The Pennsylvania State University
The Graduate School
The Harold and Inge Marcus Department of Industrial and Manufacturing Engineering

ROBUST PLANNING AND EXECUTION

A Dissertation in
Industrial Engineering
by
Yooneun Lee

© 2017 Yooneun Lee

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Doctor of Philosophy

May 2017
The dissertation of Yooneun Lee was reviewed and approved* by the following:

Vittaldas V. Prabhu
Professor of Industrial and Manufacturing Engineering
Dissertation Advisor
Chair of Committee

Uday V. Shanbhag
Professor of Industrial and Manufacturing Engineering

Tao Yao
Associate Professor of Industrial and Manufacturing Engineering

Mort D. Webster
Associate Professor of Energy Engineering

Janis Terpenny
Professor of Industrial and Manufacturing Engineering
Peter and Angela Dal Pezzo Department Head/Professor

*Signatures are on file in the Graduate School
ABSTRACT

Uncertainties in a system environment have an adverse impact on the system. The system will yield suboptimal results, inhibit real-time adjustments, and may incur excessive costs for redesign. One of the most common managerial strategies to deal with such uncertainties is to incorporate redundant buffers in planning to absorb changes in system parameters, but more often than not, this strategy leads to waste of resources. Robust optimization is a method to deal with such parameter uncertainties. It postulates the worst case scenario and obtains a solution that provides a protection against the uncertainty. Plans designed for the worst tend to be overengineered; in many cases, when uncertain parameters are revealed, the degree of uncertainty turns out to be no greater than estimated. This effect may lead to underutilization of resources and inflated costs.

Here we present methods for robust planning and execution in various applications. We first consider the problem of scheduling operating rooms for surgeries. In order to avoid allocation of excessive resources and limit the conservativeness of the robust solution, we apply a cardinality-based robust approach. This enables us to control the level of protection in accordance with the solution quality. Experimental results based on real hospital data indicate that our method outperforms the practical gap-based approach in terms of both overtime and underutilization.

We next suggest a robust productivity index (RPI) to measure organizational performance in a dynamic environment. Compared to the traditional productivity index, it can be used for monitoring time-dependent performance and detecting exceptions. Efficiencies of service providers are evaluated based on panel data on youth outcomes from a selected community prevention program. The results suggest that our approach not only recognizes patterns of productivity progression, but also enables classification of the innovators.

Lastly, we study the real-time adaptive control and monitoring algorithm of an adaptive Work in Process (WIP) method. The simulation results suggest benefits of the algorithm as a decision support tool for
managing work-in-process and throughput level. Convergence and robustness of the algorithm are also investigated.

For each of these applications, the proposed method is validated and supported in an appropriate manner with detailed discussion of accuracy.
# TABLE OF CONTENTS

LIST OF TABLES ................................................................................................................................. viii

LIST OF FIGURES ................................................................................................................................. ix

1 INTRODUCTION ...................................................................................................................................... 1

2 ROBUST SURGERY PLANNING AND EXECUTION ........................................................................ 4
   2.1 Introduction....................................................................................................................................... 4
   2.2 Literature Review............................................................................................................................... 6
   2.3 Model Formulation ............................................................................................................................ 8
      2.3.1 Baseline surgery planning model ............................................................................................ 9
      2.3.2 Robust surgery planning .......................................................................................................... 11
      2.3.3 Solution methods ...................................................................................................................... 13
   2.4 Experimental Results ...................................................................................................................... 15
      2.4.1 Quality of the robust operational plan .................................................................................... 15
      2.4.2 Sensitivity analysis of robust parameters ............................................................................. 18
   2.5 Small Hospital Case study ............................................................................................................. 20
      2.5.1 Data description ....................................................................................................................... 20
      2.5.2 Comparison of key performance indicators ............................................................................ 21
      2.5.3 Schedule disruptions in execution ......................................................................................... 25
   2.6 Large Hospital Case Study ............................................................................................................ 30
      2.6.1 Data set and experiment .......................................................................................................... 30
   2.7 Implication and Applicability of Results to Other Hospitals ....................................................... 34
3 EVALUATION AND MONITORING COMMUNITY YOUTH PREVENTION PROGRAMS USING ROBUST PRODUCTIVITY INDEX

3.1 Introduction

3.2 Background

3.2.1 Planning prevention services

3.2.2 Performance evaluation using DEA

3.2.3 Malmquist Productivity Index

3.3 Monitoring Performance

3.3.1 Robust distance functions

3.3.2 Constructing productivity bounds

3.4 An Application of Community Youth Prevention Programs

3.4.1 Data and variables

3.4.2 Results

3.5 Implementation and Integration of Monitoring Engine

3.6 Conclusions

4 REAL-TIME SUPPLY CHAIN CONTROL USING ADAPTIVE WIP ALGORITHM

4.1 Introduction

4.2 Literature Review of Card Controlling System

4.2.1 Kanban and CONWIP system

4.2.2 Self-regulated WIP (SWIP)
4.3 Adaptive WIP (AWIP) Design ................................................................. 83
4.3.1 AWIP integral controller....................................................................... 83
4.3.2 AWIP algorithm.................................................................................. 86
4.4 Performance of AWIP Algorithm............................................................. 88
4.4.1 Solution quality .................................................................................. 88
4.4.2 Fidelity of the AWIP algorithm ............................................................ 93
4.4.3 The AWIP algorithm versus the mathematical model......................... 95
4.5 Concluding Remarks............................................................................... 100
5 CONCLUSION AND FUTURE RESEARCH............................................... 101
5.1 Conclusion ............................................................................................ 101
5.2 Future Research .................................................................................... 102
REFERENCES ............................................................................................ 103
LIST OF TABLES

Table 2.1 Experimental Design................................................................. 16
Table 2.2 Instances....................................................................................... 16
Table 2.3 Summary of results for 10-surgery samples for robust method.............. 17
Table 2.4 Summary of results for 15-surgery samples for robust method.............. 18
Table 2.5 Statistics on surgery duration for both planned and exceedance cases............... 21
Table 2.6 Underutilization in robust models and fixed gap model.......................... 24
Table 2.7 Summary of comparison between FG and RO models.......................... 25
Table 2.8 Empirical distribution of surgery duration and its interval ...................... 27
Table 2.9 Probability of disruption and changes in the total cost.......................... 29
Table 2.10 Cost increase of gap planning model.............................................. 35
Table 3.1 Variables....................................................................................... 57
Table 3.2 Input and output variables by quarter .............................................. 59
Table 3.3 Comparative efficiency by fiscal year .............................................. 63
Table 3.4 Average annual productivity growth of the youth prevention program over FY 2010-FY 2015............ 65
Table 3.5 Average quarterly changes by site .............................................. 66
Table 3.6 Sites shifting the technology frontier.............................................. 67
Table 3.7 Detailed components information associated with upper and lower limits .............. 72
Table 4.1 Notations....................................................................................... 84
Table 4.2 AWIP Algorithm.......................................................................... 88
Table 4.3 Performance measures and numerical results ...................................... 90
Table 4.4 Relative results of optimization model compared to the AWIP results ............. 98
LIST OF FIGURES

Figure 1.1 Hierarchical illustrations of planning layers in the supply chain domain......................... 1
Figure 2.1 Linkage between planning and execution phases in exception analytics for surgery system .... 5
Figure 2.2 Daily surgery schedule with three shared ORs................................................................. 9
Figure 2.3 Sensitivity analysis of robustness parameter ................................................................. 19
Figure 2.4 Sensitivity analysis of deviation parameter ...................................................................... 20
Figure 2.5 Comparison of solution quality for robust (RO) and fixed-gap (FG) model....................... 23
Figure 2.6 Pictorial overview of Monte-Carlo simulation model....................................................... 26
Figure 2.7 Probability of schedule disruption ................................................................................... 28
Figure 2.8 Relative changes in total cost .............................................................................................. 29
Figure 2.9 Simulation result of the probability of schedule disruption vs loss from overengineering ...... 30
Figure 2.10 Empirical distribution of planned (left) and actual (right) surgery durations.................. 31
Figure 2.11 Planned (left) and actual (right) total costs with respect to robust parameters............... 33
Figure 2.12 Planned (left) and actual (right) overtime with respect to robust parameters............... 33
Figure 2.13 Planned (left) and actual (right) underutilization with respect to robust parameters......... 34
Figure 3.1 Hierarchical overview of planning and delivery of prevention services in Pennsylvania (Source: Kang et al. 2016)............................................................... 39
Figure 3.2 Distance functions and frontiers ..................................................................................... 43
Figure 3.3 Graphical illustration of the robust distance function...................................................... 49
Figure 3.4 Desired clinical discharges ............................................................................................... 60
Figure 3.5 Proximal outcomes at discharge ...................................................................................... 61
Figure 3.6 Yearly average efficiency scores of sites.......................................................................... 64
Figure 3.7 Cumulated results of Malmquist productivity index ......................................................... 65
Figure 3.8 Robust productivity bound and productivity index .......................................................... 70
Figure 4.1 A Self-regulating WIP (SWIP) system ............................................................................ 81
Figure 4.2 Inverse Base Stock (IBS)........................................................................................................82
Figure 4.3 Release point 0, and inventory loop \((i, j)\) in \(M\) stage production system.........................83
Figure 4.4 AWIP Integral Controller .....................................................................................................86
Figure 4.5 Flow chart of AWIP algorithm...............................................................................................87
Figure 4.6 Experimental result of medium size supply chain problem...................................................90
Figure 4.7 Dynamics of the size of the card loop \((i, j)\)........................................................................93
Figure 4.8 Performance measure with three different initial solutions..................................................95
Figure 4.9 Balance condition for \(k\)th job on machine \(m\) in the part of Gantt chart...............................97
Figure 4.10 Experimental results of AWIP vs optimization model.......................................................100
# INTRODUCTION

Planning can be carried out at three different levels, distinguished by time scale and contributions: strategic, tactical, and execution (operational) planning. From the hierarchical point of view, each planning layer has its own functional and structural attributes, as well as associated planning tasks. As shown in Figure 1.1, strategic planning includes long-term decisions, such as network design, facility location problems, capacity planning, planning policies, and so on. The tactical planning layer involves medium-term decisions on production volume, target level stock, master scheduling, and so on. At the execution level, decisions on physical process, such as production timing and purchase orders are considered. In many cases, however, due to interdependencies and structural complexities, decision problems do not fall into a single planning layer. A small disturbance in system parameters may provoke serious discrepancies between the layers, and hence, it is necessary to deal with the underlying uncertainties over the different layers.

![Figure 1.1 Hierarchical illustrations of planning layers in the supply chain domain](image)

Robust optimization has been actively researched recently because it is distribution-free, and a well-designed robust model can reduce the dimension. Instead of assuming a random distribution, uncertain
parameters are assumed to lie on the set, and this set-based approach induces a significant reduction in the size of the problem, compared to the stochastic optimization method. Most of the recent research, however, focuses largely on robust decisions on the strategic and tactical levels, without considering interrelations with the operational planning layer and its impact in real time. Therefore, it is necessary to explore robust planning and execution in a broader sense and analyze complicated aspects of the methods through applications in different fields. To that end, this dissertation addresses three quite different optimization problems.

We begin with robust surgery planning and execution in Chapter 2. Applying the cardinality-based robust optimization approach, we address the overengineering issue of the industry standard, fixed-gap planning model. The quality of the robust approach is illustrated by comparing two methods. Results of the numerical experiment indicate that our model not only surpasses the fixed gap model, but it keeps the schedule disruption at a very low level. We conduct case studies using real hospital data to show substantial reductions in total costs, overtime and underutilization. The results also suggest that the cost increase due to overengineering is relative smaller, while the probability of schedule disruption in execution decreases significantly.

Chapter 3 focuses on the monitoring and evaluation of efficiency in a dynamic environment. The Malmquist productivity index (MPI) is a traditional method for measuring time-dependent performance, where the relative organizational efficiencies are measured over two consecutive periods. We propose a robust productivity index (RPI) to estimate longitudinal efficiencies and monitor the managerial performance. The robust productivity bound is constructed to identify innovators that make technical shifts. Our approach is illustrated with panel data from a selected youth intervention program between the fiscal years 2010 and 2015. The results suggest that our method enables us to recognize the patterns of productivity changes.
While Chapters 2 and 3 study robust planning and execution at the tactical and operational levels, Chapter 4 focuses on analysis of the robustness of the feedback control method. We investigate an adaptive WIP method, a feedback control algorithm for real-time production line control and inventory management. The proposed method is a fast card adjusting procedure, where the real-time key performance measures are continuously evaluated and the number of cards is updated accordingly. Once target throughput rate, blocking, and WIP level are set, the algorithm iteratively finds the best design of the system by determining the number of cards at each inventory point. The algorithm is proven to be not only fast, but also robust, regardless of the distribution of the random data, and initial solutions.

Chapter 5 summarizes the studies, draw overall conclusions, and outlines the future work.
2 ROBUST SURGERY PLANNING AND EXECUTION

2.1 Introduction

Difficulties in operating room (OR) management are largely due to conflicting objectives among stakeholders. Hospital managers mainly aim at reducing costs, while medical staffs are concerned with surgery safety first and foremost. However, they both suffer from uncertainty in surgery duration, which has a negative impact on the stability of the surgery schedule (May et al. 2011, Cardoen et al. 2010). In many cases, in response to disturbances and other changes of surgery duration, the original planning needs to be modified or adjusted during execution to deal with such uncertainties. Frequent rescheduling harms not only the well-being of patients, but also the efficiency of the hospital, through the accrual of additional operational costs. Therefore it is critical to build a robust schedule that is insensitive to uncertainties (Herroelen and Leus 2004, Wu et al. 1993, Vieira et al. 2003).

Current practice in building an OR schedule tends to deploy a sort of cushion in the planning to absorb any impact from perturbation duration during the execution of surgical procedures. Having buffers in a surgery schedule, however, can make the planning excessively conservative. An overengineered schedule is one with more than needed capacity and excessive protections. Excessive protection burdens the planning in terms of efficiency, profitability, and throughput of patients, because, more often than not, it results in the underutilization of medical resources, such as physicians, nurses, and medical device. In this sense, the eventual goal of the stakeholders is to manage the robustness of surgery planning with regard to overengineering and underutilization in its execution.

In estimating surgery duration, the planning model considers a baseline schedule that is insensitive to uncertainty. However, there is still a possibility that the baseline schedule could turn out to be no longer optimal or feasible if parameters deviate more than expected. Thus, when such an exceptional event happens, it may cause a transformation of the original planning of the system. An exception is defined as an anomalous deviation in one or more parameters that may cause changes in the planning status of the system.
Shaikh and Prabhu (2009) discuss exception analytics in manufacturing systems for the design and implementation of monitoring and prioritization strategies for anomalies. They emphasize the importance of computerized tools for automated monitoring of the varying parameters in the planning phase.

In the surgical environment, however, there is no consensus about the usage of a general purpose software tool. This is primarily because surgery procedures and environments are so different from one system to another that it is almost impossible to implement common features. Also, the cost of such an automated computer-aided system is generally too high to persuade management to implement it, and the benefits are as yet unclear. One of the advantages of our approach is it only requires a very limited amount of data to build a robust planning model and to build an exception analytics.

![Figure 2.1 Linkage between planning and execution phases in exception analytics for surgery system](image)

The purpose of this study is to present a systematic approach for analyzing the discrepancies between surgery planning and execution phases with respect to total cost, overtime, and underutilization (see Figure 2.1). We begin by suggesting a surgery planning model for an operational day-ahead surgery plan with a given number of patients. The model considers the allocation of surgical cases with uncertain
surgery duration in multiple ORs on a single day. The planning decision is limited to assigning surgery cases into ORs on a given day without considering surgeries to be reassigned to another day.

In building our surgery planning model, we incorporate gaps in two different approaches: a fixed-gap model, and a robust model. The fixed-gap model allocates a fixed time cushion between surgery cases, no matter how much the surgery duration deviates. We then extend the robust optimization approach presented by Bertsimas and Sim (2003) to obtain a surgery schedule with “smart gaps” that makes the schedule immune to uncertainty and lessens the cost of overengineering.

We then assess the potential impact of the robustness of the planning on the schedule execution phase in terms of key performance indicators. Based on real data from two hospitals, we design experiments to account for actual surgery execution on the day of surgery. By comparing the performance of the two approaches with respect to total costs, overtime and underutilization, we demonstrate that the robust model outperforms the fixed-gap model. More importantly, the robust approach results in less frequent schedule disruptions in the execution phase, as compared to the common practice of the fixed-gap model.

