{\gamma}_i^1$, that will aid with their comparison, both in finite-time and in steady-state.

(i) True model (Pair-specific model; performance-aware model) (4.4)

(ii) Task-level model ([AP] model; Task-specific model) (4.6)

(iii) 5-Step model ([ASAPI] model) (4.19)

True model. Since $\hat{\gamma}_{ij}^0 = \hat{\mu}(T_{ij}(n)) - \hat{\gamma}_{ij}^1\hat{\mu}(F_{ij}(n+1))$ by linear regression, $\forall(i,j) \in \boldsymbol{I}(n+1) \times \boldsymbol{J}(n+1)$ we have that

$$\begin{aligned}\hat{T}_{ij}(F(n+1);\hat{\boldsymbol{\gamma}}_{ij}) &= \hat{\gamma}_{ij}^0 + \hat{\gamma}_{ij}^1 F_{ij}(n+1)\\ &= \hat{\mu}(T_{ij}(n)) + \hat{\gamma}_{ij}^1(F_{ij}(n+1) - \hat{\mu}(F_{ij}(n))).\end{aligned} \tag{4.23}$$

Task-level model. Since $\hat{\gamma}_i^0 = \hat{\mu}(T_i(n)) - \hat{\gamma}_i^1\hat{\mu}(F_i(n+1))$ by linear regression, $\forall(i,j) \in \boldsymbol{I}(n+1) \times \boldsymbol{J}(n+1)$ we have that

$$\begin{aligned}\hat{T}_i(F_{ij}(n+1);\hat{\boldsymbol{\gamma}}_i) &= \hat{\gamma}_i^0 + \hat{\gamma}_i^1 F_{ij}(n+1)\\ &= \hat{\mu}(T_i(n)) + \hat{\gamma}_i^1(F_{ij}(n+1) - \hat{\mu}(F_i(n))).\end{aligned} \tag{4.24}$$

The next goal is to transform the main prediction model (4.18) into a form that use $\hat{\gamma}_i^1$'s, so that it is easy to compare with (4.24).

5-Step model. By (4.18), $\forall (i,j) \in \boldsymbol{I}(n+1) \times \boldsymbol{J}(n+1)$ we have that

$$\begin{aligned}\hat{T}_i(F_{ij}(n+1); \hat{\boldsymbol{\Gamma}}_i) &= \hat{\mu}(T_i(n)) + \hat{\sigma}(T_i(n)) \left(\hat{\beta}^1 \frac{F_{ij}(n+1) - \hat{\mu}(F_i(n))}{\hat{\sigma}(F_i(n))} \right) \\ &= \hat{\mu}(T_i(n)) + \hat{\sigma}(T_i(n)) \left(\left(\sum_{i' \in \mathcal{I}(n)} \frac{N_{i'}(n)}{N(n)} \hat{\beta}^1_{i'} \right) \frac{F_{ij}(n+1) - \hat{\mu}(F_i(n))}{\hat{\sigma}(F_i(n))} \right),\end{aligned}$$

where the second equality is by (4.16) of Lemma 4.3.1. Furthermore, by (4.13), $\hat{\beta}^1_{i'}$ can be substituted, i.e.,

$$\begin{aligned}&\hat{T}_i(F_{ij}(n+1); \hat{\boldsymbol{\Gamma}}_i) \\ &= \hat{\mu}(T_i(n)) + \hat{\sigma}(T_i(n)) \left(\left(\sum_{i' \in \mathcal{I}(n)} \frac{N_{i'}(n)}{N(n)} \hat{\gamma}^1_{i'} \frac{\hat{\sigma}(F_{i'}(n))}{\hat{\sigma}(T_{i'}(n))} \right) \frac{F_{ij}(n+1) - \hat{\mu}(F_i(n))}{\hat{\sigma}(F_i(n))} \right) \\ &= \hat{\mu}(T_i(n)) + \frac{N_i(n)}{N(n)} \hat{\gamma}^1_i (F_{ij}(n+1) - \hat{\mu}(F_i(n))) \\ &\qquad + \frac{\hat{\sigma}(T_i(n))}{\hat{\sigma}(F_i(n))} \left(\sum_{i' \neq i} \frac{N_{i'}(n)}{N(n)} \hat{\gamma}^1_{i'} \frac{\hat{\sigma}(F_{i'}(n))}{\hat{\sigma}(T_{i'}(n))} \right) (F_{ij}(n+1) - \hat{\mu}(F_i(n))) \\ &= \hat{\mu}(T_i(n)) + (F_{ij}(n+1) - \hat{\mu}(F_i(n+1))) \\ &\qquad \left[\frac{N_i(n)}{N(n)} \hat{\gamma}^1_i + \frac{\hat{\sigma}(T_i(n))}{\hat{\sigma}(F_i(n))} \left(\sum_{i' \neq i} \frac{N_{i'}(n)}{N(n)} \hat{\gamma}^1_{i'} \frac{\hat{\sigma}(F_{i'}(n))}{\hat{\sigma}(T_{i'}(n))} \right) \right] \\ &= \hat{\mu}(T_i(n)) + (F_{ij}(n+1) - \hat{\mu}(F_i(n))) \\ &\qquad \left[p_i(n) \hat{\gamma}^1_i + \frac{\hat{\sigma}(T_i(n))}{\hat{\sigma}(F_i(n))} \left(\sum_{i' \neq i} p_{i'}(n) \hat{\gamma}^1_{i'} \frac{\hat{\sigma}(F_{i'}(n))}{\hat{\sigma}(T_{i'}(n))} \right) \right],\end{aligned} \tag{4.25}$$

where

$$p_i(n) := \frac{N_i(n)}{N(n)} \tag{4.26}$$

represents the probability of a task being task i. It depends on the arrival rates $G(I,J)$

and the total number of tasks in the system; yet, it does not depend on the assignment mechanism.

The prediction of the task-level model in (4.24) scales $(F_{ij}(n+1) - \hat{\mu}(F_i(n)))$ by $\hat{\gamma}_i^1$, which only depends on task i itself. In contrast, in (4.25) the prediction of the 5-Step model scales $(F_{ij}(n+1) - \hat{\mu}(F_i(n)))$ by

$$p_i(n)\hat{\gamma}_i^1 + \frac{\hat{\sigma}(T_i(n))}{\hat{\sigma}(F_i(n))}\left(\sum_{i' \neq i} p_{i'}(n)\hat{\gamma}_{i'}^1 \frac{\hat{\sigma}(F_{i'}(n))}{\hat{\sigma}(T_{i'}(n))}\right), \tag{4.27}$$

which captures the firm-wide effect of familiarity across all tasks in the system on task i, explicitly showing the impact of $i' \neq i$ on the prediction made for task i. It shrinks the impact of task i's own slope coefficient $\hat{\gamma}_i^1$ by $p_i(n)$ assuming $p_i(n) < 1$ for any task i (i.e., $|\boldsymbol{I}| \geq 2$). Furthermore, each slope coefficient $\gamma_{i'}^1$ is scaled by

1. $p_{i'}(n)$: The more likely that task $i' \neq i$ arrives, the higher its impact on $\hat{T}_i(F_{ij}(n+1); \hat{\mathbf{\Gamma}}_i)$.

2. $\hat{\sigma}(F_{i'}(n))/\hat{\sigma}(T_{i'}(n))$: Tasks for which $F_{i'}$ varies more relative to $T_{i'}$, across all j's, have greater impact on $\hat{T}_{ij}(n+1)$. In other words, any task $i' \neq i$ for which the completion time is less variant to familiarity has a greater impact on predicted time.[7] It is an artifact of converting the $\hat{\beta}_{i'}^1$ into $\hat{\gamma}_{i'}^1$, i.e., (4.13).

Inversely, the weighted sum of the $\hat{\gamma}_{i'}^1$'s is scaled by $\hat{\sigma}(T_i(n))/\hat{\sigma}(F_i(n))$. If T_i varies more with respect to F_i, then the effect of all other tasks $i' \neq i$ on task i's predicted time is amplified.[8]

[7] $T_{i'}$ is less variant to $F_{i'}$ if $\hat{\gamma}_{i'}^1$ is smaller; thus, offsetting (i.e., correcting for) the impact of $\hat{\gamma}_{i'}^1$ on the prediction.

[8] T_i is more variant to F_i if $\hat{\gamma}_i^1$ is larger; thus, offsetting (i.e., correcting for) the impact of $\hat{\gamma}_i^1$ on the prediction.

4.4 Policies in Steady-State

In Section 4.4.1, we derive the long-run discrepancy in the predicted completion times between (i) the task-level model (i.e., [AP] model) and the pair-level model and (ii) the [ASAPI] model and [AP] model. In Sections 4.4.2 and 4.4.3, we formulate the steady-state optimization models. Then, in Section 4.4.4, we study the behavior of a new type of policy –the egalitarian policy– which is instrumental to upper bound the prediction degradation due to being performance-blind, i.e., being equitable.

4.4.1 Steady-State Analysis Under [ASAPI]

In this context, we assume that the analyst runs the PTO loop using the [ASAPI] (i.e., 5-Step) prediction model and obtains the emerging $\mathbb{E}[F_{ij}(\infty)] = \Pi_{ij}$ $\forall(i,j)$. Thus, Π_{ij} $\forall(i,j)$ depends critically on [ASAPI].

Assumption 9 (Convergence)**.** As $n \to \infty$, the limits of $\{\hat{\mu}(T_{ij}(n)), \hat{\sigma}(T_{ij}(n)), \hat{\mu}(F_{ij}(n)), \hat{\sigma}(F_{ij}(n))\}$ and $\{\hat{\mu}(T_i(n)), \hat{\sigma}(T_i(n)), \hat{\mu}(F_i(n)), \hat{\sigma}(F_i(n))\}$ exist, and they converge to the true means and standard deviations of the corresponding variables, which exist by Assumption 7, e.g.,

$$\lim_{n\to\infty} \hat{\mu}(F_{ij}(n)) = \mathbb{E}[F_{ij}(\infty)], \quad \forall(i,j).$$

Before deriving the steady-state expressions, we first define several quantities that will be instrumental.

Definition 4.4.1. The probability of assigning j to i, conditional on doing task i, is $p_{j|i}(n) \coloneqq \frac{N_{ij}(n)}{N_i(n)}$ where its long-run average is

$$p_{j|i} \coloneqq \lim_{n\to\infty} p_{j|i}(n).$$

Conversely, the probability of assigning i to j, conditional on being performed by j, is

$$p_{i|j}(n) := \frac{N_{ij}(n)}{N_j(n)},$$

where its long-run average is

$$p_{i|j} := \lim_{n\to\infty} p_{i|j}(n).$$

The probability of doing task i was defined in (4.26) ($p_i(n) := \frac{N_i(n)}{N(n)}$) and its long-run average is

$$p_i := \lim_{n\to\infty} p_i(n).$$

By the law of large numbers, in steady-state, task i arrives with marginal probability of

$$g_i := \lim_{n\to\infty} \frac{N_i(n)}{n},$$

and, in steady-state, the marginal probability that worker j arrives is

$$g_j := \lim_{n\to\infty} \frac{N_j(n)}{n}.$$

We assume that the limits in Definition 4.4.1 exist.

Corollary 4.4.0.1 (Corollary of Lemma 4.3.1)**.** *Under Assumptions 7, 9, 10, and by Definition 4.4.1, $\hat{\gamma}_i^1$ is identified in steady-state. For a given i, it has the form*

$$\gamma_i^1 = \sum_{j\in\boldsymbol{J}} \left(\frac{\sigma^2(F_{ij}(\infty))}{\sigma^2(F_i(\infty))} p_{j|i}\gamma_{ij}^1 + \frac{\mathbb{E}[T_{ij}(\infty)] - \mathbb{E}[T_i(\infty)]}{\sigma^2(F_i(\infty))} p_{j|i}\mathbb{E}[F_{ij}(\infty)] \right),$$

which is a function of the (identified) γ_{ij}^1 $\forall(i,j)$ and of Π_{ij} $\forall(i,j)$ by Lemma 4.2.1. Further-

more, by Definition 4.4.1

$$\beta^1 = \sum_{i \in \boldsymbol{I}} p_i \beta_i^1. \tag{4.28}$$

Corollary 4.4.0.1 follows directly by Lemma 4.3.1, in the scenario when Assumptions 7, 9 and 10 hold.

Assumption 10 (Identification)**.** As $n \to \infty$, the regression parameter estimates converge to the true parameters, given Π_{ij} $\forall(i,j)$, i.e.,

1. $(\hat{\gamma}_{ij}^0, \hat{\gamma}_{ij}^1) \to (\gamma_{ij}^0, \gamma_{ij}^1)$

2. $(\hat{\gamma}_i^0, \hat{\gamma}_i^1) \to (\gamma_i^0, \gamma_i^1)$, where γ_i is a function of Π_{ij}.

3. $(\hat{\Gamma}_i^0, \hat{\Gamma}_i^1) \to (\Gamma_i^0, \Gamma_i^1)$, where Γ_i is a function of Π_{ij}.

Given that Π_{ij}'s are generated by [ASAPI] $\forall(i,j)$, we can now derive the steady-state task time predictions coming from the performance-aware (true) model and the equitable [AP] and [ASAPI] models, in terms of the model primitives.

1. True model (performance-aware).

$$\begin{aligned}
\mathbb{E}\left[\lim_{n\to\infty} \hat{T}_{ij}(F_{ij}(n+1); \hat{\boldsymbol{\gamma}}_{ij})\right] &= \mathbb{E}\left[\lim_{n\to\infty} (\hat{\mu}(T_{ij}(n)) + \hat{\gamma}_{ij}^1(F_{ij}(n+1) - \hat{\mu}(F_{ij}(n))))\right] \\
&= \mathbb{E}\left[(\gamma_{ij}^0 + \gamma_{ij}^1 \mathbb{E}[F_{ij}(\infty)]) + \gamma_{ij}^1 \left(F_{ij}(\infty) - \mathbb{E}[F_{ij}(\infty)]\right)\right] \\
&= \gamma_{ij}^0 + \gamma_{ij}^1 \Pi_{ij}, \qquad (4.29)
\end{aligned}$$

where the first equality follows by (4.23), the second by Assumptions 7, 9 and 10, and the third by Lemma 4.2.1.

