Summer 2016

Multi-period dynamic technician routing and scheduling problems with experience-based service times and stochastic customers

Xi Chen
University of Iowa

MULTI-PERIOD DYNAMIC TECHNICIAN ROUTING AND SCHEDULING PROBLEMS WITH EXPERIENCE-BASED SERVICE TIMES AND STOCHASTIC CUSTOMERS

by

Xi Chen

A thesis submitted in partial fulfillment of the
requirements for the Doctor of Philosophy
degree in Business Administration
in the Graduate College of
The University of Iowa

August 2016

Thesis Supervisors: Associate Professor Barrett Thomas
Associate Professor Mike Hewitt

Graduate College
The University of Iowa
Iowa City, Iowa

CERTIFICATE OF APPROVAL

PH.D. THESIS

This is to certify that the Ph.D. thesis of

Xi Chen

has been approved by the Examining Committee for the thesis requirement for the Doctor of Philosophy degree in Business Administration at the August 2016 graduation.

Thesis committee: ______________________________
Barrett Thomas, Thesis Supervisor

____
Mike Hewitt, Thesis Supervisor

____
Qihang Lin

____
Jeffrey Ohlmann

____
Gautam Pant

To Xiujie Zhang and Hong Chen

ACKNOWLEDGEMENTS

Many important people in my life have contributions to my thesis, I owe my sincere gratitude to all those people who love and help me during my Ph.D. study and my life. I am also fortunate to have them in my life and enjoy such a wonderful and unforgettable five-year time in Iowa City.

I would like to express my deepest appreciation to my parents Xiujie Zhang and Hong Chen. I am fortunate enough to have their support and love through all my life. Their constant love, support, and strength have enabled the completion of this thesis. I dedicate this thesis to my parents.

My deepest gratitude is to my advisor Dr. Barrett Thomas. From the first year, he guided me to the field of research, starting with countless revisions of an one-paragraph problem description. He always gave me freedom to explore on my won and provided constructive criticism when I make mistakes. He taught me how to think and express my ideas. His support and patience helped survive those tough times and complete this thesis. I will do my best to be an advisor like Barry.

My co-advisor Dr. Mike Hewitt always came up with brilliant ideas when we were stuck in the middle of our research problems. I really appreciate for his insight and direction when I encounter obstacles in my research. I am also grateful to him for the encouragement during my preparation for the qualification exams.

I am indebted to my committee members, Dr. Jeffrey Ohlmann, Dr. Qihang Lin, and Dr. Gautam Pant. I also express my appreciation to all the great faculty members in my department. During my Ph.D. years at the University of Iowa, they have offered support and help, especially when I was in difficulty situations. I would also like to thank my department secretaries Barb Carr and Scott Hansen, and the Ph.D. program coordinator Renea Jay for their constant administrative support and help.

I am grateful to Dr. Jeffrey Ohlmann for his guidance and encouragement on my teaching assistant work. I would like to be an instruction full of responsibility and enthusiasm just like Jeff.

Many friends and classmates helped and supported me my throughout my studies, especially Shu Zhang and Guanglin Xu. I greatly value their friendship and appreciate their help and support.

Finally, I would like to thank my graduate advisors Dr. Jihong Zhang and Dr. Xiaosong Ding at Beijing Foreign Studies University, for encouraging me to pursue a doctoral degree and for their constant support.

ABSTRACT

While home services is a fast growing industry, little attention has been given to the management of its workforce. To maintain growth, a key challenge for home-service companies is managing their expensive and limited labor resources. In particular, the time an employee needs to provide high quality service often depends on his/her experience. Importantly, experience increases over time, thus gradually decreasing the time required to provide service. By accounting for employee experience and the accompanying learning, managers can take advantage of capacity increases that result from experience, improving efficiency and enabling further growth.

We address the technician routing and scheduling problem from three perspectives, which in turn constitute the three parts of my dissertation. First, we introduce a model of technician routing that explicitly models individualized, experience-based learning. We convert the multi-day problem into a series of daily problems and approach the daily decision making in a myopic fashion. The results demonstrate that explicit modeling and the resulting ability to capture changes in productivity over time due to learning lead to significantly better and different solutions than those found when learning and workforce heterogeneity is ignored. We show that these differences result from the levels of specialization that occur in the workforce.

In the second part, we design solution methods that account for the fact that serving today's demand has implications, in terms of learning, for serving tomorrow's demand. We integrate the future information in the decision process to overcome the

drawback of the myopic algorithm. We introduce the multi-period technician scheduling problem with experience-based service times and stochastic customers. Then, we model the problem as a Markov decision process and introduce an approximate dynamic programming-based solution approach. The model can be adapted to handle cases of worker attrition and new task types. Using an extensive computational study, we demonstrate the value of our approach versus a myopic solution approach that views the problem as a single-period problem.

In the final part, we continue exploring the value of integrating future information into the current period decision-making process for the Multi-period Dynamic Technician Scheduling Problems with Experience-based Service Times and Stochastic Customers discussed in Chapter 3. We propose an alternate approximate dynamic programming solution approach with basis function to approximate the value function by taking the advantage of the future information for the whole planning horizon. We turn to an offline simulation procedure to recursively update the coefficient vector of the basis function, which allows fast decision making within the execution phase. Our computational results demonstrate the value of the ADP solution approach with the basis function.

PUBLIC ABSTRACT

While home service is a fast growing industry, little attention has been given to the management of its workforce. To maintain growth, a key challenge for home-service companies is managing their expensive and limited labor resources. In particular, the time an employee needs to provide high quality service often depends on his/her experience. Importantly, experience increases over time, thus gradually decreasing the time required to provide service. By accounting for employee experience and the accompanying learning, managers can take advantage of capacity increases that result from experience, improving efficiency and enabling further growth.

We address the technician routing and scheduling problem from three perspectives, which in turn constitute the three parts of my dissertation. First, we introduce a model of technician routing that explicitly models individualized, experience-based learning. We demonstrate that explicit modeling and the resulting ability to capture changes in productivity over time due to learning lead to significantly better and different solutions. In the second part, we design solution methods that integrates the future information in the decision process to overcome the drawback of the myopic algorithm. The model can be adapted to handle cases of worker attrition and new task types. In the final part, we propose an alternate approximate dynamic programming solution approach with basis function to approximate the value function by taking the advantage of the future information for the whole planning horizon. Our computational results further demonstrate the value of future information in the

current period decision making.

TABLE OF CONTENTS

LIST OF TABLES

Table

LIST OF FIGURES

Figure

CHAPTER 1
INTRODUCTION

1.1 Motivation

As the global economy recovers, there is growing pressure in the skilled labour markets. According to the Hays Global Skills Index 2014, a statistics-based study designed to assess the dynamics of skilled labour markets across 31 countries, this pressure continues to rise, particularly in economies that are returning to pre-crisis levels such as United States, Germany, and the United Kingdom (Hays plc, 2014). Particularly, home services is one of the fastest growing industries in the US. For example, revenue from heating, ventilation and air conditioning service is expected to rise at an average annual rate of 5.9% between 2012 and 2017, reaching $2.5 billion by 2017 (Panteva, 2012). Home services companies are facing the same skilled labour shortage. To maintain growth, a key challenge for companies is managing their expensive and limited labor resources. In particular, the time an employee needs to provide high quality service often depends on his/her experience. Importantly, experience increases over time, thus gradually decreasing the time required to provide service. By accounting for employee experience and the accompanying learning, managers can take advantage of capacity increases that result from experience, improving efficiency and enabling further growth. Matching the right employee with the right job can not only help a company meet its current needs, but also build capacity for meeting future demand and potentially demand growth.

In this work, we explore the issue of how companies can use immediate employee job assignments to meet current demand and build capacity for the future. We focus on service workers, particularly service technicians. The problems discussed in this paper are variants of the technician and task scheduling problem (TTSP). In the TTSP, a set of technicians serves a set of customer requests. Customers are associated with certain tasks and different tasks have different skills associated with them. In our version of the problem, technicians have different service times depending on their experience in performing a task as well as each technician's ability to transform that experience into improved productivity. We measure experience in the number of times that the technician has performed the task.

The fact that technician productivity is linked to experience suggests that what could be modeled as a single-period problem (i.e. focusing solely on making assignments to serve the current day's tasks) should instead be modeled as a multi-period problem. As such, we consider the multi-period technician scheduling (and routing) problems that account for the fact that productivity increases (or service time decreases) as technicians gain experience. These increases in productivity are often referred to as "learning." We assume that the time that it takes a technician to complete a task depends on the technician's experience in the skill associated with the task and how quickly the technician learns. How quickly a technician learns is known as the technician's learning rate. We assume that we have a set of heterogeneous technicians whose learning rates and initial experience are known. The service time depends on the amount of experience the worker has with the skill required by the task.

Chapter 2 introduces a model of technician routing that explicitly models individualized, experience-based learning. In this problem, each day, the technicians serve the day's known demand, starting and ending each day at the depot. We model the problem as a Markov decision process and introduce a daily myopic solution approach that decomposes a multi-period problem into a series daily problems and then solves the daily problems using the record-to-record (RTR) heuristics. We seek to minimize the sum of each day's makespan over a finite horizon, accounting for both travel and service times on each individual day. The objective accounts for the desire to increase the capacity available to grow the business. The results demonstrate that explicitly modeling and the resulting ability to capture changes in productivity over time due to learning lead to significantly better and different solutions than those found when learning and workforce heterogeneity is ignored. We show that these differences result from the levels of specialization that occur in the workforce.

As the first to explicitly model the impact of experience-based learning on technician productivity, this work makes several contributions to the literature. First, we introduce to the literature a Markov decision process model of the problem and introduce a myopic solution approach. In addition, this work presents several important insights. These are:

1. Explicitly modeling workforce heterogeneity and learning offers better solutions in comparison to assuming homogeneous learning curves and/or static productivity.

2. Importantly, modeling workforce heterogeneity and learning captures that fast learners have more capacity that can be used to improve solution quality.
3. Regardless of the learning rate, inexperienced technicians specialize more than more experienced technicians.

Further, we show that, in the presence of workforce heterogeneity and human learning, technician routing solutions trade-off routing and scheduling. We introduce "rules of thumb" that demonstrate which aspect is more important based on the individual characteristics of a technician.

Chapter 3 introduces the multi-period technician scheduling problem with experience-based service times and stochastic customers. In this problem, a manager must assign tasks of different types that are revealed at the start of each day to technicians who must complete the tasks that same day. As a technician gains experience with a type of task, the time that it takes to serve future tasks of that type is reduced (often referred to as experiential learning). We model the problem as a Markov decision process and introduce an approximate dynamic programming-based solution approach. The model can be adapted to handle cases of worker attrition and new task types. The solution approach relies on an approximation of the cost-to-go that uses forecasts of the next day's assignments for each technician and the resulting estimated time it will take to service those assignments given current period decisions. Using an extensive computational study, we demonstrate the value of our approach versus a myopic solution approach that views the problem as a single-period problem.

In this study, we make the following research contributions. First, we present a Markov decision process (MDP) model for the MTSP-ESTSC. We note that this model can be adapted to handle cases of worker attrition and new task types. Second, we introduce a method for approximating the future value of today's workforce assignments, and third, demonstrate how to incorporate this approximation into a tractable model for making daily decisions.

In Chapter 4, we continue exploring the value of integrating future information into the current period decision-making process for the Multi-period Dynamic Technician Scheduling Problems with Experience-based Service Times and Stochastic Customers discussed in Chapter 3. We propose an alternate approximate dynamic programming solution approach with basis function to approximate the value function by taking the advantage of the future information for the whole planning horizon. At each decision epoch, we choose each technician's aggregate experience levels on all task types as the features in the basis function. Then, we turn to an offline simulation procedure to recursively update the coefficient vector of the basis function, which allows fast decision making within the execution phase.

Our computational results demonstrate the value of the ADP solution approach with the basis function. We benchmark the performance of the ADP solution approach with the basis function against a myopic approach that converts the multi-period problem into a series of single-period problems, ignoring the impact of future information on the decision making in the current period.

The remainder of this work is organized as follows. The remaining part of this chapter reviews the literature on problems related to the technician routing and scheduling and learning. Chapter 2 studies the technician routing problem with experience-based service time, and Chapter 3 discusses the multi-period technician scheduling problem with experience-based service times and stochastic customers. Chapter 4 proposes an alternate approximate dynamic programming solution approach with basis function. Finally, Chapter 5 concludes this work and suggests areas of future research.

1.2 Literature Review

Two major fields of literature are related to the problems studied in this paper: the technician routing and scheduling problem (TRSP) and learning.

1.2.1 Technician Routing and Scheduling

The existing literature contains a variety of technician routing and scheduling problems. The TRSP was first introduced by Dutot et al. (2006) and is based on a problem faced by the telecommunications industry. In the problem as introduced in Dutot et al. (2006), technicians are grouped into teams, and tasks are assigned to teams so that skill requirements and the skill level can be matched. However, neither learning nor the extended horizon over which learning occurs is considered.

In 2007, the French Operations Research Society introduced a challenge (`http://challenge.roadef.org/2007/en/`) based on Dutot's work and offered a real-world data set for technician scheduling. The challenge resulted in a stream of papers. The papers are largely algorithmic, and none of the papers resulting from the challenge consider routing. Hurkens (2009) uses mixed integer programming to construct a day schedule and demonstrates the effectiveness of the linear programming techniques in solving scheduling problems. Fırat and Hurkens (2012) propose a solution methodology that uses a flexible match model for a special multi-skill workforce scheduling problem, in which a set of combined technicians stays together for the duration of a work day. Cordeau et al. (2010) propose a construction heuristic and an adaptive large neighborhood search heuristic for the technician and task scheduling problem arising in a large telecommunications company. The objective is to minimize a weighted combination of makespans of each priority class. Hashimoto et al. (2011) present a variant of the Greedy Randomized Adaptive Search Procedure (GRASP) for solving the technician and interventions scheduling problem for telecommunications. The authors also introduce a lower bound procedure for the problem.

Other literature has considered both the travel and service time aspects of the problem, but again, does not consider learning and a multi-period horizon. Kovacs et al. (2012) define the service technician routing and scheduling problem with and without team building. The objective is to minimize the sum of total routing and outsourcing costs. Tsang and Voudouris (1997) and Pillac et al. (2012a) propose heuristics for related problems. Alsheddy and Tsang (2011) consider a bi-objective optimization problem in which both the technician routing costs and the employees' interests are considered.

Additional papers incorporate dynamic and stochastic service requests. Similar to the work in this paper, Bostel et al. (2008) consider a multi-period planning and routing problem of technicians in the field. However, Bostel et al. (2008) do not consider learning that takes place over time. Also similar to this work, the problem is solved without incorporating information about future information. Other work considers single-day problems. Inspired by British Telecommunications plc, Lesaint et al. (2000) describe a dynamic scheduler based on a combination of heuristic search and constraint-based reasoning for dynamic workforce scheduling problem. Weintraub et al. (2012) address the routing and scheduling of service technicians for energy providers in Chile. Customers service requests are considered to be stochastic and priorities of different tasks are taken into consideration. The objective is to minimize the response time to these requests. Pillac et al. (2012b) study the Dynamic TRSP in which new requests appear over time by proposing a fast reoptimization approach based on a parallel Adaptive Large Neighborhood Search and a Multiple Plan Approach.

Home healthcare scheduling and routing is a special case of the technician routing problem. A key feature of the home healthcare literature is the need to respect patient preferences for particular healthcare workers. These preferences usually result from a patient's prior experience with a particular worker. In a sense, these preferences capture the fact that the home healthcare problems are multi-period, even if they are not explicitly modeled as such. Examples include Bertels and Fahle (2006), Bard et al. (2014), Rasmussen et al. (2012).

The author is aware of only limited work that incorporates learning in a routing context. Zhong et al. (2007) explicitly models driver learning, but in the context of familiarity with a particular geographic area and the customers found in that area.

Unlike this work, the heterogeneity of tasks at the individual customers is ignored. While learning is not explicitly modeled, work on consistency in multi-day vehicle routing often cites advantages of repeat visits to the same region or same customers. For example, Smilowitz et al. (2012) suggests that repeated visits may allow a delivery driver to “more efficiently serve her customer base.”

1.2.2 Models of Learning

That humans learn as they gain experience, “learning-by-doing,” is a well known phenomenon. The learning effect was first examined on a scientific basis by Wright (1936), who quantified learning curves with the observation that in the aircraft industry the working costs per unit declined with an increasing production output. Subsequent empirical studies confirmed the existence and importance of learning effects (see for example Conway and Schultz (1958), Venezia (1985), Cochran (1960), Moulton et al. (2006)). In 2016, the concept has become mainstream enough that it is now included in textbooks on operations management (Cachon and Terweisch (2016), Meredith and Shafer (2016)).

The mathematical descriptions of learning are often called learning curves. Reviews of the literature on learning curves can be found in Dar-El (2000), Jaber and Sikström (2004), Jaber (2006), and Anzanello and Fogliatto (2011). Because of the availability of distributions from which to generate workforces, in this research, we use the hyperbolic learning model described in Nembhard and Osothsilp (2005). We note that, while we employ the hyperbolic learning model, most learning curves have similar shapes and would support conclusions similar to those discussed in Section 4.4. We note that there also exists an extensive literature focusing on organizational learning. Argote (1999) provides an excellent reference.

Work that explicitly models individual learning and the associated heterogeneity of the workforce demonstrates the value of capturing learning. Buzacott (2002) shows that simply modeling worker heterogeneity without considering learning improves system performance versus assuming uniform workforce productivity in flow-line production. Shafer et al. (2001) extend the analysis of Buzacott (2002) to demonstrate the impact of heterogeneous learning and forgetting curves on system productivity in a assembly-line setting. Chen et al. (to appear) confirms the results of Shafer et al. (2001) for technician routing. In addition to flow lines, assembly lines, and technician routing, the value of modeling learning has also been found in call centers (Gans and Zhou (2002)), departmental assignment (Sayin and Karabati (2007)), machine scheduling (see Biskup (2008) for a review), project selection (Gutjahr et al. (2008), Gutjahr (2009)), and vehicle routing (Zhong et al. (2007)).

One of the challenges of much of the workforce planning literature that models individual learning is that the nonlinearity of the learning curves creates challenges. For this reason, work such as Gans and Zhou (2002) and Fowler et al. (2008), simplify the model of individual learning to avoid the nonlineariaties. Work such as Nembhard and Bentefouet (2012) exploits structural properties of the optimal solution to increase the size of the problem that can be solved. However, such approaches do not generalize. Work such as Corominas et al. (2010), Nembhard and Norman (2007), and Heimerl and Kolisch (2010) are limited to solving small problems. Hewitt et al. (2015) introduce a linear and integer reformulation of the learning curve that takes advantage of the fact that most work is assigned in time intervals. The reformuala-tion allows much larger problems to be solved than had been previously. We take advantage of the reformulation in this work as well. Additional review of workforce planning models that incorporate learning can be found in De Bruecker et al. (2015) and Hewitt et al. (2015).

CHAPTER 2
THE TECHNICIAN ROUTING PROBLEM WITH EXPERIENCE-BASED SERVICE TIME

2.1 Introduction

In this chapter, we consider the Technician Routing and Scheduling Problem (TRSP) over multiple periods or days and account for the fact that productivity increases (or service time decreases) as technicians gain experience. These increases in productivity are often referred to as "learning." We assume that the time that it takes a technician to complete a task depends on the technician's experience in the skill associated with the task and how quickly the technician learns. How quickly a technician learns is known as the technician's learning rate. We assume that we have a set of heterogeneous technicians whose learning rates and initial experience are known. For this problem, the experience of a technician with a skill depends on the number of times the technician has performed the task.

We assume that daily demand is not revealed until the day of service. Each day, the technicians serve the day's known demand, starting and ending each day at the depot. In this work, we seek to minimize the sum of each day's makespan over a finite horizon, accounting for both travel and service times on each individual day. The objective accounts for the desire to increase the capacity available to grow the business. We call our problem variant the technician routing problem with experience-based service times (TRP-EST).

To solve the problem, we implement a rolling-horizon procedure, creating routes for each day's known demand without regard for future demand. In the rolling-horizon framework, the objective becomes simply the minimization of the makespan for a given day. To solve the daily routing problem, we use a variant of the record-to-record travel algorithm (RTR), a heuristic first introduced by Li et al. (2005). At the end of each day, we update each technician's accumulated experience.

As the first to explicitly model the impact of experience-based learning on technician productivity, this work makes several contributions to the literature. First, we introduce to the literature a Markov decision process model of the problem and introduce a myopic solution approach. In addition, this work presents several important insights. These are:

1. Explicitly modeling workforce heterogeneity and learning offers better solutions in comparison to assuming homogeneous learning curves and/or static productivity.

2. Importantly, modeling workforce heterogeneity and learning captures that fast learners have more capacity that can be used to improve solution quality.

3. Regardless of the learning rate, inexperienced technicians specialize more than more experienced technicians.

Further, we show that, in the presence of workforce heterogeneity and human learning, technician routing solutions trade-off routing and scheduling. We introduce "rules of thumb" that demonstrate which aspect is more important based on the individual characteristics of a technician.

2.2 Problem Description, Model, and Solution Approach

In this section, we first present a formal description of the TRSP discussed in this work. We then present a model of the problem and describe our solution approach.