The rest of this chapter is organized as follows. In the next section, we provide a review of the literature relating to OR scheduling under uncertainty in surgery duration and its execution. We present two surgery scheduling models and solution methodology in Section 2.3. In Section 2.4, the quality of the robust model is presented compared to the fixed-gap model. Sensitivity analysis of the parameters is conducted as well. Section 2.5 describes a case study of a small-sized hospital to show the benefits of the robust approach in terms of schedule disruption. Section 2.6 studies the numerical experiment based on data from a large hospital. Finally, discussion and conclusions are presented in Section 2.8.

2.2 Literature Review

More general reviews on surgery planning and scheduling can be found in May et al. (2011), Guerriero et al. (2011), Cardoen et al. (2010), and Blake and Carter (1997). Several studies have particularly addressed the utilization of OR in surgery assigning and allocation problems. While overutilization is
usually measured by overtime costs, a few studies consider the underutilization of OR time in their models. Among them, Blake and Donald (2002) describe a resource allocation problem that minimizes the under supply of OR time under a block scheduling approach. In the case study of Mount Sinai Hospital, they demonstrate that the model produces an equitable schedule for each surgical department throughout the process. Strum et al. (1999) highlight both overutilization and underutilization of OR as key performance indicators in measuring OR efficiency. The quality of the surgical schedule is evaluated based on these two indicators, and the capacity planning of the surgery block allocation is considered. In the OR planning and scheduling problem, Fei et al. (2008) solve an integer programming problem for the elective surgery assignment problem that minimizes the weighted sum of overutilization and underutilization costs. All of these studies use the planning duration of a surgical case in their models, and hence the models are deterministic.

To deal with uncertain surgery durations, Hans et al. (2008) introduce planned slacks in the capacity utilization problem and hedge the uncertainties. Stochastic programming and robust optimization, however, are among the most extensively investigated methods for surgery scheduling problems. Denton et al. (2007) propose a two-stage stochastic programming model for sequencing surgical cases within a day. Using the L-shaped method and sample average approximation (SAA) algorithm they show the impact of surgery sequencing rule on performance measures such as total surgeons, OR staff waiting, OR underutilization, and overtime costs. Min and Yih (2010) present a mixed integer model for the surgery scheduling problem for elective patients considering the capacity of downstream resources. They obtain optimal scheduling using the SAA algorithm and show the benefits of their stochastic model in terms of total costs, overtime, underutilization, total number of scheduled patients, and cancellations.

The stochastic programming approach needs a strong assumption of known distribution on the OR schedule duration, and hence is heavily dependent on the availability of historical data. A robust optimization approach, on the other hand, deals with set-based uncertainty, providing protection against the
uncertainty set. Denton et al. (2010) examine robust settings of the daily OR allocation problem, and compare the results with solutions from deterministic, stochastic programming problems. For reducing computational time, they introduce symmetry-breaking constraints, which we will use in our formulation. Their numerical experiment demonstrates that the robust method runs much faster than the stochastic recourse model, while the quality of the robust optimal schedule is as good as the simple LPT heuristic.

Although the vast majority of the literature focuses on building a robust surgery schedule under uncertainty in surgery duration, a few studies investigate the impact of planning on the multiple levels, tactical, operational and executional, on hospital efficiency and utilization. Adan et al. (2011) examine two-stage planning procedures, reflecting changes between the tactical plan and the operational plan. Given the target level of resource utilization, they first develop tactical plan with operating slots. The slots are filled in the operational plan to deal with capacity excesses, emergency cases, and early/late cancellations. Ma and Demeulemeester (2013) also suggest an integrative approach to a hospital case mix and capacity planning. They studied three stages in the framework: case mix planning, master surgery scheduling, and the operational performance evaluation phase, but do not include emergency cases. Thus, while the interaction and effect between tactical and operational levels are mainly addressed, few studies tackle discrepancies between the planning and execution levels. Furthermore, both Adan et al. and Ma and Demeulemeester find the tradeoffs between hospital efficiency and service level (patient satisfaction). As we will show, those results do not tend to be consistent with our findings on the impact of robust operational planning. In contrast to their findings, our results indicate that the robust approach can actually enhance both resource efficiency and patient satisfaction.

2.3 Model Formulation

In this section, we describe surgery planning and scheduling models for daily surgery schedules. The decisions related to our models are (1) how many ORs to open on a given day, (2) which surgery to assign to which OR, and (3) the size of the gap between surgeries.
2.3.1 Baseline surgery planning model

Baseline surgery planning and scheduling usually takes place in advance. When there are multiple ORs being a shared resource among surgeons, surgeries can be processed in parallel. Recent research demonstrates the impact of parallel surgery processing and the cost-effectiveness of OR pooling (Batun et al. 2011). Figure 2.2 schematically illustrates feasible schedules for a hospital with three ORs. In the original planning, OR 1 shows an ideal situation, in which surgeries are well allocated and finish on time without overtime. If the first surgery takes longer than its planned hour, however, the following surgeries will shift to the right and hence incur overtime costs. To avoid such schedule disruptions, a fixed amount of gap is inserted between surgeries in the third figure. Not all surgeries are tardy, however, and some surgeries may be completed even earlier than planned. In the fourth figure, we observe significant underutilization in regular working hours. This schedule is overengineered, and overall efficiency seems worse than the original schedule without buffers.

Figure 2.2 Daily surgery schedule with three shared ORs

In order to address the challenges of the baseline planning method, we instead formulate the operational surgery planning model as an extensible bin packing problem. Considering overtime operating cost, the objective of the hospital manager is to minimize both fixed and operating costs for the OR in the hospital. Thus, the generated schedule not only minimizes total cost, but also determines the weekly surgery
plan of the hospital. Our model is similar to the daily surgery planning model by Denton et al. (2010), in which surgeries can be assigned to any OR regardless of surgery specialties. However, our day-ahead planning problem considers costs incurred by both overtime and underutilization. Assuming that the surgery duration embraces all pre- and post-incision time, we contain the artificial gaps to make the schedule robust.

Given the unknown surgery duration, we use the following notation in our model:

**Indices and Sets**

- $i$ OR index from 1, ..., $M$
- $j$ surgery index from 1, ..., $N$

**Parameters**

- $M$ Number of total available ORs
- $N$ Number of total surgeries
- $L$ Session length of each OR
- $c_i^f$ Fixed cost of operating OR $i$
- $c_i^o$ Overtime cost per hour in OR $i$
- $p_j$ Duration of surgery $j$
- $G_j$ Fixed gap of surgery $j$

**Decision Variables**

- $x_i$ binary variable indicating OR $i$ open
- $y_{ij}$ binary variable indicating surgery $j$ takes place in OR $i$
- $O_i$ overtime in OR $i$

The baseline planning model is:
(BP) \[
Z_{BP} = \min \sum_{i=1}^{M} (c^f x_i + c^o O_i) \quad (2.1)
\]
subject to \[
y_{ij} \leq x_i \quad \forall i, j \quad (2.2)
\]
\[
\sum_{i=1}^{M} y_{ij} = 1 \quad \forall j \quad (2.3)
\]
\[
\sum_{j=1}^{N} p_j y_{ij} - O_i \leq L x_i \quad \forall i \quad (2.4)
\]
\[
O_i \geq 0 \quad \forall i \quad (2.5)
\]
\[
x_i, y_{ij} \in \{0, 1\} \quad \forall (i, j) \quad (2.6)
\]

The objective of the deterministic model is minimization of the total cost associated with OR fixed cost and overtime cost. Constraint (2.2) ensures that surgeries can only be assigned to open ORs, and (2.3) guarantees that a single surgery can be assigned to only one OR. Constraint (2.4) calculates the overtime of the OR with a given allocation of surgeries. In the fixed gap model, Constraint (2.4) is replaced with a buffer of fixed size in addition to the nominal surgery duration.

(FG) \[
Z_{FG} = \min \sum_{i=1}^{M} (c^f x_i + c^o O_i) \quad (2.7)
\]
subject to \[
(2.2), (2.3), (2.5), (2.6)
\]
\[
\sum_{j=1}^{N} (p_j + G_j) y_{ij} - O_i \leq L x_i \quad \forall i \quad (2.8)
\]

2.3.2 Robust surgery planning

In this section, we use a robust optimization approach to build a planning model. The stochastic programming approach we discussed in the previous section relies heavily on the explicit distribution of uncertain parameters, and hence requires historical data. When a hospital manager has insufficient reliable data on surgery duration, the stochastic programming approach may not be delivered appropriately. On the
other hand, in the robust optimization method, the uncertain parameters are taken from an uncertainty set, and constraints are protected against the given uncertainty set.

In the context of our problem, a hospital manager does not necessarily know the exact distribution of the surgery duration, but can make a reasonable estimate of both the lower and upper bounds of processing time. Also, we assume that data is collected on the number of surgeries that deviate up to their upper bound. Since only some of the parameters in the constraint of the formulation are allowed to change, it is suitable to model the robust optimization formulation proposed by Bertsimas and Sim (2003). They suggest a discrete robust formulation in which both the cost coefficients and the data in the constraints of the discrete optimization problem are subject to uncertainty. The cardinality-based robust modeling approach is justified in the sense that it is unlikely for all of the realizations of uncertain parameters to hit their upper bound, so it is too conservative to provide full protection against all uncertain parameters. Furthermore, we can control the level of robustness and its adverse impacts on the solution by limiting the total number of uncertain coefficients within a constraint. Lastly, the robust counterpart, a transformation of the robust model, remains linear, and hence the problem is computationally tractable.

We assume that the uncertain surgery duration has an interval uncertainty, bounded by \( \tilde{p}_j \in [p_j - \hat{p}_j, p_j + \hat{p}_j] \). Then the solution of the following robust counterpart model will constitute the robust solution of Problem (2.1).

\[
(RC) \quad Z_{RC} = \min \sum_{i=1}^{M} (c^f x_i + c^o O_i) \\
\text{s. t. } (2.2), (2.3), (2.5), (2.6) \\
\sum_{j=1}^{N} p_j y_{ij} + \beta_i - O_i \leq L x_i \ \forall i
\]

where
\[
\beta_i = \max_{C_i} \left\{ \sum_{j \in S_i} \hat{p}_j z_{ij} + (I_i - |I_i|) \hat{p}_t z_{it} \right\},
\]

\(I_i\) is the set of coefficients \(p_j\) that are subject to parameter uncertainty and \(C_i = \{S_i \cup \{t_i\} | S_i \subseteq I_i, |S_i| = |I_i|, t_i \in I_i \setminus S_i\}.\) The formulation offers a protection against changes up to \(|I_i|\) number of coefficients, and one coefficient by \((I_i - |I_i|) \hat{p}_t\). The parameter \(I'\) indicates the pessimism of the constraint by controlling the number of coefficient changes up to its least favorable value. While Problem (RC) is nonlinear, the following lemma transforms it into a linear problem.

**Lemma 2.1** Problem (RC) has an equivalent formulation as follows (Bertsimas and Sim 2003):

\[(RO)\]
\[
Z_{RO} = \min \sum_{i=1}^{M} (c^f x_i + c^o O_i)
\]
\[\text{s. t. } (2.2), (2.3), (2.5), (2.6)\]
\[
\sum_{j=1}^{N} p_j y_{ij} + I_i w_i + \sum_{j \in I_i} q_{ij} - O_i \leq L x_i \forall i
\]
\[
w_i + q_{ij} \geq \hat{p}_j y_{ij} \forall i, j \in I_i
\]
\[
q_{ij} \geq 0 \forall i, j \in I_i
\]
\[
w_i \geq 0 \forall i
\]

Moreover, the robust solution still remains feasible with high probability even if the number of coefficient changes is more than \(|I'|\).

### 2.3.3 Solution methods

Under the assumption of identical ORs, the solution based on the baseline formulation (BP) has equivalent solutions when the sets of surgeries are switched between ORs. Due to the symmetric structure of the mixed-integer linear programming (MILP) formulation, the computational time is extended by exploring symmetric alternative solutions in the standard integer programming algorithm. To break this
symmetry of formulation, Denton et al. (2010) introduce three types of symmetry-breaking constraints, which can significantly reduce the time to obtain the optimal solution.

\[ x_i \geq x_{i+1} \quad \forall i \leq M - 1 \]  
\[ (2.16) \]

\[ \sum_{i=1}^{j} y_{ij} = 1 \quad \forall i, j \leq \min\{M, N\} \]  
\[ (2.17) \]

\[ \sum_{h=i}^{\min\{j,M\}} y_{hj} \leq \sum_{k=i-1}^{j-1} y_{i-1,k} \quad \forall i, j \geq i \geq 2 \]  
\[ (2.18) \]

Constraint (2.16) enforces that we open ORs sequentially: OR 2 cannot be opened without opening OR 1, OR 3 cannot be opened without opening OR 2 and hence OR 1, and so on. To break symmetry within surgeries, the surgeries are assigned to the opened ORs in a lexicographical order in terms of their indices. Without loss of generality, Constraint (2.17) assumes that the first surgery is allocated in OR 1, the second surgery can be placed either OR 1 or OR 2, and subsequent surgeries can be allocated similarly. Constraint (2.18) ensures that when the first \((j - 1)\) surgeries are placed in \( (i - 1) \) ORs, then the \( j \) th surgery can only be assigned to one of the first \( i \) ORs, not to \((i + 1)\)th or higher ORs. (See Denton et al. (2010) for more detail.)

We further make the following assumption on the cost parameter to avoid a solution where all surgery blocks are assigned to a single OR with incurred overtime cost:

\[ c^f < L c^o. \]  
\[ (2.19) \]

This assumption leads to the lower bound for the number of ORs to open because the cost assumption implies that each OR can serve a set of surgeries up to \((L + c^f / c^o)\). If the sum of durations exceeds the amount, it is optimal to open another OR to support the additional capacity.

\[ \sum_{j} p_j y_{ij} \leq \left( L + \frac{c^f}{c^o} \right) x_i \]  
\[ (2.20) \]
Therefore, relaxing the integrality of binary variable $y_{ij}$, the minimum number of OR openings is:

$$x_i = 1, \text{for } i = 1, \ldots, \left\lceil \frac{\sum_j p_j}{L + c_f / c_o} \right\rceil.$$  \hfill (2.21)

We will use constraints (2.16), (2.17), (2.18), and (2.21) in the numerical experiment in the case studies in Section 2.5 and 2.6 to significantly reduce computational time.

### 2.4 Experimental Results

In this section, we provide the results of numerical experiments to illustrate the optimal schedule using the robust approach, and to evaluate the proposed model. The models presented in this section were implemented and solved using CPLEX 12.2. All experiments were performed on an Intel® Core™ CPU i5-3340 Quad-Core, with 3.10GHz and 8.00 GB of RAM.

We begin by estimating the quality of the robust operational plan and performing sensitivity analysis for the robust parameters. Next, we present case studies for small and large hospitals by applying the proposed methodology. In the first case study, we illustrate the performance of the proposed approach based on the limited data provided by a small hospital. Compared to the current baseline practice where a fixed gap is inserted between surgeries, the robust approach reduces both underutilization and total cost, while keeping the probability of schedule disruption at the same level. In the second case study, we conduct a numerical experiment based on detailed data from a large hospital with a more complex surgery allocation policy.

#### 2.4.1 Quality of the robust operational plan

To evaluate the performance of the robust operational plan, we compare it to both the fixed gap model and the snapshot model presented in Problem (2.1). We begin with estimation of the parameters used for numerical experiments. We assume that regular hours are from 8 am to 8 pm, and every surgery that ends after 8 pm is considered as overtime, which costs 1.5 times the regular cost. We consider two different choices of session length ($L$), 8- and 10-hour sessions, and two types of associated fixed ($c_f$) and overtime
costs \((c^o), (c^f, c^o) = (1.0, 0.1875), (1.0, 0.15)\). The values assure that the per-hour overtime cost is 50% higher than the regular cost when the session length is 8 and 10 hours, respectively. Note that the values of the parameter comply with the assumption made in (2.19). We also take into account two different numbers of surgeries: 10 and 15 surgeries per day \((N)\). For surgery durations, we use samples from two different location \((\mu)\) and scale parameters \((\sigma)\) of lognormal distribution, which is common and widely validated by previous research in surgery practices (May et al. 2000; Strum et al. 2000). Note that the robust formulation does not specify the distribution of the uncertain parameter, but provides coverage against interval uncertainty. Table 2.1 presents the list of parameters and the values used in the experiments.