2. Task-level model ([AP]; performance-blind).

$$
\begin{aligned}
&\mathbb{E}\left[\lim_{n\to\infty}\hat{T}_i(F_{ij}(n+1);\hat{\boldsymbol{\gamma}}_i)\right]\\
&=\mathbb{E}\left[\lim_{n\to\infty}\hat{\mu}(T_i(n))+\hat{\gamma}_i^1(F_{ij}(n+1)-\hat{\mu}(F_i(n)))\right]\\
&=\mathbb{E}\left[\lim_{n\to\infty}\sum_{j'\in\mathcal{J}(n)}p_{j'|i}(n)\hat{\mu}(T_{ij'}(n))\right.\\
&\qquad\left.+\hat{\gamma}_i^1\left[F_{ij}(n+1)-\sum_{j'\in\mathcal{J}(n)}p_{j'|i}(n)\hat{\mu}(F_{ij'}(n))\right]\right] && (4.30)\\
&=\mathbb{E}\left[\lim_{n\to\infty}\left(\sum_{j'\in\mathcal{J}(n)}p_{j'|i}(n)(\hat{\gamma}^0_{ij'}+\hat{\gamma}^1_{ij'}\hat{\mu}(F_{ij'}(n)))\right.\right.\\
&\qquad\left.\left.+\hat{\gamma}_i^1\left[F_{ij}(n+1)-\sum_{j'\in\mathcal{J}(n)}p_{j'|i}(n)\hat{\mu}(F_{ij'}(n))\right]\right)\right] && (4.31)\\
&=\sum_{j'\in\boldsymbol{J}}p_{j'|i}(\gamma^0_{ij'}+\gamma^1_{ij'}\Pi_{ij'})+\gamma_i^1\left[\Pi_{ij}-\sum_{j'\in\boldsymbol{J}}p_{j'|i}\Pi_{ij'}\right], && (4.32)
\end{aligned}
$$

where the first equality follows by (4.24), and the final equality by Lemma 4.2.1 under Assumptions 7, 9 and 10.

3. 5-step model ([ASAPI]; performance-blind).

$$
\begin{aligned}
&\mathbb{E}\left[\lim_{n\to\infty}\hat{T}_i(F_{ij}(n+1);\hat{\mathbf{\Gamma}}_i)\right]\\
&=\mathbb{E}\left[\lim_{n\to\infty}\sum_{j'}p_{j'|i}(n)(\hat{\gamma}^0_{ij'}+\hat{\gamma}^1_{ij'}\hat{\mu}(F_{ij'}(n)))\right]\\
&\quad+\mathbb{E}\left[\lim_{n\to\infty}\left\{\left[p_i(n)\hat{\gamma}_i^1+\frac{\hat{\sigma}(T_i(n))}{\hat{\sigma}(F_i(n))}\left(\sum_{i'\neq i}p_{i'}(n)\hat{\gamma}^1_{i'}\frac{\hat{\sigma}(F_{i'}(n))}{\hat{\sigma}(T_{i'}(n))}\right)\right]\right.\right.\\
&\quad\left.\left.\left(F_{ij}(n+1)-\sum_{j'\in\mathcal{J}(n)}p_{j'|i}(n)\hat{\mu}(F_{ij'}(n))\right)\right\}\right]
\end{aligned}
$$

$$
\begin{aligned}
&= \sum_{j' \in \boldsymbol{J}} p_{j'|i}(\gamma^0_{ij'} + \gamma^1_{ij'}\Pi_{ij'}) \\
&\quad + \left(p_i\gamma^1_i + \frac{\sigma(T_i(\infty))}{\sigma(F_i(\infty))} \sum_{i' \neq i} p_{i'}\gamma^1_{i'} \frac{\sigma(F_{i'}(\infty))}{\sigma(T_{i'}(\infty))} \right) \left[\Pi_{ij} - \sum_{j' \in \boldsymbol{J}} p_{j'|i}\Pi_{ij'} \right],
\end{aligned}
\tag{4.33}
$$

where the first equality follows by (4.25). In addition, it follows by expressing $\{\hat{\mu}(T_i(n)), \hat{\mu}(F_i(n))\}$ in terms of $\{\hat{\mu}(T_{ij}(n)), \hat{\mu}(F_{ij}(n))\}$, respectively, as in (4.30), and by expressing $\hat{\mu}(T_{ij}(n))$ in terms of $\hat{\mu}(F_{ij}(n))$, as in (4.31). The second equality follows by Assumptions 7, 9 and 10, and Lemma 4.2.1.

We note that the performance-blind models have different adjustments on

$$
(\Pi_{ij} - \sum_{j' \in \boldsymbol{J}} p_{j'|i}\Pi_{ij'}),
$$

which is the steady-state counterpart of $(F_{ij}(n+1) - \hat{\mu}(F_i(n)))$. In the task-level (i.e., [AP]) model, this adjustment is the identified familiarity effect of the task itself, i.e., γ^1_i in (4.32). In the 5-Step model, the adjustment is the steady-state firm-wide effect of familiarity across tasks on task i, i.e., the steady-state counterpart of (4.27), as shown in the second term of (4.33).

Lemma 4.4.1 (Muting of worker performance)**.** *By Lemma 4.2.1 and Assumptions 7, 9, and 10, the steady-state discrepancy between the true model's predicted time and the task-specific predicted time for a given pair (i, j) is*

$$
\begin{aligned}
&\mathbb{E}\left[\lim_{n\to\infty} \hat{T}_{ij}(F_{ij}(n+1); \hat{\boldsymbol{\gamma}}_{ij})\right] - \mathbb{E}\left[\lim_{n\to\infty} \hat{T}_i(F_{ij}(n+1); \hat{\boldsymbol{\gamma}}_i)\right] \\
&= \sum_{j' \neq j \in \boldsymbol{J}} \left(p_{j'|i} \left[(\gamma^0_{ij} - \gamma^0_{ij'}) + \Pi_{ij}(\gamma^1_{ij} - \gamma^1_i) + \Pi_{ij'}(\gamma^1_i - \gamma^1_{ij'}) \right] \right).
\end{aligned}
\tag{4.34}
$$

Proof of Lemma 4.4.1. For a fixed pair (i, j), subtracting (4.32) from (4.29) yields,

$$\begin{aligned}
&\mathbb{E}\left[\lim_{n\to\infty} \hat{T}_{ij}(F_{ij}(n+1); \hat{\boldsymbol{\gamma}}_{ij})\right] - \mathbb{E}\left[\lim_{n\to\infty} \hat{T}_{i}(F_{ij}(n+1); \hat{\boldsymbol{\gamma}}_{i})\right] \\
&= \gamma^0_{ij} + \gamma^1_{ij}\Pi_{ij} - \sum_{j'\in\boldsymbol{J}} p_{j'|i}(\gamma^0_{ij'} + \gamma^1_{ij'}\Pi_{ij'}) - \gamma^1_i \left[\Pi_{ij} - \sum_{j'\in\boldsymbol{J}} p_{j'|i}\Pi_{ij'}\right] \\
&= (1 - p_{j|i})(\gamma^0_{ij} + \gamma^1_{ij}\Pi_{ij}) - \sum_{j'\neq j\in\boldsymbol{J}} p_{j'|i}(\gamma^0_{ij'} + \gamma^1_{ij'}\Pi_{ij'}) \\
&\quad - \gamma^1_i \left[(1 - p_{j|i})\Pi_{ij} - \sum_{j'\neq j\in\boldsymbol{J}} p_{j'|i}\Pi_{ij'}\right] \\
&= \left[\sum_{j'\neq j\in\boldsymbol{J}} p_{j'|i}\right](\gamma^0_{ij} + \gamma^1_{ij}\Pi_{ij}) - \sum_{j'\neq j\in\boldsymbol{J}} p_{j'|i}(\gamma^0_{ij'} + \gamma^1_{ij'}\Pi_{ij'}) \\
&\quad - \gamma^1_i \left[\left[\sum_{j'\neq j\in\boldsymbol{J}} p_{j'|i}\right]\Pi_{ij} - \sum_{j'\neq j\in\boldsymbol{J}} p_{j'|i}\Pi_{ij'}\right] \\
&= \sum_{j'\neq j\in\boldsymbol{J}} p_{j'|i}((\gamma^0_{ij} - \gamma^0_{ij'}) + (\gamma^1_{ij}\Pi_{ij} - \gamma^1_{ij'}\Pi_{ij'})) - \gamma^1_i \sum_{j'\neq j\in\boldsymbol{J}} p_{j'|i}(\Pi_{ij} - \Pi_{ij'}),
\end{aligned}$$

which simplifies further into,

$$\begin{aligned}
&= \sum_{j'\neq j\in\boldsymbol{J}} p_{j'|i}((\gamma^0_{ij} - \gamma^0_{ij'}) + (\gamma^1_{ij}\Pi_{ij} - \gamma^1_{ij'}\Pi_{ij'}) - \gamma^1_i(\Pi_{ij} - \Pi_{ij'})) \\
&= \sum_{j'\neq j\in\boldsymbol{J}} p_{j'|i}((\gamma^0_{ij} - \gamma^0_{ij'}) + \Pi_{ij}(\gamma^1_{ij} - \gamma^1_i) + \Pi_{ij'}(\gamma^1_i - \gamma^1_{ij'})).
\end{aligned}$$

□

When (4.34) is nonzero, the prediction from the task-specific model is either higher or lower than the true model's prediction. Each summand represents the discrepancy between worker j and some $j' \neq j$. Recall that $\gamma^1_{ij} < 0$, by construction. To interpret (4.34), fix two distinct workers j and j' and suppose that the pair-specific familiarity effects are either both smaller (or larger) than γ^1_i, e.g., $\gamma^1_{ij} < \gamma^1_i$ and $\gamma^1_{ij'} < \gamma^1_i$. Then, the last two terms offset each other; $\gamma^1_{ij} - \gamma^1_i < 0$ and $\gamma^1_i - \gamma^1_{ij'} > 0$. In this case, the interplay between the last two terms ameliorates the prediction discrepancy due to using the task-specific model. However, if γ_{ij} and $\gamma_{ij'}$ are on opposite sides of γ^1_i, this exacerbates the prediction discrepancy.

Lemma 4.4.2 (Information Sharing Across Tasks)**.** *By Lemma 4.2.1 and Assumptions 7, 9, and 10, the steady-state discrepancy between the task-specific predicted time and the predicted time of [ASAPI] for a given pair* (i, j) *is*

$$\begin{aligned}
&\mathbb{E}\left[\lim_{n\to\infty} \hat{T}_i(F_{ij}(n+1); \hat{\boldsymbol{\gamma}}_i)\right] - \mathbb{E}\left[\lim_{n\to\infty} \hat{T}_i(F_{ij}(n+1); \hat{\mathbf{\Gamma}}_i)\right] \\
&= \left[\sum_{i'\neq i\in \boldsymbol{I}} p_{i'} \left(\gamma^1_i - \gamma^1_{i'} \frac{\sqrt{(\gamma^1_i)^2\sigma^2(F_i(\infty)) + \sigma^2(\epsilon_i(\infty))}}{\sigma(F_i(\infty))} \frac{\sigma(F_{i'}(\infty))}{\sqrt{(\gamma^1_{i'})^2\sigma^2(F_{i'}(\infty)) + \sigma^2(\epsilon_{i'}(\infty))}}\right)\right] \\
&\qquad \left[\Pi_{ij} - \sum_{j'\in \boldsymbol{J}} p_{j'|i}\Pi_{ij'}\right],
\end{aligned} \tag{4.35}$$

where $\epsilon_i(n)$ *and* $\epsilon_{i'}(n)$ *are defined in (4.5).*

Proof of Lemma 4.4.2. For a fixed pair (i, j), subtracting (4.33) from (4.32) yields,

$$\begin{aligned}
&\mathbb{E}\left[\lim_{n\to\infty} \hat{T}_i(F_{ij}(n+1); \hat{\boldsymbol{\gamma}}_i)\right] - \mathbb{E}\left[\lim_{n\to\infty} \hat{T}_i(F_{ij}(n+1); \hat{\mathbf{\Gamma}}_i)\right] \\
&= \left(\left[\sum_{i'\neq i\in \boldsymbol{I}} p_{i'}\right] \gamma^1_i - \frac{\sigma(T_i(\infty))}{\sigma(F_i(\infty))} \sum_{i'\neq i} p_{i'}\gamma^1_{i'} \frac{\sigma(F_{i'}(\infty))}{\sigma(T_{i'}(\infty))}\right) \left[\Pi_{ij} - \sum_{j'\in \boldsymbol{J}} p_{j'|i}\Pi_{ij'}\right] \\
&= \sum_{i'\neq i\in \boldsymbol{I}} p_{i'} \left(\gamma^1_i - \gamma^1_{i'} \frac{\sigma(T_i(\infty))}{\sigma(F_i(\infty))} \frac{\sigma(F_{i'}(\infty))}{\sigma(T_{i'}(\infty))}\right) \left[\Pi_{ij} - \sum_{j'\in \boldsymbol{J}} p_{j'|i}\Pi_{ij'}\right] \\
&= \left[\sum_{i'\neq i\in \boldsymbol{I}} p_{i'} \left(\gamma^1_i - \gamma^1_{i'} \frac{\sqrt{(\gamma^1_i)^2\sigma^2(F_i(\infty)) + \sigma^2(\epsilon_i(\infty))}}{\sigma(F_i(\infty))} \frac{\sigma(F_{i'}(\infty))}{\sqrt{(\gamma^1_{i'})^2\sigma^2(F_{i'}(\infty)) + \sigma^2(\epsilon_{i'}(\infty))}}\right)\right] \\
&\qquad \left[\Pi_{ij} - \sum_{j'\in \boldsymbol{J}} p_{j'|i}\Pi_{ij'}\right],
\end{aligned}$$

where the first equality follows by (4.32) and (4.33). The final equality follows by (4.5), i.e.,

$$\begin{aligned}
\hat{\sigma}^2(T_i(n)) &= \hat{\sigma}^2(\hat{\gamma}_i^0 + \hat{\gamma}_i^1 F_i(n) + \hat{\epsilon}_i(n)) \\
&= (\hat{\gamma}_i^1)^2 \hat{\sigma}^2(F_i(n)) + \hat{\sigma}^2(\epsilon_i(n)) + 2\hat{\gamma}_i^1 \hat{Cov}(F_i(n), \hat{\epsilon}_i(n)) \\
&= (\hat{\gamma}_i^1)^2 \hat{\sigma}^2(F_i(n)) + \hat{\sigma}^2(\hat{\epsilon}_i(n)).
\end{aligned}$$

Under Assumptions 7 and 9, as $n \to \infty$, $\sigma^2(T_i(\infty)) = (\gamma_i^1)^2 \sigma^2(F_i(\infty)) + \sigma^2(\epsilon_i(\infty))$. □

Lemma 4.4.2 shows that the discrepancy in the steady-state predictions of [AP] and [ASAPI] have the form (4.35). If $\sigma^2(\epsilon_i(\infty)) \to 0 \ \forall i$, then (4.35) converges to 0. That is, as the uncertainty in task times vanish, the steady-state predictions of [AP] and [ASAPI] converge.