2.2.1 Problem Description

In the TRSP, we assume a horizon of T days, and let the set $\mathcal{T} = \{1, 2, \ldots, T\}$ index the days of the planning horizon. We let $\mathcal{K} = \{1, 2, \ldots, K\}$ be a set of technicians and assume $\mathcal{K}$ is invariant in t. Let $\mathcal{R} = \{1, 2, \ldots, R\}$ be the set of all possible skill types.

Associated with each customer is a task requiring skill r in $\mathcal{R}$, and associated with each technician k in $\mathcal{K}$ is a set of parameters related to the technician's ability to learn that skill. Let d_{r0} be the service time for skill r for any technician with a minimum level of training. Let D_r^k be the steady state service time for technician k performing task requiring skill r. The parameter L_r^k is the individual learning factor for technician k on skill r. These parameters are typically estimated from empirical data.

For each day t in the planning horizon, we capture the experience of technician k at the start of day in the R–dimensional vector $Q_t^k = (q_{1t}^k, q_{2t}^k, \ldots, q_{Rt}^k)$, where the r^{th} entry q_{rt}^k indicates the technician k's experience with the r^{th} skill at the start of day t. Let Q_0^k represent technician k's experience at the beginning of the horizon. Each skill r is associated with a service time d_{rt}^k for day t and technician k, which is negatively related to the technician's skill experience. Assuming the De Jong Model of learning, we compute d_{rt}^k as

$$d_{rt}^k = D_r^k + d_{r0}(q_{rt}^k)^{-L_r^k}.$$

As suggested previously, this learning function accounts for both learning and the fact that service or production times do not go to zero. Thus, the service time is divided into two parts. The first term represents the "incompressible" part of the task. As noted by De Jong (1957), the incompressible time is not related to the technician's experience level. The second term describes the learning process.

On a given day t, we seek to serve a set of customers $\mathcal{C}_t = \{1, 2, \ldots, C_t\}$. We assume that this set of customers becomes known only at the beginning of day t. Each customer requires the completion of a single task requiring skill r in $\mathcal{R}$. Every technician departs from and returns to the depot, denoted by 0 at the beginning of the day and by $C_t + 1$ at the end of the day. An arc (i, j) is associated with each pair of elements in $\mathcal{C}_t \cup \{0, C_t + 1\}$, and each arc (i, j) is associated with a travel time τ_{ij}. We assume that no driver incurs service time at the depot. Our problem objective is to minimize the expected sum of latest task completions time for each day.

2.2.2 Markov Decision Process Model

The TRSP described above is a sequential and stochastic decision making problem. A natural modeling framework for such a problem is a Markov decision process (MDP). In this section, we present a formal MDP model for the problem.

- States

In this problem, decisions are made at the beginning of each day during the planning horizon. Let $t = 1, 2, \ldots, T$ be the decision epochs, where day T is the last day in the problem horizon. The state of the system at the beginning of day t captures all the information that is needed to make a routing and scheduling decision. For this problem, the state needs to capture the experience of the technicians at the start of day t as well as the service requests on day t. Let Q_t be the matrix of the technicians' experience on day t with elements q^k_{rt} for all $k \in \mathcal{K}$ and $r \in \mathcal{R}$. To capture the customer requests, we consider a vector of tuples. We represent each tuple in the vector as $(lat_{ct}, long_{ct}, r_{ct})$, where lat_{ct} is the latitude of the c^{th} customer requesting service on day t, $long_{ct}$ the longitude of the c^{th} customer requesting service on day t, and r_{ct} is the skill required to perform the task at the c^{th} customer requesting service on day t. We denote the vector of tuples representing the requests on day t as W_t. Then, the state of the system on day t is given by $s_t = (Q_t, W_t)$.

- Actions

  Given state s_t, an action is a set of routes that serves the day t requests. For convenience we convert the vector of tuples W_t into the set of customers $\mathcal{C}_t$. We formally represent the action for day t as $a_t(s_t) = \{x^k_{ijt} : i \in \mathcal{C}_t \cup \{0\}, j \in \mathcal{C}_t \cup \{C_t + 1\}, k \in \mathcal{K}\}$ where binary variables x^k_{ijt} equal to 1 if arc (i, j) is traversed by technician k in day t. A feasible action satisfies the following constraints:

$$\textstyle\sum_{i\in\mathcal{C}_t\cup\{0\}}\sum_{k\in\mathcal{K}} x_{ijt}^k = 1 \qquad \forall j \in \mathcal{C}_t, \tag{2.1}$$

$$\textstyle\sum_{j\in\mathcal{C}_t\cup\{C_t+1\}} x_{0jt}^k = 1 \qquad \forall k \in \mathcal{K}, \tag{2.2}$$

$$\textstyle\sum_{i\in\mathcal{C}_t\cup\{0\}} x_{i(C_t+1)t}^k = 1 \qquad \forall k \in \mathcal{K}, \tag{2.3}$$

$$\textstyle\sum_{j\in\mathcal{C}_t\cup\{0\}} x_{jit}^k - \sum_{j\in\mathcal{C}_t\cup\{C_t+1\}} x_{ijt}^k = 0 \qquad \forall i \in \mathcal{C}_t, \forall k \in \mathcal{K}, \tag{2.4}$$

$$d_{rt}^k = D_r^k + d_{r0}(q_{rt}^k)^{-L_r^k} \qquad \forall r \in \mathcal{R}, \forall t \in \mathcal{T}, \forall k \in \mathcal{K}, \tag{2.5}$$

$$B_j \geq \textstyle\sum_{k\in\mathcal{K}}(B_i + \sum_{r\in\mathcal{R}} z_{ir} d_{rt}^k + \tau_{ij}) x_{ijt}^k \qquad \forall i \in \mathcal{C}_t \cup \{0\}, \forall j \in \mathcal{C}_t, \tag{2.6}$$

$$x_{ijt}^k \in \{0,1\} \qquad \forall i \in \mathcal{C}_t \cup \{0\}, \forall j \in \mathcal{C}_t \cup \{C_t+1\}, \forall k \in \mathcal{K}, \tag{2.7}$$

$$B_i \geq 0 \qquad \forall i \in \mathcal{C} \cup \{0, C_t+1\}. \tag{2.8}$$

Constraints (2.1) ensure that a customer is assigned to exactly one technician and that it is assigned only once. Constraints (2.2) and (2.3) guarantee that every technician starts and ends a day's working at the depot. Flow constraints (2.4) require the technician to enter and leave a customer if the customer has been assigned to that technician. Constraints (2.5) state the negative relationship between service time and experience level. Constraints (2.6) state precedence relationship between two consecutive customers for every technician. Constraints (2.7) through (2.8) ensure integrality and non-negativity. Let $\mathcal{A}_t(s_t)$ be the set of actions available on day t, the set of actions is the set of sets of routes that serve the day t requests.

- Transition Function

We consider the transition in two parts. The first is a deterministic transition governed by the action selected on day t. We call this new state the post-decision state. Given that the state is currently s_t and that action a_t in $\mathcal{A}_t(s_t)$ is selected, a deterministic transition is made to post-decision state $s_t^a = Q_t^a$ by updating technicians' experience as follows:

$$q^k_{r(t+1)}(s_t, a_t) = q^k_{rt} + \sum_{i \in \mathcal{C}_t} \sum_{j \in \mathcal{C}_t \cup \{C_t+1\}} x^k_{ijt} z_{ir} \quad \forall k \in \mathcal{K}, \forall r \in \mathcal{R}, \quad (2.9)$$

where $z_{ir} = 1$ if customer i requires task r and set $z_{ir} = 0$ otherwise.

On day $t+1$, a transition is made from post-decision state s_t^a to pre-decision state $s_{t+1} = (Q_{t+1}.W_{t+1})$ by observing new service requests W_{t+1} arriving at the beginning of day $t+1$. In this transition, technicians' experience remain unchanged from the post-decision state, and thus $Q_{t+1} = Q_t^a$.

- Contribution function

  At decision epoch t, given state s_t and action $a_t(s_t)$, a transition from pre-decision state s_k to post-decision state s_k^a results in a contribution

$$c(s_t, a_t) = e^t_{max} \qquad \forall a_t \in A_t(s_t), \quad (2.10)$$

  which is the time required to complete the last task on day t.

- Objective function

  The problem objective is then $\min_{\pi \in \Pi} E\left[\sum_{t=1}^{T} c(s_t, a_t^\pi(s_t))\right]$, where π is a policy that determines actions for all days t over the problem horizon T and Π is the set of all policies.

2.2.3 Myopic Solution Approach

To solve MDPs, authors often turn to the well known Bellman equation:

$$V(s_t) = \min_{a_t \in A_t(s_t)} \{c(s_t, a_t) + E\left[V(s_{t+1}) \mid s_t, a_t\right]\}. \tag{2.11}$$

Because of the size of the state space and the challenges associated with even computing the expectation in the Bellman equation, however, we propose a myopic solution approach. In the language of MDPs, we are seeking the optimal myopic policy (Powell, 2011). A myopic policy is constructed by, at each decision epoch or in our case on each day, choosing an action that minimizes the current state costs while ignoring information about the future. This approach is equivalent to using the decision rule $\operatorname{argmin}_{a_t \in A_t(s_t)}\{c(s_t, a_t)\}$. Thus, the problem becomes one of solving a series of daily routing problems.

For our problem, this approach offers several important computational advantages. First, by not considering the cost-to-go in the Bellman equation, we solve a daily routing problem that is a deterministic problem. Second, in the fashion of rolling horizon or rollout methods (for discussion, see (Goodson et al., 2015)), rather than solving for every state as is necessary in traditional backward dynamic programming, we can step forward in time and solve for only the observed demand realizations. A sketch of our solution approach can be found in Algorithm 2.1. In the next section, we describe how we solve the daily routing problem and update technician productivity based on accumulated experience.

Algorithm 2.1 Myopic Solution Approach

for day $t = 1$ TO T **do**

  Observe demand for day t

  Solve daily routing problem, minimizing the completion time of last task

  Update technician productivity based on experience gained on day t

end for

2.3 The Daily Routing Problem

In this section, we present a model and solution approach for the daily routing problem. Throughout, to make explicit the dependence of service times on accumulated experience, where appropriate, we subscript our parameters and variables with the day t.

In addition to the notation introduced in the previous section, we let $r(i) \in \mathcal{R}$ indicate the skill needed to perform the task required by customer $i \in \mathcal{C}_t$. For customers i and j in $\mathcal{C}_t$, our model uses binary variables x^k_{ijt} equal to 1 if arc (i, j) is traversed by technician k. Continuous variable $B_i \geq 0$ is the start time of service at customer i, and e^{max}_t is the completion time of the last task on day t. With this notation, the daily technician routing problem with experience-based service times can be modeled as follows:

$$\min \quad e_t^{max}$$

$$\text{s.t.} \quad \textstyle\sum_{i\in\mathcal{C}_t\cup\{0\}} \sum_{k\in\mathcal{K}} x_{ijt}^k = 1, \quad \forall j \in \mathcal{C}_t, \tag{2.12}$$

$$\textstyle\sum_{j\in\mathcal{C}_t\cup\{C_t+1\}} x_{0jt}^k = 1, \quad \forall k \in \mathcal{K}, \tag{2.13}$$

$$\textstyle\sum_{i\in\mathcal{C}_t\cup\{0\}} x_{i(C_t+1)t}^k = 1, \quad \forall k \in \mathcal{K}, \tag{2.14}$$

$$\textstyle\sum_{j\in\mathcal{C}_t\cup\{0\}} x_{jit}^k - \sum_{j\in\mathcal{C}_t\cup\{C_t+1\}} x_{ijt}^k = 0, \quad \forall i \in \mathcal{C}_t, \forall k \in \mathcal{K}, \tag{2.15}$$

$$B_j \geq \textstyle\sum_{k\in\mathcal{K}}(B_i + d_{r(i)t}^k + \tau_{ij})x_{ijt}^k, \quad \forall i \in \mathcal{C}_t \cup \{0\}, \forall j \in \mathcal{C}_t, \tag{2.16}$$

$$e_t^{max} \geq B_i + \textstyle\sum_{j\in\mathcal{C}_t\cup\{C_t+1\}} \sum_{k\in\mathcal{K}} \sum_{r\in\mathcal{R}} x_{ijt}^k(d_{r(i)t}^k + \tau_{ij}), \quad \forall i \in \mathcal{C}_t, \tag{2.17}$$

$$x_{ijt}^k \in \{0,1\}, \quad \forall i \in \mathcal{C}_t \cup \{0\}, \forall j \in \mathcal{C}_t \cup \{C_t+1\}, \forall k \in \mathcal{K}, \tag{2.18}$$

$$B_i \geq 0, \quad \forall i \in \mathcal{C} \cup \{0, C_t+1\}. \tag{2.19}$$

The objective of the model is to minimize the completion time of the last task. Constraints (2.12) ensure that a customer is assigned to exactly one technician and that it is assigned only once. Constraints (2.13) and (2.14) guarantee that every technician starts and ends a day's working at the depot. Flow constraints (2.15) require the technician to enter and leave a customer if the customer has been assigned to that technician. Constraints (2.16) state precedence relationship between two consecutive customers for every technician. Constraints (2.17) impose that e_t^{max} is no less than the completion time of any task. Constraints (2.18) through (2.19) ensure integrality and non-negativity.

To solve the daily routing problem, we modify the record-to-record travel (RTR) algorithm presented in Li et al. (2005). We chose the RTR algorithm for our problem for two reasons. First, the algorithm is known to return high-quality solutions for vehicle routing problems and does so in relatively short run times. Second, code for the algorithm is open source and available in the COIN-OR repository (`https://projects.coin-or.org/VRPH`). This open-source code shortens development times.

In the algorithm, the record represents the best solution found so far. The RTR is a multi-phase local search algorithm. The algorithm alternates between two phases. The first phase is a diversification phase in which an uphill criterion is applied. That is, non-improving neighboring solutions are accepted if they are within a particular threshold of the record. The diversification phase ends after a fixed number of iterations. In the second phase, the improvement phase, only improving moves are accepted. The improvement phase continues until a local minimum is found.

To overcome convergence to local minima, a perturbation step follows the improvement phase (see Li et al. (2005)). The sequence of the diversification, improvement, and perturbation phases is run for a minimum number of iterations before the algorithm terminates. The algorithm terminates when no solution can be found that, after a fixed number of attempts, improves the best known solution by more than ε. In our implementation, the threshold is $(1 + 0.01)$ times the value of the record or best known solution, the number of iterations in the diversification phase is 30, ε is 0.00001, and the number of attempts is 5. These values are suggested by Li et al. (2005). A detailed algorithmic description can also be found in Li et al. (2005).

Because the daily routing problem must be solved multiple times over the horizon $\mathcal{T}$, we update each technician's experience and service times based on the assignments of the previous day. For each $t \in \mathcal{T}, t \neq T$, for each $k \in \mathcal{K}$, we have:

$$q^k_{rt} = q^k_{r(t-1)} + \sum_{i \in \mathcal{C}_t : r(i)=r} \sum_{j \in \mathcal{C}_t \cup \{C_t+1\}} x^k_{ij(t-1)} \forall k \in \mathcal{K}, \forall r \in \mathcal{R} \tag{2.20}$$

and

$$d^k_{rt} = D^k_r + d_{r0}(q^k_{rt})^{-L^k_r} \quad \forall r \in \mathcal{R}, \forall k \in \mathcal{K}. \tag{2.21}$$

Equation (2.20) updates a technician's experience for a particular skill. Equation (2.21) computes the service times based on previous experience.

Our implementation of the RTR is a modification of existing RTR code, called VRPH, available in the COIN-OR repository (`https://projects.coin-or.org/VRPH`) and described in Groër et al. (2010). The code was originally developed to solve the capacitated vehicle routing problem. To solve the daily routing problem, we modify the code to account for service times. We can solve for each day in our horizon by updating the service times using the solution from the previous day (see equations (2.20) and (2.21)) and calling RTR with data for each new day.

2.4 Instances

This section describes the instances that were used to examine the impact of including learning in technician routing models. There are three sets of attributes that define an instance of the TRP-EST: (1) the number and geographic diversity of the customers requiring service, (2) the number of different skills required to perform tasks requested by customers, and, for each skill, the length of time a novice technician needs to perform a task requiring that skill, and, (3) the number of technicians and each technician's individual traits.

The TRP-EST is a multi-period/day problem, with the number of customers and their geographic distribution not known until the beginning of each day. For our experiments, we limit our analysis to a 29 day period and generate an instance by creating a series of 29 daily customers sets, where the set for day t is $\mathcal{C}_t$. The number of customers and their geographical distributions in these customer sets is based on instances taken from the Symmetric CVRP instances found in VRPLIB: A Vehicle Routing Problem LIBrary. This library of instances is provided by the Operations Research Group at the University of Bologna, Italy and can be found at: `http://www.or.deis.unibo.it/research_pages/ORinstances/VRPLIB/VRPLIB.html`. Specifically, the 29 instances "E072-04f" to "E151D14r" are used, with the number of customers in these instances ranging from 71 to 150. To generate a 29-day instance, we randomly order these 29 daily instances. We generate 10 such instances by considering 10 different orderings of the 29 daily instances. In all instances, travel times correspond to Euclidean distances.

Regarding the second set of attributes, in each instance, each customer $i \in \mathcal{C}_t$ on day t requires a task to be performed and performing that task requires a skill $r \in \mathcal{R}$. We refer to the customer diversity of an instance as the number of different skills required to serve all the customers, or $|\mathcal{R}|$. In our experiments, each instance can have one of four diversity levels: $5, 10, 25$, and 50. For each of the 29 instances "E072-04f" to "E151D14r" and each diversity level $|\mathcal{R}|$, we randomly assign each customer a skill from the set $\{1, \ldots, |\mathcal{R}|\}$.

As discussed in Section 3.2, we model the relationship between experience and productivity using the function $d_{rt}^k = D_r^k + d_{r0}(q_{rt}^k)^{-L_r^k}$. Thus, each instance requires values for the parameters d_{r0}, $\forall r \in \mathcal{R}$, which represents the time it would take a novice technician to perform a task requiring skill r. We assign the parameter d_{r0} one of five initial service time values: 100, 200, 300, 400, and 500 (time units). These values were chosen to represent values that were on the order of travel times (100 units) versus those that were significantly greater (500 units).

These values are assigned to skills in a round-robin manner. For example, with a customer diversity of ten, skills one and six ($r = 1, 6$) will be assigned an initial service time of 100 ($d_{10} = d_{60} = 100$), skills two and seven ($r = 2, 7$) will be assigned an initial service time of 200 ($d_{20} = d_{70} = 200$) and so forth. As noted earlier, the skill associated with any particular customer $\mathcal{R}$ is assigned randomly among the customers in the geography.

Regarding the third set of attributes, in each instance we consider a workforce that consists of 18 technicians. Again recalling the equation $d_{rt}^k = D_r^k + d_{r0}(q_{rt}^k)^{-L_r^k}$, we see that an instance requires values for the parameters D_r^k, q_{r0}^k, and L_r^k, $\forall r \in \mathcal{R}, k \in \mathcal{K}$. In all experiments, we set $D_r^k = 5$ $\forall r \in \mathcal{R}, k \in \mathcal{K}$. As noted previously, this value is unrelated to the technician's experience and ability to learn and reflects that fact that task times do not tend to zero. As such, while the value is chosen arbitrarily, a different parameter value would change the values of our solutions, but not the trends discussed subsequently.

We assume that an individual technician k has the same learning rate on all skills, or, $L_r^k = L_{r'}^k, \forall r, r' \in \mathcal{R}$. This choice reflects the literature on human learning that suggests that individuals tend to learn different skills at the same rate (**?**). As a result, we refer to L_r^k as L^k, and the parameter values used are given in Table 2.1. According to Dar-El (2000), the learning rates 0.515, 0.321, and 0.152 are associated with "fast," "medium," and "slow" learners, respectively. To contextualize these parameter values, Table 2.1 also notes the reduction in service times resulting from doubling a technician's experience level on a skill.

	Label		
	Fast	*Medium*	*Slow*
L^k	.515	.321	.152
Reduction in d_{rt}^k when experience doubled	30%	20%	10%

Table 2.1: Learning rates used in experiments

Finally, similar to parameter L_r^k, we have constructed our experiments so that technician k has the same initial experience, q_{r0}^k in all skills. That is, $q_{r0}^k = q_{r'0}^k$, $\forall r, r' \in \mathcal{R}$. Thus, we can refer to this parameter as q_0^k. We consider three values for this parameter, $q_0^k = 1$, which we call *Low*, $q_0^k = 25$, which we call *Medium*, and $q_0^k = 50$, which we call *High*. We illustrate in Figure 2.1 each of the three learning curves, as well as where each initial experience level puts a technician on his/her curve.

With three possible learning rates and initial experience levels, there are nine

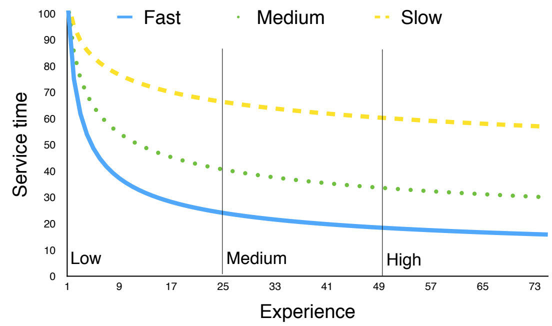


Figure 2.1: Learning curves

Technician		Learning rate		
		Fast	*Medium*	*Slow*
Initial experience	*Low*	1,10	4,13	7,16
	Medium	2,11	5,14	8,17
	High	3,12	6,15	9,18

Table 2.2: Workforce

possible combinations of these traits, and the workforce of 18 technicians that we use in our experiments consists of two technicians for each combination. We present the workforce in detail in Table 2.2. Throughout the rest of the work ,"X-Y" refers to technicians with learning rate "X" and initial experience level "Y." For example, "Slow-Low" refers to technicians 7,16 (see Table 2.2), who are slow learners ($L^k = .152$) and have low initial experience ($q_0^k = 1$).