Table 2.1 Experimental Design

<table>
<thead>
<tr>
<th>Factor</th>
<th>Possible Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Surgeries</td>
<td>High ((N = 15)), Low ((N = 10))</td>
</tr>
<tr>
<td>Surgery Duration</td>
<td>Long ((\mu = 2.70)), Short ((\mu = 2.50))</td>
</tr>
<tr>
<td>Variation</td>
<td>High ((\sigma = 1.28)), Low ((\sigma = 0.64))</td>
</tr>
<tr>
<td>Num. of Uncertain Surgeries</td>
<td>Large ((I^r = 5)), Small ((I^r = 3))</td>
</tr>
<tr>
<td>Deviation</td>
<td>Large ((\theta = 0.2)), Small ((\theta = 0.1))</td>
</tr>
</tbody>
</table>

Next, we generate sample instances for testing the robust scheduling model. Table 2.2 summarizes the model instances, each of which has a different combination in high and low values of mean and standard deviation of lognormal distribution. For each instance, we generate 1,000 samples and calculate the average of optimal solutions. To evaluate the quality of optimal solution to the robust model, we use different values in both the number of surgery deviations \((I^r = 3, 5)\), and the amount of deviation \((\theta = 0.1, 0.2)\). For the sake of simplicity, we assume the same degree of uncertainty among the different operating rooms \((I^r = I^r_i, \forall i)\), and the amount of deviation is proportional to its planned duration.

Table 2.2 Instances

<table>
<thead>
<tr>
<th>Surgery Duration</th>
<th>Variation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table 2.3 and 2.4 summarize the results of the experiments for 10- and 15-surgery instances, respectively. Values in the tables represent the ratios of the average objective values of total cost of the fixed gap model (FG) and robust model (RO) to that of no-gap model. From the results, we observe that in most cases of both the FG and RO model, the average optimal costs are higher, due to the buffer inserted between surgeries. Also, solutions of the RO model outperform those of the FG model in both $\gamma = 3$ and $5$ because it provides smarter protection against uncertain surgery durations rather than a fixed amount of buffer between surgeries. In both the 10- and 15-surgery instances, the difference between the ratios of FG and RO becomes larger in Instances 3 and 4 because the surgery duration is shorter. For similar reasons, the 15-surgery samples are overall likely to be less efficient when more surgeries are assigned to each OR.

### Table 2.3 Summary of results for 10-surgery samples for robust method

<table>
<thead>
<tr>
<th>$L$</th>
<th>$c^o$</th>
<th>$\theta$</th>
<th>$\gamma$</th>
<th>Instance 1</th>
<th>Instance 2</th>
<th>Instance 3</th>
<th>Instance 4</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>FG</td>
<td>RO</td>
<td>FG</td>
<td>RO</td>
</tr>
<tr>
<td>10</td>
<td>0.15</td>
<td>3</td>
<td>0.9615</td>
<td>0.9789</td>
<td>0.9927</td>
<td>0.9351</td>
<td>0.9557</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.9654</td>
<td>0.9872</td>
<td>0.9407</td>
<td>0.9271</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.9641</td>
<td>0.8966</td>
<td>0.8761</td>
<td>0.8855</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.8693</td>
<td>0.8899</td>
<td>0.8625</td>
<td>0.8645</td>
</tr>
<tr>
<td>8</td>
<td>0.1875</td>
<td>3</td>
<td>0.8517</td>
<td>0.8727</td>
<td>0.8769</td>
<td>1.0097</td>
<td>0.9267</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.8555</td>
<td>0.8579</td>
<td>0.9933</td>
<td>0.9124</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.8588</td>
<td>0.9993</td>
<td>0.9600</td>
<td>0.9126</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.7131</td>
<td>0.7635</td>
<td>0.9163</td>
<td>0.8151</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.7247</td>
<td>0.7160</td>
<td>0.7750</td>
<td>0.7776</td>
</tr>
</tbody>
</table>

Table 2.4 Summary of results for 15-surgery samples for robust method

\begin{table}
\centering
\begin{tabular}{cccccccc}
\hline
$L$ & $c^\alpha$ & $\theta$ & $\Gamma$ & FG & RO & FG & RO & FG & RO \\
\hline
10 & 0.15 & 3 & 0.8669 & 0.9137 & 0.8674 & 0.9175 & 0.8631 & 0.9089 & 0.8637 & 0.9141 \\
 & & 5 & 0.8932 & 0.9050 & 0.8874 & 0.9137 & 0.9175 & 0.8631 & 0.9089 & 0.8637 & 0.9141 \\
0.2 & 3 & 0.7338 & 0.8317 & 0.7347 & 0.8382 & 0.7266 & 0.8228 & 0.7274 & 0.8307 & 0.7274 & 0.8307 \\
 & & 5 & 0.7928 & 0.7967 & 0.7822 & 0.8228 & 0.7274 & 0.8307 & 0.7274 & 0.8307 & 0.7274 & 0.8307 \\
8 & 0.1875 & 3 & 0.8752 & 0.9247 & 0.8755 & 0.9276 & 0.8727 & 0.9212 & 0.8729 & 0.9239 & 0.8729 & 0.9239 \\
 & & 5 & 0.9069 & 0.9098 & 0.9029 & 0.9212 & 0.8727 & 0.9239 & 0.8729 & 0.9239 & 0.8729 & 0.9239 \\
0.2 & 3 & 0.7503 & 0.8536 & 0.7509 & 0.8606 & 0.7454 & 0.8467 & 0.7458 & 0.8531 & 0.7458 & 0.8531 \\
 & & 5 & 0.8187 & 0.8267 & 0.8110 & 0.8467 & 0.7458 & 0.8531 & 0.7458 & 0.8531 & 0.7458 & 0.8531 \\
\hline
\end{tabular}
\end{table}

So far we have discussed the objective function value of the robust approach. Across the samples of instances with different combinations of parameters, we were able to obtain a robust surgery schedule without incurring excessive cost by regulating conservativeness. Now, we illustrate the performance of the robust scheduling in terms of the amount of gap inserted and the amount of overtime in the optimal schedule. Again, compared to the fixed gap, the linear model where a fixed gap is inserted between surgeries, the robust approach has fundamentally less buffer. Moreover, by reducing such overengineering, the robust model results in less overtime and hence better objective function value, as shown in the previous two tables.

\textbf{2.4.2 Sensitivity analysis of robust parameters}

Here we consider the evolution of the robust optimal schedule with respect to the robustness parameter $\Gamma$ and deviation parameter $\theta$. As we discussed, parameter $\Gamma$ controls the trade-off between the probability of a constraint being violated and the effect to the objective function value. Thus, hospital managers can calibrate the value depending on their own level of conservativeness. If they are risk-averse, then a large value for $\Gamma$ should be chosen. On the other hand, $\Gamma$ becomes small if they are risk-taking. Fixing
$N = 10$ and $\theta = 0.1$, the sensitivity analysis of the robust parameter is conducted over four instances for two sets of parameters: $L = 10, c^o = 0.15$ and $L = 8, c^o = 0.1875$.

Figure 2.3 illustrates the average optimal solutions of the 10-surgery problem instances with variation of the robust parameter $\Gamma$. The left figure indicates the average optimal total costs of the surgical suite when $L = 10$ and $c^o = 0.15$, and right plot shows $L = 8$ and $c^o = 0.1875$. In both cases, the optimal total cost is non-decreasing, that is, lower total costs for smaller robustness and higher total costs for larger robustness. As $\Gamma$ increases within smaller values, we observe higher costs incurred by more conservative planning. In larger $\Gamma$, however, we also observe the total costs remain the same, even though $\Gamma$ increases further. Roughly speaking, given 10 surgeries in 3 ORs, there would be 3-4 surgeries assigned to a single OR, depending on their durations. Thus, providing schedule protection to more surgeries in a single OR made no difference in the total costs.

Figure 2.4 illustrates the sensitivity of deviation parameter $\theta$, which defines the range of uncertain surgery duration. If the parameter is set to 0.1, the robust approach protects against up to 10% of deviation from the nominal surgery duration. As we observe, the total cost function of the surgical suite is also non-decreasing with respect to $\theta$. Compared to the sensitivity analysis of $\Gamma$, it is clear that the impact of the
deviation parameter $\theta$ is greater on the objective function value in this specific example. Unlike the plots of the robust parameter $\Gamma'$ in Figure 2.3, however, the optimal total cost graph is ever-increasing with respect to deviation parameter $\theta$, because the robust model provides full protection no matter how much the surgery deviates within the interval. Therefore, the optimal total cost preserves the increasing tendency, in contrast to the robust parameter case.

![Figure 2.4 Sensitivity analysis of deviation parameter](image)

2.5 Small Hospital Case study

2.5.1 Data description

In this section, we apply the proposed approach to a small hospital that treats about 3,600 patients annually. On average, 12 surgeries are assigned and served a day, across 9 surgical specialties, including orthopedics (ORTHO), neurology (NEURO), ophthalmology (OPHTH), obstetrics/gynecology (OBGYN), general (GEN), plastic (PLASTIC), and urology (URO). This hospital employs a scheduling policy with a one-hour artificial gap between adjacent surgeries. The hospital does not record the planned and actual duration of each surgery, so the only available data is for the cases whose durations exceed the one-hour buffer. Over 16 months, the total number of such exceptional cases was 61 out of 4,800 (1.27%).
Table 2.5 Statistics on surgery duration for both planned and exceedance cases

<table>
<thead>
<tr>
<th>Specialty</th>
<th>Planned (min)</th>
<th>Exceedance (min)</th>
<th>No. of Observations</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mean</td>
<td>Std. Dev</td>
<td>Mean</td>
</tr>
<tr>
<td>OBGYN</td>
<td>145</td>
<td>58</td>
<td>290</td>
</tr>
<tr>
<td>GEN</td>
<td>124</td>
<td>60</td>
<td>236</td>
</tr>
<tr>
<td>ORTHO</td>
<td>163</td>
<td>30</td>
<td>274</td>
</tr>
<tr>
<td>PLASTIC</td>
<td>228</td>
<td>107</td>
<td>348</td>
</tr>
<tr>
<td>GI</td>
<td>165</td>
<td>57</td>
<td>375</td>
</tr>
<tr>
<td>Others</td>
<td>257</td>
<td>96</td>
<td>399</td>
</tr>
<tr>
<td>Total</td>
<td>162</td>
<td>77</td>
<td>297</td>
</tr>
</tbody>
</table>

Table 2.5 summarizes the statistical information on 61 cases. For the following experiments, we assume that the surgery duration has an identical distribution regardless of surgery specialty, since the available data is not enough to distinguish distributions of different surgery specialties. Even from a practical perspective, the actual duration is not dependent on the specialty, but on the difficulty of the surgery or the severity of the patient’s condition. Thus, we estimate location and scale parameters of lognormal distribution by fitting the historical data of the hospital and generate random samples based on it.

2.5.2 Comparison of key performance indicators

Based on the fitted lognormal distribution with \( \mu = 2.70 \) and \( \sigma = 1.28 \), we randomly generate 1,000 sample sets of surgeries to estimate the three key performance indicators for both the robust and fixed gap model. Using the parameters \( \theta = 0.1 \) and \( \Gamma = 1, 2 \) and 3, the experiments are independently conducted to compare the total cost, underutilization, and overtime of the two models. Figure 2.5 depicts the results of the experiment, where data points on the 45-degree line indicate that the performance metrics of the two models are equivalent. The plots in the first column illustrate that the total cost of the robust optimal solution is smaller than the fixed gap model for \( \Gamma = 1, 2 \) and 3, respectively. Again, the objective values of the robust
model are better in every single sample for every $\Gamma$, and hence the plots are located above the 45-degree line. This corresponds to what we found in the previous section (Table 2.3), but the difference in values is narrower in larger $\Gamma$. 
Comparison of results in terms of overtime are shown in the middle column. The optimal overtime of the fixed gap model is generally longer than that of the robust approach, but the difference is again
smaller in larger $\Gamma$. We note that data points marked in the bottom part of each figure indicate that only the robust model has a significant amount of overtime, while the fixed gap model has no overtime. Note that even in those samples, the optimal cost of the robust model is still lower than that of the fixed gap model. This is because the robust optimal planning method has a smaller number of ORs open than the optimal fixed gap model, and so has lower fixed costs, and incurs the relatively cheaper overtime cost, rather than the relatively higher cost of keeping the OR open.

The last column suggest that the robust model tends to have smaller underutilization than the fixed gap model. The average actual underutilized time is estimated by: $\bar{UT} = \frac{1}{M} \sum_i (Lx_i + O_i - \sum_j \tilde{p}_j y_{ij})$. The sample points marked in the upper left of the figures show that the robust optimal solution keeps underutilization low by opening fewer ORs. This explains how the robust approach is likely to have relatively shorter idle time and more overtime than the fixed gap model. More importantly, the robust model tends to have less variability across the sample instances. The data points for the robust planning method not only fall within a lower range of values, but also vary to the same of lower extent. The range of underutilization and its standard deviation from the experiment are summarized in Table 2.6. Note that the robust approach does not aim specifically to minimize the standard deviation.

Table 2.6 Underutilization in robust models and fixed gap model

<table>
<thead>
<tr>
<th></th>
<th>Robust model</th>
<th>Fixed gap model</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\Gamma = 1$</td>
<td>$\Gamma = 2$</td>
</tr>
<tr>
<td>Range</td>
<td>[0.24, 4.16]</td>
<td>[0.46, 4.74]</td>
</tr>
<tr>
<td>$\bar{UT}$</td>
<td>1.38</td>
<td>2.12</td>
</tr>
<tr>
<td>$\sigma(UT)$</td>
<td>0.92</td>
<td>0.96</td>
</tr>
</tbody>
</table>

The results are summarized in Table 2.7. For consistency, we exclude the cases where two models have different numbers of open ORs. Overall, the robust planning outperforms the fixed gap model. The ratios of overtimes under robust planning are 59.69%, 76.34% and 87.82%, while those of underutilization...
are 45.78%, 68.17% and 83.62% for $\Gamma = 1$, 2 and 3, respectively. As $\Gamma$ increases, the difference in the performance measures is decreases, but the benefits of the robust approach still dominate in all three cases.

Table 2.7 Summary of comparison between FG and RO models

<table>
<thead>
<tr>
<th></th>
<th>Robust model</th>
<th>Fixed gap model</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\Gamma = 1$</td>
<td>$\Gamma = 2$</td>
</tr>
<tr>
<td>Overtime* (hrs)</td>
<td>2.20 (59.69%)</td>
<td>2.81 (76.34%)</td>
</tr>
<tr>
<td>Underutilization* (hrs)</td>
<td>1.25 (45.78%)</td>
<td>1.86 (68.17%)</td>
</tr>
<tr>
<td>Total Cost</td>
<td>2.94 (94.19%)</td>
<td>3.02 (96.67%)</td>
</tr>
</tbody>
</table>

* For consistency, we compare only the samples where the optimal solutions of the two models have same number of open OR.