4.4.2 Optimal Assignments in Steady-State

In Section 4.2, we built a model that minimizes the total time of all pairs by assigning pairs optimally. Now, we will introduce a long-run version of this model, which solves for $p_{j|i}$, i.e., the limiting fraction of time any worker j should be assigned to any task i, given that the task has arrived. In the performance-aware model, the analyst (hypothetically) uses the pair-specific predictions, denoted by $\mathbb{E}[\lim_{n\to\infty} \hat{T}_{ij}(F_{ij}(n+1); \hat{\boldsymbol{\gamma}}_{ij})]$. We represent its steady-state counterpart by $\mathbb{E}[\hat{T}_{ij}(F_{ij}(\infty); \hat{\boldsymbol{\gamma}}_{ij})]$. The total steady-state predictions across all encounters is

$$\sum_{i \in \boldsymbol{I}} \sum_{j \in \boldsymbol{J}} \Pi_{ij} \mathbb{E}[\hat{T}_{ij}(F_{ij}(\infty); \hat{\boldsymbol{\gamma}}_{ij})]. \tag{4.36}$$

The limiting probability can be written as follows,

$$\Pi_{ij} = \lim_{n\to\infty} \frac{N_{ij}(n)}{n} = \frac{N_{ij}(n)}{N_i(n)} \cdot \frac{N_i(n)}{n} = p_{j|i} \cdot g_i, \quad \forall (i,j), \tag{4.37}$$

where the first equality follows by Assumption 8, and the third equality follows by Definition 4.4.1. Then, by (4.37), the objective function in (4.36) becomes

$$\sum_{i\in \boldsymbol{I}}\sum_{j\in \boldsymbol{J}} p_{j|i}\cdot g_i \mathbb{E}[\hat{T}_{ij}(F_{ij}(\infty);\hat{\gamma}_{ij})]. \tag{4.38}$$

It is possible to rewrite $\Pi_{ij} = \lim_{n\to\infty}\frac{N_{ij}(n)}{n} = \frac{N_{ij}(n)}{N_j(n)}\cdot\frac{N_j(n)}{n} = p_{i|j}\cdot g_j$. Thus,

$$p_{j|i}\cdot g_i = p_{i|j}\cdot g_j, \quad \forall (i,j). \tag{4.39}$$

We now formulate the optimal (i.e., performance-aware) program in terms of the decision variable $p_{j|i}$, using estimates from data,

$$\begin{aligned} \hat{Z}^*_{min} = \quad \underset{p}{\text{minimize}} \quad & \sum_{i\in \boldsymbol{I}} g_i \sum_{j\in \boldsymbol{J}} p_{j|i}\mathbb{E}[\hat{T}_{ij}(F_{ij}(\infty);\hat{\gamma}_{ij})] \\ \text{subject to} \quad & \sum_{j\in \boldsymbol{J}} p_{j|i} = 1, \qquad \forall i : i\in \boldsymbol{I}, \\ & \sum_{i\in \boldsymbol{I}} p_{j|i} g_i \le g_j, \qquad \forall j : j\in \boldsymbol{J}, \\ & 0\le p_{j|i}\le g_{j|i}, \quad \forall (i,j) : i\in \boldsymbol{I}, j\in \boldsymbol{J}. \end{aligned} \tag{4.40}$$

The first set of constraints in (4.40) guarantee that given task i has arrived, it is covered, i.e., $\sum_{j\in \boldsymbol{J}} p_{j|i} = 1\ \forall i$. The second set of constraints guarantee that workers are not assigned beyond their availability, i.e., $\sum_{i\in \boldsymbol{I}} p_{i|j} \le 1\ \forall j$.[9] But then, by (4.39), this implies $\sum_{i\in \boldsymbol{I}} p_{j|i}\cdot g_i \le g_j\ \forall j$.[10] Finally, the constraints should adhere to pair-specific availability constraints, i.e., $p_{j|i} \le \frac{g_{ij}}{g_i} := g_{j|i}\,\forall (i,j)$, where $g_{j|i}$ denotes the probability that worker j arrives, given that

[9] If $|\boldsymbol{I}(n)| = |\boldsymbol{J}(n)|\ \forall n$, then it becomes $\sum_{i\in \boldsymbol{I}} p_{i|j} = 1\ \forall j$.

[10] If $|\boldsymbol{I}(n)| = |\boldsymbol{J}(n)|\ \forall n$, then it becomes $\sum_{i\in \boldsymbol{I}} p_{j|i} g_i = g_j\ \forall j$.

task i has arrived, in an arbitrary period. We denote the set of constraints by

$$\begin{aligned} \mathcal{P} = \{p_{j|i} : & \sum_{j \in \boldsymbol{J}} p_{j|i} = 1, \quad \forall i \in \boldsymbol{I} \\ & \sum_{i \in \boldsymbol{I}} p_{j|i} g_i \leq g_j, \quad \forall j \in \boldsymbol{J} \\ & 0 \leq p_{j|i} \leq g_{j|i}, \quad \forall i \in \boldsymbol{I}, \forall j \in \boldsymbol{J}\}. \end{aligned} \tag{4.41}$$

We emphasize that (4.40) is a relaxation of the true steady-state behavior as it does not fully capture the information in $G(\boldsymbol{I}, \boldsymbol{J})$ and only uses marginal arrival probabilities g_i, g_j and $g_{j|i}$. Nonetheless, it is a plausible approximation for the steady-state behavior of the performance-aware model. Although its solution may not be implementable, since we use it to provide bounds on performance, it suffices that it is as an approximate program in our setting. Note that the objective function of (4.40) is quadratic in $p_{j|i}$, i.e.,

$$\begin{aligned} \sum_{i \in \boldsymbol{I}} g_i \sum_{j \in \boldsymbol{J}} p_{j|i} \mathbb{E}[\hat{T}_{ij}(F_{ij}(\infty); \hat{\boldsymbol{\gamma}}_{ij})] &= \sum_{i \in \boldsymbol{I}} g_i \sum_{j \in \boldsymbol{J}} p_{j|i} \mathbb{E}[\hat{\gamma}^0_{ij} + \hat{\gamma}^1_{ij} F_{ij}(\infty)] \\ &= \sum_{i \in \boldsymbol{I}} g_i \sum_{j \in \boldsymbol{J}} p_{j|i} \left(\hat{\gamma}^0_{ij} + \hat{\gamma}^1_{ij} \Pi_{ij}\right) \\ &= \sum_{i \in \boldsymbol{I}} g_i \sum_{j \in \boldsymbol{J}} p_{j|i} \left(\hat{\gamma}^0_{ij} + \hat{\gamma}^1_{ij} p_{j|i} g_i\right) \\ &= \sum_{i \in \boldsymbol{I}} g_i \sum_{j \in \boldsymbol{J}} p_{j|i} \hat{\gamma}^0_{ij} + \sum_{i \in \boldsymbol{I}} (g_i)^2 \sum_{j \in \boldsymbol{J}} \hat{\gamma}^1_{ij} (p_{j|i})^2, \end{aligned} \tag{4.42}$$

where the second equality follows by Lemma 4.2.1, and the third equality follows by (4.37). Furthermore, under Assumption 10, the (exact) performance-aware model becomes

$$\begin{aligned} Z^*_{min} = \underset{p \in \mathcal{P}}{\text{minimize}} \quad & \sum_{i \in \boldsymbol{I}} g_i \sum_{j \in \boldsymbol{J}} p_{j|i} \mathbb{E}[\hat{T}_{ij}(F_{ij}(\infty); \boldsymbol{\gamma}_{ij})] \\ \text{subject to} \quad & \mathcal{P}. \end{aligned} \tag{4.43}$$

We denote the exact maximization corresponding to (4.43) by Z^*_{max}, i.e., Z^*_{max} gives the worst-case outcome when the same objective function is maximized, subject to $\mathcal{P}$. The maximization program corresponding to (4.40) is denoted by $\hat{Z}^*_{max}$.

4.4.3 Equitable Assignments in Steady-State

Replacing the predicted time in (4.40) with the two other models' predictions yields the *performance-blind models*. The prediction of the [AP] model is $\mathbb{E}[\hat{T}_i(F_{ij}(\infty); \hat{\boldsymbol{\gamma}}_i)]$, and $\mathbb{E}[\hat{T}_i(F_{ij}(\infty); \hat{\boldsymbol{\Gamma}}_i)]$ denotes the prediction of the [ASAPI] model. Respectively, these models are,

$$\begin{aligned} \hat{Z}^*_{[AP]} = \min_{p \,\in\, \mathcal{P}} & \quad \sum_{i \in \boldsymbol{I}} g_i \sum_{j \in \boldsymbol{J}} p_{j|i} \mathbb{E}[\hat{T}_i(F_{ij}(\infty); \hat{\boldsymbol{\gamma}}_i)] \\ \text{s.t.} & \quad \mathcal{P}, \end{aligned} \tag{4.44}$$

for [AP], and for [ASAPI], it is

$$\begin{aligned} \hat{Z}^*_{[ASAPI]} = \min_{p \,\in\, \mathcal{P}} & \quad \sum_{i \in \boldsymbol{I}} g_i \sum_{j \in \boldsymbol{J}} p_{j|i} \mathbb{E}[\hat{T}_i(F_{ij}(\infty); \hat{\boldsymbol{\Gamma}}_i)] \\ \text{s.t.} & \quad \mathcal{P}. \end{aligned} \tag{4.45}$$

By (4.6) and Lemma 4.2.1, the objective of (4.44) becomes

$$\begin{aligned} \sum_{i \in \boldsymbol{I}} g_i \sum_{j \in \boldsymbol{J}} p_{j|i} \mathbb{E}[\hat{T}_i(F_{ij}(\infty); \hat{\boldsymbol{\gamma}}_i)] &= \sum_{i \in \boldsymbol{I}} g_i \sum_{j \in \boldsymbol{J}} p_{j|i} \mathbb{E}[\hat{\gamma}^0_i + \hat{\gamma}^1_i F_{ij}(\infty)] \\ &= \sum_{i \in \boldsymbol{I}} g_i \sum_{j \in \boldsymbol{J}} p_{j|i} (\hat{\gamma}^0_i + \hat{\gamma}^1_i \Pi_{ij}) \\ &= \sum_{i \in \boldsymbol{I}} g_i \sum_{j \in \boldsymbol{J}} p_{j|i} (\hat{\gamma}^0_i + \hat{\gamma}^1_i p_{j|i} g_i) \\ &= \sum_{i \in \boldsymbol{I}} g_i \hat{\gamma}^0_i + \sum_{i \in \boldsymbol{I}} (g_i)^2 \hat{\gamma}^1_i \sum_{j \in \boldsymbol{J}} (p_{j|i})^2. \end{aligned} \tag{4.46}$$

Similarly, it follows by (4.20) and Lemma 4.2.1 that the objective of (4.45) is

$$\begin{aligned}\sum_{i\in\boldsymbol{I}} g_i \sum_{j\in\boldsymbol{J}} p_{j|i}\mathbb{E}[\hat{T}_i(F_{ij}(\infty);\hat{\boldsymbol{\Gamma}}_i)] &= \sum_{i\in\boldsymbol{I}} g_i \sum_{j\in\boldsymbol{J}} p_{j|i}\mathbb{E}[\hat{\Gamma}_i^0 + \hat{\Gamma}_i^1 F_{ij}(\infty)] \\ &= \sum_{i\in\boldsymbol{I}} g_i \sum_{j\in\boldsymbol{J}} p_{j|i}(\hat{\Gamma}_i^0 + \hat{\Gamma}_i^1 \Pi_{ij}) \\ &= \sum_{i\in\boldsymbol{I}} g_i \sum_{j\in\boldsymbol{J}} p_{j|i}(\hat{\Gamma}_i^0 + \hat{\Gamma}_i^1 p_{j|i} g_i) \\ &= \sum_{i\in\boldsymbol{I}} g_i \hat{\Gamma}_i^0 + \sum_{i\in\boldsymbol{I}} (g_i)^2 \hat{\Gamma}_i^1 \sum_{j\in\boldsymbol{J}} (p_{j|i})^2. \qquad (4.47)\end{aligned}$$

As (4.40), (4.44), and (4.45) have the same constraint matrix $\mathcal{P}$, the gap between their objective functions corresponds to the approximate *cost of performance-privacy.*

Proposition 2. Let Π_{ij} $\forall(i,j)$ be an optimal solution to (4.40), where $\Pi_{ij} = p_{j|i} \cdot g_i$ by (4.37) for $p \in \mathcal{P}$. If Π_{ij}'s are implementable, i.e., if there exists a policy such that $\mathbb{E}[F_{ij}(\infty)] = \Pi_{ij}$ $\forall(i,j)$, then by Lemma 4.2.1 and under Assumptions 7, 9 and 10, the objective functions of the performance-aware and equitable models are equal. Otherwise, they are not equal.

In Proposition 2, we show that the objective functions of the three models are equal *assuming* that the analyst is able to recover the true $\boldsymbol{\gamma}_i$ $\forall i$, corresponding with Π_{ij}, by running the PTO loop (defined in Algorithm 1) where the predictions are computed in an equitable manner. However, in our setting, there is no guarantee to recover the identified parameters because it is unclear how to implement the steady-state solution; that is, it is unclear what the "right" $p_{j|i}$ $\forall(i,j)$ are (what the "right" Π_{ij} are), which corresponds to the steady-state, performance-aware model. To see this, observe that the equitable, steady-state objective functions (4.46) and (4.47) treat workers interchangeably. These models are fundamentally "defective" because they allow assigning workers to tasks in an interchangeable manner, such that there are multiple alternative optimal solutions[11]. Each solution may result in a different Π_{ij} $\forall(i,j)$ solution, and thus, different $\boldsymbol{\gamma}_i$ $\forall i$, as seen by Corollary 4.4.0.1. Thus, the equitable models lack the fidelity to drive the system to the true (pair-specific) optimal. In contrast, we observe from the performance-aware objective function (4.42) that, in reality, the workers are *not* interchangeable, i.e., the assignment of workers matters.