In summary, we consider 10 different orderings of the 29 daily customer sets, and for each ordering, we consider four diversity levels. For each ordering and diversity level, we consider one random assignment of skills to customers. For all experiments, we consider the same workforce. As such, our experimental setup consists of 40 different instances.

2.5 Computational Analysis

In this section, we perform a computational study to analyze whether, and at what fidelity, learning should be considered when making daily planning decisions. Specifically, we begin our analysis by running a series of experiments to quantify the magnitude of the benefits associated with recognizing both that learning occurs and that each technician's individual traits impact how much they learn when making daily assignments of technicians to tasks. We then seek to understand which is the source of these benefits: recognizing that learning occurs or that each technician has individual traits that impact how they learn.

We then turn our analysis to specialization. Specifically, we study whether when making daily assignments a planner should focus on giving a technician a high level of experience in a limited set of skills (making them "experts") or some experience in many skills (making them "jacks-of-all-trades"). We also study whether a technician's traits (learning rate, initial experience levels) should determine whether they become "experts" or "jacks-of-all-trades." Finally, we study whether the number of different skills on which technicians can gain experience impacts whether technicians should specialize and/or which technicians should specialize.

2.5.1 Value of Heterogeneity and Learning

In this section, we first consider the value of our model versus models that do not incorporate workforce heterogeneity and individual learning. We then demonstrate the value of our model versus models that consider workforce heterogeneity but not individual learning

2.5.1.1 Comparison versus Not Modeling Heterogeneity and Learning

Our analysis begins by studying whether a service organization should track their technicians' experience levels on different skills and then recognize both experience level and each technician's learning rate when performing daily assignments of technicians to tasks. To explore this question, we first run Algorithm 2.1 on each of the 40 instances described above and record the makespan e_t^{max} for each day. As learning is explicitly recognized when Algorithm 2.1 is executed, we label these values e_t^{max-L}.

Next, we compare the values e_t^{max-L} with the daily makespan values seen when both learning and a technician's individual traits (learning rate, experience levels) are ignored when making daily planning decisions. To do so, we run a variant of Algorithm 2.1 that ignores individual learning rates and individually accumulated experience when determining daily assignments. Specifically, recalling that technician $k's$ time to complete a task requiring skill r in period t is modeled by the equation $d_{rt}^k = D_r^k + d_{r0}(q_{rt}^k)^{-L_r^k}$, in this variant of Algorithm 2.1, daily routes are created with the assumption that $d_{rt}^k = \bar{d} \quad \forall r \in \mathcal{R}, t = 1, \ldots, \mathcal{T}, k \in \mathcal{K}$. To calculate the value $\bar{d}$, we first assume that all 18 technicians described in Table 2.2 have *Medium* initial experience ($q_0^k = \bar{q}_0 = 25, \forall k \in \mathcal{K}$) and learn at the *Medium* rate ($L^k = \bar{L} = 0.321, \forall k \in \mathcal{K}$). Next, to calculate $\bar{d}$, we model that, while an organization may not track each technician's actual accumulated experience level, a planner may still use a forecast of accumulated experience when making daily decisions. We presume that this forecast of accumulated experience in each skill depends on the number of different skills required by the customers, or $|\mathcal{R}|$, and report in Table 2.3 the values $\bar{q}_{|\mathcal{R}|}$. As such, we have that technician $k's$ time to complete a task requiring skill r in period t is dictated by the equation $d^k_{rt} = D^k_r + d_{r0}(\bar{q}_{|\mathcal{R}|})^{-\bar{L}}$.

	$\lvert\mathcal{R}\rvert$			
	5	*10*	*25*	*50*
Fixed experience levels	43	33	28	27

Table 2.3: Fixed experience levels

While this variant of Algorithm 2.1 determines daily routes without acknowledging that technicians accumulate experience or differ in their individual traits, to make a fair comparison to e^{max-L}_t, we calculate daily makespans for those routes using the appropriately updated values for q^k_{rt}, d^k_{rt} as specified by Equations (2.20) and (2.21). That is, when calculating the daily makespan for the solutions returned by the variant of Algorithm 2.1, we recognize that experience accumulates and that the time a technician needs to perform a task requiring a specific skill decreases at a rate that depends on his/her individual traits. Because in these experiments we do not consider learning and essentially assume the workforce is homogeneous when making daily planning decisions, we refer to these daily makespan values as $e^{max-NL-H}_t$.

As an example, consider an instance of the TRP-EST for which $|\mathcal{R}| = 5$. Per Table 2.3, on all days, all technicians are assumed to have the experience level 43 on all skills. Suppose that the variant of Algorithm 2.1 described above prescribed that Technician 3 performs three tasks that required skill 2 on day 0 and is currently determining the task assignments for day 1. When executing the RTR code to solve the routing problem for day 1, d^1_{23}, or the time Technician 3 needs to perform a task requiring skill 2, is set to the value $D^3_2 + d_{20}(43)^{-.321}$ (i.e. the experience accumulated on day 0 is ignored and the *Medium* learning rate is assumed). However, after the routes have been created, the makespan value, $e^{max-NL-H}_1$ is calculated with the value d^1_{23} set to the value $D^3_2 + d_{20}(50 + 3)^{-.515}$ (recall that Technician 3 is a fast learner with high initial experience).

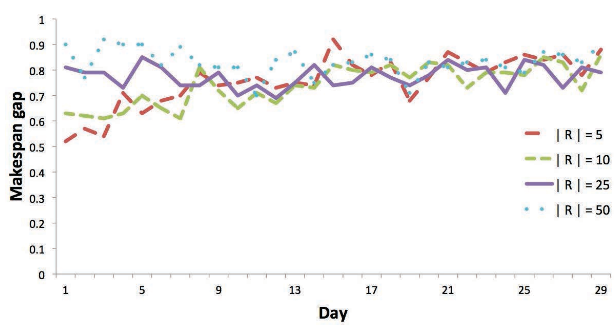
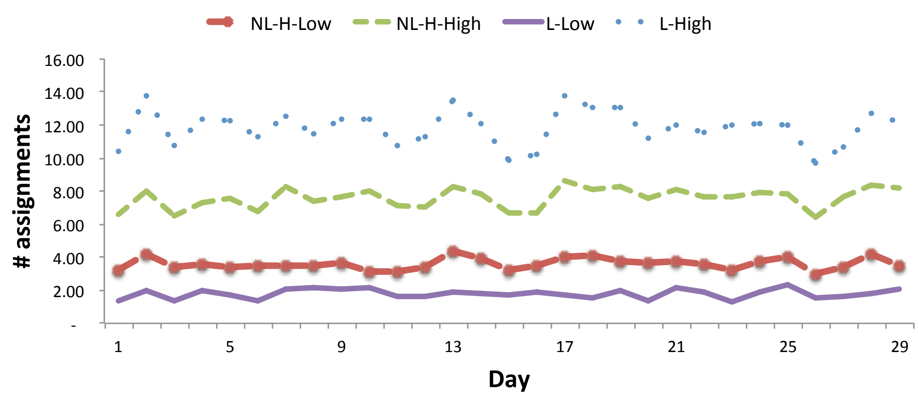


Figure 2.2: Impact of planning without recognizing individual traits or learning

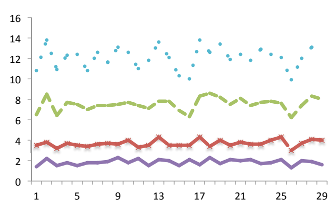
Figure 2.2 presents the daily relative gap in the makespans (calculated as $\frac{e_t^{max-NL-H} - e_t^{max-L}}{e_t^{max-NL-H}}$), averaged over instances with the same customer diversity level. We see in this figure that assuming a homogeneous workforce that does not learn leads to significantly worse routes across all diversity levels and that these routes tend to get worse over time.

To try to understand the source of these large gaps, Figures 2.3(a)-2.3(d) present the degree to which the daily assignment of tasks to technicians is imbalanced. Specifically, for the case in which the workforce is assumed to be homogeneous and learning is not recognized, the line "NL-H-Low" represents the number of tasks assigned to the technician that is assigned the smallest number of tasks, averaged over the 10 instances associated with the respective diversity level. The line associated with "L-Low" is the analogous result for the workforce when recognizing learning and technician heterogeneity. For the case in which the workforce is assumed to be homogeneous and learning is not recognized, the line "NL-H-High" represents the number of tasks assigned to the technician that is assigned the largest number of tasks, averaged over the 10 instances associated with the respective diversity level. The line "L-High" is the analogous result for the workforce when recognizing learning and technician heterogeneity. The results show that, across all diversity levels, there is a larger imbalance in assigned work when learning is recognized.

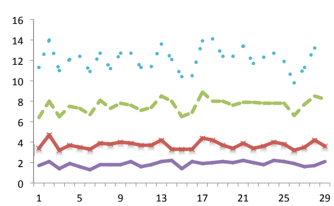
This imbalance results from assignments in the case of recognizing learning and heterogeneity that take advantage of the productivity associated with fast learners. Figure 2.4 shows, by technician type (combination of learning rate and initial experience), the total number of tasks performed. The results show that the solutions considering both learning and workforce heterogeneity assign more tasks to technicians with the *Fast* learning rate and fewer to those that have a *Slow* learn-



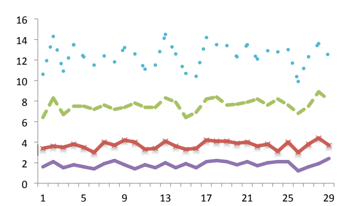
(a) $|\mathcal{R}| = 5$



(b) $|\mathcal{R}| = 10$



(c) $|\mathcal{R}| = 25$



(d) $|\mathcal{R}| = 50$

Figure 2.3: Workload imbalance by diversity

ing rate. Alternatively, the poorer quality solutions resulting from not considering heterogeneity distribute tasks evenly across the workforce. In summary, when not recognizing learning or individual traits, one ignores the extra capacity associated with *Fast* learners.

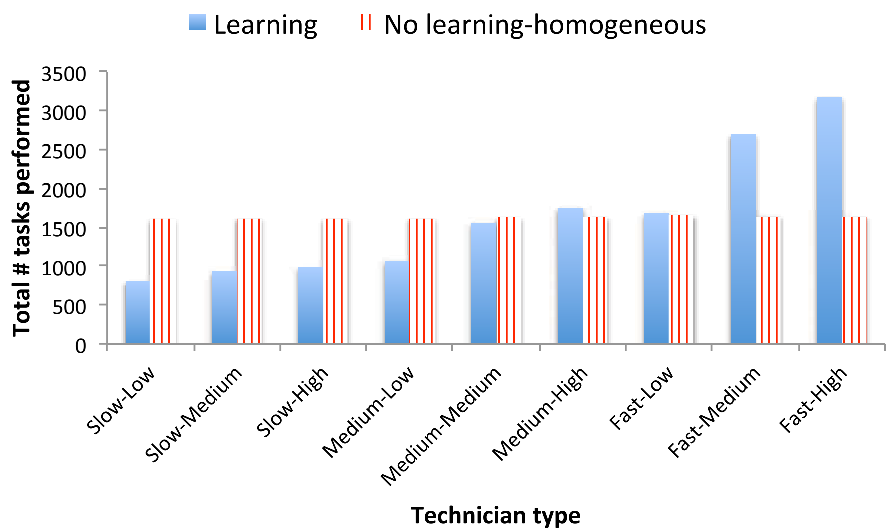
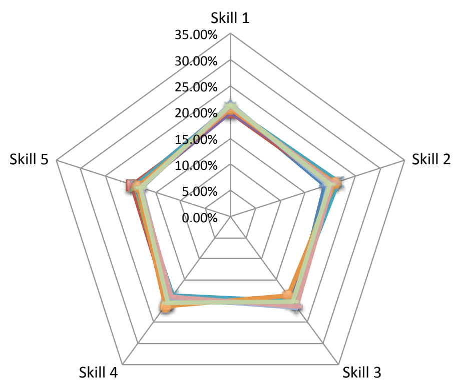


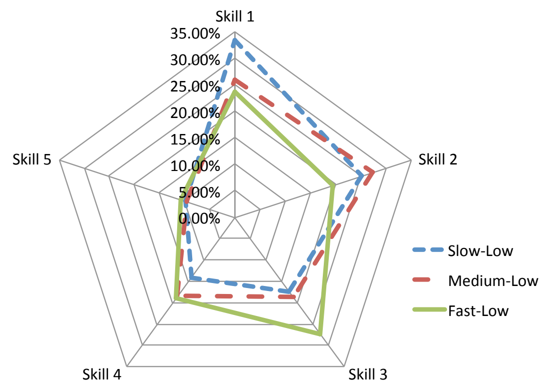
Figure 2.4: Total number of tasks performed by technician type

While our results are from a 29-day planning horizon, we believe the problems associated with ignoring learning and assuming a homogeneous workforce will persist over longer horizons. As evidence, Figures 2.5(a)-2.5(d) present the distribution of technicians' experience levels at the end of the planning horizon, averaging over all 10 instances with a task diversity of 5 and by technicians with the same learning rate and initial experience. Specifically, to calculate the "Skill 1" point on the "Slow-

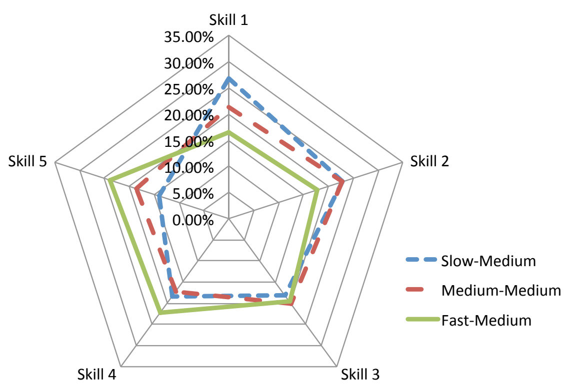
Low" curve, we first calculate the quantities $f_7^1 = \frac{q_{1,7}^{29}}{\sum_{r=1}^{5} q_{r,7}^{29}}$ and $f_{16}^1 = \frac{q_{1,16}^{29}}{\sum_{r=1}^{5} q_{r,16}^{29}}$, with the first representing the fraction of technician 7's assignments over the 29-day period that were to a task that required Skill 1 and the second is the same only for technician 16. Then, the "Skill 1" point for a "Slow-Low" technician is calculated as $f_{SL}^1 = (f_7^1 + f_{16}^1)/2$. Other points in the figures are calculated similarly.



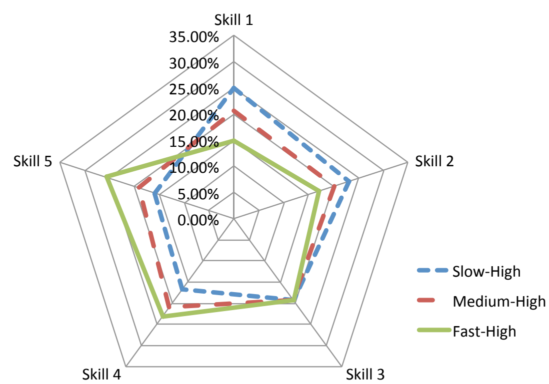
(a) Ignore learning and individual traits



(b) Recognize learning (Low initial exp.)



(c) Recognize learning (Medium initial exp.)



(d) Recognize learning (High initial exp.)

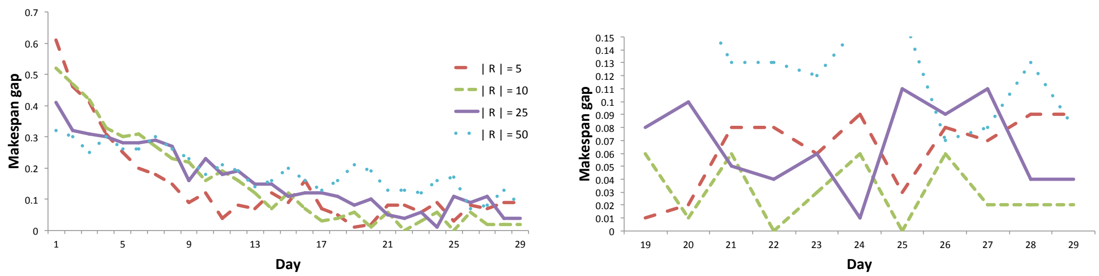
Figure 2.5: Distribution of experience levels at end of 29 days ($|\mathcal{R}| = 5$)

The figures show that the distribution of experience levels for technicians are different when learning and individual traits are recognized from when they are not. For example, when learning and individual traits are recognized, the experience levels for technicians with *Low* initial experience (Figure 2.5(b)) are not evenly distributed across all skills as they are when they are ignored. As these experience levels impact the productivity of technicians in subsequent days, we conclude that ignoring learning and individual traits will leave a workforce poorly prepared for the future. We note that similar graphs for other diversity levels ($|\mathcal{R}| = 10, 25, 50$) exhibit a similar pattern. Recognizing learning and individual traits leads to a workforce with experience unevenly distributed across all skills whereas not recognizing these factors does not. We explore this issue further in the next subsection.

2.5.1.2 Comparison versus Modeling of Heterogeneity but not Learning

In the previous analysis, recognizing learning and each technician's individual traits leads to much better daily decisions and a different workforce at the end of the 29-day planning horizon, in terms of skill proficiency, than not doing so. We next discuss experiments that seek to understand whether recognizing workforce heterogeneity, but still not recognizing learning, will close the gaps seen in Figure 2.2.

Specifically, we again execute a variant of Algorithm 2.1 where daily routes are created with the assumption that each technician has a static experience level on each skill. In contrast to the previous experiments in which experience was fixed based



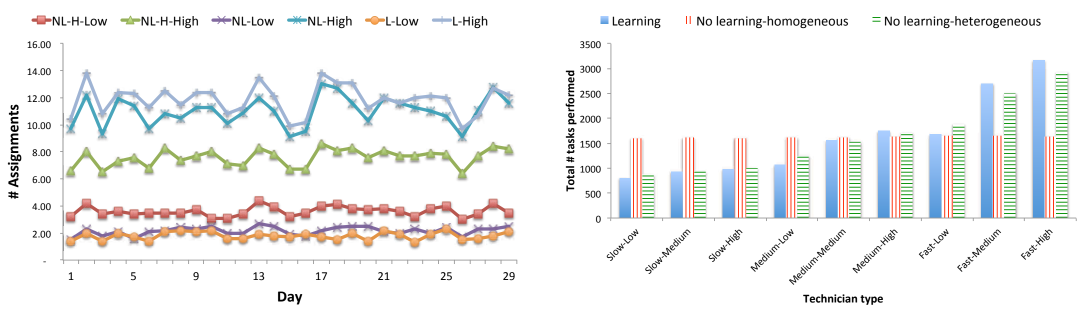
(a) All 29 days (b) Last 10 days

Figure 2.6: Impact of planning without recognizing learning but recognizing individual traits

on the assumption that everyone began with a *Medium* experience level, in these experiments, the fixed experience level for a technician is based on the appropriate value q_0^k according to Table 2.2. Similarly, in the previous experiments, the time a technician needed to perform a task $\bar{d}$ was calculated based on the assumption that the technician learned at the *Medium* learning rate. In these experiments, the value $\bar{d}_k$ for technician k is calculated based on their individual value L^k from Table 2.1 and according to Table 2.2. As in the previous experiment, we also assume that each technician reaches a level of forecasted experience. Finally, analogous to the previous experiment, while we generate solutions that ignore learning, we evaluate the solutions returned by the algorithm by including learning and accumulated experience. We denote these solutions values by e_t^{max-NL} for each day t in the horizon.

We now perform an analysis similar to what was done for the previous experiments. Figures 2.6(a) and 2.6(b) present results calculated in a manner similar to those presented in Figure 2.2. We see that, while recognizing individual traits reduces the gaps seen in Figure 2.2, it does not close them. The gaps still range from 2% to 10% at the end of the 29-day planning horizon.

Evidence for why the gaps close can be seen in Figures 2.7(a) and 2.7(b), which display results similar to those seen in Figures 2.3(a) and 2.4. The figures show that recognizing workforce heterogeneity, yet still ignoring learning, leads to a much better appreciation of the capacity associated with *Fast* learners and thus a similar imbalance of work as when learning is recognized. While we only show figures for $|\mathcal{R}| = 5$, results for other diversity levels exhibit a similar pattern.