2.5.3 Schedule disruptions in execution

Here, we explain some intriguing aspects of the robust framework through two experiments. In the first experiment, we investigate the schedule disruption and overengineering costs of the robust planning model. We present a problem with 3 ORs, the robust parameter ($\Gamma$), and the number of surgeries ($N$). By subjecting the surgery duration to random perturbations, we estimate the operational costs and probability of surgery schedule disruption. Initially, we run the robust planning problem (3) and obtain a robust solution for different sets of ($\Gamma, N$). Then, for each surgery procedure, a random number is generated within its interval uniformly using Monte Carlo simulation. Figure 2.6 shows a pictorial overview of the simulation phases.
In the first phase, we select surgery durations from the empirical distribution and solve the robust surgery planning problem (2.3). Given the surgery duration, we obtain the robust optimal solutions, $\mathbf{x}^*(\Gamma), \mathbf{y}^*(\Gamma)$ and $\mathbf{O}^*(\Gamma)$ with respect to the robust parameter $\Gamma$. We take integer values of $\Gamma$ in $[0,N]$. In the second phase, we generate 10,000 random surgery durations, each of which independently deviates within its interval $[p_j^L, p_j^U]$, where $p_j^L$ and $p_j^U$ are lower and upper bounds for procedure duration of surgery $j$. From the empirical distribution presented in Table 2.8, we randomly select surgery intervals up to the total number of surgeries. Next, we estimate the probability of schedule disruption and average total cost based on a random sample of surgery durations in the second phase and robust optimal planning from the first phase.
Table 2.8 Empirical distribution of surgery duration and its interval

<table>
<thead>
<tr>
<th>$p_j^L$</th>
<th>$p_j^U$</th>
<th>$\bar{p}_j$</th>
<th>Probability</th>
<th>Cumulative</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>1</td>
<td>0.75</td>
<td>0.1266</td>
<td>0.1266</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>1.5</td>
<td>0.2658</td>
<td>0.3924</td>
</tr>
<tr>
<td>1.5</td>
<td>2</td>
<td>1.75</td>
<td>0.1519</td>
<td>0.5443</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>2.5</td>
<td>0.2532</td>
<td>0.7975</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>3</td>
<td>0.0380</td>
<td>0.8354</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>3.5</td>
<td>0.1013</td>
<td>0.9367</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td>4</td>
<td>0.0253</td>
<td>0.9620</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>4.5</td>
<td>0.0127</td>
<td>0.9747</td>
</tr>
<tr>
<td>4</td>
<td>6</td>
<td>5</td>
<td>0.0127</td>
<td>0.9873</td>
</tr>
<tr>
<td>6</td>
<td>7</td>
<td>6.5</td>
<td>0.0127</td>
<td>1.0000</td>
</tr>
</tbody>
</table>

Let $Z_{RO}$ be the robust optimal cost that is achievable when no deviation occurs in surgery durations. Depending on how long the surgery actually takes, however, the surgery schedule may or may not be disrupted. The probability of surgery schedule disruptions in OR $i$ can be measured by $\sum_{j=1}^{n} \bar{p}_j y_{ij}^* - Lx_i^* > O_i^*$, where $\bar{p}_j$ is actual duration of surgery $j$. The relative cost change due to overengineering in the robust planning is calculated by $(\bar{Z} - Z_{RO}^*)/Z_{RO}^*$, where $Z_{RO}^*$ is the robust optimal cost, and the cost of the planned execution is estimated by:

$$\bar{Z} = \sum_{i=1}^{n} \left( c_i^f x_i^* + \max \left( 0, \sum_{j=1}^{n} \bar{p}_j y_{ij}^* - Lx_i^* \right) \right).$$
Figure 2.7 Probability of schedule disruption
Figure 2.8 Relative changes in total cost

Simulation results are depicted in Figure 2.7 and 2.83. Both the probability of a surgery schedule disruption and the relative changes in cost decrease with respect to $\Gamma$. However, we note that the probability of the surgery schedule disruption decreases exponentially, while the relative change in cost does not change much in larger $\Gamma$. With $\Gamma = 1$, for instance, the total cost increase due to overengineering is only 0.01%, while the probability of schedule disruption is as large as 50%. As $\Gamma$ increases, the chances of schedule disruption drop significantly, but the total costs increases only slightly. At $\Gamma = 3$, the schedule has about 3% possibility of disruptions with 14% increase in total cost. The disruption probability drops even less than 1% with $\Gamma > 4$, as summarized in Table 2.9. On the other hand, the impact of robustness on the total cost seems to fade for larger values, as relative changes in total cost remain within a smaller range.

Table 2.9 Probability of disruption and changes in the total cost

<table>
<thead>
<tr>
<th>$\Gamma$</th>
<th>Prob. of disruption</th>
<th>Relative changes in total cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.5005</td>
<td>-0.0001</td>
</tr>
<tr>
<td>2</td>
<td>0.1629</td>
<td>-0.0777</td>
</tr>
<tr>
<td>3</td>
<td>0.0304</td>
<td>-0.1383</td>
</tr>
<tr>
<td>4</td>
<td>0.0065</td>
<td>-0.1573</td>
</tr>
<tr>
<td>5</td>
<td>0.0009</td>
<td>-0.1881</td>
</tr>
<tr>
<td>6</td>
<td>$8.3333 \times 10^{-5}$</td>
<td>-0.2013</td>
</tr>
<tr>
<td>7</td>
<td>$3.3333 \times 10^{-5}$</td>
<td>-0.2055</td>
</tr>
<tr>
<td>8</td>
<td>-</td>
<td>-0.2145</td>
</tr>
<tr>
<td>9</td>
<td>-</td>
<td>-0.2154</td>
</tr>
<tr>
<td>10</td>
<td>-</td>
<td>-0.2324</td>
</tr>
</tbody>
</table>

In the second experiment, we examine the relationship between the probability of disruption and overengineering costs. We follow the simulation steps of the first experiment. Random samples for surgery durations for given $\Gamma$ are generated based on the empirical distribution. We iterate the entire procedure by
increasing $\Gamma$ by 0.01 in each step. Figure 2.9 reveals the relationship between the probability of schedule disruption and the changes in the total cost. This suggests that robust surgery planning plays an important role in execution by keeping the overengineering cost to within a manageable amount.

![Figure 2.9 Simulation result of the probability of schedule disruption vs loss from overengineering](image)

2.6 Large Hospital Case Study

2.6.1 Data set and experiment

In this section, we test the proposed approach based on surgery data from one of the largest medical institutes in Pennsylvania. Our parameter estimations are based on historical data as well as discussion with a hospital administrator. The institution consists of 34 ORs serving more than 24,000 surgery cases across 40 surgery specialties, including orthopaedics, general, cardiac, obstetrics and gynecology, neurosurgery, pediatric, ophthalmology, otolaryngology, oncology, plastic, and urology. The data contains both planned
and actual surgery duration of surgeries, with the OR assigned for each surgery. Empirical distributions of both planned and actual duration are illustrated in Figure 2.10.

Figure 2.10 Empirical distribution of planned (left) and actual (right) surgery durations

After in-depth statistical investigation of the data set, we can make following observations:

- Planned surgery duration is significantly overengineered. The statistical test emphasizes that, on average, the planned surgery duration is more than 40 minutes longer than the actual duration.
- However, there is a proportion of surgery cases (13.51%, or 3,251 out of 24,056) whose actual durations take longer than planned. At the same time, there are also a few cases (3.84%, or 923 out of 24,056) whose procedure lagged more than 1 hour behind the planned duration, which disrupted the surgery schedule.
- The difference between planned and actual duration is larger for longer surgery duration. Those values are highly correlated, with a correlation coefficient 0.8024. Therefore, we assume that the duration of a surgery is intentionally overestimated, proportional to the planned duration.
• Surgery workload varies from day to day, but can be categorized into two different levels of demand: low demand on Thursday and Friday (11.07 cases per day) and high demand on Monday through Wednesday and weekends (87.38 cases per day).

No strict restrictions exist in assigning surgeries to ORs, except for pulmonary and pediatric surgeries being exclusively assigned to three ORs. There is, however, a clear indication that some surgeries are assigned preferentially to a specific set of ORs. This might be due to either the accessibility of staffs and devices or to purely clinical decisions, but it is common in many hospitals. Therefore, we begin the experiment by classifying the surgeries into 12 groups, each of which performs procedures in specific surgery specialties. This practice is widely deployed, especially in large hospitals, according to the author’s investigation. With high demand on Monday through Wednesday, Saturday and Sunday, we solve the surgery scheduling problem by surgery group.

We also made reasonable modifications to the historical data. Instead of using the actual starting and ending time of the surgery schedule, we assume that the elective surgery cases are optimally allocated based on a simple linear programming model. For each day, we estimate total cost, amount of overtime, and underutilization, with 10 regular hours. We compare the results with those from the robust surgery scheduling by changing the amount of protection: the robust parameter $\Gamma$ changes from 1 to 2 by 0.1, and the amount of buffer changes from 0.1 to 0.5 of the estimated surgery duration. In order to improve the computation time, we preprocess the number of ORs per day by selecting the minimum required OR openings from the linear programming model. The three key performance measures are evaluated based on both the planned and actual surgery durations. The planned and actual underutilizations are estimated by $\sum_{i=1}^{m} \left( \max \{0, Lx_{i}^{\ast} - \sum_{j=1}^{n} p_{j}y_{ij}^{\ast}\}\right)$ and $\sum_{i=1}^{m} \left( \max \{0, Lx_{i}^{\ast} - \sum_{j=1}^{n} \tilde{p}_{j}y_{ij}^{\ast}\}\right)$, respectively.
Figure 2.11 Planned (left) and actual (right) total costs with respect to robust parameters

Figure 2.12 Planned (left) and actual (right) overtime with respect to robust parameters
The robust total costs based on both planned and actual surgery duration are higher with larger protections, but the actual cost increase gets smaller in Figure 2.11. This is similar in the underutilization; larger protection results in longer buffers within the schedule, but the degree of actual underutilization becomes narrower, as shown in Figure 2.13. However, the schedule may suffer from overtime. As pointed out earlier, a few surgeries take significantly longer than the estimated duration, and the linear scheduling model does not cope with this issue properly. The amount of overtime is smaller in the entire parameter region as the relative changes become negative in Figure 2.12. We note that the overtime in the execution gets smaller with the larger parameters, because the assignment has enough buffer to absorb the time lag.

2.7 Implication and Applicability of Results to Other Hospitals

The results of our case studies provide a sharp indication of the benefits that hospitals can obtain from a robust scheduling scheme. First, the results show that hospitals can take advantage of the robust approach in building surgery schedules when the data is not readily available. Recall that the approach with one-hour buffer (current practice) faces a probability of schedule disruption of 1.27%. The surgery schedule of the hospital is not optimally designed, and data on non-exception surgeries are not readily available. Assuming that the schedule is obtained by the fixed gap model (a surgery schedule with one-hour gaps), we estimate the probability and cost loss incurred by the fixed gaps. Depending on different $\Gamma$, the results
listed in Table 2.10 show that the total cost of the fixed gap model ranges from 2.0% to 5.7% higher than the robust planning model. Thus, all in all, the robust approach shows practicability, especially when the hospital is not fully equipped for data collection.

Table 2.10 Cost increase of gap planning model

<table>
<thead>
<tr>
<th>( \Gamma )</th>
<th>RO</th>
<th>Gap</th>
<th>Cost increase (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3.40</td>
<td>3.60</td>
<td>5.66</td>
</tr>
<tr>
<td>2</td>
<td>3.45</td>
<td>3.56</td>
<td>3.15</td>
</tr>
<tr>
<td>3</td>
<td>3.58</td>
<td>3.69</td>
<td>2.00</td>
</tr>
<tr>
<td>4</td>
<td>3.83</td>
<td>3.98</td>
<td>3.88</td>
</tr>
<tr>
<td>5</td>
<td>3.84</td>
<td>3.96</td>
<td>3.18</td>
</tr>
</tbody>
</table>

Our approach may be particularly useful under the circumstances where the duration of a surgery is likely to be over-estimated. One can reduce the chance of schedule disruption, while enjoying a significant cost savings both in terms of overtime and underutilization of resources. Compared to the scheduling environments in other areas, such as manufacturing and production processes, the findings have significance because the resources of the healthcare system tend to be both expensive and scarce.

2.8 Conclusion

Here we consider a surgery assignment problem under uncertain surgery duration. A robust surgery assignment system is developed to obtain a surgery schedule that is resilient to uncertainty in surgery duration. As the robust schedule provides protection against the perturbation of surgery duration and conserves its original planning, the total costs grow higher. Our model reduces conservativeness by controlling the number of deviating surgeries. Numerical experiments show that the robust approach results in smarter protection as compared to a fixed gap model where every surgery is overestimated by some amount. The robust approach can be an efficient approximation of the fixed gap model in terms of providing
a robust assignment of surgeries to ORs. The impact of robust parameters and amount deviation to the total cost is analyzed from the sensitivity analysis.

Based on a case study of a small-sized hospital, we illustrate the relationship between the robustness parameter and the chances of surgery schedule disruption. The experimental results demonstrate that the robust approach significantly reduces the probability of schedule disruptions, and the cost increases only insignificantly. This method thus can be particularly beneficial to small hospitals that are incapable of collecting enough data on surgery duration. From the simulation results, even if the number of deviating surgical cases is not accurate, the robust surgery approach still performs satisfactorily in terms of surgery disruption.

In the large hospital case study, we take into account various performance measures that are widely employed by many medical institutes. Based on the planned and actual surgery durations, we examine total cost, overtime, and underutilization by comparing them with the nominal optimal surgery planning. The results indicate that the schedule from a nominal model may suffer from large overtime in its execution when the surgery duration is mostly overestimated. Therefore, we would suggest that the hospital manager control the level of robustness by finding the minimum operating costs, overtime, and underutilization of the surgical suite based on the actual surgery duration.

Future work will study the dynamic reallocation of surgery cases when the original plan is disrupted. This will entail an increase in problem complexity, as further algorithm will be necessary for reallocation of such cases, but it will make an impact on the robustness of the system. In this chapter, we do not consider the scheduling of surgery personnel. Incorporating physicians’ schedules would offer the opportunity to assess the impact of robust planning in more realistic scheduling scenarios.
3 EVALUATION AND MONITORING COMMUNITY YOUTH PREVENTION PROGRAMS USING ROBUST PRODUCTIVITY INDEX

3.1 Introduction

Prevention and intervention programs target youth behavioral and mental problems in the early stage of development, and reduce future social costs. Effective prevention programs can reduce problems such as delinquency, aggression, violence, bullying, substance abuse, and school failure in the youth population through better social and emotional health. Several evidence-based programs (EBP) have undergone randomized trials (Aos et al. 2006).

In the United States, prevention programs are generally implemented at the state level, and are put into practice by local community sites. The objectives and the target population are different for each prevention program (Mrazek and Haggerty 1994), but these kinds of programs tend to have processes for enrollment/participation, treatment/intervention, evaluation/assessment, discharge, and discharge follow-up if needed. Once a client (youth) enrolls in the program, parents or caregivers provide proper treatment with assistance from trained therapists. Assessment of outcome consists of measuring changes based on the critical criteria of the given program. With the aid of a web-based database system, clinicians can aggregate client data and monitor the overall result of intervention by regularly generating outcome reports.

Evaluating and comparing the efficiencies of community sites, however, present some challenges. Even though the overall economic benefit of some prevention programs has been shown (Jones et al. 2008), there is no consensus in assessing the efficiency of local providing sites. Since the specifications of community sites vary, it is hard to measure and compare the performance of one prevention program provider against that of another. Simply comparing the percentage of youths that are “successfully discharged,” for instance, does not take into account the condition of the youths when they are first referred. Furthermore, any data collected in the programs must be carefully considered; since the available data is largely dependent upon input from individual therapists, attention should be paid to the nuances of reporting.
Missing data, for instance, may skew comparisons. Some aspects of performance are time-dependent, and monitoring the developmental phase of the provider site will be needed.

In this article, therefore, we develop a decision support tool to assess the performance of prevention program sites. Our focus lies in the following aspects of the problem: First, rather than finding a single overall measure of organizational performance, we identify key factors that are critical in determining the organizational performance and youth outcomes from evidence-based programs using Data Envelopment Analysis (DEA). Next, we evaluate the time-dependent performance of the sites by using the Malmquist productivity index (MPI), and suggest a novel way to evaluate the development phase of the site. By evaluating managerial performance of a selected prevention program provider, the model finds the “best practices,” which implies the maximum set of outputs to be efficient with a given input level. We estimate the DEA score of prevention program providers using the data set collected and stored through a web application system. Finally, we suggest a real-time monitoring and notification functionality for the system based on previous results. This approach will enable program managers and policy makers to make fair comparisons among service providers and to have a better managerial understanding of the tendency of the performance of providers over a period of time.

The remainder of this paper is as follows: in Section 3.2, we present the background of DEA and MPI. Section 3.3 introduces the selected prevention programs and variables. Section 3.4 demonstrates the experimental results using real data. Finally, conclusions are given in Section 3.5.

3.2 Background

3.2.1 Planning prevention services

Prevention services are designed to address both behavioral and mental problems of youth. Effective delivery of prevention services significantly reduces delinquency, aggression, violence, bullying, and substance abuse in the youth population (Chilenski et al. 2007). Prevention programs differ in their objectives, scopes, and target populations, but there are three primary roles in common for planning and
delivery of a prevention program: (1) state government policymakers, (2) prevention service providers, and (3) youth or families who participate in the service. The policymakers plan the budget allocation based on the request and the proposal from service providers. Service providers propose a funding request based on the needs of their communities, and establish local community sites to implement the selected prevention program, once their proposals are accepted. Youths and families can be viewed as customers of the prevention service. This hierarchical structure is illustrated in Figure 3.1.