Proof of Proposition 2. The proof has two parts; the steady-state prediction of the performance-aware model is compared against the steady-state predictions of [AP] and [ASAPI], respectively.

1. Performance-aware vs. [AP]. By taking the weighted sum of (4.32), we show that it is equal to the weighted sum of (4.29), where the weights $p_{j|i}$ are summed across j. For a fixed (i,j),

$$\begin{aligned}&\sum_{j\in\boldsymbol{J}} p_{j|i}\left(\sum_{j'\in\boldsymbol{J}} p_{j'|i}\left(\gamma^0_{ij'}+\gamma^1_{ij'}\Pi_{ij'}\right)+\gamma^1_i\left[\Pi_{ij}-\sum_{j'\in\boldsymbol{J}} p_{j'|i}\Pi_{ij'}\right]\right)\\ &=\sum_{j'\in\boldsymbol{J}} p_{j'|i}(\gamma^0_{ij'}+\gamma^1_{ij'}\Pi_{ij'})\sum_{j\in\boldsymbol{J}} p_{j|i}+\gamma^1_i\left[\sum_{j\in\boldsymbol{J}} p_{j|i}\Pi_{ij}-\sum_{j'\in\boldsymbol{J}} p_{j'|i}\Pi_{ij'}\sum_{j\in\boldsymbol{J}} p_{j|i}\right]\\ &=\sum_{j'\in\boldsymbol{J}} p_{j'|i}(\gamma^0_{ij'}+\gamma^1_{ij'}\Pi_{ij'}).\end{aligned}$$

But then, this implies that taking the weighted sums across i, weighted by g_i, these quantities are equal. That is, the objective functions of the performance-aware and [AP] policies are equal to

$$\sum_{i\in\boldsymbol{I}} g_i\sum_{j\in\boldsymbol{J}} p_{j|i}(\gamma^0_{ij}+\gamma^1_{ij}\Pi_{ij}). \tag{4.48}$$

[11] If $|\boldsymbol{I}|=|\boldsymbol{J}|$, there are $|\boldsymbol{J}|!$ ways in which workers can be assigned. In general, there are $\frac{|\boldsymbol{J}|!}{(|\boldsymbol{J}|-|\boldsymbol{I}|)!}$ alternative optimal solutions.

2. Performance-aware vs. [ASAPI]. Starting with the weighted sum of (4.33), we show that it is equal to the weighted sum of (4.29), where the weights $p_{j|i}$ are summed across j. For a fixed (i, j),

$$\sum_{j \in \boldsymbol{J}} p_{j|i} \left\{ \sum_{j' \in \boldsymbol{J}} p_{j'|i}(\gamma^0_{ij'} + \gamma^1_{ij'}\Pi_{ij'}) + \left(p_i\gamma^1_i + \frac{\sigma(T_i(\infty))}{\sigma(F_i(\infty))} \sum_{i' \neq i} p_{i'}\gamma^1_{i'} \frac{\sigma(F_{i'}(\infty))}{\sigma(T_{i'}(\infty))} \right) \right.$$
$$\left. \left[\Pi_{ij} - \sum_{j' \in \boldsymbol{J}} p_{j'|i}\Pi_{ij'} \right] \right\}$$
$$= \sum_{j' \in \boldsymbol{J}} p_{j'|i}(\gamma^0_{ij'} + \gamma^1_{ij'}\Pi_{ij'}) \sum_{j \in \boldsymbol{J}} p_{j|i}$$
$$+ \left(p_i\gamma^1_i + \frac{\sigma(T_i(\infty))}{\sigma(F_i(\infty))} \sum_{i' \neq i} p_{i'}\gamma^1_{i'} \frac{\sigma(F_{i'}(\infty))}{\sigma(T_{i'}(\infty))} \right) \sum_{j \in \boldsymbol{J}} p_{j|i} \left[\Pi_{ij} - \sum_{j' \in \boldsymbol{J}} p_{j'|i}\Pi_{ij'} \right]$$
$$= \sum_{j' \in \boldsymbol{J}} p_{j'|i}(\gamma^0_{ij'} + \gamma^1_{ij'}\Pi_{ij'}) \sum_{j \in \boldsymbol{J}} p_{j|i}$$
$$+ \left(p_i\gamma^1_i + \frac{\sigma(T_i(\infty))}{\sigma(F_i(\infty))} \sum_{i' \neq i} p_{i'}\gamma^1_{i'} \frac{\sigma(F_{i'}(\infty))}{\sigma(T_{i'}(\infty))} \right) \left[\sum_{j \in \boldsymbol{J}} p_{j|i}\Pi_{ij} - \sum_{j' \in \boldsymbol{J}} p_{j'|i}\Pi_{ij'} \sum_{j \in \boldsymbol{J}} p_{j|i} \right]$$
$$= \sum_{j' \in \boldsymbol{J}} p_{j'|i}(\gamma^0_{ij'} + \gamma^1_{ij'}\Pi_{ij'}).$$

Similar to the first case, taking the weighted sums across i, weighted by g_i, these quantities are equal. That is, the objective functions of the performance-aware and [ASAPI] policies are equal to (4.48). □

We aim to compare the steady-state, performance-aware model to running the PTO loop with an equitable prediction algorithm to find the cost of equity. In order to do so, we use an alternative policy, again in steady-state, which we show yields the worst-case solution under certain assumptions on the primitives. Under weaker assumptions, this policy is not the worst-case solution, yet it serves as a reasonable benchmark. We call this alternative policy the *egalitarian* policy.

4.4.4 Egalitarian Assignments in Steady-State

We now introduce the egalitarian policy.

Definition 4.4.2 (Egalitarian Policy)**.** Choose the conditional assignment probabilities such that;

- No worker has a greater propensity to execute a task than any other worker, i.e., $p_{j|i} = \frac{1}{|\boldsymbol{J}|}$ $\forall i \in \boldsymbol{I}$,
- No task has a greater propensity to be performed by a worker than any other task, i.e., $p_{i|j} = \frac{1}{|\boldsymbol{I}|}$ $\forall j \in \boldsymbol{J}$.

To bound the difference between the (steady-state) performance-aware policy and any other policy (including the egalitarian policy), we adopt the following structure on worker-task arrivals.

Assumption 11. We suppose that the schedule is set up (by an exogenous agent) in a way such that

(a) $|\boldsymbol{I}(n)| = |\boldsymbol{J}(n)|$, $\forall n$

(b) $g_i = g_j = g$, $\forall (i, j)$

(c) $g_{j|i} = 1$, $\forall (i, j)$ (full availability)

Under part (a) of Assumption 11, the number of arriving workers and tasks in each period are equal. In (4.3a)–(4.3d), this amounts to exchanging the inequality sign in (4.3c) with equality, i.e., the single-period assignment problem is transformed into a *perfect* matching problem. Part (b) is to assume equal (marginal) arrival probabilities for all (i, j). Since $g_{j|i} \cdot g_i = g_{i|j} \cdot g_j$, (b) implies that $g_{j|i} = g_{i|j}$, which is 1 by part (c). Part (c) indicates that workers and tasks are synchronized, i.e., the system either operates in full capacity or not at all. The rate at which tasks and workers (collectively) arrive is g.

Under the egalitarian policy, Assumption 11 (a) implies (b), since by the Bayes' theorem,

$$\begin{aligned} & p_{i|j} \cdot g_j = p_{j|i} \cdot g_i, \quad \forall (i,j) \\ \Longrightarrow & \frac{1}{|\boldsymbol{I}|} g_j = \frac{1}{|\boldsymbol{J}|} g_i, \quad \forall (i,j) \\ \Longrightarrow & g_j = g_i, \quad \forall (i,j). \end{aligned}$$

Conversely, under the egalitarian policy, part (b) implies $\frac{1}{|\boldsymbol{I}|} = \frac{1}{|\boldsymbol{J}|}$, but this does not necessarily imply that the number of workers is equal to the number of total tasks, initially. That is, there may be an "excess" number of workers who need to be "fired" before implementing the egalitarian policy. To avoid this extra cost, we do not assume only part (b) of Assumption 11 when assessing the egalitarian policy, i.e, (a) is necessary. On the other hand, for the performance-aware policy, both (a) and (b) are needed to obtain $\sum\limits_{i \in \boldsymbol{I}} p_{j|i} = 1$ $\forall j$. Only (a) yields $\sum\limits_{i \in \boldsymbol{I}} p_{j|i} \cdot g_i = g_j \ \forall j$ (since by part (a), we have $\sum\limits_{i \in \boldsymbol{I}} p_{i|j} = 1 \ \forall j$), allowing for different marginal arrival probabilities. Only (b) yields

$$\begin{aligned} & p_{i|j} \cdot g_j = p_{j|i} \cdot g_i, \quad \forall (i,j) \\ \Longrightarrow & p_{i|j} = p_{j|i}, \quad \forall (i,j), \end{aligned}$$

which is pair-specific. For example, suppose that $g_i = g_j = g = 1 \ \forall (i,j)$, and $\boldsymbol{I} = \{A\}$ and $\boldsymbol{J} = \{1,2\}$. Then, $p_{A|1} = p_{1|A} = 1$ and $p_{A|2} = p_{2|A} = 0$ implies more workers arrive than needed.

Under Assumption 11, (4.40) simplifies into

$$\begin{aligned} \hat{Z}^*_{min} = \quad & \underset{p}{\text{minimize}} && \sum_{i\in\boldsymbol{I}} g \sum_{j\in\boldsymbol{J}} p_{j|i}\hat{\gamma}^0_{ij} + \sum_{i\in\boldsymbol{I}} g^2 \sum_{j\in\boldsymbol{J}} \hat{\gamma}^1_{ij}(p_{j|i})^2 \\ & \text{subject to} && \sum_{j\in\boldsymbol{J}} p_{j|i} = 1, \quad \forall i : i \in \boldsymbol{I}, \\ &&& \sum_{i\in\boldsymbol{I}} p_{j|i} = 1, \quad \forall j : j \in \boldsymbol{J}, \\ &&& \quad p_{j|i} \geq 0, \quad \forall (i,j) : i \in \boldsymbol{I}, j \in \boldsymbol{J}, \end{aligned} \tag{4.49}$$

where the second constraint is transformed from $\sum\limits_{i\in\boldsymbol{I}} p_{j|i} \cdot g_i \leq g_j \ \forall j$ to $\sum\limits_{i\in\boldsymbol{I}} p_{j|i} = 1 \ \forall j$.

For the egalitarian policy, under Assumptions 10 and 11, the objective in (4.42) becomes

$$\begin{aligned} Z^E &= \sum_{i\in\boldsymbol{I}} g_i \left[\sum_{j\in\boldsymbol{J}} p_{j|i}\gamma^0_{ij} + g_i \sum_{j\in\boldsymbol{J}} \gamma^1_{ij}(p_{j|i})^2 \right] \\ &= g \sum_{i\in\boldsymbol{I}} \sum_{j\in\boldsymbol{J}} \frac{\gamma^0_{ij}}{|\boldsymbol{J}|} + g^2 \sum_{i\in\boldsymbol{I}} \sum_{j\in\boldsymbol{J}} \frac{\gamma^1_{ij}}{|\boldsymbol{J}|^2} \\ &= \frac{g}{|\boldsymbol{J}|} \sum_{i\in\boldsymbol{I}} \sum_{j\in\boldsymbol{J}} \gamma^0_{ij} + \frac{g^2}{|\boldsymbol{J}|^2} \sum_{i\in\boldsymbol{I}} \sum_{j\in\boldsymbol{J}} \gamma^1_{ij}. \end{aligned} \tag{4.50}$$

Before concluding this section, we illustrate the egalitarian policy by an example in Table 4.2 with 5 tasks and workers. We highlight that there is no optimization problem under this policy.

4.5 Performance Bounds on the Equitable Policy

Under Assumption 11, we will derive the performance-aware and egalitarian policies under the following regimes, ordered from the most general to the most specific. In reality, there are more cases than those we list; however, since they are similar, we only study the following five cases.

Table 4.2: Sample schedule under Assumption 11, which is driven by the egalitarian policy. Suppose that $g = 0.6$. Each task is rotated among all workers. The lighter-color assignments beyond period $n = 8$ represent the continuation of the same matching schedule.

$n = 1$	$n = 2$	$n = 3$	$n = 4$	$n = 5$
task 1–worker 1		task 1–worker 2		task 1–worker 3
task 2–worker 2		task 2–worker 3		task 2–worker 4
task 3–worker 3		task 3–worker 4		task 3–worker 5
task 4–worker 4		task 4–worker 5		task 4–worker 1
task 5–worker 5		task 5–worker 1		task 5–worker 2
$n = 6$	$n = 7$	$n = 8$		
	task 1–worker 4	task 1–worker 5		task 1–worker1
	task 2–worker 5	task 2–worker 1		task 2–worker2
	task 3–worker 1	task 3–worker 2		task 3–worker3
	task 4–worker 2	task 4–worker 3		task 4–worker 4
	task 5–worker 3	task 5–worker 4		task 5–worker 5

1. **Heterogeneous Pairs:** *The base performance and familiarity effect can differ across pairs* $\{\gamma_{ij}^0, \gamma_{ij}^1\}$.

2. **Identical task or worker familiarity effect**: *The familiarity effect depends on workers* γ_j^1 *or tasks* γ_i^1, *base performance can differ across pairs* γ_{ij}^0.

3. **Identical pair familiarity effect**: *The familiarity effect is the same across pairs* γ^1, *base performance can differ across pairs* γ_{ij}^0.

4. **Identical tasks or workers**: *The base performance and familiarity effect depend on workers* $\{\gamma_j^0, \gamma_j^1\}$ *or tasks* $\{\gamma_i^0, \gamma_i^1\}$.

5. **Identical pairs**: *The base performance and familiarity effect are the same across pairs* $\{\gamma^0, \gamma^1\}$.