(a) Imbalance (b) Total distribution by technician type

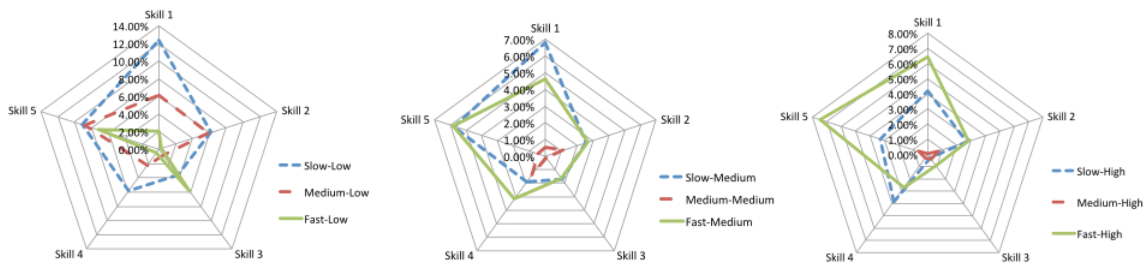
Figure 2.7: Workload analysis when recognizing individual traits but not learning ($|\mathcal{R}| = 5$)

The reason that the gaps do not close completely can be seen by analyzing two differences in the distributions of experience levels at the end of the 29-day period: (1)

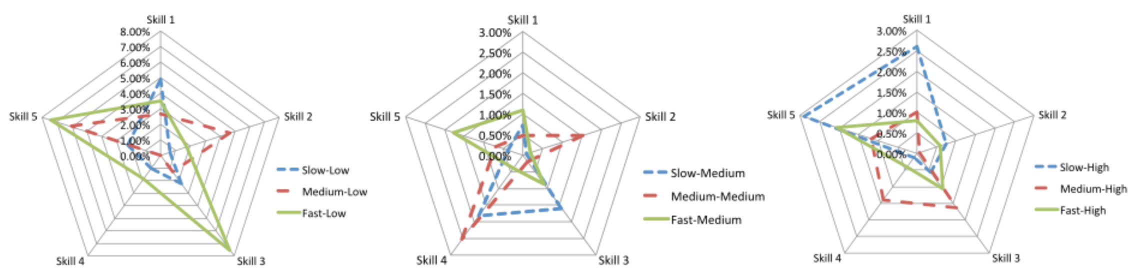
the differences between when learning is recognized and when both learning and individual traits are ignored, and, (2) the differences between when learning is recognized and when learning is ignored but individual traits are not. We calculate the quantities f_{XY}^{r} (e.g. f_{SL}^{1} as discussed previously) for each of the three runs of Algorithm 2.1: (1) when learning is recognized, which we label f_{XY}^{r-L}, (2) when learning and individual traits are ignored, which we label f_{XY}^{r-NL-H}, and (3) when learning is ignored but individual traits are not, which we label f_{XY}^{r-NL}. Figure 2.8(a) presents the quantities $|f_{XY}^{r-L} - f_{XY}^{r-NL-H}|, \ \forall r \in \mathcal{R}$ and all technician types, "X-Y" (e.g. "Slow-Low"). Figure 2.8(b) presents the quantities $|f_{XY}^{r-L} - f_{XY}^{r-NL}|, \ \forall r \in \mathcal{R}$ and all technician types. Similar to what was seen in Figures 2.5(a) to 2.5(d), we see that ignoring learning and individual traits leaves a workforce with a very different set of skill levels than recognizing learning does. Figure 2.8(b) suggests that recognizing individual traits reduces the differences in skill levels but does not eliminate them. Again, while we only display results for $|\mathcal{R}| = 5$, the pattern is similar for other diversity levels. In summary, while recognizing individual traits is better than not doing so, it is best to recognize learning as it has benefits in the short term and the impact of doing so persists through differences in the distribution of final experience levels.

2.5.2 Workforce Specialization

Having established that an organization should consider both an up-to-date accounting of their technician's experience levels at different skills as well as their individual learning rates, we next turn our attention to what insights can be gained



(a) Also ignore individual traits



(b) Recognize individual traits

Figure 2.8: Differences in distribution of experience levels at end of 29 days when recognizing learning and not($|\mathcal{R}| = 5$)

into how experience levels and learning rates should be mapped to daily scheduling and routing decisions. Our analysis primarily focuses on the concept of *specialization* and looks at whether a technician should become an expert in a small set of skills and whether a technician's traits impact this decision.

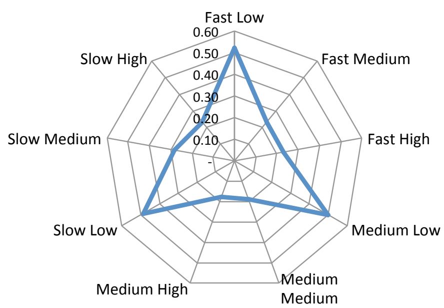
To measure specialization, we use the Coefficient of Variation of the number of times a technician performs tasks requiring each skill. For example, suppose that at the end of 29 days, a technician has performed tasks that require Skill 1 five times, tasks that require Skill 2 one time, tasks that require Skill 3 two times, tasks that require Skill 4 three times, and tasks that require Skill 5 four times. Then, the expected number of times that technician performs a task requires a given skill is $\mu = (5+1+2+3+4)/5 = 3$. Similarly, the standard deviation of the number of times that technician performs tasks requiring each skill can be calculated as $\sigma = \sqrt{\frac{(5-3)^2+(1-3)^2+(2-3)^2+(3-3)^2+(4-3)^2}{5}} = 1.414$, leaving a $CV = \frac{1.414}{3} = 0.471$. In the analysis that follows, we calculate CV values based on a technician's experience at the end of the 29-day planning horizon. Specifically, $CV_k = \frac{\sigma_k}{\mu_k}$ where $\mu_k = \frac{\sum_{r\in\mathcal{R}} q^k_{r29}}{|\mathcal{R}|}$ and $\sigma_k = \sqrt{\frac{\sum_{r\in\mathcal{R}}(q^k_{r29}-\mu_k)^2}{|\mathcal{R}|}}$. When referring to a CV value for a technician type, such as a "Slow-Low" technician, we refer to the average of the CVs of the technicians that are of that type; e.g. $CV_{Slow-Low} = (CV_7 + CV_{16})/2$. Higher CV values represent greater specialization as they indicate a wider disparity between the tasks done the most times and those done the least.

Figures 2.9(a)-2.9(d) present CV values averaged over all 10 instances of each customer diversity and by technician type. The figures show that, regardless of diversity level, technicians with Low initial experience specialize the most, and their degree of specialization increases as diversity increases (note the scale is different for each figure). On the other hand, technicians with High initial experience specialize the least.

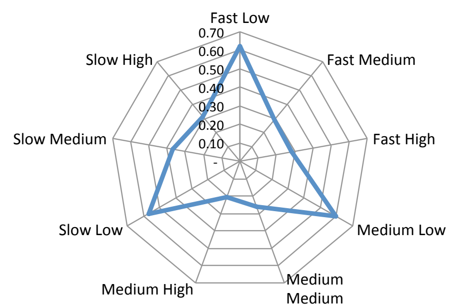
To understand the result, consider that workers with low initial experience are relatively inefficient in all tasks, and more importantly, see a very large marginal benefit from additional experience on a task. As a result, the workers with Low initial experience see significant gains in productivity on the tasks to which they are assigned in the first few days of the horizon. These gains make Low initial experience workers relatively more attractive on those tasks relative to workers with High initial experience. Thus, the best solutions see Low initial experience workers frequently assigned to the tasks on which they gain the most experience in the first few days.

We next look at specialization from a different perspective. If daily routes were evaluated based only on travel times, one expects specialization to only occur due to the geographic distribution of customers (e.g. a cluster of customers requesting tasks that require the same skill). Conversely, if daily routes were evaluated based only on service times, one would expect a high degree of specialization. We next seek to understand whether, when making daily task assignment decisions, some technician types can be treated as individuals for whom one should focus on routing (minimizing travel time) or scheduling (minimize service time).

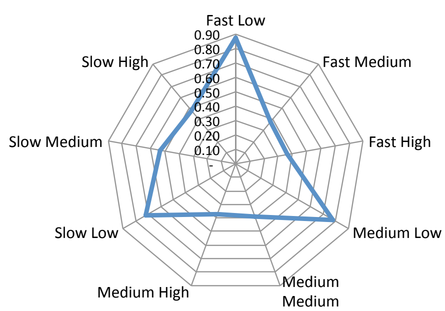
To do so, we run two different variants of Algorithm 2.1 for each instance; each variant differs from Algorithm 2.1 only in how e_t^{max} is calculated. The first variant



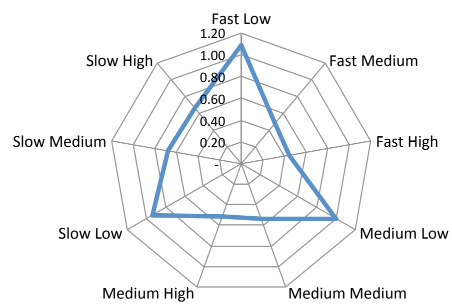
(a) $|\mathcal{R}| = 5$



(b) $|\mathcal{R}| = 10$



(c) $|\mathcal{R}| = 25$



(d) $|\mathcal{R}| = 50$

Figure 2.9: Specialization by customer diversity and technician type

only counts travel times when minimizing e_t^{max}. We treat the CV values (which we label CV_{X-Y}^{TT}) in solutions produced with this variant as lower bounds on the amount of specialization one should see in a solution produced by Algorithm 2.1. Similarly, the second variant only counts service times when minimizing e_t^{max}, and we interpret the CV values (which we label CV_{X-Y}^{ST}) in solutions produced with this variant as upper bounds. We present an example of these three sets of CV values in Figure 2.10. With these values, we hypothesize that a planner should focus on routing technician type "X-Y" when

$$CV_{X-Y} - CV_{X-Y}^{TT} < CV_{X-Y}^{ST} - CV_{X-Y}, \tag{2.22}$$

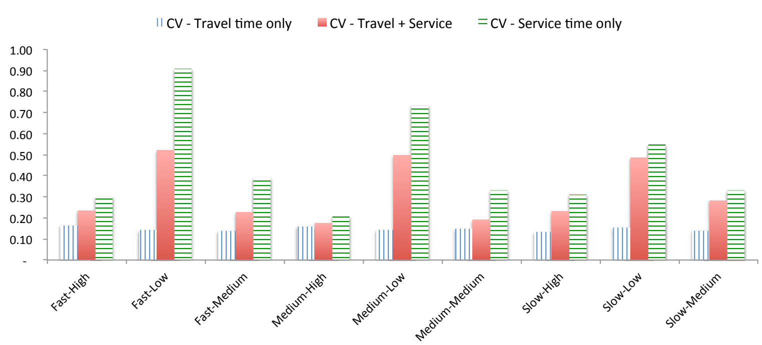


Figure 2.10: Determining assignment rules ($|\mathcal{R}| = 5$)

or, the CV value produced when running Algorithm 2.1 is closer to the one produced when only counting travel times than the one produced when only counting service times. We then say that a planner should focus on scheduling technician type

“X-Y” when Inequality (2.22) does not hold. For example, referring to Figure 2.10, we conclude that a planner should focus on routing “Medium-Medium” technicians and scheduling “Slow-Low” technicians. Essentially, specializing matters more for “Slow-Low” technicians than “Medium-Medium” technicians.

We then calculate average CV, CV^{TT}_{X-Y}, and CV^{ST}_{X-Y} values over all 10 instances for each diversity level and apply this line of reasoning to derive heuristics or “rules of thumb” for what a planner should focus on when making daily task assignments. Tables 2.11(a)-2.11(d) present these rules for all technician types and by customer diversity level. We see that the rules are fairly static across customer diversity levels, with the only differences being the degree to which a planner should focus on scheduling individuals with “Low” initial experience. For all cases of a “High” or “Medium” initial experience and either “Fast” or “Medium” learning rate, routing is the dominant factor in determining the assignment. Further, when initial experience is “Low” and the learning rate is “Slow,” the scheduling dominates. Further, for diversity levels of 25 and 50, the assignments of the technicians with “Low” initial experience are dominated by the scheduling concerns. Differentiation comes at the lower diversity levels and the case of “Low” initial experience. For the case of a diversity of 10, technicians with “Medium” initial experience and “Slow” or “Medium” learning rates are dominated by routing. For the diversity of five, technicians with “Medium” initial experience, and “Slow” or “Medium” learning rates, the assignments are dominated by routing and scheduling, respectively. We believe that the difference in the “Low” initial experience and “Medium” learning rate is the result of sampling error and that there is no significant difference to focusing assignments on the routing or scheduling of “Slow” or “Medium” learners with “Low” initial experience.

Assignment rule		Learning rate *Fast*	*Medium*	*Slow*
Init. exp.	*Low*	Route	Schedule	Schedule
	Medium	Route	Route	Schedule
	High	Route	Route	Schedule

(a) $|\mathcal{R}| = 5$

Assignment rule		Learning rate *Fast*	*Medium*	*Slow*
Init. exp.	*Low*	Route	Route	Schedule
	Medium	Route	Route	Schedule
	High	Route	Route	Schedule

(b) $|\mathcal{R}| = 10$

Assignment rule		Learning rate *Fast*	*Medium*	*Slow*
Init. exp.	*Low*	Schedule	Schedule	Schedule
	Medium	Route	Route	Schedule
	High	Route	Route	Schedule

(c) $|\mathcal{R}| = 25$

Assignment rule		Learning rate *Fast*	*Medium*	*Slow*
Init. exp.	*Low*	Schedule	Schedule	Schedule
	Medium	Route	Route	Schedule
	High	Route	Route	Schedule

(d) $|\mathcal{R}| = 50$

Figure 2.11: Assignment rules by customer diversity and technician type

2.6 Conclusions

In this chapter, we present a model and approach for accounting for on-the-job learning in the daily routing of technicians. Our objective minimizes the completion time of the last task. We solve the daily routing problems using the record-to-record travel (RTR) heuristic. Our results offer the following key insights:

1. Explicitly modeling both learning and technician heterogeneity leads to better and different solutions in comparison to assuming homogeneous learning curves and/or static productivity.

2. Relatedly, explicitly modeling both learning and technician heterogeneity leads to different distributions of skills in comparison to assuming homogeneous learn-

ing curves and/or static productivity.

3. Inexperienced technicians specialize the most and experienced technicians the least.

The differences in the solutions to the model that incorporates workforce heterogeneity and human learning versus those that do not demonstrates the importance of incorporating workforce characteristics into the model when applicable. While the importance of such characteristics have been discussed in the manufacturing literature, this work is the first to show it for routing problems.

Ultimately, our results reveal the need to balance routing and scheduling when making assignments for a heterogeneous workforce that learns. Our experiments and "rules of thumb" identify that the scheduling aspect is particularly dominant for workers who are "Slow" learners or who are of "Low" initial experience. Routing dominates the assignments for "Fast" or "Medium" learners coupled with either "High" or "Medium" initial experience. This rule-based approach can help managers develop effective daily plans with less computational efforts than what is required by our optimization.

CHAPTER 3
MULTI-PERIOD TECHNICIAN SCHEDULING WITH EXPERIENCE-BASED SERVICE TIMES AND STOCHASTIC CUSTOMERS

3.1 Introduction

In this chapter, we study the the multi-period technician scheduling problem with experience-based service times and stochastic customers (MTSP-ESTSC). We assume that daily demand is not revealed until the day of service. Each day, the technicians serve the day's known demand. In this work, we seek to minimize the expected sum of each day's total service times over a finite horizon. The objective accounts for the desire to increase the capacity available to grow the business.

To solve the problem, we propose an approximate dynamic programming (ADP) approach that, at each stage, solves a mixed integer program (MIP) to assign technicians to tasks. In addition to recognizing the resulting service times in the current period, the MIP approximates the impact of those assignments on future technician service times (the "cost-to-go") with a forecast of each technician's task assignments in the next period. Assignment decisions in the next period are partially driven by each technician's service times on each type of task, which are in turn driven by their experience level on each type of task.To capture this fact, the forecasting model embedded in the MIP is a function of the assignment decisions in the current period.

One of the challenges associated with solving optimization models that recognize that humans learn is that the quantitative models of human learning proposed by the psychology community are non-linear. As a result, to solve a MIP at each stage of the ADP, we adapt an exact reformulation method from the literature that relies on the fact that the function we use to map experience to service time has a finite domain.

In this study, we make the following research contributions. First, we present a Markov decision process (MDP) model for the MTSP-ESTSC. We note that this model can be adapted to handle cases of worker attrition and new task types. Second, we introduce a method for approximating the future value of today's workforce assignments, and third, demonstrate how to incorporate this approximation into a tractable model for making daily decisions.

We demonstrate the value of the proposed solution approach with three experiments. In one experiment, the set of technicians and the set of task types remain the same over the problem horizon. In the second, we introduce a workforce disruption in the middle of the horizon in which one technician leaves the workforce and a new technician is added. The third variant adds an additional task type in the middle of the horizon. For each of the three problem variants, we compare the proposed solution approach to a myopic solution approach that views the problem as a single-period problem, ignoring the impact of current period decisions on future service times. Our comparisons demonstrate that the proposed solution approach leads to higher-quality solutions by better positioning technicians to meet future demands.

The remainder of this chapter is organized as follows. Section 3.2 presents a model for the problem. Section 3.3 describes the solution approach. Section 3.4 discuss the design of the experiments, and Section 3.5 presents our computational results. Finally, Section 3.6 concludes this work and suggests areas of future research.

3.2 Problem Formulation

In this section, we present a formal model for the multi-period technician scheduling problem with experience-based service times and stochastic customers (MTSP-ESTSC). We also discuss how the model can be adapted to a number of problem variants.

Let the set $\mathcal{T} = \{1, 2, \ldots, T\}$, be the set of T days in the planning horizon. Let $\mathcal{K} = \{1, 2, \ldots, K\}$ be the set of technicians, and let $\mathcal{R} = \{1, 2, \ldots, R\}$ be the set of all possible task types. In this paper, we assume that the technicians learn independently on each task. However, both our model and our solution approach can easily be modified to account for the transfer of experience on one task to another particularly as described by Olivella (2007).

Each technician k in $\mathcal{K}$ requires a particular time to complete each task r in $\mathcal{R}$. The service time for a particular task depends on the experience a technician has in performing the task as well as the technician's ability to learn from that experience. We let d_{r0} be the maximum service time for a task r, and D_r^k be the maximum productivity rate for technician k performing task r. The parameter L_r^k describes how quickly technician k can learn from experience on task r. The parameter L_r^k is often called a learning rate. In practice, these individual parameters can be estimated from data.

Given these parameters, we can determine technician k's service times for each task in $\mathcal{R}$ at the start of day t using a learning curve. The learning curve requires a technician's experience on a given task up to day t. We capture this experience in the R–dimensional vector $\mathcal{Q}_t^k = (q_{1t}^k, q_{2t}^k, \ldots, q_{Rt}^k)$, where the r^{th} entry q_{rt}^k indicates technician k's experience with task r^{th} at the start of day t. We let $\mathcal{Q}_0^k$ represent technician k's experience at the beginning of the horizon. Assuming the hyperbolic learning curve (Nembhard and Osothsilp, 2005), technician k's service time for task r on day t is:

$$d_{rt}^k = (D_r^k(\frac{q_{rt}^k}{q_{rt}^k + L_r^k}))^{-1}. \tag{3.1}$$

On a given day t, we seek to serve a set of tasks $\mathcal{N}_t = \{1, 2, \ldots, N_t\}$ by assigning the tasks to individual technicians. We assume that there is a limit C to the amount of time that a technician can work in a day. We also assume that this set of tasks becomes known only at the beginning of day t. Our problem objective is to minimize the expected sum of each day's service times.

We model the optimization problem as a Markov decision process (MDP). A decision is made at the beginning of each day during the planning horizon. Let $t = 1, 2, \ldots, T$ be the decision epochs, where day T is the last day in the problem horizon. The state of the system captures all the information that we need to make a scheduling decision. In this case, to make task assignments for the day, we must know the set of tasks that need to be served for the day and the experience of the technicians at the start of day. Then, the state of the system at the beginning of day

t is $s_t = \{\mathcal{N}_t, \mathcal{Q}_t\}$.

Given state s_t, an action is a set of assignments that serves the day t requests. Let action $a_t(s_t) = \{x_{it}^k : i \in \mathcal{N}_t, k \in \mathcal{K}\}$, where binary variables x_{it}^k equals 1 if customer i is served by technician k in day t. For notational convenience, let $r(i)$ map a task i into the set of task types $\mathcal{R}$. Then, given a day t, a feasible action satisfies the following constraints:

$$\sum_{k\in\mathcal{K}} x_{it}^k = 1 \qquad \forall i \in \mathcal{N}_t, \tag{3.2}$$

$$\sum_{i\in\mathcal{N}_t} x_{it}^k d_{r(i)t}^k \leq C \quad \forall k \in \mathcal{K} \tag{3.3}$$

$$x_{it}^k \in \{0, 1\} \qquad \forall i \in \mathcal{N}_t, \forall k \in \mathcal{K}. \tag{3.4}$$

Constraints (3.2) ensure that a task is assigned to exactly one technician and that the task is assigned only once. Constraints (3.3) ensure that all technicians finish working by the end of the day, and Constraints (3.4) ensure integrality. Let $\mathcal{A}_t(s_t)$ be the set of all actions available on day t given state s_t.

Given that the state is currently s_t and that action a_t in $\mathcal{A}_t(s_t)$ is selected, a deterministic transition is made to post-decision state $s_t^a = \{\mathcal{N}_t, \mathcal{Q}_t^a\}$ by updating technicians' experience as follows:

$$q_{r,t+1}^k(s_t, a_t) = q_{rt}^k + \sum_{i\in\mathcal{N}_t:r(i)=r} x_{it}^k \quad \forall k \in \mathcal{K}, \forall r \in \mathcal{R} \tag{3.5}$$

Equation (3.5) adds each technician's existing experience the experience gained from the day t assignments.