![Hierarchical overview of planning and delivery of prevention services in Pennsylvania](image)

Figure 3.1 Hierarchical overview of planning and delivery of prevention services in Pennsylvania
(Source: Kang et al. 2016)

However, decisions of a stakeholder in one planning stage are often made without measuring its impact on the other side in a quantitative way. Such impact is not thoroughly investigated and revealed, and most of the part remain as a “black box”. Also, the need to optimize research trials in prevention service has been recognized and techniques to enhance its efficiency have been proposed (Collins et al. 2007).
Since prevention practice largely operates on a limited budget, the implementation of EBP needs to be significantly improved under resource constraints.
Recently, we studied the planning and delivery of prevention services in our previous work, considering the actual impact of funding decision on youths and families in the communities (Kang et al. 2016). In a two-stage, multi-criteria decision making problem, we find the optimal funding allocation of state government and effective prevention programs based on the needs from local communities. Given the state government as a supplier of fund, the financial resources are distributed across the local sites upon request and the proposal. Each prevention service provider delivers prevention services to youths and families, so that they can meet the demand in their communities. The actual estimation of communities’ prevention needs are estimated by investigating the results of a biennial survey, the PA Youth Survey (PAYS). We examined the risk and protective factors, and revealed which communities are at risk. Each prevention program makes an impact on multiple number of risk and protective factors, and the magnitude of the impact on every factors is determined by using analytic hierarchical process (AHP). The optimal solution of the two-stage planning problem prioritizes the prevention programs and assigns the limited budget effectively.

In context of prevention services, however, evaluating the performance of local community sites remains still unknown to prevention practitioners. Thus, we need to measure how efficiently the prevention program is delivered to its customers by the local community sites that are being put into place. This will eventually fill the gap at the community level by clarifying the relationship between implementation of a prevention program and its outcomes. In the following sections, we will investigate the evaluation and monitoring framework that leads the improved efficiency among the sites.

3.2.2 Performance evaluation using DEA

Data Envelopment Analysis (DEA) is a decision support tool to evaluate organizational performance given multiple inputs and multiple outputs. Given \( N \) number of inputs and \( M \) number of outputs, measures of efficiency of organization \( k' \) commonly have the form of a ratio,
\[ \frac{\text{Output}}{\text{Input}} = \frac{\sum_{m=1}^{M} u_m y_{km}}{\sum_{n=1}^{N} v_n x_{kn}}, \quad (3.1) \]

which reflects that smaller inputs and larger outputs are preferable in principle. In a multiple-input-and-multiple-output case, however, the question is how to determine ‘weights’ for the various inputs and outputs. The easiest way is to fix pre-selected weights and yield an index for evaluating efficiencies. This fixed-weight method simplifies the matter, but at the same time can raise significant questions, such as how one can justify \textit{a priori} assumptions about the weights and whether inefficiencies are due to the weights or observations.

In DEA, the weights are variable, and are chosen in a manner that assigns a best set of weights to each organization. An optimization problem is solved to determine the optimal weights for each organization, called a Decision Making Unit (DMU), and hence total \( n \) optimization problems are needed to obtain the values of weights for all inputs and outputs of all organizations. The following mathematical model is a fractional programming problem to obtain the weights of inputs and outputs that maximize the ratio (\( \theta \)) of DMU \( k' \):

\[
\max_{\nu, u} \quad \theta = \frac{\sum_{m=1}^{M} u_m y_{km}}{\sum_{n=1}^{N} v_n x_{kn}} \quad (3.2)
\]

\[ \text{s. t.} \quad \frac{\sum_{m=1}^{M} u_m y_{km}}{\sum_{n=1}^{N} v_n x_{kn}} \leq 1 \quad \forall k = 1, \ldots, K \\

v_n \geq 0 \quad \forall n = 1, \ldots, N \\
u_n \geq 0 \quad \forall n = 1, \ldots, M \]

The equivalent linear optimization problem suggested by Charnes et al. (1978) is as follows:

\[
\max_{\mu, \nu} \quad \theta = \sum_{m=1}^{M} \mu_m y_{km} \quad (3.3)
\]
\[ \begin{align*}
\text{s. t.} \quad \sum_{n=1}^{N} v_{n}x_{kn} &= 1 \\
\sum_{m=1}^{M} \mu_{m}y_{km} &\leq \sum_{n=1}^{N} v_{n}x_{kn} \quad \forall k = 1, \ldots, K \\
v_{n} &\geq 0 \quad \forall n = 1, \ldots, N \\
\mu_{m} &\geq 0 \quad \forall m = 1, \ldots, M
\end{align*} \]

We note that measurement units of inputs and outputs do not need to be congruent; they may involve number of people, or amount of money invested, or other variables, and they may even be unitless.

### 3.2.3 Malmquist Productivity Index

The Malmquist Productivity Index (MPI) is a non-parametric approach to discover the productivity difference of a single organization between two periods of time. The difference is measured by the geometric mean of the Malmquist index at periods t and t+1, both of which estimate the ratio of the distances of each data point relative to a common technology. The index can be constructed either by an output-oriented or by an input-oriented model, where the former uses input distance functions to estimate the productivity index, and the latter consists of output distance function. The output distance function indicates the relative efficiency to the point onto the frontier line, which implies that the inverse of the distance function is the maximum possible increase in outputs with given input level.

The output distance function used for the output-oriented Malmquist productivity index is defined at t as

\[ D_{o}^{t}(x^{t}, y^{t}) = \min \{ \delta | y/\delta \in P^{t}(x) \}, \quad (3.4) \]

where \( P^{t}(x) = \{ y^{t}/x^{t} | x^{t} \text{ produces } x^{t} \text{ in period } t \} \) is the production possibility set as defined by Shepherd (1970), where the technology consists of the set of all possible input/output vectors for each period \( t = 1, \ldots, T \). The value of the output distance function \( D_{o}^{t}(x^{t}, y^{t}) \) varies between 0 and 1 where
$D_o^t(x^t, y^t) < 1$ indicates that DMU is inefficient and under the frontier curve, and $D_o^t(x^t, y^t) = 1$ implies the production technology is efficient and hence on the boundary. In other words, the inverse of the distance function indicates the proportional increase in the output levels for the DMU to be efficient.

Figure 3.2 depicts the product possibility set and the distance function for a DMU with a single input $x$ and two outputs $y_1$ and $y_2$. Points A and B represent the production point of the DMU for periods $t$ and $t+1$, respectively, and the frontiers of periods $t$ and $t+1$ are constructed by other efficient DMUs in periods $t$ and $t+1$, respectively. In this particular example, the DMU seems to be inefficient in both periods $t$ and $t+1$, as it is not on the frontier in either period, even though the DMU is located in a better position in period $t+1$ with improved outputs in both $y_1$ and $y_2$. Since other DMUs, especially those that were efficient, also improved their relative position in period $t+1$, this makes the DMU once again inefficient. Therefore we must determine the impact of the shift in frontier and changes in efficiency on the productivity index. The former is called ‘efficiency change’ or ‘catch-up effect,’ and the latter ‘technology change’ or ‘frontier-shift effect.’ We further discuss these below.
Finally, based on the DMU’s output production points and frontiers, the distance functions are estimated as follows:

\[ D_0^t(x^t, y^t) = \frac{OA}{OC}, \quad D_0^t(x^{t+1}, y^{t+1}) = \frac{OB}{OC}, \quad D_0^{t+1}(x^t, y^t) = \frac{OA}{OD}, \quad \text{and} \quad D_0^{t+1}(x^{t+1}, y^{t+1}) = \frac{OB}{OD}. \]

By using the four distance functions, the output-oriented Malmquist productivity index is calculated by:

\[
M_0(x^t, y^t, x^{t+1}, y^{t+1}) = \sqrt{\frac{D_0^{t+1}(x^{t+1}, y^{t+1}) D_0^t(x^{t+1}, y^{t+1})}{D_0^t(x^t, y^t) D_0^{t+1}(x^t, y^t)}}, \tag{3.5}
\]

where \( D_0^t(y^t, x^t) \) and \( D_0^{t+1}(y^{t+1}, x^{t+1}) \) measure the technical efficiencies for periods \( t \) and \( t + 1 \), respectively, and \( D_0^t(y^{t+1}, x^{t+1}) \) and \( D_0^{t+1}(y^t, x^t) \) indicate inter-temporal efficiency.

The productivity index can be interpreted as the geometric mean of two efficiency ratios: one being the efficiency change measured at period \( t \), and the other the efficiency change measured at period \( t + 1 \). The distance function \( D_0^t(y^{t+1}, x^{t+1}) \) measures an efficiency score based on the observation in period \( t + 1 \) compared to the frontier technology in period \( t \). Similarly, \( D_0^{t+1}(y^t, x^t) \) shows the efficiency measure based on the observation at period \( t \) compared to the frontier technology in period \( t + 1 \). The output MPI equal to 1 indicates no change in productivity, an index greater (less) than one implies productivity growth (decrease).

As we mentioned earlier, the Malmquist productivity index can capture the impact of efficiency change and technical change separately, as the index can also be interpreted as a product of two components: technical efficiency change (EC) from period \( t \) to \( t + 1 \) and technical change (TC) from period \( t \) to \( t + 1 \). The former measures how the DMU catches up to the frontier, while the latter measures the shift in the best-practice technology. The index can be decomposed multiplicatively into two terms as:

\[
M_0(x^t, y^t, x^{t+1}, y^{t+1}) = (EC) \cdot (TC),
\]

where
EC = \frac{D_{o}^{t+1}(x^{t+1},y^{t+1})}{D_{o}^{t}(x^{t},y^{t})}

and

TC = \frac{D_{o}^{t+1}(x^{t+1},y^{t+1}) D_{o}^{t}(x^{t},y^{t})}{D_{o}^{t+1}(x^{t+1},y^{t+1}) D_{o}^{t+1}(x^{t},y^{t})}.

If EC is greater than 1, efficiency is increased from period $t$ to period $t+1$. If EC is equal to 1, there is no change in efficiency from period $t$ to $t+1$. If EC is less than 1, efficiency is decreased from period $t$ to $t+1$. Similarly, TC greater than 1 indicates that efficiency is increased from period $t$ to period $t+1$, while TC equal to 1 and TC less than 1 indicate no change and regress in the relative efficiency, respectively.

To estimate the four distance functions, Färe et al (1994) suggested a DEA-type of distance function based on the constant returns to scale (CRS) assumption. The following four linear programming problems are solved for each DMU to obtain an MPI for a pair of adjacent periods. For each pair of periods, $t$ and $t+1$, the DEA-type distance functions of DMU $k'$ are:

\begin{equation}
\left( D_{o}^{t}(x_{k'}^{t},y_{k'}^{t}) \right)^{-1} = \max \theta
\end{equation}

s.t. \quad x_{kn}^{t} \geq \sum_{k=1}^{K} \lambda_{k} x_{kn}^{t}, \forall n = 1, ..., N

\theta y_{km}^{t} \leq \sum_{k=1}^{K} \lambda_{k} y_{km}^{t}, \forall m = 1, ..., M

\lambda_{k} \geq 0, \forall k = 1, ..., K,

\begin{equation}
\left( D_{o}^{t+1}(x_{k'}^{t+1},y_{k'}^{t+1}) \right)^{-1} = \max \theta
\end{equation}

s.t. \quad x_{kn}^{t+1} \geq \sum_{k=1}^{K} \lambda_{k} x_{kn}^{t}, \forall n = 1, ..., N
\[
\theta y_{km}^{t+1} \leq \sum_{k=1}^{K} \lambda_k y_{km}^t, \forall m = 1, ..., M
\]

\[
\lambda_k \geq 0, \forall k = 1, ..., K,
\]

\[
\left( D_{o}^{t+1}(x_{k'}, y_{k'}^t) \right)^{-1} = \max \theta \tag{3.8}
\]

\[
\text{s.t. } x_{km}^t \geq \sum_{k=1}^{K} \lambda_k x_{kn}^{t+1}, \forall n = 1, ..., N
\]

\[
\theta y_{km}^t \leq \sum_{k=1}^{K} \lambda_k y_{km}^{t+1}, \forall m = 1, ..., M
\]

\[
\lambda_k \geq 0, \forall k = 1, ..., K,
\]

and

\[
\left( D_{o}^{t+1}(x_{k'}, y_{k'}^{t+1}) \right)^{-1} = \max \theta \tag{3.9}
\]

\[
\text{s.t. } x_{km}^{t+1} \geq \sum_{k=1}^{K} \lambda_k x_{kn}^{t+1}, \forall n = 1, ..., N
\]

\[
\theta y_{km}^{t+1} \leq \sum_{k=1}^{K} \lambda_k y_{km}^{t+1}, \forall m = 1, ..., M
\]

\[
\lambda_k \geq 0, \forall k = 1, ..., K,
\]

where \( \lambda \in R^K \) is the intensity vector that forms a production possibility set by a convex combination with data \((x^t, y^t)\).

Models (3.6) and (3.7) estimate the DEA score for DMU \( k' \) of period \( t \) by means of the frontier at periods \( t \) and \( t + 1 \), respectively, while models (3.8) and (3.9) estimate the DEA score for DMU \( k' \) of periods \( t \) and \( t + 1 \) by means of the frontier at period \( t \). Note that the optimal \( \theta^* \) obtained by the optimization model is greater than or equal to 1 in models (3.6) and (3.9), where production points are compared to technologies in the same time period. It is not necessary for \( \theta^* \) to be greater than or equal to
one in models (3.7) and (3.8) as the production point can lie above the production frontier of the other period, as in production point B in Figure 3.2.

3.3 Monitoring Performance

In this section, we present a model to monitor the performance of a DMU in real time. The Malmquist productivity index obtained by formulation (3.5) indicates an increasing or decreasing tendency by estimating a measure of technical efficiency and inter-temporal performance. However, when we want to predict the efficiency of the next period and monitor the trend of performance ‘on the fly,’ the conventional model only provides an inexact evaluation based on estimated input and output values. Furthermore, analyzing the exact impact of changes in variables is even more complicated, because the efficiency score of a DMU is determined not only by itself, but in relation to others, and hence it is unclear whether variable changes in the next period result in changes of overall MPI score, and by how much. We first present a robust MPI model using a robust optimization approach, and suggest an efficiency bound with lower and upper control limits to implement online monitoring.

3.3.1 Robust distance functions

Let us consider a scenario in which a site has a target of 5% decreased input and 5% increased output for the next period. Practically, it is unlikely to see every site achieve its goal. In such cases, the estimation of productivity of a DMU is almost random, because the efficiency of the DMU is relative to others whose outcome is also unknown. Not only is the conventional productivity index estimated after the performance of the site is revealed, but the variables of others are also aggregated. To overcome these limitations, we suggest a prediction model using a robust distance function.

We begin by revisiting the mathematical formulation of the distance function in models (3.6) through (3.9), and consider the robust distance function. In general, robust optimization incorporates parameter uncertainties and provides an optimal solution that is immune to changes in the parameters. Here, in the context of the DEA type of distance function, we assume that the performance of a DMU in period $t + 1$ is unknown, but takes values in the interval set; the input and output variables of period $t + 1$, $x_{kn}^{t+1}$
and \( y_{km}^{t+1} \), have values in the interval \([x_{kn}^{t+1} - \tilde{x}_{kn}^{t+1}, x_{kn}^{t+1} + \tilde{x}_{kn}^{t+1}]\) and \([y_{kn}^{t+1} - \tilde{y}_{kn}^{t+1}, y_{kn}^{t+1} + \tilde{y}_{kn}^{t+1}]\), respectively. Then we need to consider the robust distance functions for \( D_o^{t}(x_{k'}^{t+1}, y_{k'}^{t+1}), D_o^{t+1}(x_{k'}^{t+1}, y_{k'}^{t+1}) \) and \( D_o^{t+1}(x_{k'}^{t+1}, y_{k'}^{t+1}) \) that contain the uncertain terms for period \( t + 1 \).