Figure 4.7 shows how the five cases relate to each other. The two possibilities under cases 2 and 4 give the same results regardless of whether the parameters differ across workers or tasks. We will study the identical tasks regime; i.e., the parameters differ across workers, and we argue that this is without loss of generality.

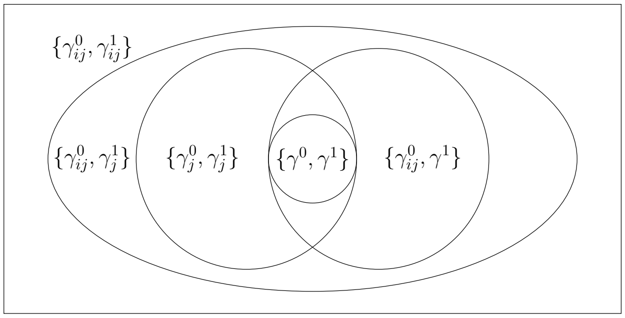

$\{\gamma^0_{ij}, \gamma^1_{ij}\}$
$\{\gamma^0_{ij}, \gamma^1_j\}$
$\{\gamma^0_j, \gamma^1_j\}$
$\{\gamma^0, \gamma^1\}$
$\{\gamma^0_{ij}, \gamma^1\}$

Figure 4.7: The relation of the cases we study.

We denote the discrepancy between the egalitarian policy and the performance-aware policy by $\Delta^{E}_{min} := Z^{\mathrm{E}} - Z^*_{min}$, and the discrepancy between the worst-case (objective-maximizing) policy and the egalitarian policy by $\Delta^{max}_{E} := Z^*_{max} - Z^{E}$. It follows that $\Delta^{max}_{min} := Z^*_{max} - Z^*_{min} = \Delta^{E}_{min} + \Delta^{max}_{E}$.

Notice that (4.49) is the LP relaxation of an assignment program, with a quadratic objective function, i.e.,

$$g \sum_{i \in \boldsymbol{I}} \sum_{j \in \boldsymbol{J}} p_{j|i} \hat{\gamma}^0_{ij} + g^2 \sum_{i \in \boldsymbol{I}} \sum_{j \in \boldsymbol{J}} \hat{\gamma}^1_{ij} (p_{j|i})^2. \tag{4.51}$$

Proposition 3. Under Assumption 11, the optimal solution of (4.49) is attained at an integral solution.

Given its brevity, we provide the proof here.

Proof of Proposition 3. The first step is to validate the concavity of (4.51) through its Hessian. As there are no cross-terms, the Hessian matrix is a diagonal matrix of dimension

$|\boldsymbol{I}|^2 \times |\boldsymbol{I}|^2$, and it has the form

$$\begin{bmatrix} 2g\hat{\gamma}^1_{11} & 0 & 0 & \cdots & 0 \\ 0 & 2g\hat{\gamma}^1_{12} & 0 & \cdots & 0 \\ 0 & 0 & 2g\hat{\gamma}^1_{13} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & \cdots & 0 & 2g\hat{\gamma}^1_{IJ} \end{bmatrix} \tag{4.52}$$

Because the diagonal elements of (4.52) are strictly negative[12], it follows that the Hessian is negative definite. Hence, (4.51) is strictly concave. Second, because the constraint matrix of (4.49) is totally unimodular and the right-hand side vector is integral, it follows that every extreme point solution of the feasible region are integral. Consolidating these results, we argue that the optimal solution of (4.49) is integral. For this, we employ a well-known result regarding the minimization of a concave function over a convex set, i.e.,

Theorem 4.5.1 (Horst and Tuy [2013])**.** *Let $f : D \to \mathbb{R}$ be concave and let $D \subset \mathbb{R}^n$ be nonempty, compact and convex. Then, the global minimum of f over D is attained at an extreme point of D.*

In our setting, the domain D of (4.51) is $[0,1]^{(|\boldsymbol{I}|^2)}$. D is a nonempty, compact and convex subset of $\mathbb{R}^n$. Thus, the optimal solution of (4.49) is a perfect matching of pairs, i.e., 0-1 solution. This manifests itself such that each task has its dedicated worker and each worker has their dedicated task. This concludes the proof. □

4.5.1 Heterogeneous Pairs

By Proposition 3, the optimal solution to (4.49) is attained under 1-1 matching. Let us denote the matched, i.e., "dedicated", pairs by $\mathcal{D}$. Then, the objective function value of the

[12] By definition, $g > 0$ and $\gamma^1_{ij} < 0 \ \forall (i,j)$.

exact performance-aware policy is

$$g \sum_{(i,j)\in\mathcal{D}} \gamma_{ij}^{0} + g^{2} \sum_{(i,j)\in\mathcal{D}} \gamma_{ij}^{1}. \tag{4.53}$$

Then, (4.50) captures the objective function value of the egalitarian policy under the most general regime, i.e., heterogeneous pairs. Let $\mathcal{D}'$ denote the set of pairs that are not assigned, such that $\mathcal{D}\cup\mathcal{D}' = \boldsymbol{I}\times\boldsymbol{J}$.

Corollary 4.5.1.1 (Heterogeneous Pairs)**.** *The gap between the performance-aware and the egalitarian policy is*

$$\begin{aligned}\Delta_{min}^{E} = g &\left[\frac{1}{|\boldsymbol{J}|}\sum_{(i,j)\in\mathcal{D}'}\gamma_{ij}^{0} + \frac{1-|\boldsymbol{J}|}{|\boldsymbol{J}|}\sum_{(i,j)\in\mathcal{D}}\gamma_{ij}^{0}\right] \\ &+ g^{2}\left[\frac{1}{|\boldsymbol{J}|^{2}}\sum_{(i,j)\in\mathcal{D}'}\gamma_{ij}^{1} + \frac{1-|\boldsymbol{J}|^{2}}{|\boldsymbol{J}|^{2}}\sum_{(i,j)\in\mathcal{D}}\gamma_{ij}^{1}\right].\end{aligned} \tag{4.54}$$

The cardinality of $\mathcal{D}$, i.e., $|\mathcal{D}|$, is equal to number of tasks (or workers) and $|\mathcal{D}'| = |\boldsymbol{I}|^{2} - |\boldsymbol{I}| = |\boldsymbol{J}|^{2} - |\boldsymbol{J}|$.

Corollary 4.5.1.1 follows by (4.50), (4.53) and Proposition 3. We emphasize that (4.54) holds for the general case, and it simplifies when certain assumptions are imposed on the primitives. It captures the performance degradation when the organization adopts an egalitarian approach.

4.5.2 Identical Task Familiarity Effect

In the identical task familiarity effect regime, i.e., γ_{ij}^{0} and γ_{j}^{1}, the objective function becomes

$$g\sum_{i\in\boldsymbol{I}}\sum_{j\in\boldsymbol{J}} p_{j|i}\gamma_{ij}^{0} + g^{2}\sum_{j\in\boldsymbol{J}}\gamma_{j}^{1}\sum_{i\in\boldsymbol{I}}(p_{j|i})^{2},$$

and, by Proposition 3, the solution is $g \sum_{(i,j)\in\mathcal{D}} \gamma^0_{ij} + g^2 \sum_{j\in\boldsymbol{J}} \gamma^1_j$, where $\mathcal{D}$ is defined as in the heterogeneous pairs regime. Furthermore, the egalitarian policy (4.50) becomes $Z^E = \frac{g}{|\boldsymbol{J}|} \sum_{i\in\boldsymbol{I}} \sum_{j\in\boldsymbol{J}} \gamma^0_{ij} + \frac{g^2}{|\boldsymbol{J}|} \sum_{j\in\boldsymbol{J}} \gamma^1_j$.

Corollary 4.5.1.2 (Identical Task Familiarity)**.** *The gap between the performance-aware and the egalitarian policy is*

$$\Delta^E_{min} = g\left[\frac{1}{|\boldsymbol{J}|}\sum_{(i,j)\in\mathcal{D}'} \gamma^0_{ij} + \frac{1-|\boldsymbol{J}|}{|\boldsymbol{J}|}\sum_{(i,j)\in\mathcal{D}} \gamma^0_{ij}\right] + g^2\frac{1-|\boldsymbol{J}|}{|\boldsymbol{J}|}\sum_{j\in\boldsymbol{J}} \gamma^1_j,$$

where $\mathcal{D}$ is the set of dedicated pairs under the performance-aware policy and $\mathcal{D}'$ is the set of pairs that are not assigned together.

4.5.3 Identical Pair Familiarity Effect

In the identical pair familiarity effect regime, i.e., γ^0_{ij} and γ^1, the objective function becomes

$$g\sum_{i\in\boldsymbol{I}}\sum_{j\in\boldsymbol{J}} p_{j|i}\gamma^0_{ij} + g^2\gamma^1 \sum_{j\in\boldsymbol{J}}\sum_{i\in\boldsymbol{I}} (p_{j|i})^2,$$

and, by Proposition 3, the solution is $g \sum_{(i,j)\in\mathcal{D}} \gamma^0_{ij} + g^2|\boldsymbol{J}|\gamma^1$, where $\mathcal{D}$ is defined as in the heterogeneous pairs regime. The egalitarian policy in (4.50) becomes $Z^E = \frac{g}{|\boldsymbol{J}|} \sum_{i\in\boldsymbol{I}} \sum_{j\in\boldsymbol{J}} \gamma^0_{ij} + g^2\gamma^1$.

Corollary 4.5.1.3 (Identical Pair Familiarity)**.** *The gap between the performance-aware and the egalitarian policy is*

$$\Delta^E_{min} = g\left[\frac{1}{|\boldsymbol{J}|}\sum_{(i,j)\in\mathcal{D}'} \gamma^0_{ij} + \frac{1-|\boldsymbol{J}|}{|\boldsymbol{J}|}\sum_{(i,j)\in\mathcal{D}} \gamma^0_{ij}\right] + g^2(1-|\boldsymbol{J}|)\gamma^1,$$

where $\mathcal{D}$ is the set of dedicated pairs under the performance-aware policy and $\mathcal{D}'$ is the set *of pairs that are not assigned together.*

The form of the final two cases allows making stronger statements regarding the bounds on the equitable policy. In particular, we find that the egalitarian policy is the objective-maximizing policy in the final two cases.

4.5.4 Identical Tasks

In the identical tasks regime, i.e., γ_j^0 and γ_j^1, the objective function becomes

$$g \sum_{j \in \boldsymbol{J}} \gamma_j^0 \sum_{i \in \boldsymbol{I}} p_{j|i} + g^2 \sum_{j \in \boldsymbol{j}} \gamma_j^1 \sum_{i \in \boldsymbol{I}} (p_{j|i})^2. \tag{4.55}$$

By Proposition 3, it follows that (4.55) becomes $g \sum_{j \in \boldsymbol{J}} \gamma_j^0 + g^2 \sum_{j \in \boldsymbol{J}} \gamma_j^1$. We note that it does not matter which pairs are assigned, so long as they are dedicated to each other. The egalitarian policy (4.50) becomes $Z^E = g \sum_{j \in \boldsymbol{J}} \gamma_j^0 + \frac{g^2}{|\boldsymbol{J}|} \sum_{j \in \boldsymbol{J}} \gamma_j^1$.

Proposition 4. In the identical tasks regime, Z^*_{max} is attained under the egalitarian policy, i.e., $Z^E = Z^*_{max}$ and $\Delta_E^{max} = 0$.

The proof is deferred to Section 4.7.2.

Corollary 4.5.1.4 (Identical Tasks)**.** *The gap between the performance-aware and the egalitarian policy is*

$$\Delta^E_{min} = g^2 \frac{1 - |\boldsymbol{J}|}{|\boldsymbol{J}|} \sum_{j \in \boldsymbol{J}} \gamma_j^1.$$

By Proposition 4, $\Delta_E^{max} = 0$; thus, we have $\Delta^{max}_{min} = \Delta^E_{min}$.

4.5.5 Identical Pairs

In the identical pairs regime, i.e., γ^0 and γ^1, the objective function becomes

$$g\gamma^0 \sum_{i \in \boldsymbol{I}} \sum_{j \in \boldsymbol{J}} p_{j|i} + g^2\gamma^1 \sum_{i \in \boldsymbol{I}} \sum_{j \in \boldsymbol{J}} (p_{j|i})^2,$$

and, by Proposition 3, the solution is $g|\boldsymbol{J}|\gamma^0 + g^2|\boldsymbol{J}|\gamma^1$. Again, it does not matter which pairs are assigned so long as they are always assigned together. The egalitarian policy (4.50) becomes $Z^E = g|\boldsymbol{J}|\gamma^0 + g^2\gamma^1$.

Corollary 4.5.1.5 (Identical Pairs)**.** *The gap between the performance-aware and the egalitarian policy is*

$$\Delta^E_{min} = g^2(1 - |\boldsymbol{J}|)\gamma^1. \tag{4.56}$$

By Proposition 4, $\Delta^{max}_E = 0$*; thus, we have* $\Delta^{max}_{min} = \Delta^E_{min}$.

We emphasize that both cases of Sections 4.5.4 and 4.5.5 yield a true bound, because the egalitarian policy is the worst-case policy, thus Δ^E_{min} constitutes the cost of preserving performance-privacy, i.e., cost of equity.

In Table 4.3, we extend the example of Table 4.2 to the performance-aware policy under Assumption 11.

We summarize Δ^E_{min} in Table 4.4, under Assumption 11. The egalitarian policy drives each pair to accrue the same level of familiarity. On the other hand, the performance-aware policy takes advantage of the increasing returns to specialization. We reiterate that $Z^E = Z^*_{max}$ in the bottom two rows of Table 4.4. In these two cases, Z^E scales down the benefit of familiarity, and Δ^E_{min} is equal to the difference in the familiarity effect terms only. Δ^E_{min} increases in arrival probability g and system size, i.e., $|\boldsymbol{I}| = |\boldsymbol{J}|$. Among all cases in Table 4.4, except for heterogeneous pairs, higher $|\gamma^1|$ (or $|\gamma^1_j|\ \forall j$) leads to a larger Δ^E_{min}.

Table 4.3: Sample schedule under Assumption 11, which is driven by the performance-aware policy. Suppose that $g = 0.6$. Tasks are not rotated among workers; each worker-task pair gains experience exclusively with each other.