Following day t, we observe exogenous information consisting of the new service requests for day $t+1$, denoted as W_{t+1}. The random vector W_{t+1} is such that

$W_{t+1} = (w_{1,t+1}, \ldots, w_{\check{N}_{t+1},t+1})$ where $w_{i,t+1}$ is a random variable representing the customer type of the i^{th} service request on day $t+1$ and $\check{N}_{t+1}$ is a random variable representing the random number of requests on day $t+1$, neither of which is known prior to day $t+1$. Following day t, we observe the exogeneous information for day $t+1$ and transition to decision epoch $t+1$. Decision epoch $t+1$ requires a transition from post-decision state s_t^a to pre-decision state $s_{t+1} = \{\mathcal{N}_{t+1}, \mathcal{Q}_{t+1}\}$, where $\mathcal{N}_{t+1}$ is the realization of the exogenous process for day $t+1$. In this transition, technicians' experience remain unchanged, thus $\mathcal{Q}_{t+1} = \mathcal{Q}_t^a$. The technicians' productivity for day $t+1$ can be calculated according to the learning curve given the technicians' experience levels.

At decision epoch t, given state s_t and action $a_t(s_t)$, a transition from pre-decision state s_k to post-decision state s_k^a results in a contribution

$$C(s_t, a_t) = \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{N}_t} x_{it}^k d_{r(i)t}^k \qquad \forall a_t \in \mathcal{A}_t(s_t), \tag{3.6}$$

which is the total service time to complete all tasks on day t. The problem objective is then $\min_{\pi \in \Pi} E\left[\sum_{t=1}^{T} C(s_t, \delta_t^\pi(s_t))\right]$, where π is a policy that determines actions for all days t over the problem horizon T, Π is the set of all policies, and $\delta_t^\pi(s_t)$ is a decision rule specific to policy π that maps the state s_t to an action a_t.

3.2.1 Problem Variants

We next discuss how the model presented above can accommodate two problem variants: (1) where there is attrition and hiring in the technician pool, and, (2) when there is a new task type to be serviced. First, we can model that technician k leaves the workforce and is replaced by a new technician, k', on day t by setting $q^k_{r,t}(s_t, s_t) = I^{k'}_r$, where $I^{k'}_r$ is the initial experience level of technician k' on tasks of type r. For the second variant, wherein a new task type (labeled $R+1$) is to be serviced starting on day t, we expand the state to include the experience levels $q^k_{R+1,t}, \forall k \in \mathcal{K}$ starting on day t. We can then initialize this experience level, $q^k_{R+1,t}$ to whatever the appropriate value should be for each technician and the algorithm can proceed as before.

3.3 Solution Approach

Given our assumptions, an optimal policy can be found by using the well known Bellman equation and backward dynamic programming. The general form of the Bellman equation and the form sufficient for our problem is given by setting

$$V(s_t) = \min_{a \in \mathcal{A}_t(s_t)} \{C(s_t, a) + E\left[V(s_{t+1}) \mid s_t, a\right]\}. \tag{3.7}$$

While we can easily state the Bellman equation for our problem, because of the need to capture each day's set of requests in the state space, the problem is far too large for an exact solution approach. Instead, we introduce an approximate dynamic programming (ADP) approach that approximates the cost-to-go, the second term in Equation (3.7), using a forecaste of technician assignments for one day in the future. Using this approximation, we then step forward in time. Analogous to rollout algorithms (see Goodson et al. (2015)), by stepping forward in time, we need only find actions for states that are realized. In the remainder of this section, we first present our approximation of the cost-to-go. We then demonstrate how to use this approximation to solve an approximate form of the Bellman Equation, Equation (3.7).

Finally, we discuss our forward ADP algorithm.

3.3.1 Cost-to-Go Approximation and Solving the Approximate Bellman Equation

For each technician, this forecast is based on the technician's previous assignments and a forecast of the next day's tasks. Specifically, for each technician and each customer type, we first predict the demand requests for the next period with an exponential smoothing forecasting: $F^k_{rt} = \alpha F^k_{r,t-1} + (1-\alpha)A^k_{r,t-1}$, where $F^k_{r,t-1}$ and $A^k_{r,t-1}$ are the numbers of predicted and actual type r demand requests assigned to technician k for period $t-1$, respectively, and F^k_{rt} is the forecasted number of assignments of task type r to technician k on day t. The parameter α is the smoothing factor. Given an assignment of technician k for each task i on day $t-1$, $x^k_{i,t-1}$, we can compute $A^k_{r,t-1}$ as $\sum_{i\in\mathcal{N}_{t-1}:r(i)=r} x^k_{i,t-1}$.

With the forecasted number of assignments for day t, F^k_{rt}, for each technician k and task type r, we can write an approximate Bellman equation as:

$$\hat{V}(s_{t-1}) = \min_{a\in\mathcal{A}_t(s_{t-1})} \{C(s_{t-1}, a) + \sum_{k\in\mathcal{K}}\sum_{r\in\mathcal{R}} F^k_{rt}\hat{d}^k_{rt}\}, \tag{3.8}$$

where $\hat{d}^k_{rt}$ is the forecasted service time on day t for task type r and technician k. Similar to $A^k_{r,t-1}$, for technician k and task type r, $\hat{d}^k_{rt}$ can be calculated using the assignments for day $t-1$ as:

$$\hat{d}^k_{rt} = (D^k_r(\frac{q^k_{r,t-1} + \sum_{i\in\mathcal{N}_{t-1}:r(i)=r} x^k_{i,t-1}}{q^k_{r,t-1} + \sum_{i\in\mathcal{N}_{t-1}:r(i)=r} x^k_{i,t-1} + L^k_r}))^{-1}. \tag{3.9}$$

We can rewrite Equation (3.8) as a math program as follows:

$$\textbf{(F)} \quad \min \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{N}_{t-1}} x^k_{i,t-1} d^k_{r(i),t-1} + \sum_{k \in \mathcal{K}} \sum_{r \in \mathcal{R}} F^k_{rt} \hat{d}^k_{rt}$$

subject to (3.2), (3.3), (3.4) ,

$$F^k_{rt} = \alpha F^k_{r,t-1} + (1-\alpha) \sum_{i \in \mathcal{N}_t : r(i) = r} x^k_{i,t-1} \qquad , \forall k \in \mathcal{K}, \forall r \in \mathcal{R}, \quad (3.10)$$

$$\hat{d}^k_{rt} = (D^k_r(\frac{q^k_{r,t-1} + \sum_{i \in \mathcal{N}_{t-1} : r(i) = r} x^k_{i,t-1}}{q^k_{r,t-1} + \sum_{i \in \mathcal{N}_{t-1} : r(i) = r} x^k_{i,t-1} + L^k_r}))^{-1}, \forall r \in \mathcal{R}, , \forall k \in \mathcal{K}, \quad (3.11)$$

$$x^k_{i,t-1} \text{ binary} \qquad , \forall k \in \mathcal{K}, \forall i \in \mathcal{N}_{t-1}. \quad (3.12)$$

The objective is to minimize the sum of total service time and the forecasted service time for the next period. As presented earlier, Constraints (3.2) to (3.4) define feasible actions. Constraints (3.10) are replaced with $F^k_{rt} = \sum_{i \in \mathcal{N}_t : r(i) = r} x^k_{i,t-1}$ when $t - 1 = 1$. Constraints (3.11) determine the service time for the forecasted assignments.

Model (**F**) is non-linear. We use the reformulation presented by Hewitt et al. (2015) to overcome this challenge. The reformulation takes advantage of the fact that the task assignments and thus experience are discrete and that the maximum number of tasks of each type is known for a given day $t - 1$. Thus, we can enumerate the set of potential forecasted service times for each technician on each customer type for day t. We describe the steps of the enumeration in Algorithm 3.1.

To present the reformulation, we first present some additional notation. We let $z^{kj}_{r,t-1}$ be a binary variable indicating whether technician k did j tasks of type r on day $t - 1$. Let $\bar{d}^{kj}_{rt}$ represent technician k's estimated service times for serving tasks of type r on day t having done j jobs of task r on day $t-1$. Then, having completed j tasks of type r on day $t-1$ and given $\bar{d}_{rt}^{kj}$ for technician k on task type r for day t, the total time required for k to complete the forecasted number of tasks of type r on day t is $\overline{ST}_{rt}^{kj} = \bar{d}_{rt}^{kj} F_{rt}^{k}$. We introduce ST_{rt}^{k} as a decision variable that gives the total service time of technician k on tasks of type r for day t. Finally, let $M_{t-1} = (m_{1,t-1}, m_{2,t-1}, \ldots, m_{R,t-1})$ be a vector of the number of each task type r for day $t-1$. Then, the reformulation model is as follows:

$$\textbf{(R)} \quad \min \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{N}_{t-1}} x_{i,t-1}^{k} d_{r(i),t-1}^{k} + \sum_{k \in \mathcal{K}} \sum_{r \in \mathcal{R}} ST_{rt}^{k}$$

$$\text{subject to } (3.2), (3.3), (3.4), (3.10) \qquad ,$$

$$ST_{rt}^{k} = \sum_{j=1}^{m_{r,t-1}} \overline{ST}_{rt}^{kj} z_{r,t-1}^{kj} \qquad ,\forall r \in \mathcal{R}, \forall k \in \mathcal{K}, \tag{3.13}$$

$$\sum_{j=1}^{m_{r,t-1}} j z_{r,t-1}^{kj} = \sum_{i \in \mathcal{N}_{t-1} : r(i) = r} x_{i,t-1}^{k} \qquad ,\forall r \in \mathcal{R}, \forall k \in \mathcal{K}, \tag{3.14}$$

$$\sum_{j=1}^{m_{r,t-1}} z_{r,t-1}^{kj} = 1 \qquad ,\forall r \in \mathcal{R}, \forall k \in \mathcal{K}, \tag{3.15}$$

$$z_{rt}^{kj} \text{ binary} \qquad ,\forall r \in \mathcal{R}, \forall k \in \mathcal{K}, j = 1, \ldots, m_{rt}. \tag{3.16}$$

Constraints (3.13) assign one forecasted total service time for each technician on each task type for day t. Constraints (3.14) ensure that the next day's forecasted total service time of a technician on each task type corresponds to the number of this type of customers he/she performed today. Constraints (3.15) ensure that each technician is assigned exactly one forecasted total service time for each task type on each day.

3.3.2 Forward Approximate Dynamic Programming Algorithm

Algorithm 3.1 presents our forward ADP algorithm for solving instances of the MTSP-ESTSC. Unlike the traditional backward dynamic programming algorithm, Algorithm 3.1 steps forward in time. Stepping forward in means offers the advantage that we need only solve the Bellman equation for those states that are actually visited. However, stepping forward in time also means that we have not computed values for the cost-to-go in the Bellman equation. We instead approximate the cost-to-go in the manner described in the previous section. Using that approximation and the previously described reformulation, the approximate Bellman equation given in Equation (3.8) can be solved using the math program given in **R**.

Algorithm 3.1 outputs a series of actions, or technician assignments, one for each state visited over the planning horizon. The algorithm takes as input the set of technicians, their learning parameters, and the set of task types. Line (3) initializes the state by setting each technicians experience on each task type to that technician's given initial experience.

Then, the algorithm seeks to determine the set of assignments for each technician for each day. At the beginning of each day, the day's service requests are observed (Line (5)). Then, Lines (6) to (13) enumerate the possible total forecasted service times for day $t + 1$. With these enumerated values, Line (14) solves the resulting instance of math program **R**.

Algorithm 3.1 Forward Approximate Dynamic Programming Algorithm

Output: Assignments of technicians over the planning horizon

Input: Technician $\mathcal{K}$, customer type $\mathcal{R}$, parameters of learning functions L_r^k, D_r^k, I_r^k for each technician k and each task type r

$q_{r1}^k = I_r^k, \forall r \in \mathcal{R}, \forall k \in \mathcal{K}$

for $t = 1$ **to** T **do**

  Observe $\mathcal{N}_t$

  for $r = 1$ **to** R **do**

    for $j = 1$ **to** m_{rt} **do**

      $\overline{q}_{r,t+1}^{kj} \leftarrow q_{rt}^k + j$

      $\overline{d}_{r,t+1}^{kfj} \leftarrow (D_r^k(\frac{\overline{q}_{r,t+1}^{kj}}{\overline{q}_{r,t+1}^{kj} + L_r^k}))^{-1}$

      $\overline{F}_{r,t+1}^{kj} = \alpha F_{r,t}^k + (1 - \alpha) j$

      $\overline{ST}_{r,t+1}^{kj} = \overline{F}_{r,t+1}^{kj} \overline{d}_{r,t+1}^{kfj}$

    end for

  end for

  Solve Model (**R**) resulting in the action a_t

  Update technician productivity based on experience gained on day t according to Equations (3.1) and (3.5)

end for

3.4 Experimental Design

In this section, we describe the instances that we create to computationally test the value of our solution approach. Our instances all involve one of 30 workforces of five technicians with heterogeneous learning parameters. In particular, the learning rate and steady state productivity rate are sampled from a multivariate normal distribution with the mean and covariance matrix derived from empirical data by Nembhard and Osothsilp (2005), given in Table 3.1.

$$\mu^T = \begin{array}{c} \begin{matrix} \log D & \log I & \log L & F \end{matrix} \\ \begin{pmatrix} 1.448 & 1.963 & 2.0446 & 2.269 \end{pmatrix} \end{array}$$

$$\Sigma = \begin{array}{c|cccc} & \log D & \log I & \log L & F \\ \hline \log D & 0.0045 & 0.0167 & 0.0191 & -0.0214 \\ \log I & 0.0167 & 0.4621 & 0.2443 & 0.2380 \\ \log L & 0.0191 & 0.2443 & 0.1905 & 0.2326 \\ F & -0.0214 & -0.2380 & -0.2326 & 0.3638 \end{array}$$

Table 3.1: Mean and covariance of workforce parameters

For each workforce, we consider four levels of capacity (C). These levels are seven, eight, nine and 10, where seven is the tightest capacity that ensures feasibility. These levels represent the maximum amount of time that a technician can work in a day. We also assume four different cardinalities for the set of task types $\mathcal{R}$. We refer to these levels as task diversity. We consider four levels of task diversity (R): five, 10, 15, and 20. With 30 workforces, four levels of workforce capacity, and four levels of task diversity we have 480 instances in total.

We test the 480 instances on 120-day horizons. To limit the number of factors in our experimental design, we assume 50 customer requests per day. For each of the 480 instances, we generate 100 trials of the 120-day horizon. To generate a trial, for each day, each of the 50 requests is randomly assigned a task type using a discrete uniform distribution.

We also conduct experiments regarding the two problem variants discussed above. For the first, we model a disruption in the workforce wherein one technician leaves the workforce and a new technician is hired as a replacement in the middle of the planning horizon. Specifically, Technician 1 is replaced with a new technician with initial experience level on all types of tasks on day 61. Specifically, on day 61, the transition function for Technician 1 is $q^1_{r,61}(s_{60}, a_{60}) = I^1_r$ for every r in $\mathcal{R}$.

For the second variant, we consider the situation in which a new task type is introduced in the middle of the planning horizon. We assume that all technicians are inexperienced on this new task type, and we set their experience levels on the new task type to be one on the day of the introduction (day 61). Specifically, on day 61, the transition function for the new task type is $q^k_{R+1,61}(s_{60}, a_{60}) = 1 \ \ \forall k \in \mathcal{K}$. In both the case of workforce disruption and the new task type, we use capacity levels of nine, 10, 11, and 12 to ensure the feasibility.

With these instances, we consider three separate experiments. In all three experiments, we seek to demonstrate the value of our method for approximating the future value of current decisions. In the first set of experiments, we consider the 480 instances described above and solve each with both the proposed ADP approach as well as an approach that approximates the Bellman equation by setting the second term of Equation (3.7) to 0. The resulting daily optimization problem can then be solved by solving a simplified version of the math program **R**. The second and third experiments are analogous to those just described except that we solve the workforce disruption and new product variants, respectively.

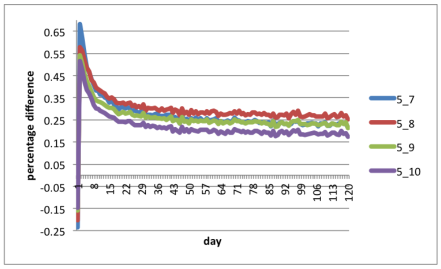
All computations are performed by Gurobi 5.6 with Python 2.7 interface on Intel Xeon processor running at CentOS 6.3. In addition, for all experiments, we set the smoothing factor (α) to $\alpha = 0.9$. This value was chosen after examining the results of various choices of α over many different instances.

3.5 Computational Analysis

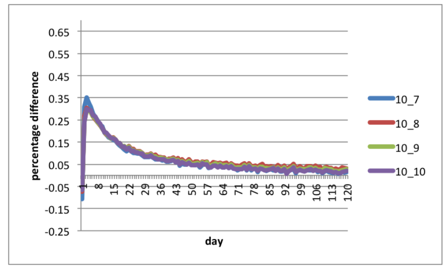
In this section, we present the results of our computational experiments. We do not report runtimes as the math program **R** solves nearly instantaneously and is thus capable of providing daily technician assignments in a timely fashion.

3.5.1 The Value of Incorporating Information about Future Requests

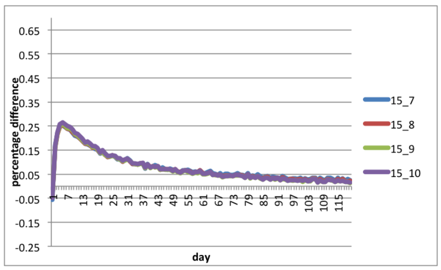
In this first experiment, we solve the 120-day MTSP-ESTSC with the ADP algorithm and the previously described myopic approach. Figures 3.1 and 3.2 present the results of the experiment. The figures present the average daily differences be-



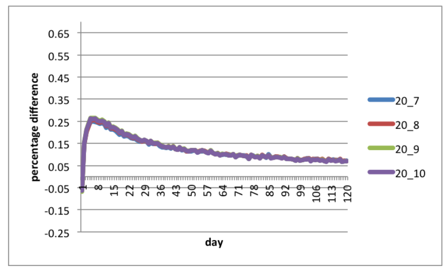
(a) Diversity=5



(b) Diversity=10



(c) Diversity=15



(d) Diversity=20

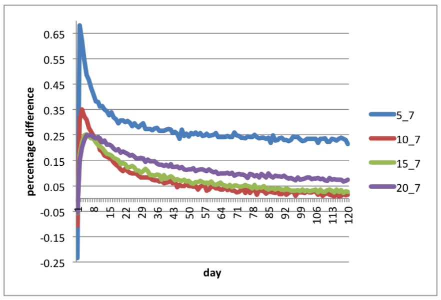
Figure 3.1: Average daily gaps by task diversity (%)

tween the two solution approaches, averaged over all capacity and diversity levels, respectively. Specifically, for both ADP and myopic approaches, we record the total service time for each day of the 120-day horizon for each trial (using common trials between the two approaches). We record these values as Obj_m^t and Obj_p^t for the myopic and ADP approaches, respectively. We then compute the objective gap on day t as $gap_{m\_p}^t = \frac{obj_m^t - obj_p^t}{obj_p^t} \times 100\%$ for each trial. For Figure 3.1, for each diversity level, the results are presented for each diversity level averaged over all capacity levels. Analogously, for Figure 3.2, the results are presented for each capacity level averaged over all diversity levels

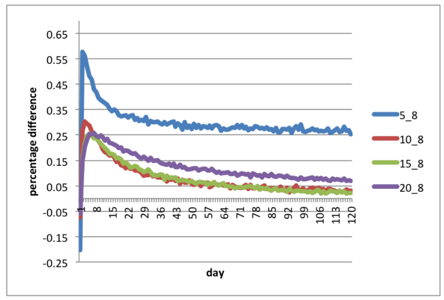
For all task diversities and all capacity levels, the average daily gap follows a similar trend. From Figures 3.1 and 3.2, we see a sharp increase from negative to positive gaps for the first three to eight days depending on the diversity levels. The myopic approach focuses on optimizing only current period service times while the ADP algorithm balances current and next periods so as to achieve the best assignment over both current and next periods. Then, the objective values on day one is not optimal with the ADP algorithm. In addition, there is no history information on day one, so the ADP approach has no advantage over the myopic approach. The above two reasons explain the negative gap on day one. As it accumulates more information over time, the ADP approach shows an increasing advantage, which is reflected by the sharp increase of the positive gaps between two approaches. According to learning curves, most learning effects occur during the early periods, which corresponds to the larger gaps between the myopic and ADP earlier in the horizon. Then, as the technicians accumulate more experience and approach the plateau of the learning curves, the average daily gaps decrease slowly to a relatively stable level.

Comparing the performance across the task diversities (Figure 3.1), we see sharper changes with lower task diversity and milder changes with higher task diversity. With lower task diversity, there are more tasks of each type, and experience accumulates much more quickly. Thus, learning occurs intensely over a short period. On the contrary, with higher task diversity, experience increases more slowly due to fewer tasks of each type.