We begin with the robust transformation of the distance function, \( D_o^{t}(x_{k'}^{t+1}, y_{k'}^{t+1}) \) in model (3.5). Since the inequalities in the constraints should hold for any values within the interval set \([x_{kn}^{t+1} - \tilde{x}_{kn}^{t+1}, x_{kn}^{t+1} + \tilde{x}_{kn}^{t+1}]\) and \([y_{kn}^{t+1} - \tilde{y}_{kn}^{t+1}, y_{kn}^{t+1} + \tilde{y}_{kn}^{t+1}]\), the robust formulation of the inter-temporal distance function is as follows:

\[
(D_o^r(x_{k'}^{t+1}, y_{k'}^{t+1}))^{-1} = \max \theta \tag{3.10}
\]

s.t. \( x_{kn}^{t+1} \geq \sum_{k=1}^{K} \lambda_k x_{kn}^t, \forall n = 1, ..., N \)

\( \theta \gamma_{km}^{t+1} \leq \sum_{k=1}^{K} \lambda_k y_{km}^t, \forall m = 1, ..., M \)

\( \lambda_k \geq 0, k = 1, ..., K, \)

where \( x_{kn}^{t+1} = x_{kn}^{t+1} - \tilde{x}_{kn}^{t+1} \) and \( y_{km}^{t+1} = y_{km}^{t+1} + \tilde{y}_{km}^{t+1} \) for all \( k = 1, ..., K \). Compared to the constraints of the nominal model in (3.5), the input constraint must satisfy at its lower bound, while the output constraint must satisfy at its upper bound. In other words, the robust distance function estimates the efficiency in the worst scenario with minimum possible input and maximum possible output levels in period \( t + 1 \).

Let \( \theta^* \) and \( \hat{\theta}^* \) be the optimal value of models (3.7) and (3.10), respectively. Then we can examine \( \hat{\theta}^* \leq \theta^* \), followed by \( D_o^{t+1}(x_{k'}^{t+1}, y_{k'}^{t+1}) \). Since the distance function measures the maximal proportional change in outputs to make the production \((x^{t+1}, y^{t+1})\) feasible, the relationship indicates that the robust distance function requires greater change to make the production feasible with respect to the technology at period \( t \). This relationship is illustrated in Figure 3.3. The robust distance function is estimated based on the virtual production point located in the upper left corner of the set. In both
cases, the robust distance function is defined as $\hat{D}_o^t(x_{k_{k'}}^{t+1}, y_{k_{k'}}^{t+1}) = \frac{y^{t+1}}{y^D}$, while the nominal distance function is $D_o^t(x_{k_{k'}}^{t+1}, y_{k_{k'}}^{t+1}) = \frac{y^{t+1}}{y^C}$. Since $y^D \leq y^C$ and $\frac{y^{t+1}}{y^D} \geq \frac{y^{t+1}}{y^C}$, the robust distance function is greater than or equal to the nominal distance function.

![Graphical illustration of the robust distance function](image)

**Figure 3.3** Graphical illustration of the robust distance function

**Lemma 3.1** The robust distance function $\hat{D}_o^t(x_{k_{k'}}^{t+1}, y_{k_{k'}}^{t+1})$ with interval uncertainty in period $t+1$ is greater than or equal to the nominal distance function.

$$\hat{D}_o^t(x_{k_{k'}}^{t+1}, y_{k_{k'}}^{t+1}) \geq D_o^t(x_{k_{k'}}^{t+1}, y_{k_{k'}}^{t+1}).$$

So far, we have discussed the robust distance of one inter-temporal function, which is the relative distance of the production point of period $t+1$ to the frontier in period $t$. We now consider the other inter-temporal measure presented in model (3.8). In this case, the frontier of period $t+1$ needs to be reformulated in order to compensate for the uncertainty in the next period. Thus, the robust optimal distance function that will be obtained by the robust transformation indicates the relative distance from the production point of period $t$ to the frontier in period $t+1$. Here we assume that only a subset of input and output variables changes simultaneously. Rather than providing the full protection against changes in all variables, the robust transformation will take into account only a selected number of uncertain variables. This concept of
robustness is first suggested by Bertsimas and Sim (2004), who tackle the cardinality uncertainty to adjust the level of conservativeness.

Let us assume that \( J^x_n \) and \( J^y_m \) are the index sets of uncertain input and output variables, respectively. We introduce parameters \( \gamma^x_n \) and \( \gamma^y_m \), representing the number of uncertain input and output variables in period \( t+1 \). The values of the parameters are in the bounded intervals \([0, |J^x_n|]\) and \([0, |J^y_m|]\), respectively, where \(|\cdot|\) is the cardinality of a set. That is, among the input and output variables \( x_{kn}^{t+1} \) and \( y_{km}^{t+1} \), the robust formulation only provides protection up to the values of \( \gamma^x_n \) and \( \gamma^y_m \), respectively. The values are not necessarily integers, and fractional protection is guaranteed against changes in \( x_{kn}^{t+1} \), \( y_{km}^{t+1} \) by \((\gamma^x_n - \lfloor \gamma^x_n \rfloor) \hat{x}_{kn}^t \) and \((\gamma^y_m - \lfloor \gamma^y_m \rfloor) \hat{y}_{km}^t\), respectively. Thus, the role of parameters \( \gamma^x_n \) and \( \gamma^y_m \) is to control the level of robustness, based on the assumption that it is less likely that all DMUs deviate beyond their targeted input and output values at the same time.

The robust transformation of model (3.8) is as follows:

\[
(\hat{B}_{o}^{t+1}(x_k^t,y_k^t))^{-1} = \max \quad \theta \tag{3.11}
\]

subject to

\[
x_{km}^t \geq \sum_{k=1}^{K} \lambda_k x_{kn}^{t+1} + \beta^x_n, \forall n = 1, \ldots, N
\]

\[
\theta y_{km}^t \leq \sum_{k=1}^{K} \lambda_k y_{kn}^{t+1} + \beta^y_m, \forall m = 1, \ldots, M
\]

where

\[
\beta^x_n = \max_{c_n^x} \left\{ \sum_{k \in S_n^x} \lambda_k \left( x_{kn}^{t+1} - x_{kn}^t \right) + (\gamma^x_n - \lfloor \gamma^x_n \rfloor) \lambda_{t_n^x} \left( x_{t_n^x,n}^{t+1} - x_{t_n^x,n}^t \right) \right\},
\]

\[
c_n^x = \{ S_n^x \cup \{t_n^x\} | S_n^x \subseteq J^x_n, |S_n^x| \leq |J^x_n|, t_n^x \in J^x_n \setminus S_n^x \}.
\]
\[ \beta_m^y = \max_{c_m^y} \left\{ \sum_{k \in S_m^y} \lambda_k \left( \bar{y}_{km}^{t+1} - y_{km}^{t+1} \right) + \left( r_m^y - |r_m^y| \right) \lambda_{t_m^y} \left( \bar{y}_{t_m^ym}^{t+1} - y_{t_m^ym}^{t+1} \right) \right\} \]

and

\[ C_m^y = \left\{ S_m^y \cup \{ t_m^y \} | \sum_{k \in S_m^y} \lambda_k \leq |r_m^y|, t_m^y \in J_m \setminus S_m^y \right\}. \]

Applying the proposition of Bertsimas and Sim (2004) to transform the nonlinear model (3.11) into the following linear model:

\[
\left( \tilde{D}_{o}^{t+1}(x_{k^t}, y_{k^t}) \right)^{-1} = \max \theta (3.12)
\]

s.t. \[ x_{km}^{t} \geq \sum_{k=1}^{K} \lambda_{k} x_{km}^{t+1} + z_n x_n^{t} + \sum_{k=1}^{K} p_{kn}, \forall n = 1, ..., N \]

\[ \theta y_{km}^{t} \leq \sum_{k=1}^{K} \lambda_{k} y_{km}^{t+1} - z_m y_m^{t} - \sum_{k=1}^{K} q_{km}, \forall m = 1, ..., M \]

\[ z_n^{x} + p_{kn} \geq \lambda_{k} x_{kn}^{t+1}, \forall k = 1, ..., K, \forall n = 1, ..., N \]

\[ z_m^{y} + q_{km} \geq \lambda_{k} y_{kn}^{t+1}, \forall k = 1, ..., K, \forall m = 1, ..., M \]

\[ \lambda_{k} \geq 0, \forall k = 1, ..., K, \]

\[ z_n^{x}, z_m^{y} \geq 0, \forall n = 1, ..., N, \forall m = 1, ..., N, \]

\[ p_{kn}, q_{km} \geq 0, \forall k = 1, ..., K, \forall n = 1, ..., N, \forall m = 1, ..., M, \]

where \( p_{kn} \) and \( q_{km} \) are auxiliary decision variables and \( x_n^{X} \) and \( y_n^{X} \) are robust parameters. Model (3.12) solves the DEA score of site \( k^t \) by measuring based on the projected frontier in period \( t+1 \).

Similarly, the distance function \( \tilde{D}_{o}^{t+1}(x_{k^t+1}, y_{k^t+1}) \) has the following linear formulations:

\[
\left( \tilde{D}_{o}^{t+1}(x_{k^t+1}, y_{k^t+1}) \right)^{-1} = \max \theta (3.13)
\]

s.t. \[ x_{kn}^{t+1} \geq \sum_{k=1}^{K} \lambda_{k} x_{kn}^{t+1} + z_n x_n^{t} + \sum_{k=1}^{K} p_{kn} + p_{n}^0, \forall n = 1, ..., N \]
\[ \theta y_{k'}^{t+1} \leq \sum_{k=1}^{K} \lambda_k y_{kn}^{t+1} - z_{m}^{t+1} - \sum_{k=1}^{K} q_{km} - q_{m}^{0}, \forall m \]
\[ = 1, ..., M \]
\[ z_{n}^{x} + p_{kn} \geq \lambda_k z_{kn}^{t+1}, \forall k = 1, ..., K, \forall n = 1, ..., N \]
\[ z_{n}^{x} + p_{n}^{0} \geq z_{kn}^{t+1}, \forall n = 1, ..., N \]
\[ z_{m}^{y} + q_{km} \geq \lambda_k z_{kn}^{t+1}, \forall k = 1, ..., K, \forall m = 1, ..., M \]
\[ z_{m}^{y} + q_{m}^{0} \geq \theta z_{kn}^{t+1}, \forall m = 1, ..., M \]
\[ \lambda_k \geq 0, \forall k = 1, ..., K, \]
\[ z_{n}^{x}, z_{m}^{y}, p_{n}^{0}, q_{m}^{0} \geq 0, \forall n = 1, ..., N, \forall m = 1, ..., N, \]
\[ p_{kn}, q_{km} \geq 0, \forall k = 1, ..., K, \forall n = 1, ..., N, \forall m = 1, ..., M. \]

Note that model (3.13) has two more auxiliary constraints, as compared to model (3.12). Again, the robust optimal distance function obtained by this formulation measures the technical efficiency of DMU \( k' \) relative to the frontier constructed by other efficient DMUs within the same period \( t+1 \). While the nominal distance function \( D_{o}^{t+1}(x_{k'}^{t+1}, y_{k'}^{t+1}) \) cannot be greater than 1 as \( \theta \) is greater than or equal to 1, the robust distance function \( \hat{D}_{o}^{t+1}(x_{k'}^{t+1}, y_{k'}^{t+1}) \) can be greater than 1, as both the data point of DMU \( k' \) and the frontier are determined upon the uncertainty sets.

**Lemma 3.2** The robust distance functions \( \hat{D}_{o}^{t+1}(x_{k'}^{t+1}, y_{k'}^{t+1}) \) and \( \hat{D}_{o}^{t+1}(x_{k'}^{t+1}, y_{k'}^{t+1}) \) are greater than or equal to the nominal distance functions:

\[ \hat{D}_{o}^{t+1}(x_{k'}^{t}, y_{k'}^{t}) \geq D_{o}^{t+1}(x_{k'}^{t}, y_{k'}^{t}) \text{ and } \hat{D}_{o}^{t+1}(x_{k'}^{t+1}, y_{k'}^{t+1}) \geq D_{o}^{t+1}(x_{k'}^{t+1}, y_{k'}^{t+1}). \]

Finally, we obtain the following robust productivity index (RPI) based on the robust distance functions, (3.10), (3.12) and (3.13).

**Definition 3.1** The robust productivity index (RPI) is defined as
\[
\hat{M}_o(x^t, y^t, x^{t+1}, y^{t+1}) = \sqrt{\frac{\hat{D}_o^t(x^{t+1}, y^{t+1}) \hat{D}_o^{t+1}(x^{t+1}, y^{t+1})}{D_o^t(x^t, y^t) D_o^{t+1}(x^t, y^t)}}, \tag{3.14}
\]

where \(\hat{D}_o^{t+1}(x^{t+1}, y^{t+1})\) are robust measures of technical efficiency and \(\hat{D}_o^t(x^{t+1}, y^{t+1})\) and \(\hat{D}_o^{t+1}(x^t, y^t)\) are robust measures of inter-temporal distance function, all of which are based on the output-oriented evaluation.

**Theorem 3.1** The robust productivity index (RPI) is greater than or equal to the Malmquist productivity index (MPI):

\[
\hat{M}_o(x^t, y^t, x^{t+1}, y^{t+1}) \geq M_o(x^t, y^t, x^{t+1}, y^{t+1})
\]

### 3.3.2 Constructing productivity bounds

We next consider a bounded interval of the productivity index. The upper and lower bounds represent the productivity limits that a DMU can achieve under a given uncertainty intervals of the input and output variables. The robust productivity index discussed in the previous section suggests a productivity score that hedges against uncertain data in the next period. Based on the robust index formulation, we build a bounded interval of productivity by fixing the input and output values of the DMU at their limits. The efficiency of the DMU becomes greater with less input and more output, smaller with larger input and less output. Therefore, the upper limit of the productivity bound is obtained from the optimistic scenario and the lower limit from the pessimistic scenario.

The lower limit of the bounded interval is defined by the following linear programming formulations:

\[
\hat{M}_o(x^t, y^t, x^{t+1}, y^{t+1}) = \sqrt{\frac{D_o^t(x^t, y^t)}{D_o^t(x^t, y^t)}}
\]

where
\[(D_0^t(\bar{x}^{t+1}_{k'}, \bar{y}^{t+1}_{k'}))^{-1} = \max \theta \]

s.t. \[\bar{x}^{t+1}_{kn} \geq \sum_{k=1}^{K} \lambda_k x^t_{kn}, \forall n = 1, \ldots, N\]

\[\theta \bar{y}^{t+1}_{km} \leq \sum_{k=1}^{K} \lambda_k y^t_{km}, \forall m = 1, \ldots, M\]

\[\lambda_k \geq 0, k = 1, \ldots, K,\]

\[(D_0^t(\bar{x}^{t+1}_{k'}, \bar{y}^{t+1}_{k'}))^{-1} = \max \theta \]

s.t. \[x^t_{kn} \geq \sum_{k \in \{1, \ldots, K\}\setminus\{k'\}} \lambda_k x^t_{kn} + \lambda_{k'} \bar{x}^{t+1}_{kn} + z^x n^x + \sum_{k \in \{1, \ldots, K\}\setminus\{k'\}} p_{kn}, \forall n = 1, \ldots, N\]

\[\theta y^t_{km} \leq \sum_{k \in \{1, \ldots, K\}\setminus\{k'\}} \lambda_k y^t_{km} + \lambda_{k'} \bar{y}^{t+1}_{km} - z^y m^y - \sum_{k \in \{1, \ldots, K\}\setminus\{k'\}} q_{km}, \forall m = 1, \ldots, M\]

\[z^x_n + p_{kn} \geq \lambda_k x^t_{kn}, \forall k \in \{1, \ldots, K\}\setminus\{k'\}, \forall n = 1, \ldots, N\]

\[z^y_m + q_{km} \geq \lambda_k y^t_{km}, \forall k \in \{1, \ldots, K\}\setminus\{k'\}, \forall m = 1, \ldots, M\]

\[\lambda_k \geq 0, \forall k = 1, \ldots, K,\]

\[z^x_n, z^y_m \geq 0, \forall n = 1, \ldots, N, \forall m = 1, \ldots, N,\]

\[p_{kn}, q_{km} \geq 0, \forall k \in \{1, \ldots, K\}\setminus\{k'\}, \forall n = 1, \ldots, N, \forall m = 1, \ldots, M,\]

and

\[(D_0^{t+1}(\bar{x}^{t+1}_{k'}, \bar{y}^{t+1}_{k'}))^{-1} = \max \theta \]

s.t. \[\bar{x}^{t+1}_{kn} \geq \sum_{k \in \{1, \ldots, K\}\setminus\{k'\}} \lambda_k x^t_{kn} + \lambda_{k'} \bar{x}^{t+1}_{kn} + z^x n^x + \sum_{k \in \{1, \ldots, K\}\setminus\{k'\}} p_{kn}, \forall n = 1, \ldots, N\]
\[
\theta \frac{y_{km}^{t+1}}{y_{km}^t} \leq \sum_{k \in \{1, \ldots, K\} \setminus \{k\}} \lambda_k y_{km}^{t+1} + \lambda_{k'} y_{km}^{t+1} - z_{m}^{y_{km}^t} - \sum_{k \in \{1, \ldots, K\} \setminus \{k\}} q_{km}, \forall m = 1, \ldots, M
\]

\[
z_{n}^{x} + p_{kn} \geq \lambda_{k} \hat{x}_{km}^{t+1}, \forall k \in \{1, \ldots, K\} \setminus \{k'\}, \forall n = 1, \ldots, N
\]

\[
z_{m}^{y} + q_{km} \geq \lambda_{k} \hat{y}_{km}^{t+1}, \forall k \in \{1, \ldots, K\} \setminus \{k'\}, \forall m = 1, \ldots, M
\]

\[
\lambda_{k} \geq 0, \forall k = 1, \ldots, K,
\]

\[
z_{n}^{x}, z_{m}^{y} \geq 0, \forall n = 1, \ldots, N, \forall m = 1, \ldots, N,
\]

\[
p_{kn}, q_{km} \geq 0, \forall k \in \{1, \ldots, K\} \setminus \{k'\}, \forall n = 1, \ldots, N, \forall m = 1, \ldots, M.
\]