$n = 1$	$n = 2$	$n = 3$	$n = 4$	$n = 5$
task 1–worker 1 task 2–worker 2 task 3–worker 3 task 4–worker 4 task 5–worker 5		task 1–worker 1 task 2–worker 2 task 3–worker 3 task 4–worker 4 task 5–worker 5		task 1–worker 1 task 2–worker 2 task 3–worker 3 task 4–worker 4 task 5–worker 5
$n = 6$	$n = 7$	$n = 8$		
	task 1–worker 1 task 2–worker 2 task 3–worker 3 task 4–worker 4 task 5–worker 5	task 1–worker 1 task 2–worker 2 task 3–worker 3 task 4–worker 4 task 5–worker 5		task 1–worker1 task 2–worker2 task 3–worker3 task 4–worker 4 task 5–worker 5

Table 4.4: The steady-state discrepancy Δ^E_{min} between the performance-aware policy and the egalitarian policy, from general to specific.

Heterogeneous pairs	$g\left[\frac{1}{\lvert\boldsymbol{J}\rvert}\sum_{(i,j)\in\mathcal{D}'}\gamma^0_{ij} + \frac{1-\lvert\boldsymbol{J}\rvert}{\lvert\boldsymbol{J}\rvert}\sum_{(i,j)\in\mathcal{D}}\gamma^0_{ij}\right] + g^2\left[\frac{1}{\lvert\boldsymbol{J}\rvert^2}\sum_{(i,j)\in\mathcal{D}'}\gamma^1_{ij} + \frac{1-\lvert\boldsymbol{J}\rvert^2}{\lvert\boldsymbol{J}\rvert^2}\sum_{(i,j)\in\mathcal{D}}\gamma^1_{ij}\right]$
Identical task familiarity	$g\left[\frac{1}{\lvert\boldsymbol{J}\rvert}\sum_{(i,j)\in\mathcal{D}'}\gamma^0_{ij} + \frac{1-\lvert\boldsymbol{J}\rvert}{\lvert\boldsymbol{J}\rvert}\sum_{(i,j)\in\mathcal{D}}\gamma^0_{ij}\right] + g^2\frac{1-\lvert\boldsymbol{J}\rvert}{\lvert\boldsymbol{J}\rvert}\sum_{j\in\boldsymbol{J}}\gamma^1_j$
Identical pair familiarity	$g\left[\frac{1}{\lvert\boldsymbol{J}\rvert}\sum_{(i,j)\in\mathcal{D}'}\gamma^0_{ij} + \frac{1-\lvert\boldsymbol{J}\rvert}{\lvert\boldsymbol{J}\rvert}\sum_{(i,j)\in\mathcal{D}}\gamma^0_{ij}\right] + g^2(1-\lvert\boldsymbol{J}\rvert)\gamma^1$
Identical tasks	$g^2\frac{1-\lvert\boldsymbol{J}\rvert}{\lvert\boldsymbol{J}\rvert}\sum_{j\in\boldsymbol{J}}\gamma^1_j$
Identical pairs	$g^2(1-\lvert\boldsymbol{J}\rvert)\gamma^1$

When we allow base performances to vary across pairs, Δ^E_{min} is amplified in γ^0_{ij} of non-dedicated pairs (in lower γ^0_{ij} of dedicated pairs). It follows that Δ^E_{min} shrinks in γ^0_{ij} of dedicated pairs (in lower γ^0_{ij} of non-dedicated pairs), assuming $\mathcal{D}$ is unaltered. If $\mathcal{D}$ changes, Δ^E_{min} relies on other primitives as well. Similarly, for heterogeneous pairs, Δ^E_{min} grows in $\lvert\gamma^1_{ij}\rvert$ of dedicated pairs (in lower $\lvert\gamma^1_{ij}\rvert$ of non-dedicated pairs). In addition, it shrinks in $\lvert\gamma^1_{ij}\rvert$ of non-dedicated pairs (in lower $\lvert\gamma^1_{ij}\rvert$ of dedicated pairs), assuming $\mathcal{D}$ remains the same. Again, the change in Δ^E_{min} may be in either direction, depending on the other primitives.

4.6 Concluding Remarks and Discussion

The novel [ASAPI] (5-Step) method that we develop serves to respect the performance-privacy of workers, while also dealing with the small sample size issue using a meaningful "universal" familiarity effect coefficient. In this work, we develop a practical method for equitably assigning workers to tasks, i.e., by treating any two workers with the same familiarity with a particular task as interchangeable. We uncover three main insights. First, that the steady-state, equitable policy is fundamentally defective, due to the interchangeability of the workers. It lacks sufficient fidelity to drive the system to the true optimal solution of the steady-state, performance-aware model. Second, under certain assumptions on worker-task arrivals, we find that the steady-state, performance-aware policy is optimized by 1-1 matching. This is an artifact of the increasing benefits to specialization. Third, under certain assumptions on the primitives, we find that the egalitarian policy is the objective-maximizing (worst-case) policy; it lies at the other extreme of 1-1 matching. In this case we can bound the cost of equity.

There are other potential avenues to be investigated. We study a sequential assignment problem, solving myopic assignment problems in each period. A future direction is to model and analyze the dynamic version of the assignment model. In this work, we assumed a linear learning curve. Further work can study more complex forms of learning and experience evolution. We assessed the egalitarian policy; alternative policies can be characterized and compared against other policies. One can study the performance outcomes when workers and tasks are allowed to arrive with different marginal probabilities. We assume the existence of rich historical data at beginning of the horizon, i.e., starting in period 0, so that the assignments are sensible from the first period onwards. Assuming that the analyst starts running the PTO loop by setting $F_{ij}(0)$ $\forall(i,j)$ equal to feasible Π_{ij} $\forall(i,j)$ solutions from the steady-state model, there is an interesting question of whether the PTO loop would keep the initial Π_{ij} solution or deviate away from it.

Although, under our assumptions, being equitable results in performance degradation, in reality the organizational benefits of being equitable may outweigh this loss. For instance, in the operating room setting, inequitable treatment of the nursing staff may result in strikes or the quitting of staff, and thus, the operations may come to a halt. To avoid such unwanted situations, managers may prefer to lower performance to achieve the benefits.

4.7 Appendix

4.7.1 Derivation of the Estimators in Section 4.3

Estimators for Pair-Level Data

Mean time. An estimator of mean completion time is the empirical average, i.e.,

$$\hat{\mu}(T_{ij}(n)) := \frac{\sum_{\{n' \leq n : X_{ij}(n')=1\}} T_{ij}(n')}{N_{ij}(n)}.$$

Standard deviation of time. An unbiased estimator of standard deviation of task time is

$$\hat{\sigma}(T_{ij}(n)) := \left(\frac{1}{N_{ij}(n)-1} \sum_{\{n' \leq n : X_{ij}(n')=1\}} \left(T_{ij}(n') - \hat{\mu}(T_{ij}(n))\right)^2 \right)^{1/2}.$$

Mean familiarity. An estimator of mean familiarity is the empirical average, i.e.,

$$\hat{\mu}(F_{ij}(n)) := \frac{\sum_{\{n' \leq n : X_{ij}(n')=1\}} F_{ij}(n')}{N_{ij}(n)}.$$

Standard deviation of familiarity. An unbiased estimator of standard deviation of task familiarity is

$$\hat{\sigma}(F_{ij}(n)) \coloneqq \left(\frac{1}{N_{ij}(n)-1} \sum_{\{n' \le n : X_{ij}(n')=1\}} \left(F_{ij}(n') - \hat{\mu}(F_{ij}(n)) \right)^2 \right)^{1/2}.$$

Estimators for Task-Level Data

Mean time.

$$\hat{\mu}(T_i(n)) \coloneqq \frac{\sum\limits_{j \in \mathcal{J}(n)} \sum\limits_{\{n' \le n : X_{ij}(n')=1\}} T_{ij}(n')}{N_i(n)}.$$

Standard deviation of time.

$$\hat{\sigma}(T_i(n)) \coloneqq \left(\frac{1}{N_i(n)-1} \sum_{j \in \mathcal{J}(n)} \sum_{\substack{\{n' \le n : \\ X_{ij}(n')=1\}}} \left(T_{ij}(n') - \hat{\mu}(T_i(n)) \right)^2 \right)^{1/2}.$$

Mean familiarity.

$$\hat{\mu}(F_i(n)) \coloneqq \frac{\sum\limits_{j \in \mathcal{J}(n)} \sum\limits_{\{n' \le n : X_{ij}(n')=1\}} F_{ij}(n')}{N_i(n)}.$$

Standard deviation of familiarity.

$$\hat{\sigma}(F_i(n)) \coloneqq \left(\frac{1}{N_i(n)-1} \sum_{j \in \mathcal{J}(n)} \sum_{\substack{\{n' \le n : \\ X_{ij}(n')=1\}}} \left(F_{ij}(n') - \hat{\mu}(F_i(n)) \right)^2 \right)^{1/2}.$$

We can relate the estimators in $\forall i \in \mathcal{I}(n)$,

$$\hat{\mu}(T_i(n)) = \sum_{j \in \mathcal{J}(n)} \frac{N_{ij}(n)}{\sum\limits_{j' \in \mathcal{J}(n)} N_{ij'}(n)} \hat{\mu}(T_{ij}(n))$$

$$\hat{\sigma}^2(T_i(n)) = \frac{\sum\limits_{j \in \mathcal{J}(n)} (N_{ij}(n)-1)\hat{\sigma}^2(T_{ij}(n)) + N_{ij}(n)(\hat{\mu}(T_{ij}(n)) - \hat{\mu}(T_i(n)))^2}{\sum\limits_{j \in \mathcal{J}(n)} N_{ij}(n) - 1}$$

$$\hat{\mu}(F_i(n)) = \sum_{j \in \mathcal{J}(n)} \frac{N_{ij}(n)}{\sum\limits_{j' \in \mathcal{J}(n)} N_{ij'}(n)} \hat{\mu}(F_{ij}(n))$$

$$\hat{\sigma}^2(F_i(n)) = \frac{\sum\limits_{j \in \mathcal{J}(n)} (N_{ij}(n)-1)\hat{\sigma}^2(F_{ij}(n)) + N_{ij}(n)(\hat{\mu}(F_{ij}(n)) - \hat{\mu}(F_i(n)))^2}{\sum\limits_{j \in \mathcal{J}(n)} N_{ij}(n) - 1}.$$

4.7.2 Proofs of Lemmas and Proposition 4

Proof of Lemma 4.2.1. When task familiarity evolves according to (4.2), we can write

$$\begin{aligned} F_{ij}(n+1) &= \alpha F_{ij}(n) + (1-\alpha)\mathbf{1}_{\{X_{ij}(n)=1\}} \\ &= \alpha \left[\alpha F_{ij}(n-1) + (1-\alpha)\mathbf{1}_{\{X_{ij}(n-1)=1\}} \right] + (1-\alpha)\mathbf{1}_{\{X_{ij}(n)=1\}} \\ &= \alpha^2 F_{ij}(n-1) + \alpha(1-\alpha)\mathbf{1}_{\{X_{ij}(n-1)=1\}} + (1-\alpha)\mathbf{1}_{\{X_{ij}(n)=1\}}, \end{aligned}$$

for any (i,j). Expanding the recursion in the same fashion into the negative periods, we obtain

$$F_{ij}(n+1) = \lim_{k\to\infty} (\alpha^k F_{ij}(n-k+1)) + (1-\alpha) \sum_{k=0}^{\infty} \alpha^k \mathbf{1}_{\{X_{ij}(n-k)=1\}}, \tag{4.57}$$

where the first term drops since $\alpha \in (0,1)$ and $F_{ij}(n) \in [0,1]$ $\forall n$. By Assumption 7, if $\mathbf{1}_{\{X_{ij}(n)=1\}}$ is stationary, then $\mathbb{E}[\mathbf{1}_{\{X_{ij}(n)=1\}}]$ is constant over n, $\forall n$. Thus, (4.57) becomes

$$\mathbb{E}[F_{ij}(n+1)] = (1-\alpha)\sum_{k=0}^{\infty} \alpha^k \mathbb{E}[\mathbf{1}_{\{X_{ij}(n-k)=1\}}] = \Pi_{ij}(1-\alpha)\sum_{k=0}^{\infty}\alpha^k = \Pi_{ij}.$$

Similarly, by Assumption 7, $\mathbb{E}[F_{ij}(n+1)] = \mathbb{E}[F_{ij}(\infty)] = \Pi_{ij}$. □

Proof of Lemma 4.3.1. By virtue of simple linear regression, we have that $\hat{\gamma}^1_{ij}$ has the following formula,

$$\begin{aligned}
\hat{\gamma}^1_{ij} &= \frac{\sum\limits_{\{n' \le n : X_{ij}(n')=1\}} (F_{ij}(n') - \hat{\mu}(F_{ij}(n)))(T_{ij}(n') - \hat{\mu}(T_{ij}(n)))}{\sum\limits_{\{n' \le n : X_{ij}(n')=1\}} \big(F_{ij}(n') - \hat{\mu}(F_{ij}(n))\big)^2} \\
&:= \frac{\sum\limits_{\{n' \le n : X_{ij}(n')=1\}} \delta_{ij}(n')}{\hat{\sigma}^2(F_{ij}(n))(N_{ij}(n)-1)} \\
&:= \frac{\delta_{ij}}{\hat{\sigma}^2(F_{ij}(n))(N_{ij}(n)-1)},
\end{aligned} \tag{4.58}$$

where $\delta_{ij}(n') := (F_{ij}(n') - \hat{\mu}(F_{ij}(n)))(T_{ij}(n') - \hat{\mu}(T_{ij}(n)))$ and $\delta_{ij} := \sum\limits_{\{n' \le n : X_{ij}(n')=1\}} \delta_{ij}(n')$. Note that the summation operation in (4.58) indicates summing over the observations that belong to the (i,j) pair. Similarly, the task-specific effect of familiarity has the formula,

$$\begin{aligned}
\hat{\gamma}^1_{i} &= \frac{\sum\limits_{j \in \mathcal{J}(n)} \sum\limits_{\{n' \le n : X_{ij}(n')=1\}} (F_{ij}(n') - \hat{\mu}(F_{i}(n)))(T_{ij}(n') - \hat{\mu}(T_{i}(n)))}{\sum\limits_{j \in \mathcal{J}(n)} \sum\limits_{\{n' \le n : X_{ij}(n')=1\}} \big(F_{ij}(n') - \hat{\mu}(F_{i}(n))\big)^2} \\
&:= \frac{\sum\limits_{\{n' \le n : X_{ij}(n')=1\}} \delta_{i}(n')}{\hat{\sigma}^2(F_{i}(n))(N_{i}(n)-1)}
\end{aligned}$$

where $\delta_i(n') := \sum\limits_{j \in \mathcal{J}(n)} (F_{ij}(n') - \hat{\mu}(F_i(n)))(T_{ij}(n') - \hat{\mu}(T_i(n)))$.