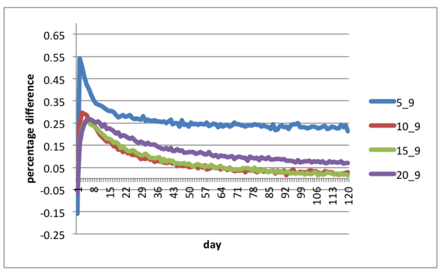
We also study the distribution of the average daily gap among instances. Table 3.2 shows the numbers of instances with positive, negative and zero average daily gap



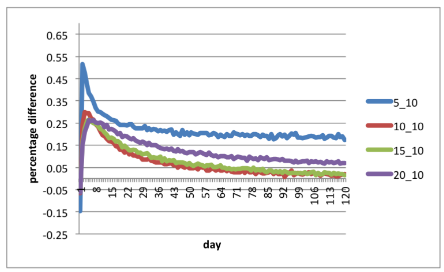
(a) Capacity=7



(b) Capacity=8



(c) Capacity=9



(d) Capacity=10

Figure 3.2: Average daily gaps by capacity (%)

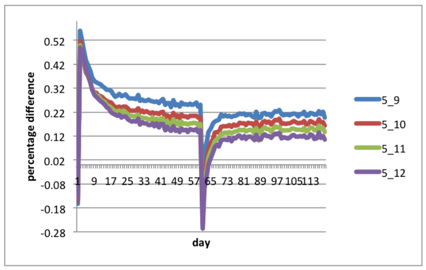
for all diversities and capacities, respectively. Results indicate that there are more instances with positive gap than those with negative gap and two approaches are more likely to perform the same with loose capacity. This result is not surprising as, when technicians are allowed to serve more customers per day, the faster learners and more experienced technicians are assigned to more customers. Thus, both approaches make similar assignments.

	Positive gap				Negative gap				Zero gap			
	R=5	R=10	R=15	R=20	R=5	R=10	R=15	R=20	R=5	R=10	R=15	R=20
C=7	1074	1055	926	1237	455	518	599	365	1471	1427	1475	1398
C=8	1104	907	793	1112	306	517	586	385	1590	1576	1621	1503
C=9	977	836	761	1065	348	511	572	352	1675	1653	1667	1583
C=10	773	792	760	1064	401	501	573	361	1826	1707	1703	1575

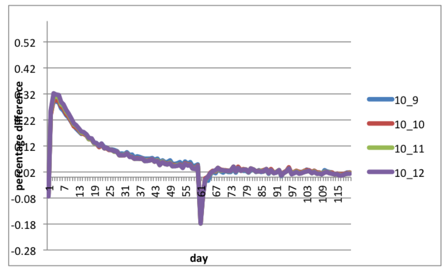
Table 3.2: Distribution of the average daily gaps

3.5.2 Responding to a Disruption in the Workforce

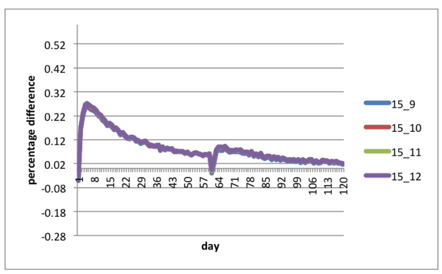
In this subsection, we present the results of the experiments designed to identify the impact of a disruption in the workforce. Figure 3.3 illustrates the changes of average daily gap by diversity level. The gaps are computed analogously to those presented in Figures 3.1 and 3.2. The general trends of the average daily gap changes are similar to those in the previous experiments, except for the temporary fluctuations on and after the day in which one technician departs and a new technician joins the workforce (day 61-70). The performance of the ADP approach after the disruption mimics its performance at the beginning of the planning horizon, where we see a transition from a negative to a positive daily gap as it positions this new technician for future demands by accepting longer service times in the near-term.



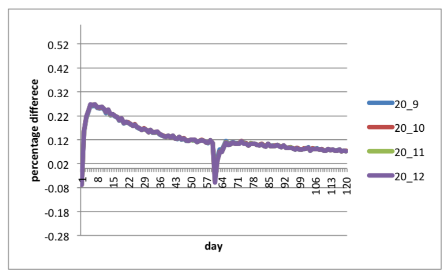
(a) Diversity=5



(b) Diversity=10



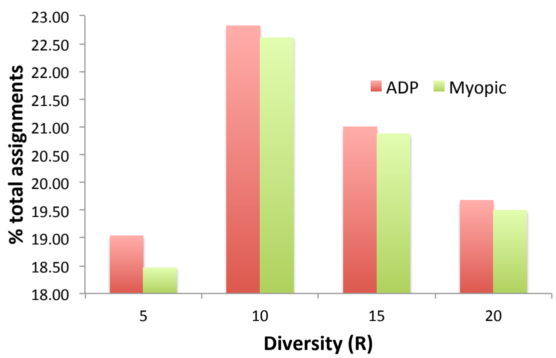
(c) Diversity=15



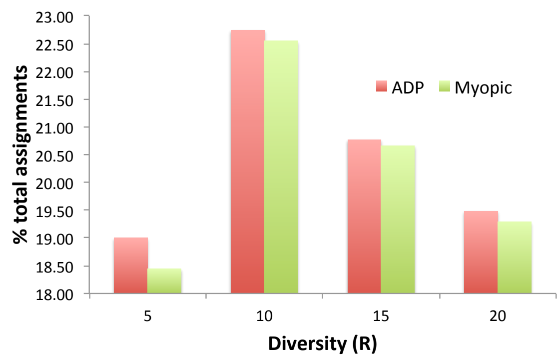
(d) Diversity=20

Figure 3.3: Average daily gaps by task diversity (%) with workforce fluctuation

Next, we study the new technician's workload (i.e. Technician 1 in our experiment). Figures 3.4(a) and 3.4(b) present the average workload of the new technician (as a percentage of the 50 tasks served) over the horizon (days 61-120) and on the day of disruption (day 61). We see that in both cases the ADP algorithm assigns more work to the new technician than the myopic approach does. Just as at the beginning of the planning horizon, the ADP algorithm is positioning the new technician to have sufficient experience for customer requests on future days, something the myopic approach ignores. In fact, in some cases the new technician does more than the average (20%) of the work. We hypothesize that in these cases the ADP approach is aggressively building up the experience level of the new technician.



(a) Over horizon



(b) On day of disruption

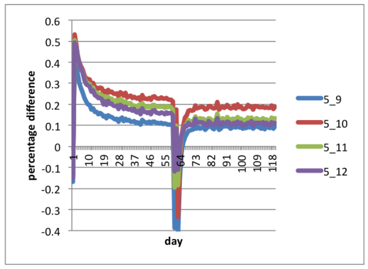
Figure 3.4: Workload of new technician

We also note that the lower the task diversity the larger the negative gaps when introducing the new technician (see the deep plunge on day 61). With a lower task diversity there are more customers for each task type, which in turn leads to technicians rapidly accumulating enough experience to operate at their maximum productivity. As such, there is a greater difference in the service times between the new technician and the existing workforce on the day the new technician begins.

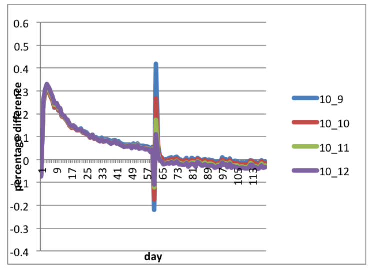
Yet, as discussed above, the ADP approach assigns more work to the new technician than the myopic approach to better prepare the new technician for future customer requests. This impact of diversity is most apparent at the lowest level (R=5). Here we see the greatest difference in the workload assigned to the new technician as the myopic approach is focused solely on leveraging the productivity of the existing, highly-experienced technicians.

3.5.3 Responding to a New Task Type

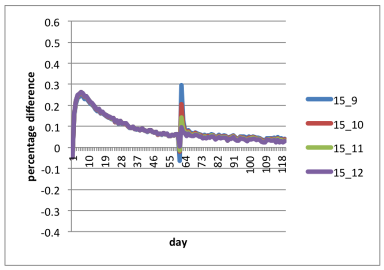
We next present results for the experiments that examine the impact of introducing a new task. Figure 3.5 illustrates average daily gaps by day and diversity. The gaps are computed analogously to those in Section 3.5.1. We notice that the general trends of the average daily gap remain the same as seen in Section 3.5.2. As with attrition, after a new task type is introduced the ADP algorithm accepts longer service times to better position the workforce to handle customer requests for the new task type in the future.



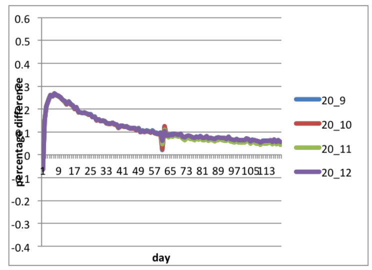
(a) Diversity=5



(b) Diversity=10



(c) Diversity=15



(d) Diversity=20

Figure 3.5: Average daily gaps by task diversity (%) with a new task type

We next examine the distribution of experience in tasks of the new type for each approach (the myopic and the ADP). Specifically, for each workforce and each instance, we calculate the percentage of times tasks of the new type are performed by the technician who performs them the most often (who we will call the most frequent technician). Similarly, we calculate the percentage of times tasks of the new type are performed by the technician who performs them the second most often (and third most, fourth most, fifth most). We report in Figure 3.6 the percentage of times for each approach and diversity level tasks of the new type are performed by the most frequent technician. As these new type tasks are very rarely (e.g. $< .65\%$ of the time) performed by the technician who performs it the third (or fourth or fifth) most often, one can reasonably conclude that when the most frequent technician does not perform a task of the new type it is performed by the second most frequent technician.

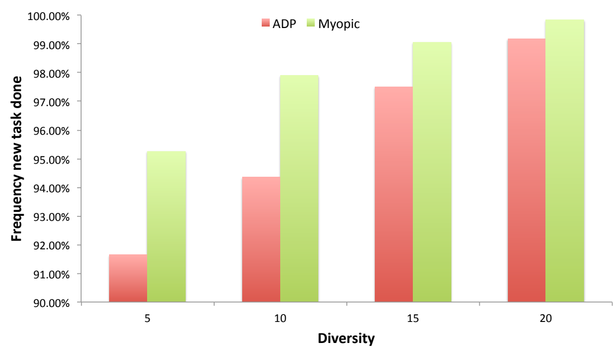


Figure 3.6: How often most frequent technician performs tasks of new type

What we observe in Figure 3.6 is that both approaches favor having one individual specialize on tasks of the new type and that specialization is directly correlated to the diversity of the instance (i.e. the number of different task types). However, the ADP approach also builds proficiency in the new task type in a second technician and does so more often when diversity is low. Similar to the discussion above, we hypothesize this is because at lower diversity levels technicians can quickly develop proficiency in the original task types and thus it is less impactful to divert two of them towards building proficiency in the new task type.

3.6 Conclusions

In this chapter, we study the scheduling problem of assigning tasks to technicians wherein the tasks to perform each day are not known until the day they are to be done. Such a problem can be modeled as a daily, task assignment problem. However, we also model that technicians learn; i.e. service times decrease as technicians gain experience performing tasks. As such, we model the problem as a multi-period problem wherein each day's decisions are made with an eye to building capacity (via experience accumulation) for the future. This problem, which we call the multi-period technician scheduling problem with experience-based service times and stochastic customers is new to the literature. We present a Markov decision process model for the problem and discuss how it can be adapted to disruptions in the workforce and the introduction of new task types to service.

We present an approximate dynamic programming-based solution approach for this model that repeatedly solves optimization problems to make daily assignments that recognize the impact of technicians gaining experience on future service times. The optimization problem approximates the cost-to-go with forecasts of the next day's assignments for each technician and the resulting estimated time it will take to service those assignments given the current day's decisions. As quantitative models of human learning are non-linear we use a technique from the literature to reformulate these optimization problems to mixed integer programs which can then be solved (nearly) instantaneously.

We demonstrate the value of both the model and the solution approach with an extensive computational study. We benchmark the performance of the model and ADP-based solution approach against a myopic approach that models the problem as a single-period problem, ignoring the impact of experience gained today on service times on later days. We see that across all variants (the baseline problem, disruption in workforce, and introduction of new task types) the ADP-based approach outperforms the myopic approach by sacrificing solution quality in a few near-term periods in order to better position the workforce for future customer service requests.

CHAPTER 4
APPROXIMATE DYNAMIC PROGRAMMING APPROACH WITH BASIS FUNCTION

4.1 Introduction

The proposed cost-to-go approximation in Chapter 3 estimates the value of experience accumulation only one day into the future. Motivated by the promising results, we propose a different approximate dynamic programming method for the Multi-Period Technician Scheduling with Experience-based Service Times and Stochastic Customers. Instead of the lookahead one-day, we use a parameterized linear model to approximate the value function of accumulated experience. The linear model uses a basis function that maps the state into a small number of features. Importantly, the basis function reduces the large number of state variables into a smaller number. Our basis function value function approximation reflects essential information in the state variable, such as each technician's aggregate experience levels over all task types. To achieve a good approximation of the cost-to-go, we are facing two challenges. The first important step is to construct a set of basis functions that reflect important elements of the state variable and capture essential information about the cost-to-go. The second step is the recursive update mechanism of the parameter of the basis function. The parameters of the basis function are determined using an offline simulation proposed by Powell (2011) and it is analogous to estimating the parameters in a linear regression. With the new method, we are able to take advantage of future information over the whole planning horizon.

In this study, we make the following research contributions. First, we develop a tractable ADP approach that uses a parameterized linear model to approximate the value function for the Multi-Period Technician Scheduling with Experience-based Service Times and Stochastic Customers. Our approach is integrating future information over the whole planning horizon into current period decision-making. Second, we provide computational experiments illustrating the value of integrating more future information. We compare the proposed ADP solution approach to the myopic solution approach. Our comparisons demonstrate that the ADP approach with linear basis function outperforms the myopic solution approach for most of the instances.

The remainder of this chapter is organized as follows. Section 4.2 reviews related literature in the field of approximate dynamic programming. Section 4.3 describes the solution approach. Section 4.4 discusses the design of the experiments and presents our computational results. Finally, Section 4.5 concludes this work.

4.2 Literature Review

In the past decade, there has been an increasing interest in approximate dynamic programming. Sutton and Barto (1998) and Powell (2011) provide an introduction of this field and a review of the literature.

The most popular strategy adopted for value function approximation is via linear basis functions. There are two major challenges in this method. First, we need to construct a set of basis functions that capture important quantities from the state variable. Because the basis functions are highly problem-specific, choosing the right set of basis functions is more of an art than science. For example, Maxwell et al. (2010) study an ambulance redeployment problem in an emergency medical service system. They approximate the value function as a set of basis functions including unreachable calls, uncovered call rate, missed call rate, future uncovered call rate, and future missed call rate. Once a set of basis functions is chosen, a number of analytical methods can be used to tune the parameter. Recursive least squares is a valuable tool often used in iteratively updating functional approximation. Lagoudakis et al. (2002) and Bradtke and Barto (1996) present least squares methods in the context of reinforcement learning. Nedić and Bertsekas (2003) describes the use of least squares equation with a linear value function approximation using policy iteration. We also use this method in our work to update our value function approximations. Other researchers including Schweitzer and Seidmann (1985), de Farias and Van Roy (2003), Adelman and Mersereau (2008) use linear programming approach to get the good estimate of parameters.

Approximate dynamic programming has been widely applied in the fields of inventory management (Godfrey and Powell, 2001), option pricing (Tsitsiklis and Van Roy, 2001), network revenue management (Adelman, 2007, Zhang and Adelman, 2009), game playing (Yan et al., 2004), and vehicle routing (Novoa and Storer, 2009, Kleywegt et al., 2004). We are the first to solve the multi-period dynamic technician scheduling problem with learning using the value function approximation. The most closely related literature is the ADP's applications in assignment problem and vehicle routing problem, because both of them involve the scheduling component.

Godfrey and Powell (2002) introduce an adaptive dynamic programming algorithm that depends on estimating separable nonlinear approximations of value functions for stochastic dynamic resource allocation problems. Spivey and Powell (2004) propose an adaptive, nonmyopic algorithm that involves linear value function approximations and iteratively solving sequences of assignment problems to solve dynamic assignment problems. Topaloglu and Powell (2006) consider a stochastic and time-dependent version of the min-cost integer multicommodity-flow problem that arises in the dynamic resource allocation context. They propose an iterative, adaptive dynamic-programming-based methodology that makes use of linear or nonlinear approximations of the value function. He et al. (2012) also use the piecewise linear approximation ADP algorithm in their dosage control problem in the COH treatment to overcome the curses of dimensionality in Markov decision processes. Their numerical experiments indicate that the piecewise linear approximation ADP algorithms can obtain policies that are very close to the one obtained by the MDP benchmark with significantly less solution time. Papageorgiou et al. (2014) develop an ADP framework for generating good solutions quickly to a class of maritime inventory routing problem with a long planning horizon. Their algorithm solves MIP subproblems and uses separable piecewise linear continuous value function approximations and requires no off-line training.

One major difference between our work and the aforementioned literature is that the general shape of their value functions is usually known. Thus, the specific linear or nonlinear value function approximation methods are quite effective. However, the assumption of the special structure does not apply to our value function. In addition, our previous testing results demonstrate that the piecewise linear approximation does not work for our problem. This is also the challenge of our problem, because it is difficult to accurately choose an appropriate form of value function to approximate.

4.3 Solution Methodology

As mentioned in the previous section, most literature on value function approximation is based on the special structure (convexity or concavity) of the value function. Special structure is beneficial because it is easy to find an appropriate form of the value function approximation. However, this assumption does not apply to our value function, we can only assume the general form of our value function. In this section, we first use three counter examples to show that the value function of our problem is not convex in general either on an individual level or on aggregate level. Then, we discuss the detail of the basis function.

4.3.1 Counter Examples

Because a technician's service time per task decreases as the technician gains experience, it is intuitive to think that a technician's total service time will decrease as a technician's experience increases. However, with the following examples, we will show that the value function is not necessarily monotonically decreasing either on an individual level or on aggregate level. That is, the total service times is not decreasing as the experience level increases.

It is helpful if we can minimize the number of parameters that we need to tune during the value function approximation. Thus, we would prefer to aggregate experience levels over all types of tasks and over all technicians. Example 4.3.1 shows that the total service time of all technicians on all task types is not necessarily decreasing on total experience of all technicians on all task types.

Example 4.3.1. *Suppose we have two technicians and two tasks. The initial setting of the learning parameters are described as follows:*

	L		D	
	Task 1	Task 2	Task 1	Task 2
Technician 1	1	1	1	0.5
Technician 2	1	1	1	0.5

Table 4.1: Initial setting of the learning parameters

where L is the learning rate and D is the steady state productivity.

We view the total service times of all technicians on all task types as a function of the total experience of all technicians on all task types. Consider the current state of the technicians' experience level on each task type summarized in Table 4.2:

	Task 1	Task 2	Total ST
Technician 1	10	3	1.1
Technician 2	5	9	2.22

Table 4.2: Current state

where column 2 and 3 demonstrate each technician's experience level on every task type and the last column (Total ST) represents each technician's total service times on all task types.

Then, the total experience level for all technicians on all task types is 27 (=10+3+5+9). The optimal solution is to assign task 1 to technician 1 and assign task 2 to technician 2 with the total service times of 3.32 (=1.1+2.22).

Then, consider a different state summarized in Table 4.3:

	Task 1	Task 2	Total ST
Technician 1	9	7	1.11
Technician 2	7	8	2.25

Table 4.3: State with experience increase

Now, the total experience level for all technicians on all task types is 31. The optimal solution is to assign task 1 to technician 1 and assign task 2 to technician 2 with the total service times of 3.36 . We can conclude that the total service time for all technicians on all task types is not necessarily decreasing on total experience levels.

Then we consider the case that, for each technician, the total service time on all task types is a function of each technician's total experience on all task types. Example 4.3.2 shows that for individual technician, the total service time on all task types is not necessarily decreasing on his/her total experience over all task types.

Example 4.3.2. *Assume the same setting in Example (4.3.1). Consider the current state summarized in Table 4.4:*

	Task 1	Task 2	Total exp	Total ST
Technician 1	10	3	13	1.1
Technician 2	5	4	9	2.5

Table 4.4: Current state

where column 2 and 3 demonstrate each technician's experience level on every task type, and column 4 (Total exp) and 5 (Total ST) represent each technician's total experience levels and total service times on all task types, respectively.

Then, the optimal solution is to assign task 1 to technician 1 and assign task 2 to technician 2 with the total service times of 1.1 and 2.5 for technician 1 and 2, respectively.

If we increase technician 1's experience on task 2 by 2 units, then the new state is summarized in Table 4.5:

	Task 1	Task 2	Total exp	Total ST
Technician 1	10	5	15	3.5
Technician 2	5	4	9	0

Table 4.5: State with experience increase

Now, the optimal solution is to assign tasks 1 and 2 to technician 1 with the total service times of 3.5 and 0 for technician 1 and 2, respectively. Thus, we show that the total service time of an individual technician is not necessarily decreasing on his/her experience level.

Finally, example 4.3.3 shows that, for each task type, the total service time for all technicians is not necessarily decreasing on total experience levels of all technicians.