Similarly, the upper productivity limit is defined as

\[
\overline{M}_o \left( x_{kt}^t, y_{kt}^t, x_{kt}^{t+1}, y_{kt}^{t+1} \right) = \sqrt{\frac{D_o \left( x_{kt}^{t+1}, y_{kt}^{t+1} \right) D_o \left( x_{kt}^{t+1}, y_{kt}^{t+1} \right)}{D_o \left( x_{kt}^{t}, y_{kt}^t \right) D_o \left( x_{kt}^{t}, y_{kt}^t \right)}}
\]

(3.19)

**Theorem 3.2** The upper productivity limit is greater than or equal to the lower productivity limit:

\[
\overline{M}_o \left( x_{kt}^t, y_{kt}^t, x_{kt}^{t+1}, y_{kt}^{t+1} \right) \geq M_o \left( x_{kt}^t, y_{kt}^t, x_{kt}^{t+1}, y_{kt}^{t+1} \right).
\]

The lower productivity bound (3.15) yields the minimum productivity where a DMU is assumed to radially produce less output with more input, while only a restricted number of sites performs at its estimated bound. Similarly, the upper bound is calculated by (3.19), with an estimate of more input and less output for the next period from the DMU. Bounds for EC and TC are constructed in a similar fashion, using the estimated distance functions.

### 3.4 An Application of Community Youth Prevention Programs

This section illustrates the proposed approach using the data from a real community youth prevention program. We begin with an overview of the selected intervention program and acquisition of data. Input and output variables collected throughout the lifetime of the program will be explained. Among the input and output variables, we first select key variables based on the selective DEA model. Using
selected variables, we perform an evaluation of the productivity of all providers and present a monitoring scheme for the time-varying efficiency of the sites.

### 3.4.1 Data and variables

The data set used in this study was obtained from one of three well-established intervention programs supported by the Evidence-based Prevention and Intervention Support Center (EPISCENTER), a collaborative partnership between the Pennsylvania Commission on Crime and Delinquency (PCCD) and the Prevention Research Center at Penn State University.\(^1\) Among the menu of evidence-based prevention and intervention programs served by EPISCENTER, we single out one intervention program that focuses on youth with behavioral offenses and substance abuse, and addresses the acting-out behavior of youth by improving family functioning. Intervention is typically conducted in family’s home by a trained therapist as a home-based model. A team accepts dysfunctional or at-risk youth aged 11-18 and their families in the treatment process and delivers the program over 3-5 months through qualified therapists. On average, a small site of two full-time therapists for the community-based program serves 90 youths per year, while a large site with eight clinicians can serve almost 300 clients per year.

Data in this study comes from the INtegrated System for Program Implementation and Real-time Evaluation (INSPIRE), a web-based data collection and reporting system, developed by the Penn State College of Engineering and EPISCENTER with funding from PCCD. The system enables providers to easily collect, store, document, analyze and report data, and facilitates model fidelity and successful outcomes across providers. Data in INSPIRE comes from two sources: regular import of national data on the program and direct data entry from therapists using a web-based interface. Variables in the database include demographic information on youth, treatment outcomes, and provider-level information. A report is generated periodically to summarize the treatment outcomes of youths, including youth and family satisfaction about the program experience, improved family functioning, successful discharge rate,

\(^{1}\) For more information, please refer to the EPISCENTER at [http://www.episcenter.psu.edu/](http://www.episcenter.psu.edu/).
successful completion rate, and the number of youth who remain drug-free, avoid re-arrest, show no new offenses, remain in the community, and improve school attendance and performance.

Among the outcome variables, we select some key variables to evaluate longitudinal productivity, as summarized in Table 3.1. The set of input variables contains the number of participants: number of youth served, number of new enrollments, and number of new families. The output variables include the number of discharged cases (number of youth discharged) and outcomes at discharge (percentage of youths successfully discharged, percentage of youths who complete the program, percentage of youths with no new offenses, percentage of youths remaining in the community, and percentage of youths improving family functioning).

Table 3.1 Variables

| N1   | Number of youth served                      |
| N2   | Number of new youth enrolled               |
| N3   | Number of new parents enrolled             |
| N4   | Number of youth clinically discharged      |
| N5   | Number of therapists with Global Therapist Review (GTR) |
| N6   | Number of therapists satisfying minimum fidelity standards |
| N7   | Number of therapists satisfying minimum adherence standards |
| P1   | Percentage of youth satisfied              |
| P2   | Percentage of families satisfied           |
| P3   | Percentage of youth who completed the program |
| P4   | Percentage of successful discharges        |
| P5   | Percentage of youth with no new offenses   |
| P6   | Percentage of youth living in community    |
| P7   | Percentage of youth improving school attendance |
| P8   | Percentage of youth improving academic performance |
| P9   | Percentage of families with improved functioning |
Note that N1 counts the total number of cases open and active during the period, and hence one youth can be counted multiple times in different periods until s/he completes the program. Variables N2 and N3 are the number of referred youth and families who enroll in the program and have a first session during the period. N4 counts only youth who are clinically discharged, that is, youth with the opportunity to complete treatment, who are not discharged for administrative reasons unrelated to case progress. Variables N5 through N7 are related to implementation quality, demonstrating the qualification of therapists who complete required training and certification. Site supervisors complete weekly ratings of therapist fidelity (N5) and dissemination adherence (N6), and these ratings are averaged to Global Therapist Review (GTR) (N7) three times a year.

P1 and P2 are the percentage of youth and families who make positive answers in questionnaires on the program. P3 measures the rate of discharges among youths who completed treatment, while P4 measures the rate of successful discharges. Percentage variables P5 through P9 are performance measures of proximal outcomes for all clinically discharged youth. P5 indicates decreases in delinquent behavior and general behavior problems, while P6 is the percentage of youth living in the community at discharge. School-related measures P7 and P8 indicate positive results in school attendance and performance, respectively, but only include youth presenting with concerns in those areas at enrollment. As an indicator of improvement in family functioning, P9 reports increased family communication and cohesion, and less verbal aggression and conflict.
<table>
<thead>
<tr>
<th>FY</th>
<th>Quarter</th>
<th>N1</th>
<th>N2</th>
<th>N3</th>
<th>N4</th>
<th>P1</th>
<th>P2</th>
<th>P3</th>
<th>P4</th>
<th>N5</th>
<th>N6</th>
<th>N7</th>
<th>P5</th>
<th>P6</th>
<th>P7</th>
<th>P8</th>
<th>P9</th>
</tr>
</thead>
<tbody>
<tr>
<td>2010</td>
<td>1</td>
<td>713</td>
<td>344</td>
<td>0</td>
<td>279</td>
<td>81.3</td>
<td>68.8</td>
<td>71.0</td>
<td>59.1</td>
<td>14</td>
<td>9</td>
<td>12</td>
<td>78.5</td>
<td>88.5</td>
<td>20.0</td>
<td>0.0</td>
<td>70.4</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>716</td>
<td>321</td>
<td>0</td>
<td>317</td>
<td>90.9</td>
<td>90.9</td>
<td>71.6</td>
<td>54.3</td>
<td>23</td>
<td>15</td>
<td>20</td>
<td>75.9</td>
<td>87.4</td>
<td>33.3</td>
<td>75.0</td>
<td>60.9</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>655</td>
<td>307</td>
<td>0</td>
<td>283</td>
<td>76.5</td>
<td>94.1</td>
<td>66.8</td>
<td>55.5</td>
<td>24</td>
<td>16</td>
<td>22</td>
<td>83.9</td>
<td>85.5</td>
<td>40.0</td>
<td>33.3</td>
<td>66.1</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>694</td>
<td>350</td>
<td>0</td>
<td>277</td>
<td>90.9</td>
<td>94.3</td>
<td>77.6</td>
<td>63.2</td>
<td>27</td>
<td>17</td>
<td>23</td>
<td>82.7</td>
<td>88.1</td>
<td>68.8</td>
<td>52.6</td>
<td>69.1</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>1691</td>
<td>1322</td>
<td>0</td>
<td>1156</td>
<td>87.9</td>
<td>90.9</td>
<td>71.7</td>
<td>57.9</td>
<td>42</td>
<td>25</td>
<td>35</td>
<td>80.0</td>
<td>87.4</td>
<td>57.8</td>
<td>47.2</td>
<td>66.3</td>
</tr>
<tr>
<td>2011</td>
<td>1</td>
<td>627</td>
<td>259</td>
<td>0</td>
<td>248</td>
<td>93.8</td>
<td>96.0</td>
<td>75.8</td>
<td>64.9</td>
<td>24</td>
<td>14</td>
<td>19</td>
<td>87.5</td>
<td>91.9</td>
<td>60.0</td>
<td>61.5</td>
<td>72.5</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>621</td>
<td>288</td>
<td>0</td>
<td>237</td>
<td>93.3</td>
<td>96.2</td>
<td>67.1</td>
<td>59.1</td>
<td>39</td>
<td>25</td>
<td>30</td>
<td>82.9</td>
<td>89.0</td>
<td>62.5</td>
<td>66.0</td>
<td>63.0</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>618</td>
<td>271</td>
<td>0</td>
<td>254</td>
<td>88.5</td>
<td>91.4</td>
<td>76.4</td>
<td>67.3</td>
<td>29</td>
<td>23</td>
<td>24</td>
<td>81.0</td>
<td>88.2</td>
<td>60.5</td>
<td>64.9</td>
<td>69.9</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>554</td>
<td>238</td>
<td>162</td>
<td>234</td>
<td>91.4</td>
<td>94.0</td>
<td>80.8</td>
<td>73.1</td>
<td>17</td>
<td>16</td>
<td>14</td>
<td>89.0</td>
<td>91.9</td>
<td>63.2</td>
<td>63.5</td>
<td>73.6</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>1424</td>
<td>1056</td>
<td>891</td>
<td>973</td>
<td>91.4</td>
<td>94.1</td>
<td>75.0</td>
<td>66.1</td>
<td>51</td>
<td>32</td>
<td>36</td>
<td>85.0</td>
<td>90.2</td>
<td>61.8</td>
<td>64.2</td>
<td>69.9</td>
</tr>
<tr>
<td>2012</td>
<td>1</td>
<td>530</td>
<td>248</td>
<td>188</td>
<td>206</td>
<td>91.4</td>
<td>93.6</td>
<td>82.5</td>
<td>79.6</td>
<td>25</td>
<td>19</td>
<td>24</td>
<td>89.7</td>
<td>91.3</td>
<td>58.3</td>
<td>68.2</td>
<td>77.3</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>508</td>
<td>223</td>
<td>197</td>
<td>177</td>
<td>94.2</td>
<td>93.0</td>
<td>75.7</td>
<td>67.2</td>
<td>36</td>
<td>20</td>
<td>31</td>
<td>84.1</td>
<td>86.4</td>
<td>59.3</td>
<td>61.5</td>
<td>78.4</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>555</td>
<td>254</td>
<td>210</td>
<td>198</td>
<td>89.4</td>
<td>89.3</td>
<td>78.3</td>
<td>72.2</td>
<td>19</td>
<td>12</td>
<td>17</td>
<td>87.8</td>
<td>89.9</td>
<td>67.7</td>
<td>72.7</td>
<td>78.7</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>585</td>
<td>273</td>
<td>301</td>
<td>186</td>
<td>89.9</td>
<td>89.9</td>
<td>76.9</td>
<td>69.9</td>
<td>21</td>
<td>16</td>
<td>18</td>
<td>85.0</td>
<td>90.9</td>
<td>71.0</td>
<td>65.9</td>
<td>70.6</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>1280</td>
<td>998</td>
<td>896</td>
<td>767</td>
<td>91.0</td>
<td>91.2</td>
<td>78.5</td>
<td>72.5</td>
<td>41</td>
<td>22</td>
<td>33</td>
<td>86.8</td>
<td>89.7</td>
<td>64.6</td>
<td>67.6</td>
<td>76.2</td>
</tr>
<tr>
<td>2013</td>
<td>1</td>
<td>595</td>
<td>250</td>
<td>216</td>
<td>208</td>
<td>92.1</td>
<td>94.4</td>
<td>83.2</td>
<td>74.5</td>
<td>16</td>
<td>14</td>
<td>16</td>
<td>89.7</td>
<td>93.3</td>
<td>69.0</td>
<td>68.3</td>
<td>78.3</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>561</td>
<td>210</td>
<td>213</td>
<td>230</td>
<td>88.7</td>
<td>94.3</td>
<td>80.0</td>
<td>72.6</td>
<td>14</td>
<td>12</td>
<td>12</td>
<td>87.6</td>
<td>92.2</td>
<td>53.7</td>
<td>45.9</td>
<td>69.2</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>523</td>
<td>227</td>
<td>248</td>
<td>181</td>
<td>83.8</td>
<td>88.0</td>
<td>69.6</td>
<td>62.4</td>
<td>12</td>
<td>9</td>
<td>8</td>
<td>81.5</td>
<td>89.5</td>
<td>51.4</td>
<td>43.2</td>
<td>61.9</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>555</td>
<td>236</td>
<td>281</td>
<td>211</td>
<td>85.4</td>
<td>91.0</td>
<td>72.5</td>
<td>66.4</td>
<td>33</td>
<td>25</td>
<td>29</td>
<td>88.8</td>
<td>88.6</td>
<td>58.0</td>
<td>55.4</td>
<td>66.3</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>1268</td>
<td>923</td>
<td>958</td>
<td>830</td>
<td>87.7</td>
<td>92.1</td>
<td>76.6</td>
<td>69.3</td>
<td>38</td>
<td>28</td>
<td>33</td>
<td>87.1</td>
<td>91.0</td>
<td>58.8</td>
<td>53.9</td>
<td>69.2</td>
</tr>
<tr>
<td>2014</td>
<td>1</td>
<td>552</td>
<td>243</td>
<td>274</td>
<td>208</td>
<td>96.6</td>
<td>95.2</td>
<td>75.5</td>
<td>72.6</td>
<td>34</td>
<td>23</td>
<td>32</td>
<td>86.2</td>
<td>89.9</td>
<td>63.3</td>
<td>50.0</td>
<td>73.6</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>526</td>
<td>207</td>
<td>233</td>
<td>204</td>
<td>94.2</td>
<td>94.8</td>
<td>78.4</td>
<td>73.5</td>
<td>10</td>
<td>8</td>
<td>10</td>
<td>88.5</td>
<td>90.2</td>
<td>60.4</td>
<td>67.7</td>
<td>70.2</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>499</td>
<td>213</td>
<td>239</td>
<td>171</td>
<td>97.4</td>
<td>96.6</td>
<td>83.0</td>
<td>76.6</td>
<td>14</td>
<td>12</td>
<td>9</td>
<td>89.7</td>
<td>90.6</td>
<td>53.9</td>
<td>53.3</td>
<td>75.6</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>514</td>
<td>221</td>
<td>220</td>
<td>172</td>
<td>93.8</td>
<td>93.8</td>
<td>83.7</td>
<td>79.1</td>
<td>22</td>
<td>20</td>
<td>20</td>
<td>89.9</td>
<td>93.6</td>
<td>74.1</td>
<td>70.0</td>
<td>77.8</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>1193</td>
<td>884</td>
<td>966</td>
<td>755</td>
<td>95.5</td>
<td>95.1</td>
<td>79.9</td>
<td>75.2</td>
<td>45</td>
<td>35</td>
<td>36</td>
<td>88.5</td>
<td>91.0</td>
<td>62.7</td>
<td>60.1</td>
<td>74.1</td>
</tr>
<tr>
<td>2015</td>
<td>1</td>
<td>544</td>
<td>233</td>
<td>247</td>
<td>191</td>
<td>88.4</td>
<td>89.2</td>
<td>76.4</td>
<td>70.2</td>
<td>19</td>
<td>19</td>
<td>17</td>
<td>85.9</td>
<td>90.6</td>
<td>54.6</td>
<td>54.4</td>
<td>69.8</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>522</td>
<td>197</td>
<td>186</td>
<td>176</td>
<td>95.2</td>
<td>94.4</td>
<td>83.0</td>
<td>81.3</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>87.9</td>
<td>90.3</td>
<td>58.8</td>
<td>63.2</td>
<td>70.2</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>525</td>
<td>205</td>
<td>201</td>
<td>188</td>
<td>90.3</td>
<td>95.6</td>
<td>78.2</td>
<td>76.1</td>
<td>21</td>
<td>17</td>
<td>19</td>
<td>88.0</td>
<td>91.0</td>
<td>73.7</td>
<td>70.0</td>
<td>72.1</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>519</td>
<td>213</td>
<td>193</td>
<td>149</td>
<td>92.3</td>
<td>94.4</td>
<td>77.2</td>
<td>70.5</td>
<td>12</td>
<td>11</td>
<td>12</td>
<td>85.5</td>
<td>89.9</td>
<td>72.2</td>
<td>57.1</td>
<td>68.7</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>1159</td>
<td>848</td>
<td>827</td>
<td>704</td>
<td>91.5</td>
<td>93.2</td>
<td>78.7</td>
<td>74.6</td>
<td>30</td>
<td>27</td>
<td>26</td>
<td>86.9</td>
<td>90.5</td>
<td>61.7</td>
<td>61.2</td>
<td>70.3</td>
</tr>
</tbody>
</table>
We will refer to quarterly data on the prevention program from the first quarter of FY 2010, starting on 1 July 2010 and including data through the last quarter of FY 2015, ending June 30 2016. Analysis of time-dependent efficiency is based on the total 16 sites that implemented the prevention program during that period. Not all sites were active during the whole period: some of them stopped operating upon termination of funding, while others started the program in the middle of the period. Table 3.2 presents statistics on the data over the period. The quarterly data indicate that the number of participants in the program gradually decreases as N1 and N2 decrease over the years. Served by 10.17 opening sites on average, the six-year averages of N1 and N2 are 575.46 and 251.29, respectively. Similarly, N4 declines over the period, with a six-year average of 216.04 youths. Note that there is no N3 data for the first six quarters, but we do not know whether this is because the data were not collected properly or simply because there were no new families during the period.