Different than (4.58), pairs' observations are summed across workers in (4.59). Then, the numerator of (4.59) $\forall n' \leq n$ is

$$\delta_i(n') = F_{ij}(n')T_{ij}(n') + \hat{\mu}(F_i(n))\hat{\mu}(T_i(n)) - \hat{\mu}(F_i(n))T_{ij}(n') - F_{ij}(n')\hat{\mu}(T_i(n)),$$

and the numerator of (4.58) $\forall n' \leq n$ is

$$\delta_{ij}(n') = F_{ij}(n')T_{ij}(n') + \hat{\mu}(F_{ij}(n))\hat{\mu}(T_{ij}(n)) - \hat{\mu}(F_{ij}(n))T_{ij}(n') - F_{ij}(n')\hat{\mu}(T_{ij}(n)).$$

But then, these two relate to each other in the following way,

$$\begin{aligned}\delta_i(n') = \delta_{ij}(n') & \\ & - \hat{\mu}(F_{ij}(n))\hat{\mu}(T_{ij}(n)) + \hat{\mu}(F_{ij}(n))T_{ij}(n') + F_{ij}(n')\hat{\mu}(T_{ij}(n)) \\ & + \hat{\mu}(F_i(n))\hat{\mu}(T_i(n)) - \hat{\mu}(F_i(n))T_{ij}(n') - F_{ij}(n')\hat{\mu}(T_i(n)).\end{aligned} \tag{4.59}$$

We can simplify (4.59) as

$$\delta_i(n') \coloneqq \delta_{ij}(n') + \omega_{ij}(n'),$$

where $\omega_{ij}(n') \coloneqq -\hat{\mu}(F_{ij}(n))\hat{\mu}(T_{ij}(n)) + \hat{\mu}(F_{ij}(n))T_{ij}(n') + F_{ij}(n')\hat{\mu}(T_{ij}(n)) + \hat{\mu}(F_i(n))\hat{\mu}(T_i(n)) - \hat{\mu}(F_i(n))T_{ij}(n') - F_{ij}(n')\hat{\mu}(T_i(n))$.

Rewriting (4.59) gives

$$\begin{aligned}\hat{\gamma}_i^1 &= \frac{\sum\limits_{j\in\mathcal{J}(n)} \sum\limits_{\{n'\leq n : X_{ij}(n')=1\}} \delta_{ij}(n') + \omega_{ij}(n')}{\hat{\sigma}^2(F_i(n))(N_i(n)-1)} \\ &= \frac{\sum\limits_{j\in\mathcal{J}(n)} \delta_{ij}}{\hat{\sigma}^2(F_i(n))(N_i(n)-1)} + \frac{\sum\limits_{j\in\mathcal{J}(n)} \sum\limits_{\{n'\leq n : X_{ij}(n')=1\}} \omega_{ij}(n')}{\hat{\sigma}^2(F_i(n))(N_i(n)-1)} \\ &= \sum_{j\in\mathcal{J}(n)} \frac{\hat{\sigma}^2(F_{ij}(n))(N_{ij}(n)-1)}{\hat{\sigma}^2(F_i(n))(N_i(n)-1)} \frac{\delta_{ij}}{\hat{\sigma}^2(F_{ij}(n))(N_{ij}(n)-1)} + \frac{\sum\limits_{j\in\mathcal{J}(n)} \sum\limits_{\{n'\leq n : X_{ij}(n')=1\}} \omega_{ij}(n')}{\hat{\sigma}^2(F_i(n))(N_i(n)-1)}\end{aligned}$$

$$= \sum_{j\in\mathcal{J}(n)} \frac{\hat{\sigma}^2(F_{ij}(n))(N_{ij}(n)-1)}{\hat{\sigma}^2(F_i(n))(N_i(n)-1)}\hat{\gamma}^1_{ij} + \frac{\sum\limits_{j\in\mathcal{J}(n)}\sum\limits_{\{n'\le n: X_{ij}(n')=1\}} \omega_{ij}(n')}{\hat{\sigma}^2(F_i(n))(N_i(n)-1)}.$$

To complete the first part of the lemma statement, it remains to show that

$$\sum_{j\in\mathcal{J}(n)}\sum_{\{n'\le n: X_{ij}(n')=1\}} \omega_{ij}(n') = (\hat{\mu}(T_{ij}(n)) - \hat{\mu}(T_i(n)))(N_{ij}(n)-1)\hat{\mu}(F_{ij}(n)).$$

From (4.59) we have

$$\begin{aligned}\sum_{j\in\mathcal{J}(n)}\sum_{\{n'\le n: X_{ij}(n')=1\}} \omega_{ij}(n') \coloneqq \sum_{j\in\mathcal{J}(n)}\sum_{\{n'\le n: X_{ij}(n')=1\}} \\ &- \hat{\mu}(F_{ij}(n))\hat{\mu}(T_{ij}(n)) + \hat{\mu}(F_{ij}(n))T_{ij}(n') + F_{ij}(n')\hat{\mu}(T_{ij}(n)) \\ &+ \hat{\mu}(F_i(n))\hat{\mu}(T_i(n)) - \hat{\mu}(F_i(n))T_{ij}(n') - F_{ij}(n')\hat{\mu}(T_i(n)).\end{aligned}$$

The first and second terms sum to 0, since

$$\begin{aligned}&\sum_{j\in\mathcal{J}(n)}\sum_{\{n'\le n: X_{ij}(n')=1\}} -\hat{\mu}(F_{ij}(n))\hat{\mu}(T_{ij}(n)) + \hat{\mu}(F_{ij}(n))T_{ij}(n') \\ &= \sum_{j\in\mathcal{J}(n)} \hat{\mu}(F_{ij}(n)) \sum_{\{n'\le n: X_{ij}(n')=1\}} (T_{ij}(n') - \hat{\mu}(T_{ij}(n))) = 0.\end{aligned}$$

The fourth and fifth terms also vanish, i.e.,

$$\begin{aligned}&\sum_{j\in\mathcal{J}(n)}\sum_{\{n'\le n: X_{ij}(n')=1\}} \hat{\mu}(F_i(n))\hat{\mu}(T_i(n)) - \hat{\mu}(F_i(n))T_{ij}(n') \\ &= (N_i(n)-1)\hat{\mu}(F_i(n))\hat{\mu}(T_i(n)) - (N_i(n)-1)\hat{\mu}(F_i(n))\hat{\mu}(T_i(n)) = 0.\end{aligned}$$

The only terms that remain are the third and sixth,

$$\sum_{j \in \mathcal{J}(n)} \sum_{\{n' \leq n : X_{ij}(n')=1\}} F_{ij}(n')\hat{\mu}(T_{ij}(n)) - F_{ij}(n')\hat{\mu}(T_i(n)),$$

which can be written as

$$\sum_{j \in \mathcal{J}(n)} \sum_{\{n' \leq n : X_{ij}(n')=1\}} \omega_{ij}(n') = (\hat{\mu}(T_{ij}(n)) - \hat{\mu}(T_i(n)))(N_{ij}(n) - 1)\hat{\mu}(F_{ij}(n)).$$

This completes the proof of the first statement of Lemma 4.3.1. To complete the proof, let us first define the following estimators:

Mean time.

$$\hat{\mu}(Z_i^T(n)) := \frac{\sum\limits_{j \in \mathcal{J}(n)} \sum\limits_{\{n' \leq n : X_{ij}(n')=1\}} Z_{ij}^T(n')}{N_i(n)},$$

where $Z_{ij}^T(n')$ was defined in (4.8).

Mean familiarity.

$$\hat{\mu}(Z_i^F(n)) := \frac{\sum\limits_{j \in \mathcal{J}(n)} \sum\limits_{\{n' \leq n : X_{ij}(n')=1\}} Z_{ij}^F(n')}{N_i(n)},$$

where $Z_{ij}^T(n')$ was defined in (4.7).

Standard deviation of familiarity.

$$\hat{\sigma}(Z_i^F(n)) := \left(\frac{1}{N_i(n) - 1} \sum_{j \in \mathcal{J}(n)} \sum_{\substack{\{n' \leq n : \\ X_{ij}(n')=1\}}} \left(Z_{ij}^F(n') - \hat{\mu}(Z_i^F(n)) \right)^2 \right)^{1/2}.$$

Mean time.

$$\hat{\mu}(Z^T(n)) \coloneqq \frac{\sum\limits_{i\in\mathcal{I}(n)} \sum\limits_{j\in\mathcal{J}(n)} \sum\limits_{\{n'\le n: X_{ij}(n')=1\}} Z^T_{ij}(n')}{N(n)}.$$

Mean familiarity.

$$\hat{\mu}(Z^F(n)) \coloneqq \frac{\sum\limits_{i\in\mathcal{I}(n)} \sum\limits_{j\in\mathcal{J}(n)} \sum\limits_{\{n'\le n: X_{ij}(n')=1\}} Z^F_{ij}(n')}{N(n)}.$$

Standard deviation of familiarity.

$$\hat{\sigma}(Z^F(n)) \coloneqq \left(\frac{1}{N(n)-1} \sum_{i\in\mathcal{I}(n)} \sum_{j\in\mathcal{J}(n)} \sum_{\substack{\{n'\le n:\\ X_{ij}(n')=1\}}} \Big(Z^F_{ij}(n') - \hat{\mu}(Z^F(n)) \Big)^2 \right)^{1/2}.$$

So far, we have established the link between $\hat{\gamma}^1_{ij}$ and $\hat{\gamma}^1_i$. Analogous to (4.15), the relation between $\hat{\beta}^1_i$ and $\hat{\beta}^1$ is

$$\begin{aligned} \hat{\beta}^1 = \sum_{i\in\mathcal{I}(n)} & \frac{\hat{\sigma}^2(Z^F_i(n))(N_i(n)-1)}{\hat{\sigma}^2(Z^F(n))(N(n)-1)} \hat{\beta}^1_i \\ & + \frac{(\hat{\mu}(Z^T_i(n)) - \hat{\mu}(Z^T(n)))(N_i(n)-1)\hat{\mu}(Z^F_i(n))}{\hat{\sigma}^2(Z^F(n))(N(n)-1)}. \end{aligned} \tag{4.60}$$

To see this, observe that $\hat{\mu}(Z^F_i(n)) = 0$ and $\hat{\sigma}(Z^F_i(n)) = 1$, by construction. Finally,

$$\begin{aligned} \hat{\mu}(Z^F(n)) &= \sum_{i\in\mathcal{I}(n)} \frac{N_i(n)}{\sum\limits_{i'\in\mathcal{I}(n)} N_{i'}(n)} \hat{\mu}(Z^F_i(n)) \\ \hat{\sigma}^2(Z^F(n)) &= \frac{\sum\limits_{i\in\mathcal{I}(n)} (N_i(n)-1)\hat{\sigma}^2(Z^F_i(n)) + N_i(n)(\hat{\mu}(Z^F_i(n)) - \hat{\mu}(Z^F(n)))^2}{\sum\limits_{i\in\mathcal{I}(n)} N_i(n) - 1}. \end{aligned} \tag{4.61}$$

By the relationship in (4.61), we have $\hat{\mu}(Z^F(n)) = 0$, and given this, $\hat{\sigma}(Z^F(n)) = 1$. □

Proof of Proposition 4. The program that we aim to solve is

$$\begin{aligned} Z^*_{max} = \quad & \underset{p_{j|i}}{\text{maximize}} && g \sum_{j \in \boldsymbol{J}} \gamma^0_j + g^2 \sum_{j \in \boldsymbol{J}} \gamma^1_j \sum_{i \in \boldsymbol{I}} (p_{j|i})^2 \\ & \text{subject to} && \sum_{j \in \boldsymbol{J}} p_{j|i} = 1, \quad \forall i : i \in \boldsymbol{I}, \\ & && \sum_{i \in \boldsymbol{I}} p_{j|i} = 1, \quad \forall j : j \in \boldsymbol{J}, \\ & && p_{j|i} \geq 0, \quad \forall (i,j) : (i,j) \in \boldsymbol{I} \times \boldsymbol{J}. \end{aligned}$$

We employ the method of Lagrange multipliers. First, we write the Lagrangian function, i.e.,

$$\begin{aligned} \mathscr{L}(\boldsymbol{p}, \hat{\boldsymbol{\lambda}}, \bar{\boldsymbol{\lambda}}) = & -\left[g \sum_{j \in \boldsymbol{J}} \gamma^0_j + g^2 \sum_{j \in \boldsymbol{J}} \gamma^1_j \sum_{i \in \boldsymbol{I}} (p_{j|i})^2 \right] \\ & + \sum_{j \in \boldsymbol{J}} \hat{\lambda}_j \left[1 - \sum_{i \in \boldsymbol{I}} p_{j|i} \right] + \sum_{i \in \boldsymbol{I}} \bar{\lambda}_i \left[1 - \sum_{j \in \boldsymbol{J}} p_{j|i} \right]. \end{aligned}$$

Then, we solve for the partial derivative with respect to $p_{j|i}$ $\forall (i,j) \in \boldsymbol{I} \times \boldsymbol{J}$,

$$\nabla_{p_{j|i}} \mathscr{L}(\boldsymbol{p}, \hat{\boldsymbol{\lambda}}, \bar{\boldsymbol{\lambda}}) = 0 \implies -2g^2 \gamma^1_j p_{j|i} - \hat{\lambda}_j - \bar{\lambda}_i = 0. \tag{4.62}$$

Similarly, we solve for the partial derivative with respect to $\hat{\lambda}_j$, $\forall (i,j) \in \boldsymbol{I} \times \boldsymbol{J}$,

$$\nabla_{\hat{\lambda}_j} \mathscr{L}(\boldsymbol{p}, \hat{\boldsymbol{\lambda}}, \bar{\boldsymbol{\lambda}}) = 0 \implies \sum_{i \in \boldsymbol{I}} p_{j|i} = 1. \tag{4.63}$$