Example 4.3.3. *Assume the same setting as in Example (4.3.1). For each task type, we view the total service times of all technicians as a function of the total experience of technicians on this task type. Consider the current state summarized in Table 4.6:*

	Technician1	Technician 2	Total Exp	Total ST
Task 1	10	5	15	1.1
Task 2	3	9	12	2.22

Table 4.6: Current state

Then, the optimal solution is to assign task 1 to technician 1 and assign task 2 to technician 2 with the total service times of 1.1 and 2.22 for task 1 and 2, respectively.

Then, we change both technicians' experience with the new state summarized in Table 4.7:

	Technician 1	Technician 2	Total Exp	Total ST
Task 1	9	7	16	1.11
Task 2	7	8	15	2.25

Table 4.7: State with experience increase

Now, the optimal solution is to assign task 1 to technician 1 and assign task 2 to technician 2 with the total service times of 1.11 and 2.25 for tasks 1 and 2, respectively. In this example, although both technicians gain experience, it is the distribution of the experience among different types of task that causes the total service time to increase.

We can conclude that, for each task type, the total service time for all technicians is not necessarily decreasing on total experience levels of all technicians.

The above counter examples show that the value function is not convex in general on technicians' experience either on an individual level or on aggregate level. Therefore, we assume the value function as a general form.

4.3.2 Value Function Approximation

From previous discussion, we notice that the value function in our problem is not convex. However, assuming certain form of the value function will take advantage of the information from state variables and help us better approximate the value function. More importantly, in our problem, for any time period and any technician, we do not need to approximate the cost-to-go for every experience level because each technician's experience level will be only in a range of values and a huge portion of the state space will not be visited (Ulmer et al., 2014). Our choices should lead us to be in a specific and continuous range of experience levels. Over that range, a linear approximation would be valid. Our way of constructing the basis function, as illustrated in detail in Section 4.3.2.1, takes advantage of this special character of our problem. Thus, we choose a basic adaptation of approximate value iteration using a linear basis function to approximate our value function. The strategy is popular because it is easy to implement and the basis function will reduce potentially large number state variables into a small number of features.

We propose a dynamic programming methodology that uses a linear model with basis function $\hat{V}_t$ to approximate the value function V_t. We consider the value function approximation of the following form:

$$\hat{V}_t(s_t) = \sum_{j=1}^{J} \theta_{jt}\phi_{jt}(s_t), \forall t \in \mathcal{T}, \tag{4.1}$$

where each $\phi_{jt}(s_t)$ is a basis function that maps a state $s_t \in S$ to a real number and each θ_{jt} is a real-valued weight, or parameter, associated with a basis function. Then equation (3.7) can be written as

$$V(s_t) = \min_{a_t \in \mathcal{A}_t(s_t)} \{C(s_t, a_t) + \sum_{j=1}^{J} \theta_{jt}\phi_{jt}(s_t, a_t)\}. \tag{4.2}$$

To achieve a good approximation of the cost-to-go, the second term in Equation (3.7), we are facing two challenges. The first important step is to construct a set of basis functions that reflect important elements of the state variable and capture essential information about the cost-to-go. We discuss our choice of basis functions in section 4.3.2.1. The second step is the recursive update mechanism of the parameter θ. In section 4.3.2.2, we describe an offline simulation procedure to determine parameters $\theta_{1t}, \ldots, \theta_{Jt}$.

4.3.2.1 Approximating the Value Function

In our problem, each technician's experience level is the most important component because it depends on the assignment from previous periods and it also impacts the assignment in the future periods. It is the connection between current period and future. Thus, connecting the cost-to-go with the information of technicians' experience level is fundamental in our value function approximation. Therefore, at each decision epoch t, we define the basis functions $\phi_{1t}(\cdot), \ldots, \phi_{Jt}(\cdot)$, each of which maps a state $s_t = \{\mathcal{N}_t, \mathcal{Q}_t\}$ to a real number, based on each technician's total experience levels on all types of tasks by the end of the day. For any state $s_t \in \mathcal{S}$, let $\phi_{1t}(s_t) = 1$. Then, θ_{1t} serves as a location parameter. At each decision epoch t, for each technician k, we construct a basis function that reflects the total experience levels of this technician on all types of tasks:

$$\phi_{1+k,t} = \sum_{r \in \mathcal{R}} q^k_{r,t+1}, \forall k \in \mathcal{K}, \forall t \in \mathcal{T}, \tag{4.3}$$

where $q^k_{r,t+1}$ is the experience level of technician k on type r task by the end of day t. Then (3.7) can be written as

$$
\begin{aligned}
(\mathbf{P})\quad \tilde{V}^n_t(\mathcal{Q}^n_t, \mathcal{N}^n_t) = \min_{a_t \in \delta^\pi_t(s_t)} & \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{N}_t} x^k_{it} d^k_{r(i)t} + \sum_{k \in \mathcal{K}} \theta^{n-1}_{k+1,t} \sum_{r \in \mathcal{R}} q^k_{r,t+1} + \theta^{n-1}_{1t} \\
\text{s.t.} \quad & (3.2), (3.3), (3.4), \\
& \sum_{r \in \mathcal{R}} q^k_{r,t+1} = \sum_{r \in \mathcal{R}} q^k_{rt} + \sum_{i \in \mathcal{N}_t} x^k_{it}, \forall k \in \mathcal{K}. \qquad (4.4)
\end{aligned}
$$

where constraints (4.4) state the transition from the pre-decision state to the post-decision state.

4.3.2.2 Offline Simulation Procedure

Once we have a set of basis functions, our next challenge is to devise a method to determine the parameters of the basis functions. We turn to an offline simulation discussed in Powell (2011). The simulation is a double pass version of approximate value iteration. Algorithm 4.1 summarizes the proposed solution approach, which consists of a forward pass, a backward pass, and a recursive updating procedure. In Step 1, we initialize the parameters of the basis function θ^0. Then, Step 2 describes the forward pass. Given a request realization, starting from the initial state (Step 2(a)), the forward pass creates a trajectory of states, actions, and outcomes using the current parameter (θ^n) and the MDP model (Step 2(b) and 2(c)). The forward pass steps through the whole planning horizon. Then, in Step 3, we step backward through time, updating the cost-to-go of being in a state using information from the same trajectory in the future. Finally, in Step 4, a recursive updating procedure (U) follows the forward and backward pass. In this procedure, we recursively update the parameters of the basis functions using the information of the current parameters, the post-decision states visited in the current iteration, and the calculated cost-to-go. We perform a total of N iterations. We will discuss the detail of the updating procedure in section 4.3.2.3.

We tested two methods to initialize θ^0. The first method is to set each element of θ^0 to be zero, which is the common practice in the literature (Powell, 2011). The second method samples the features-value pairs by solving the problem with the myopic algorithm and uses regression to get the initial values of θ^0. Testing results show that the second method provide better solutions. Thus, we use the regression on the myopic algorithm solutions to initialize θ^0.

We choose the basis function based on the post-decision state variables because it provides tremendous computational advantages. Using the post-decision state variable avoids the need to approximate the expectation explicitly within the optimization problem (Powell, 2011).

4.3.2.3 Recursive Least Squares Regression

During the forward and backward pass, the simulation uses the Monte Carlo samples of the service requests to update the value function approximation. Suppose we are at iteration n of Algorithm 4.1. $\{\hat{\mathcal{N}}_t^n, t \in \mathcal{T}\}$ is the sequence of request realizations and $\{\hat{V}_t^n, t \in \mathcal{T}\}$ is the sequence of value-function approximations. $\{\mathcal{Q}_t^n, t \in \mathcal{T}\}$ is the sequence of system states generated by solving the following problems with the

Algorithm 4.1 The Approximate Dynamic Programming Algorithm

1. Initialization: Initialize θ^0 . Set iteration counter $n = 1$.
2. Forward Pass:
   (a) Initialize the formal pass: initialize $\mathcal{Q}_1^n$ to the initial state of the experience of the technicians at the beginning of day 1. Obtain a sample realization of $\{\mathcal{N}_t : t \in \mathcal{T}\}$, say $\{\hat{\mathcal{N}_t^n} : t \in \mathcal{T}\}$.
   (b) Solve the subproblem: for time period t solve model (**P**) to get $x_{it}^k, \forall k \in \mathcal{K}, \forall i \in \mathcal{N}_t$.
   (c) Apply the system dynamics: set
   $$q_{r,t+1}^k(s_t, a_t) = q_{rt}^k + \sum_{i \in \mathcal{N}_t^n : r(i) = r} x_{it}^k \forall k \in \mathcal{K}, \forall r \in \mathcal{R}.$$
   (d) Advance time: set $t = t + 1$. If $t \in \mathcal{T}$, go to Step 2(b).
3. Backward Pass:
   (a) Initialize the backward pass: initialize $\hat{v}_{T+1}^n = 0$.
   (b) Value function update: for time period t set $\hat{v}_t^n = C(s_t, a_t) + \hat{v}_{t+1}^n$
4. Recursive least square update: $\theta^n = U(\theta^{n-1}, \hat{v}^n, s^a)$
5. Advance iteration counter: set $n = n + 1$. If $n \leq N$, go to Step 2.

current value-function approximations and request realizations

$$\tilde{V}_t^n(\mathcal{Q}_t^n, \hat{\mathcal{N}}_t^n) = \min_{a_t \in \delta_t^{\pi}(s_t)} \{C(s_t, a_t) + \hat{V}_{t+1}^{n-1}(\mathcal{Q}_{t+1})\}. \tag{4.5}$$

Then, we can get a features-value pair. For each post-decision state, we can get a set of features in the basis functions and a corresponding estimated objective value. Consider those features as the independent variables and the estimated objective value as the dependent variable, we can relate our offline simulation procedure to the parameter estimation in a linear regression.

We design our recursive least squares regression according to Powell (2011). The goal is to get a set of parameters that minimize the squared errors of the states traversed through the current iteration. Powell (2011) offers a way to use the estimates $\hat{V}^n$ to estimate a value function approximation. The following formulas can be used recursively to update the coefficient vector θ. The updating equation for θ is

$$\theta^n = \theta^{n-1} - H^n \phi^n \hat{\epsilon}^n, \tag{4.6}$$

where H^n is a matrix computed using

$$H^n = \frac{1}{\gamma^n} B^{n-1}. \tag{4.7}$$

The error $\hat{\epsilon}^n$ is computed using

$$\hat{\epsilon}^n = \tilde{V}_s(\theta^{n-1}) - \hat{v}^n, \tag{4.8}$$

and B^{n-1} is updated recursively using

$$B^n = B^{n-1} - \frac{1}{\gamma^n}(B^{n-1}\phi^n(\phi^n)^T B^{n-1}), \tag{4.9}$$

where γ^n is a scalar computed using

$$\gamma^n = 1 + (\phi^n)^T B^{n-1} \phi^n. \tag{4.10}$$

One strategy to initialize B^n is to use $B^0 = \lambda I$, where I is the identity matrix and λ is a "small constant" (Powell, 2011). In our computational experiments, we tested various values of λ and finally choose $\lambda = 0.9$ because it gives the best results.

4.4 Preliminary Results

In this section, we describe the instances that we create to computationally test the value of our solution approach. We follow a similar experimental design to what we did in Chapter 3. The difference between the design in this chapter and that in Chapter 3 is the choice of the task diversity and the capacity levels. Because we generate a new set of 30 workforces with the heterogeneous learning parameters described in Chapter 3, we change the task diversity and the capacity accordingly.

For each workforce, we consider four levels of task diversity (R): five, seven, 10, and 13. We consider four levels of capacity (C). These levels are five, six, seven, and eight, where five is the tightest capacity that ensures feasibility. These levels represent the maximum amount of time that a technician can work in a day. The 30 workforces, four levels of capacity, and four levels of task diversity result in 480 instances.

We test the 480 instances on 120-day horizons. To limit the number of factors in our experimental design, we assume 50 customer requests per day. For each day, each of the 50 requests is randomly assigned a task type, and we generate the number of each task type according to the discrete uniform distribution on each day.

In our experiments, we solve the 120-day MTSP-ESTSC with the approximate dynamic programming algorithm and the myopic approach. In detail, testing our ADP solution approach involves two stages. In the training stage, we implement Algorithm 4.1 to tune the value-function approximation. By the end of this stage, we get a set of parameters of the basis functions. Then, in the testing stage, we fix the value-function approximation with the parameters provided by the training stage and test the solution quality for the current set of value-function approximations for different customer realizations. In our experiment, we use 1000 training iterations ($N = 1000$) and 1000 testing iterations. For comparison, we also solve the problem of the same group of customer realizations used in the testing stage for the ADP approach with the myopic approach.

Figures 4.1 to 4.4 present the results of the experiment. Similar to what we did in Chapter 3, the figures present the average daily differences between the two solution approaches by diversity levels. Specifically, for both the approximate dynamic programming and myopic approaches, we record the total service time for each day of the 120-day horizon for each trial (using common trials between the two approaches). We record these values as Obj_m^t and Obj_p^t for the myopic and approximate dynamic programming approaches, respectively. We then compute the objective gap on day t as $gap_{m\_p}^t = \frac{obj_m^t - obj_p^t}{obj_p^t} \times 100\%$ for each trial.

Figures 4.1 to 4.4 show that the changes of the daily objective gap follow the similar trend across diversities and capacity levels. During the early periods of the

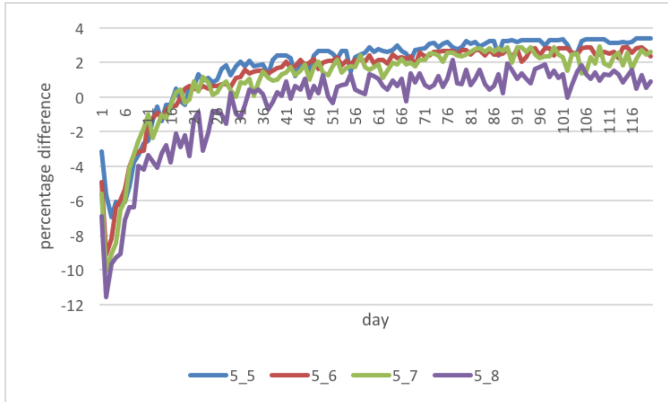


Figure 4.1: Average daily gaps for Diversity=5 (%)

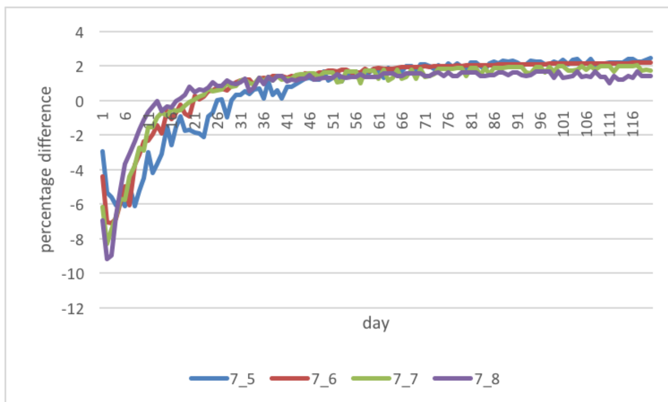


Figure 4.2: Average daily gaps for Diversity=7 (%)

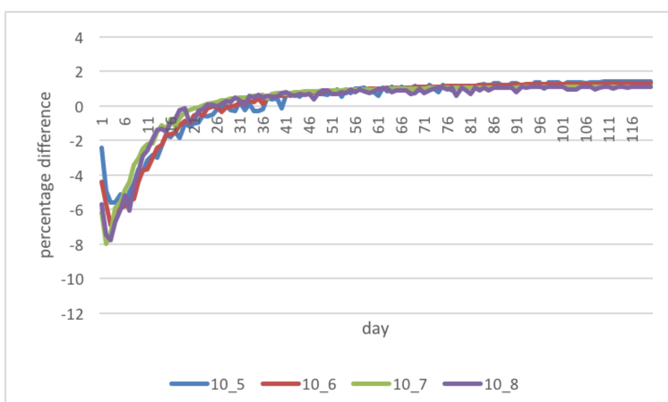


Figure 4.3: Average daily gaps for Diversity=10 (%)

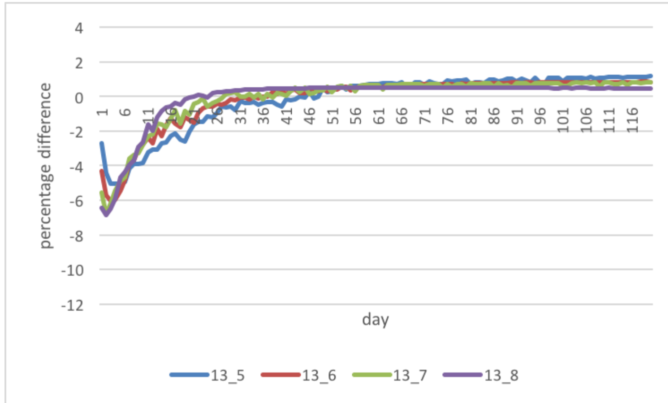


Figure 4.4: Average daily gaps for Diversity=13 (%)

planning horizon, the ADP performs worse than the myopic approach. The daily gap turns positive between day 20 to day 30, depending on the customer diversities, and keeps increasing to the end of the planning horizon.

One major reason for the performance changes of the ADP solution approach over the planning horizon is the trade-off between the current period cost and future period benefit. Throughout the planning horizon, the myopic approach seeks to assign each task to the technician that can perform it the quickest. However, the ADP algorithm will build its workforce so as to get the best solution for the whole planning horizon. The myopic approach focuses on optimizing only current period service times while the ADP algorithm balances current and future periods so as to achieve the best assignment over the whole planning horizon. Then, the objective values on early periods is not optimal with the ADP algorithm. For the ADP solution approach, we can consider the early periods as the preparation stage. In this stage, the ADP algorithm is trading-off current periods cost for future benefit. In order to achieve the best assignment over the whole planning horizon, the ADP algorithm needs to position its workforce for future demands by accepting longer service times in the near-term. Therefore, we see the worse performance of the ADP algorithm during the first 20 to 30 days. Then, once the workforce builds up sufficient experience level for future demands, the ADP algorithm demonstrates its advantage over the myopic algorithm and enjoys this benefit till the end of the planning horizon. The sacrifice it made during the preparation stage now pays off. However, the myopic algorithm, only focusing on current periods, cannot prepare its workforce for future demands. With its insufficient workforce, the myopic algorithm cannot handle the uncertain future demands well.

In addition, we have following observations similar to what we have seen in Chapter 3. First, comparing the performance of the ADP algorithm and the myopic approach across task diversity levels, we see that the benefits associated with looking into the future when performing task assignments are declining with the higher diversity level, which indicates the worse performance of the ADP algorithm with the higher diversity level. We hypothesize that there are two reasons for this behavior: (1) with a lower task diversity level the ADP algorithm is more accurate as there is more samples for each task type, and, (2) with a lower task diversity level, as there are many tasks of each type, we are able to fully leverage the benefits of learning more quickly by looking into the future.

Second, comparing the performance of the ADP algorithm and the myopic approach across capacities, we see that the benefits associated with looking into the future when performing task assignments are declining with the larger capacity, which indicates the worse performance of the ADP algorithm with the larger capacity. An examination of the results suggests that the myopic approach seeks to assign each task to the technician that can perform it the quickest. However, the ability of the approach to do so is especially limited when capacity is tight. We hypothesize that, by looking into the future, the ADP algorithm recognizes the value of developing the proficiency level of multiple technicians so that many of them are able to perform each task quickly. As a result, as it positions the workforce so that it can assign tasks to different technicians that have the same proficiency, the ADP algorithm outperforms the myopic approach the most at lower capacity levels.

4.5 Conclusions

In this chapter, we continue exploring the value of integrating future information into the current period decision-making process for the Multi-period Dynamic Technician Scheduling Problems with Experience-based Service Times and Stochastic Customers discussed in Chapter 3. We propose an alternate approximate dynamic programming solution approach with basis function to approximate the value function by taking the advantage of the future information for the whole planning horizon. At each decision epoch, we choose each technician's aggregate experience levels on all task types as the features in the basis function. Then, we turn to an offline simulation procedure to recursively update the coefficient vector of the basis function, which allows fast decision making within the execution phase.

Our computational results demonstrate the value of the ADP solution approach with the basis function. We benchmark the performance of the ADP solution approach with the basis function against a myopic approach that converts the multi-period problem into a series of single-period problems, ignoring the impact of future information on the decision making in the current period.

CHAPTER 5
CONCLUSIONS AND FUTURE WORK

5.1 Conclusions

In this work, we study the technician routing and scheduling problems in which technicians learn and explore the issue of how companies can use immediate employee job assignments to meet current demand and build capacity for the future. We explicitly model the impact of individualized, experience-based learning on technician routing and scheduling problems. Facing with the increases in product and service diversity and accelerating product and service innovation, businesses need greater agility and flexibility in their workforce management decisions (Pine, 1993). In this changing environment, how an individual is expected to respond to change becomes as important as how efficiently they are performing a current assignment (Nembhard, 2001). Due to individuals' different learning traits, workforce management based on individualized, experience-based learning has significant potential to improve productivity in organizations. Thus, our models will facilitate organizational productivity improvement by allowing for a more efficient workforce management.

In Chapter 2, we present a model and approach for accounting for on-the-job learning in the daily routing of technicians. Our objective minimizes the completion time of the last task. We solve the daily routing problems using a modified version of the record-to-record travel (RTR) heuristic. Our results offer the following key insights:

1. Explicitly modeling both learning and technician heterogeneity leads to better and different solutions in comparison to assuming homogeneous learning curves and/or static productivity.
2. Relatedly, the explicitly modeling both learning and technician heterogeneity leads to different distributions of skills in comparison to assuming homogeneous learning curves and/or static productivity.
3. Inexperienced technicians specialize the most and experienced technicians the least.

In Chapter 3, we study the scheduling problem of assigning tasks to technicians wherein the tasks to perform each day are not known until the day they are to be done. Such a problem can be modeled as a daily, task assignment problem. However, we also model that technicians learn; i.e. service times decrease as technicians gain experience performing tasks. As such, we model the problem as a multi-period problem wherein each day's decisions are made with an eye to building capacity (via experience accumulation) for the future. This problem, which we call the multi-period technician scheduling problem with experience-based service times and stochastic customers is new to the literature. We present a Markov decision process model for the problem and discuss how it can be adapted to disruptions in the workforce and the introduction of new task types to service.

We present an approximate dynamic programming-based solution approach for this model that repeatedly solves optimization problems to make daily assignments that recognize the impact of technicians gaining experience on future service times.

The optimization problem approximates the cost-to-go with forecasts of the next day's assignments for each technician and the resulting estimated time it will take to service those assignments given the current day's decisions. As quantitative models of human learning are non-linear we use a technique from the literature to reformulate these optimization problems to mixed integer programs which can then be solved (nearly) instantaneously.

We demonstrate the value of both the model and the solution approach with an extensive computational study. We benchmark the performance of the model and ADP-based solution approach against a myopic approach that models the problem as a single-period problem, ignoring the impact of experience gained today on service times on later days. We see that across all variants (the baseline problem, disruption in workforce, and introduction of new task types) the ADP-based approach outperforms the myopic approach by sacrificing solution quality in a few near-term periods in order to better position the workforce for future customer service requests.

In Chapter 4, we continue exploring the value of integrating future information into the current period decision-making process for the Multi-period Dynamic Technician Scheduling Problems with Experience-based Service Times and Stochastic Customers discussed in Chapter 3. We propose an alternate approximate dynamic programming solution approach with basis function to approximate the value function by taking the advantage of the future information for the whole planning horizon. At each decision epoch, we choose each technician's aggregate experience levels on all task types as the basis function. Then, we turn to an offline simulation procedure to recursively update the coefficient vector of the basis function, which allows fast decision making within the execution phase.

Our computational results demonstrate the value of the ADP solution approach with the basis function. We benchmark the performance of the ADP solution approach with the basis function against a myopic approach that converts the multi-period problem into a series of single-period problems, ignoring the impact of future information on the decision making in the current period.

5.2 Future Work

Because on-the-job learning is a new addition to the technician routing and scheduling literature, there are many directions for future research. Next, we discuss some possible avenues of future work.

First, we would like to compare the performance of the ADP with basis function with that of the ADP only look one day ahead in Chapter 3. We want to see the value of integrating the future information over the whole planning horizon instead of looking only one day ahead. Recall that the average daily gap between the one-day lookahead ADP and the daily myopic is in the scale of 0.1% and the largest gap is 0.68%. With the future information for the whole planning horizon, the average daily gap between the ADP with the basis function and the daily myopic is in the scale of 1%. Although the results are based on different instances, we expect that, with more future information, our ADP with basis function solution approach will outperform the ADP one-day lookahead because it is better to balance current period costs with those of the future.

Further, the multi-period nature of our problems results in a combinatorial state space and action space. One effective method to address this kind of "curse of dimensionality" is aggregation. Powell (2011) states the idea as to aggregate the original problem, solve it exactly, and then disaggregate it back to obtain an approximate solution to the original problem. Aggregation allows us to define the value function over the smaller state space, and we do not have to simplify the state of the system.

As for our problems, learning is a crucial factor that impacts the action/solution structure of our problems. Most of the learning effects occur at the early periods of the planning horizon, which corresponds to the instability of the workforce and unpredictable assignments during this periods. Therefore, we see very different assignments from day to day at the early periods of the planning horizon. As technicians accumulate experience levels by serving customers over time, most of them will reach the plateau of the learning curve, where the marginal effect is small. Thus, workforces become relatively stable and assignments do not change much from day to day in this later period of the planning horizon. Based on the above observation, future work could explore integrating the temporal aggregation in our approximate dynamic programing approach. In particular, we are able to spot the time when the workforce becomes stable and then consider all the periods after that time as one aggregated period.

There are also other avenues for future work. First, as noted in previous chapters, there are a number of factors other than experience that can affect an individual's ability to perform a task. Such considerations include learning from similar tasks (Olivella, 2007), learning by transfer from other technicians (Nembhard and Bentefouet, 2015), the effect of training (Hendrickson and Schroeder, 1941, Bianco et al., 2010), and even the technician's workload (Batt and Terwiesch, 2012, Brown et al., 2005). Including elements such as these in the formulation could offer additional insights into the problem.

Also, many technician-related scheduling problems include a routing component. It would be interesting to test the value of the proposed (or a similar) approach on the technician routing problem discussed in Chapter 2. However, the inclusion of the routing component will complicate both the state space and action space. The decision-making process not only involves the balance between current period cost and those of the future, but also has to make a trade-off between routing costs and scheduling costs. Thus, both the value function approximation and the updating process need to be elaborated so as to accommodate the additional complexity of the problem.

REFERENCES

D. Adelman. Dynamic bid prices in revenue management. *Operations Research*, 55(4):647–661, 2007.

D. Adelman and A. J. Mersereau. Relaxations of weakly coupled stochastic dynamic programs. *Operations Research*, 56(3):712–727, 2008.

A. Alsheddy and E. P. K. Tsang. Empowerment scheduling for a field workforce. *Journal of Scheduling*, 14 (6):639–654, 2011.

M. J. Anzanello and F. S. Fogliatto. Learning curve models and applications: Literature review and research directions. *International Journal of Industrial Ergonomics*, 41(5):573–583, Sept. 2011.

L. Argote. *Organizational Learning: Creating, Retaining, and Transferring Knowledge*. Kluwer, Norwell, MA, 1999.

J. F. Bard, Y. Shao, X. Qi, and A. I. Jarrah. The traveling therapist scheduling problem. *IIE Transactions*, 46(7):683–706, 2014.

R. J. Batt and C. Terwiesch. Doctors Under Load : An Empirical Study of Service Time as a Function of Census. working paper, available from `https://opimweb.wharton.upenn.edu/files/?whdmsaction=public:main.file\&fileID=5133`, 2012.

S. Bertels and T. Fahle. A hybrid setup for a hybrid scenario: combining heuristics for the home health care problem. *Computers & Operations Research*, 33(10):2866–2890, 2006.

F. J. Bianco, A. M. Cronin, E. A. Klein, E. E. Pontes, P. T. Scardino, and A. J. Vickers. Fellowship training as a modifier of the surgical learning curve. *Academic Medicine: Journal of the Association of American Medical Colleges*, 85(5):863–868, 2010.

D. Biskup. Single-machine scheduling with learning considerations. *European Journal Of Operational Research*, 188:315–329, 2008.

N. Bostel, P. Dejax, P. Guez, and F. Tricoire. Multiperiod Planning and Routing on a Rolling Horizon for Field Force Optimization Logistics. In B. Golden, S. Raghavan, and E. Wasil, editors, *The Vehicle Routing Problem: Latest Advances and New Challenges*, volume 43 of *Operations Research/Computer Science Interfaces*, pages 503–525. Springer US, Boston, MA, 2008.

S. J. Bradtke and A. G. Barto. Linear least-squares algorithms for temporal difference learning. *Machine Learning*, 22(1-3):33–57, 1996.

L. Brown, N. Gans, A. Mandelbaum, A. Sakov, H. Shen, S. Zeltyn, and L. Zhao. Statistical Analysis of a Telephone Call Center. *Journal of the American Statistical Association*, 100(469):36–50, 2005. ISSN 0162-1459. doi: 10.1198/016214504000001808.

J. A. Buzacott. The impact of worker differences on production system output. *International Journal of Production Economics*, 78(1):37–44, 2002.

G. Cachon and C. Terweisch. *Operations Management*. McGraw-Hill Education, New York, 2016.

X. Chen, B. W. Thomas, and M. Hewitt. The Technician Routing Problem with Experience-based Service Times. *Omega*, to appear.

E. Cochran. New concepts of the learning curve. *The Journal of Industrial Engineering*, 11(4):317–327, 1960.

R. W. Conway and A. Schultz. *The manufacturing progress function*. Dept. of Industrial and Engineering Administration, Cornell University, 1958.

J.-F. Cordeau, G. Laporte, F. Pasin, and S. Ropke. Scheduling technicians and tasks in a telecommunications company. *Journal of Scheduling*, 13(4):393–409, 2010.

A. Corominas, J. Olivella, and R. Pastor. A model for the assignment of a set of tasks when work performance depends on experience of all tasks involved. *International Journal of Production Economics*, 126(2): 335–340, 2010.

E. M. Dar-El. *Human Learning: From Learning Curves to Learning Organizations*, volume 29 of *International Series in Operations Research & Management Science*. Kluwer Academic Publishers, Boston, 2000.

P. De Bruecker, J. Van den Bergh, J. Beliën, and E. Demeulemeester. Workforce planning incorporating skills: state of the art. *European Journal of Operational Research*, 243(1):1–16, 2015.

D. P. de Farias and B. Van Roy. The linear programming approach to approximate dynamic programming. *Operations Research*, 51(6):850–865, 2003.

J. De Jong. The effects of increasing skill on cycle time and its consequences for time standards. *Ergonomics*, 1(1):51–60, 1957.

P.-F. Dutot, A. Laugier, and A.-M. Bustos. Technicians and Interventions Scheduling for Telecommunications. Technical report, France Telcom Research and Development, 2006. URL `http://www.g-scop.inpg.fr/ChallengeROADEF2007/en/sujet/sujet2.pdf`.

M. Fırat and C. Hurkens. An improved mip-based approach for a multi-skill workforce scheduling problem. *Journal of Scheduling*, 15(3):363–380, 2012.

J. Fowler, P. Wirojanagud, and E. Gel. Heuristics for workforce planning with worker differences. *European Journal of Operational Research*, 190(3):724–740, nov 2008. ISSN 03772217. doi: 10.1016/j.ejor.2007.06.038. URL `http://linkinghub.elsevier.com/retrieve/pii/S0377221707006017`.

N. Gans and Y.-P. Zhou. Managing Learning and Turnover in Employee Staffing. *Operations Research*, 50 (6):991–1006, nov 2002. ISSN 0030-364X. doi: 10.1287/opre.50.6.991.343. URL `http://or.journal.informs.org/cgi/doi/10.1287/opre.50.6.991.343`.

G. A. Godfrey and W. B. Powell. An adaptive, distribution-free algorithm for the newsvendor problem with censored demands, with applications to inventory and distribution. *Management Science*, 47(8): 1101–1112, 2001.

G. A. Godfrey and W. B. Powell. An adaptive dynamic programming algorithm for dynamic fleet management, i: Single period travel times. *Transportation Science*, 36(1):21–39, 2002.

J. C. Goodson, B. W. Thomas, and J. W. Ohlmann. A rollout algorithm framework for heuristic solutions to finite-horizon stochastic dynamic programs. Available from `http://slu.edu/~goodson/papers/GoodsonRolloutFramework.pdf`, June 2015.

C. Groër, B. Golden, and E. Wasil. A library of local search heuristics for the vehicle routing problem. *Mathematical Programming Computation*, 2(2):79–101, 2010.

W. J. Gutjahr. Optimal dynamic portfolio selection for projects under a competence development model. *OR Spectrum*, 33(1):173–206, aug 2009. ISSN 0171-6468. doi: 10.1007/s00291-009-0180-9. URL `http://www.springerlink.com/index/10.1007/s00291-009-0180-9`.

W. J. Gutjahr, S. Katzensteiner, P. Reiter, C. Stummer, and M. Denk. Competence-driven project portfolio selection, scheduling and staff assignment. *Central European Journal of Operations Research*, 16(3): 281–306, 2008.

H. Hashimoto, S. Boussier, M. Vasquez, and C. Wilbaut. A GRASP-based approach for technicians and interventions scheduling for telecommunications. *Annals of Operations Research*, 183(1):143–161, 2011.

Hays plc. The hays global skills index, 2014. Available from `http://www.hays.com.mx/cs/groups/hays_common/@mx/@content/documents/digitalasset/hays_1253572.pdf`.

M. He, L. Zhao, and W. B. Powell. Approximate dynamic programming algorithms for optimal dosage decisions in controlled ovarian hyperstimulation. *European Journal of Operational Research*, 222(2): 328–340, 2012.

C. Heimerl and R. Kolisch. Work assignment to and qualification of multi-skilled human resources under knowledge depreciation and company skill level targets. *International Journal of Production Research*, 48(13):3759–3781, 2010.

G. Hendrickson and W. B. Schroeder. Transfer of training in learning to hit a submerged target. *Journal of Educational Psychology*, 32(3):205–213, 1941.

M. Hewitt, A. Chacosky, S. E. Grasman, and B. W. Thomas. Integer programming techniques for solving non-linear workforce planning models with learning. *European Journal of Operational Research*, 242(3): 942–950, 2015.

C. A. Hurkens. Incorporating the strength of MIP modeling in schedule construction. *RAIRO - Operations Research*, 43(04):409–420, 2009.

M. Y. Jaber. Learning and Forgetting Models and Their Applications. In A. B. Badiru, editor, *Handbook of Industrial and Systems Engineering*, Industrial Innovation Series, chapter 30, pages 30–1–30–27. CRC Press, Boca Raton, FL, USA, 2006.

M. Y. Jaber and S. Sikström. A numerical comparison of three potential learning and forgetting models. *International Journal of Production Economics*, 92(3):281–294, Dec. 2004.

A. J. Kleywegt, V. S. Nori, and M. W. Savelsbergh. Dynamic programming approximations for a stochastic inventory routing problem. *Transportation Science*, 38(1):42–70, 2004.

A. Kovacs, S. Parragh, K. Doerner, and R. Hartl. Adaptive large neighborhood search for service technician routing and scheduling problems. *Journal of Scheduling*, 15:579–600, 2012.

M. G. Lagoudakis, R. Parr, and M. L. Littman. Least-squares methods in reinforcement learning for control. In *Methods and Applications of Artificial Intelligence*, pages 249–260. Springer, 2002.

D. Lesaint, C. Voudouris, and N. Azarmi. Dynamic Workforce Scheduling for British Telecommunications plc. *Interfaces*, 30(1):45–56, 2000.

F. Li, B. Golden, and E. Wasil. Very large-scale vehicle routing: new test problems, algorithms, and results. *Computers & Operations Research*, 32(5):1165 – 1179, 2005.

M. S. Maxwell, M. Restrepo, S. G. Henderson, and H. Topaloglu. Approximate dynamic programming for ambulance redeployment. *INFORMS Journal on Computing*, 22(2):266–281, 2010.

J. R. Meredith and S. M. Shafer. *Operations and Supply Chain Management for MBAs*. John Wiley & Sons, Hoboken, New Jersey, sixth edition, 2016.

C.-A. E. Moulton, A. Dubrowski, H. Macrae, B. Graham, E. Grober, and R. Reznick. Teaching surgical skills: what kind of practice makes perfect?: a randomized, controlled trial. *Annals of surgery*, 244(3):400–9, sep 2006. ISSN 0003-4932. doi: 10.1097/01.sla.0000234808.85789.6a. URL `http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=1856544{\&}tool=pmcentrez{\&}rendertype=abstract`.

A. Nedić and D. P. Bertsekas. Least squares policy evaluation algorithms with linear function approximation. *Discrete Event Dynamic Systems*, 13(1-2):79–110, 2003.

D. Nembhard and N. Osothsilp. Learning and forgetting-based worker selection for tasks of varying complexity. *Journal of the Operational Research Society*, 56(5):576–587, 2005.

D. A. Nembhard. Heuristic approach for assigning workers to tasks based on individual learning rates. *International Journal of Production Research*, 39(9):1955–1968, 2001.

D. A. Nembhard and F. Bentefouet. Parallel system scheduling with general worker learning and forgetting. *International Journal of Production Economics*, 139:533–542, 2012.

D. A. Nembhard and F. Bentefouet. Selection , grouping , and assignment policies with learning-by-doing and knowledge transfer. *Computers & Industrial Engineering*, 79:175–187, 2015. ISSN 0360-8352. doi: 10.1016/j.cie.2014.11.006. URL `http://dx.doi.org/10.1016/j.cie.2014.11.006`.

D. A. Nembhard and B. A. Norman. Cross Training in Production Systems with Human Learning and Forgetting. In D. A. Nembhard, editor, *Workforce Cross Training*, chapter 4, pages 111–129. CRC Press, Boca Raton, FL, USA, 2007.

C. Novoa and R. Storer. An approximate dynamic programming approach for the vehicle routing problem with stochastic demands. *European Journal of Operational Research*, 196(2):509–515, 2009.

J. Olivella. An Experiment on Task Performance Forecasting Based on the Experience of Different Tasks. In *Proceedings of I7th International Conference on Knowledge Management*, pages 305–312, Graz, Austria, 2007.

N. Panteva. Heating up: Demand for hvac services will rise along with construction market recovery, 2012. IBISWorld Industry Report OD5583.

D. J. Papageorgiou, M.-S. Cheon, G. Nemhauser, and J. Sokol. Approximate dynamic programming for a class of long-horizon maritime inventory routing problems. *Transportation Science*, 49(4):870–885, 2014.

V. Pillac, C. Guéret, and A. Medaglia. A parallel matheuristic for the technician routing and scheduling problem. *Optimization Letters*, 7(7):1525–1535, 2012a.

V. Pillac, C. Guéret, A. Medaglia, et al. On the dynamic technician routing and scheduling problem. Technical Report 12/5/AUTO, École des Mines de Nantes, Nantes, France, 2012b.

B. J. Pine. *Mass customization: the new frontier in business competition*. Harvard Business Press, 1993.

W. B. Powell. *Approximate Dynamic Programming: Solving the Curses of Dimensionality*. Wiley Series in Probability and Statistics. Wiley, Hoboken, New Jersey, second edition, 2011.

M. S. Rasmussen, T. Justesen, A. Dohn, and J. Larsen. The home care crew scheduling problem: Preference-based visit clustering and temporal dependencies. *European Journal Of Operational Research*, 219(3): 598–610, 2012.

S. Sayin and S. Karabati. Assigning cross-trained workers to departments: A two-stage optimization model to maximize utility and skill improvement. *European Journal of Operational Research*, 176(3):1643–1658, Feb. 2007. ISSN 03772217. doi: 10.1016/j.ejor.2005.10.045. URL `http://linkinghub.elsevier.com/retrieve/pii/S0377221705008970`.

P. J. Schweitzer and A. Seidmann. Generalized polynomial approximations in markovian decision processes. *Journal of mathematical analysis and applications*, 110(2):568–582, 1985.

S. M. Shafer, D. A. Nembhard, and M. V. Uzumeri. The effects of worker learning, forgetting, and heterogeneity on assembly line productivity. *Management Science*, 47(12):1639–1653, 2001.

K. Smilowitz, M. Nowak, and T. Jiang. Workforce Management in Periodic Delivery Operations. *Transportation Science*, 47(2):214–230, 2012. ISSN 0041-1655. doi: 10.1287/trsc.1120.0407.

M. Z. Spivey and W. B. Powell. The dynamic assignment problem. *Transportation Science*, 38(4):399–419, 2004.

R. S. Sutton and A. G. Barto. *Reinforcement learning: An introduction*. MIT press, 1998.

H. Topaloglu and W. B. Powell. Dynamic-programming approximations for stochastic time-staged integer multicommodity-flow problems. *INFORMS Journal on Computing*, 18(1):31–42, 2006.

E. Tsang and C. Voudouris. Fast local search and guided local search and their application to british telecom's workforce scheduling problem. *Operations Research Letters*, 20(3):119–127, 1997.

J. N. Tsitsiklis and B. Van Roy. Regression methods for pricing complex american-style options. *Neural Networks, IEEE Transactions on*, 12(4):694–703, 2001.

M. W. Ulmer, D. C. Mattfeld, and F. Köster. Budgeting time for dynamic vehicle routing with stochastic customer requests. Technical report, Working paper, 2014.

I. Venezia. On the statistical origins of the learning curve. *European Journal of Operational Research*, 19 (2):191–200, 1985.

A. Weintraub, J. Aboud, C. Fernandez, G. Laporte, and E. Ramirez. An emergency vehicle dispatching system for an electric utility in Chile. *Journal of Operational Research Society*, 44(10):690–696, 2012.

T. P. Wright. Factors affecting the cost of airplanes. *Journal of Aeronautical Sciences*, 4(3):122–128, 1936.

X. Yan, P. Diaconis, P. Rusmevichientong, and B. V. Roy. Solitaire: Man versus machine. In *Advances in Neural Information Processing Systems*, pages 1553–1560, 2004.

D. Zhang and D. Adelman. An approximate dynamic programming approach to network revenue management with customer choice. *Transportation Science*, 43(3):381–394, 2009.

H. Zhong, R. W. Hall, and M. Dessouky. Territory planning and vehicle dispatching with driver learning. *Transportation Science*, 41(1):74–89, 2007.