Finally, we solve for the partial derivative with respect to $\bar{\lambda}_i$ $\forall(i,j) \in \boldsymbol{I} \times \boldsymbol{J}$,

$$\nabla_{\bar{\lambda}_i} \mathscr{L}(\boldsymbol{p}, \hat{\boldsymbol{\lambda}}, \bar{\boldsymbol{\lambda}}) = 0 \implies \sum_{j \in \boldsymbol{J}} p_{j|i} = 1. \tag{4.64}$$

Altogether, (4.62), (4.63) and (4.64) constitute a system of $|\boldsymbol{I}||\boldsymbol{J}| + |\boldsymbol{J}| + |\boldsymbol{I}|$ equations in the same number of unknowns. We want to show that the maximizer $p^*_{j|i}$ is equal to $\frac{1}{|\boldsymbol{J}|}$ when $|\boldsymbol{I}| = |\boldsymbol{J}|$, i.e., that the egalitarian policy performs the worst. Solving for (4.62), we get

$$-2g^2\gamma_j^1 p_{j|i} = \hat{\lambda}_j + \bar{\lambda}_i \implies p_{j|i} = -\frac{\hat{\lambda}_j + \bar{\lambda}_i}{2g^2\gamma_j^1}. \tag{4.65}$$

Substituting $p_{j|i}$ into (4.63) and solving for $\hat{\lambda}_j$, we get

$$\begin{aligned} -\sum_{i \in \boldsymbol{I}} \frac{\hat{\lambda}_j + \bar{\lambda}_i}{2g^2\gamma_j^1} = 1 &\implies \frac{-1}{2g^2\gamma_j^1}\left[|\boldsymbol{I}|\hat{\lambda}_j + \sum_{i \in \boldsymbol{I}} \bar{\lambda}_i\right] = 1 \\ &\implies |\boldsymbol{I}|\hat{\lambda}_j + \sum_{i \in \boldsymbol{I}} \bar{\lambda}_i = -2g^2\gamma_j^1 \\ &\implies \hat{\lambda}_j = \frac{-2g^2\gamma_j^1}{|\boldsymbol{I}|} - \frac{\sum_{i \in \boldsymbol{I}} \bar{\lambda}_i}{|\boldsymbol{I}|}, \quad \forall j \in \boldsymbol{J}. \end{aligned} \tag{4.66}$$

Then, substituting $p_{j|i}$ into (4.64) and solving for $\bar{\lambda}_i$, we get

$$\begin{aligned} -\sum_{j \in \boldsymbol{J}} \frac{\hat{\lambda}_j + \bar{\lambda}_i}{2g^2\gamma_j^1} = 1 &\implies \frac{-1}{2g^2}\left[\sum_{j \in \boldsymbol{J}} \frac{\hat{\lambda}_j}{\gamma_j^1} + \sum_{j \in \boldsymbol{J}} \frac{\bar{\lambda}_i}{\gamma_j^1}\right] = 1 \\ &\implies \sum_{j \in \boldsymbol{J}} \frac{\hat{\lambda}_j}{\gamma_j^1} + \bar{\lambda}_i \sum_{j \in \boldsymbol{J}} \frac{1}{\gamma_j^1} = -2g^2 \\ &\implies \bar{\lambda}_i = \frac{-2g^2}{\sum_{j \in \boldsymbol{J}} \frac{1}{\gamma_j^1}} - \frac{\sum_{j \in \boldsymbol{J}} \frac{\hat{\lambda}_j}{\gamma_j^1}}{\sum_{j \in \boldsymbol{J}} \frac{1}{\gamma_j^1}} \end{aligned}$$

$$\implies \bar{\lambda}_i = \bar{c}, \quad \forall i \in \boldsymbol{I}, \tag{4.67}$$

where $\bar{c}$ represents a constant, i.e., it is independent of i. Then, it follows from (4.66) and (4.67) $\forall (i,j) \in \boldsymbol{I} \times \boldsymbol{J}$ that (4.65) becomes

$$p_{j|i} = -\frac{\hat{\lambda}_j + \bar{c}}{2g^2\gamma^1} \implies p_{j|i} = p_j.$$

Then, the constraint $\sum\limits_{i \in \boldsymbol{I}} p_{j|i} = 1$ becomes $\sum\limits_{i \in \boldsymbol{I}} p_j = 1$. Thus, $|\boldsymbol{I}|p = 1$ and $p = |\boldsymbol{I}|^{-1}$. In addition, $\sum\limits_{j \in \boldsymbol{J}} p_j = 1$ becomes $\sum\limits_{j \in \boldsymbol{J}} \frac{1}{|\boldsymbol{I}|} = 1$. This constraint is satisfied by Assumption 11, i.e., it holds since $|\boldsymbol{I}| = |\boldsymbol{J}|$. Finally, this solution satisfies the non-negativity constraints of Z^*_{max}. Thus, we conclude that it is a solution to Z^*_{max}, under Assumption 11. □

CHAPTER 5

CONCLUSION AND FUTURE DIRECTIONS

In this thesis, we explored different types of learning, in operational settings. In the second chapter, we studied the Thompson sampling algorithm in the context of a discrete-time MDP with general state-control spaces. In the latter chapters, we focused on statistical learning; in the third chapter we adapted a supervised learning model, i.e., random coefficients, and an unsupervised learning model, i.e., hierarchical clustering, into the novel setting of operating room surgical cases. Finally, the fourth chapter consolidated regression-based learning with a learning-by-doing model.

In the second chapter, we constructed a novel metric of expected regret of a policy, which is suitable to evaluate the performance under any chain structure. Building this new metric hinged on using the discounted-reward criterion, which allowed us to consolidate machinery from adaptive learning with the regret literature. We provided performance guarantees of the policy we evaluate, i.e., Thompson sampling, using the new notion of residual regret, under chain settings where the transition kernel is strictly positive. This assumption is an artifact of focusing on general state-control spaces. An interesting direction of research is to weaken this assumption in order to find how well residual regret behaves under broader chain structures.

In the third chapter, we extended two statical methods to our setting to estimate the mean and variance of surgical cases, which are comprised of one or more surgical procedures. Even though we focus on surgical cases, the methods we adapt and develop can be used in any domain where tasks are a collection of subtasks, such that subtasks are performed sequentially. We replicated the analysis for three distinct procedure coding systems. We found that neither of the statistical methods nor the coding systems is superior to the others; each configuration has at least one service line under which it performs the best. One limitation of the study was that surgeries with more than four procedures were automatically eliminated, as we eliminated infrequently-occurring surgery types for the assessment of the goodness-of-fit. A potential area of extension is to look at a bigger population of data which will allow to retain surgeries with more than four procedures, to see if the results are robust to the case when surgeries are more complicated, i.e., when they involve more procedures.

The latter part of the dissertation (Chapters 3 and 4) originated by the idea of measuring the impact of task familiarity on task (i.e., surgery) completion time. "Surgery time" is the time between the first incision and the closing of the patient. This implies that we studied the impact of worker-task familiarity on the time between the first incision and closing. Even though we restrict our focus to this interval, which is the center of a surgical encounter, the surgical encounter includes other components that surround this interval. The main event preceding the first incision is the patient's entry into the OR. Conversely, the main event that follows the closing of the patient is the exit of the patient. Thus, finding the impact of nurses' familiarity with the surgery on the duration of the entire surgical encounter, including the time before incision and after closing, is highly relevant to the question that we focused on. In addition to the surgical encounter duration, the successful flow of a hospital's operations are dependent on the OR turnover time, which includes the setup and cleanup of the OR's. Similarly, estimating the impact of nurse familiarity with the surgery on the turnover time is a potential area of extension.

In the fourth chapter, we formulated the steady-state version of the daily assignment problem, under certain assumptions on the primitives. The objective is to minimize the total task completion time, across all pairs, in steady state. An alternative, reasonable goal would be to minimize the total task variance, across all pairs. Since longer-duration tasks typically have more uncertainty involved, and thus, have longer variance, a risk-averse decision maker may opt for the variance minimization problem. In the scenario of tasks being rare objects that do not occur frequently in the data, the methodologies we developed in the third chapter will be useful.

REFERENCES

Shipra Agrawal and Navin Goyal. Analysis of thompson sampling for the multi-armed bandit problem. In *Conference on Learning Theory*, pages 39–1, 2012.

Alessandro Arlotto, Stephen E Chick, and Noah Gans. Optimal hiring and retention policies for heterogeneous workers who learn. *Management Science*, 60(1):110–129, 2014.

Emmanouil Avgerinos and Bilal Gokpinar. Team familiarity and productivity in cardiac surgery operations: The effect of dispersion, bottlenecks, and task complexity. *Manufacturing & Service Operations Management*, 19(1):19–35, 2017.

Dragan Banjević and Michael Jong Kim. Thompson sampling for stochastic control: The continuous parameter case. *IEEE Transactions on Automatic Control*, 64(10):4137–4152, 2019.

Rajiv D Banker and Sandra A Slaughter. The moderating effects of structure on volatility and complexity in software enhancement. *Information Systems Research*, 11(3):219–240, 2000.

MA Bezem and Maarten Keijzer. Generalizing hamming distance to finite sets. *Logic Group Preprint Series*, 163, 1996.

Tadeusz Caliński and Jerzy Harabasz. A dendrite method for cluster analysis. *Communications in Statistics-theory and Methods*, 3(1):1–27, 1974.

Hang Cheng, Jeffrey W Clymer, Brian Po-Han Chen, Behnam Sadeghirad, Nicole C Ferko, Chris G Cameron, and Piet Hinoul. Prolonged operative duration is associated with complications: a systematic review and meta-analysis. *Journal of Surgical Research*, 229: 134–144, 2018.

Kai Lai Chung. *A course in probability theory*. Academic press, 2001.

David A Freedman. On the asymptotic behavior of bayes' estimates in the discrete case. *The Annals of Mathematical Statistics*, pages 1386–1403, 1963.

C Gini. Variabilità e mutabilità reprinted in memorie di metodologica statistica ed e pizetti and t salvemini (rome: Libreria eredi virgilio veschi) go to reference in article, 1912.

Paul S Goodman and Dennis P Leyden. Familiarity and group productivity. *Journal of applied psychology*, 76(4):578, 1991.

Aditya Gopalan and Shie Mannor. Thompson sampling for learning parameterized markov decision processes. In *Conference on Learning Theory*, pages 861–898, 2015.

Onésimo Hernández-Lerma. *Adaptive Markov control processes*, volume 79. Springer Science & Business Media, 2012.

Onésimo Hernández-Lerma and Jean B Lasserre. *Discrete-time Markov control processes: basic optimality criteria*, volume 30. Springer Science & Business Media, 2012.

Reiner Horst and Hoang Tuy. *Global optimization: Deterministic approaches.* Springer Science & Business Media, 2013.

Ronald S Jarmin. *Learning by doing and competition in the early rayon industry*, volume 93. Bureau of the Census, 1993.

Ramesh Johari, Vijay Kamble, and Yash Kanoria. Matching while learning. *Operations Research*, 69(2):655–681, 2021.

Cem Kalkanli and Ayfer Ozgur. Asymptotic convergence of thompson sampling. *arXiv e-prints*, pages arXiv–2011, 2020.

Anand Kalvit and Assaf Zeevi. Dynamic learning in large matching markets. *ACM SIGMETRICS Performance Evaluation Review*, 50(2):18–20, 2022.

Michael Kearns and Satinder Singh. Near-optimal reinforcement learning in polynomial time. *Machine learning*, 49:209–232, 2002.

Michael Jong Kim. Thompson sampling for stochastic control: The finite parameter case. *IEEE Transactions on Automatic Control*, 62(12):6415–6422, 2017.

Tze Leung Lai and Herbert Robbins. Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics*, 6(1):4–22, 1985.

Jan Leike, Tor Lattimore, Laurent Orseau, and Marcus Hutter. Thompson sampling is asymptotically optimal in general environments. *arXiv preprint arXiv:1602.07905*, 2016.

Ying Li, Saijuan Zhang, Reginald F Baugh, and Jianhua Z Huang. Predicting surgical case durations using ill-conditioned cpt code matrix. *Iie Transactions*, 42(2):121–135, 2009.

Richard J Light and Barry H Margolin. An analysis of variance for categorical data. *Journal of the American Statistical Association*, 66(335):534–544, 1971.

Glenn Littlepage, William Robison, and Kelly Reddington. Effects of task experience and group experience on group performance, member ability, and recognition of expertise. *Organizational behavior and human decision processes*, 69(2):133–147, 1997.

David A Nembhard. Heuristic approach for assigning workers to tasks based on individual learning rates. *International journal of production research*, 39(9):1955–1968, 2001.

Gary P Pisano, Richard MJ Bohmer, and Amy C Edmondson. Organizational differences in rates of learning: Evidence from the adoption of minimally invasive cardiac surgery. *Management science*, 47(6):752–768, 2001.

Martin L Puterman. *Markov Decision Processes.: Discrete Stochastic Dynamic Programming.* John Wiley & Sons, 2014.

Ray Reagans, Linda Argote, and Daria Brooks. Individual experience and experience working together: Predicting learning rates from knowing who knows what and knowing how to work together. *Management science*, 51(6):869–881, 2005.

Manfred Schäl. Estimation and control in discounted stochastic dynamic programming. *Stochastics: An International Journal of Probability and Stochastic Processes*, 20(1):51–71, 1987.

Steven E Shreve. *Stochastic optimal control: The discrete time case.* Academic Press, 1978.

Bożena Staruch and Bogdan Staruch. Competence-based assignment of tasks to workers in factories with demand-driven manufacturing. *Central European Journal of Operations Research*, 29(2):553–565, 2021.

David P Strum, Jerrold H May, Allan R Sampson, Luis G Vargas, and William E Spangler. Estimating times of surgeries with two component procedures: comparison of the lognormal and normal models. *The Journal of the American Society of Anesthesiologists*, 98(1): 232–240, 2003.

Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction.* MIT press, 2018.

William R Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4):285–294, 1933.

Hunter DD Witmer, Çağla Keçeli, Joshua A Morris-Levenson, Ankit Dhiman, Amber Kratochvil, Jeffrey B Matthews, Dan Adelman, and Kiran K Turaga. Operative team familiarity and specialization at an academic medical center. *Annals of Surgery*, pages 10–1097, 2022.

Jeffrey M Wooldridge. *Econometric analysis of cross section and panel data.* MIT press, 2010.