![Figure 3.4 Desired clinical discharges](image)
Figure 3.5 Proximal outcomes at discharge

Completion rate (P3) reflects the percentage of clinically discharged youth who completed the program, while the rate of successful discharge (P4) is the percentage of youth whose discharge outcome is successful among completers. Over the six-year period, among 5,185 youth who were clinically discharged from the program, 3,954 youth (76.3%) completed the three phases of the program and 3,536 youth (68.2%) were successfully discharged. Figure 3.4 demonstrates that across providers, quarterly average P3 and P4 ranged from 66.8% to 83.7% and from 54.3% to 81.3%, respectively during the six fiscal years.

Figure 3.5 presents treatment outcomes reported for all clinically discharged youth. While P5 and P6 are measures of youth who had concerns at intake, and indicate achievement of goals at a relatively high rate, the other three indicators, P7, P8, and P9, for improved academic attendance/performance and family functioning, fluctuate at lower percentages. Moreover, P7 and P8 show similar tendencies, indicating that school attendance and academic performance are highly correlated.
3.4.2 Results

The data set presents some challenging issues. First, there is no consensus as to the key input and output variables. Furthermore, there are some variables with zeroes, partly because the variable entry is mainly dependent on the data entry by therapists through the system. The efficiency scores obtained from such data may be less reliable than scores from more complete data. Technically speaking, when the number of performance measures is relative large compared to the number of active sites there is a problem involving degrees of freedom and relative efficiency. The following rule of thumb for the envelopment model is suggested by Cooper et al. (2007).

\[ K \geq \max\{3(M + N), MN\}. \]  

(3.20)

To address these issues we start by identifying key input and output variables using the selective DEA model proposed by Toloo et al. (2015). They suggest an MILP model by imposing the maximum number of input and output variables holds the condition. Assuming that selecting N2 as an input variable is inevitable, we apply the selective model for each quarter. Given that the number of sites satisfying \( K \geq 3 \ (M + N) \), we solve the following selective DEA model that maximizes the aggregate efficiency:

\[
\max_{\mu, \nu} \theta = \sum_{m=1}^{M} \mu_m \bar{y}_m \\
\text{s. t.} \quad \sum_{n=1}^{N} v_n x_n = 1 \\
\sum_{m=1}^{M} \mu_m y_{km} \leq \sum_{n=1}^{N} v_n x_{kn} \quad \forall k = 1, \ldots, K \\
\sum_{m=1}^{M} q_m + \sum_{n=1}^{N} p_n \leq \left\lfloor \frac{K}{3} \right\rfloor \\
\sum_{m=1}^{M} q_m \geq 1
\]  

(3.21)
\[ \epsilon p_n \leq v_n \leq Lp_n \quad \forall n = 1, \ldots, N \]

\[ \epsilon q_m \leq \mu_m \leq Lq_m \quad \forall m = 1, \ldots, M \]

\[ v_n \geq \epsilon \quad \forall n = 1, \ldots, N \]

\[ \mu_m \geq \epsilon \quad \forall m = 1, \ldots, M \]

\[ p_n \in \{1, 0\} \quad \forall n = 1, \ldots, N \]

\[ q_m \in \{1, 0\} \quad \forall m = 1, \ldots, M, \]

where \( \tilde{x}_n = \sum_{k=1}^{K} x_{kn} \forall n = 1, \ldots, N \), \( \tilde{y}_m = \sum_{k=1}^{K} y_{km} \forall m = 1, \ldots, M \), \( L \) is a large positive number and \( \epsilon \) is a small positive fraction number. Binary variables \( p_n \) and \( q_m \) indicate whether input variable \( n \) or output variable \( m \) is selected or not, respectively. The third constraint ensures that the optimal selection of input and output variables does not exceed \([K/3]\), which leads to satisfaction of formulation (3.20). With the average aggregated efficiency score being 0.71, N1 and N4 are determined as selective input and output variables, as those variables are most frequently selected.

<table>
<thead>
<tr>
<th>FY</th>
<th>No. Active Sites</th>
<th>Before applying selective model</th>
<th>After selecting variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>2010</td>
<td>12</td>
<td>8</td>
<td>0.9565</td>
</tr>
<tr>
<td>2011</td>
<td>12</td>
<td>6</td>
<td>0.9251</td>
</tr>
<tr>
<td>2012</td>
<td>10</td>
<td>5</td>
<td>0.8963</td>
</tr>
<tr>
<td>2013</td>
<td>11</td>
<td>5</td>
<td>0.9424</td>
</tr>
<tr>
<td>2014</td>
<td>11</td>
<td>6</td>
<td>0.9166</td>
</tr>
<tr>
<td>2015</td>
<td>9</td>
<td>9</td>
<td>1.0000</td>
</tr>
<tr>
<td>Total</td>
<td>65</td>
<td>39</td>
<td>0.9395</td>
</tr>
</tbody>
</table>
Table 3.3 presents the number of efficient sites and year-by-year efficiency obtained for each of the six fiscal years. Given the selection of input and output variables, the efficiency score seems more reasonable, with a six year average of 0.82 compared to the average 0.94 without selecting variables. Overall, approximately 12.3% of the sites (8 of the total 65 active sites) are evaluated as efficient, while 60% (39 out of 65 active sites) were recognized as efficient before applying the selective model. The overall average efficiency score of all 65 sites for the period is 0.82, which implies that given the level of inputs, sites should have increased their output by 18% proportionally in order to be efficient. In this specific case, since the number of clinical discharges is the only selected output, sites should have achieved more clinical discharges for a given number of enrollments and number served. Over the six-year time span, the aggregated efficiency score of the program presents a sequence with a weak increasing trend. As shown in the dashed line in Figure 3.6, the change in efficiency is slightly positive over the period, with the largest efficiency in FY 2015. After dropping significantly in the second fiscal year, the score displays improvement, except for relatively poor performance in FY 2014.
To assess productivity growth over time, we need to investigate quarter-by-quarter evaluations. We calculated the Malmquist productivity indices including the efficiency change and technical change components by quarter. Instead of using yearly aggregated data, we used quarterly data on the prevention program over time. For each consecutive two-quarter period, four linear programming (LP) models were solved to obtain distance functions as described in the previous section. (In other words, we calculate 1,120 LP models in total using CPLEX v.12.2.) We then take a geometric mean for each site index to evaluate cumulated productivity, cumulated efficiency change, and cumulated technical change.

![Figure 3.7 Cumulated results of Malmquist productivity index](image)

Table 3.4 Average annual productivity growth of the youth prevention program over FY 2010-FY 2015

<table>
<thead>
<tr>
<th>Year</th>
<th>MPI</th>
<th>EC</th>
<th>TC</th>
</tr>
</thead>
<tbody>
<tr>
<td>FY10Q1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY10Q2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY10Q3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY10Q4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY10Q1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY10Q2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY10Q3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY10Q4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY11Q1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY11Q2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY11Q3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY11Q4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY11Q1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY11Q2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY11Q3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY11Q4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY12Q1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY12Q2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY12Q3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY12Q4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY12Q1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY12Q2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY12Q3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY12Q4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY13Q1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY13Q2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY13Q3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY13Q4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY13Q1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY13Q2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY13Q3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY13Q4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY14Q1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY14Q2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY14Q3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY14Q4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY14Q1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY14Q2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY14Q3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY14Q4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY15Q1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY15Q2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY15Q3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FY15Q4</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

65
Figure 3.7 illustrates cumulated changes for MPI, EC, and TC for entire program. Average annual changes of those components are summarized in Table 3.4. Overall, MPI increased by 120.62% over a 6-year period, with geometric mean of 3.27% a year. EC increased 140.58% over the 6-year period, with geometric mean of 3.46% a year, but changes in TC were small. In fact, TC decreased over the 6-year period by 8.30%. These results show that a significant component of the growth in productivity was due to improvement in efficiency level (EC), rather than innovation (TC).

Table 3.5 Average quarterly changes by site

<table>
<thead>
<tr>
<th>Site</th>
<th>MPI</th>
<th>EC</th>
<th>TC</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Site-01</td>
<td>0.9850</td>
<td>0.9938</td>
<td>0.9911</td>
<td>13</td>
</tr>
<tr>
<td>Site-02</td>
<td>1.0533</td>
<td>1.0000</td>
<td>1.0533</td>
<td>8</td>
</tr>
<tr>
<td>Site-03</td>
<td>1.0093</td>
<td>1.0072</td>
<td>1.0021</td>
<td>9</td>
</tr>
<tr>
<td>Site-04</td>
<td>1.1094</td>
<td>1.0228</td>
<td>1.0847</td>
<td>4</td>
</tr>
<tr>
<td>Site-05</td>
<td>0.9766</td>
<td>1.0297</td>
<td>0.9484</td>
<td>15</td>
</tr>
<tr>
<td>Site-06</td>
<td>0.9757</td>
<td>0.9821</td>
<td>0.9935</td>
<td>16</td>
</tr>
<tr>
<td>Site-07</td>
<td>0.9929</td>
<td>0.9983</td>
<td>0.9946</td>
<td>12</td>
</tr>
<tr>
<td>Site-08</td>
<td>1.1010</td>
<td>1.1226</td>
<td>0.9808</td>
<td>5</td>
</tr>
<tr>
<td>Site-09</td>
<td>0.9952</td>
<td>0.9949</td>
<td>1.0004</td>
<td>11</td>
</tr>
<tr>
<td>Site-10</td>
<td>0.9986</td>
<td>1.0204</td>
<td>0.9787</td>
<td>10</td>
</tr>
<tr>
<td>Site-11</td>
<td>0.9809</td>
<td>0.9966</td>
<td>0.9842</td>
<td>14</td>
</tr>
</tbody>
</table>
To give some idea of the productivity of an individual program provider, we turn to average quarterly results for each site. Table 3.5 exhibits the average productivity, average efficiency change and average technical change components for each site over the entire FY 2010 – FY 2015 time period. The results tell another story about productivity growth. At some sites, changes in productivity were due to efficiency change (EC) rather than technology shift (TC), but the pattern was different at other sites. For instance, the productivity changes of sites such as Site-02, Site-12 and Site-15 were largely due to the innovation (TC); at other sites improvements in both efficiency and technology shift made an impact on the productivity. Site-12, Site-14, and Site-15 were the three top performers in terms of productivity, but not all of them were good at shifting the frontier. In fact, the rate of technology change of Site-14 was second highest, but technology change regressed (TC < 1) at the other two sites.

Overall, the technical change (TC) component given in Table 3.5 signifies a shift in frontier technology for each country, but the average results do not fully identify which site actually causes the technical frontier to shift between period t and t + 1. Färe et al. (1994) suggested a set of criteria to identify the “innovators” that contribute to the frontier shift over the pair of periods. In addition to TC greater than 1, the criteria include two distance functions for period t + 1. The innovators classified by these conditions are shown in the second column of Table 3.6.

To give some idea of the productivity of an individual program provider, we turn to average quarterly results for each site. Table 3.5 exhibits the average productivity, average efficiency change and average technical change components for each site over the entire FY 2010 – FY 2015 time period. The results tell another story about productivity growth. At some sites, changes in productivity were due to efficiency change (EC) rather than technology shift (TC), but the pattern was different at other sites. For instance, the productivity changes of sites such as Site-02, Site-12 and Site-15 were largely due to the innovation (TC); at other sites improvements in both efficiency and technology shift made an impact on the productivity. Site-12, Site-14, and Site-15 were the three top performers in terms of productivity, but not all of them were good at shifting the frontier. In fact, the rate of technology change of Site-14 was second highest, but technology change regressed (TC < 1) at the other two sites.

Overall, the technical change (TC) component given in Table 3.5 signifies a shift in frontier technology for each country, but the average results do not fully identify which site actually causes the technical frontier to shift between period t and t + 1. Färe et al. (1994) suggested a set of criteria to identify the “innovators” that contribute to the frontier shift over the pair of periods. In addition to TC greater than 1, the criteria include two distance functions for period t + 1. The innovators classified by these conditions are shown in the second column of Table 3.6.

Table 3.6 Sites shifting the technology frontier

<table>
<thead>
<tr>
<th>Period</th>
<th>Innovators</th>
<th>Disruptive innovators</th>
<th>Sustaining innovators</th>
</tr>
</thead>
<tbody>
<tr>
<td>Site-12</td>
<td>1.3664</td>
<td>1.3681</td>
<td>0.9987</td>
</tr>
<tr>
<td>Site-13</td>
<td>1.0975</td>
<td>1.1213</td>
<td>0.9727</td>
</tr>
<tr>
<td>Site-14</td>
<td>1.2913</td>
<td>1.1937</td>
<td>1.0730</td>
</tr>
<tr>
<td>Site-15</td>
<td>1.6355</td>
<td>1.6407</td>
<td>0.9969</td>
</tr>
<tr>
<td>Site-16</td>
<td>1.0951</td>
<td>1.1500</td>
<td>0.9822</td>
</tr>
</tbody>
</table>

To give some idea of the productivity of an individual program provider, we turn to average quarterly results for each site. Table 3.5 exhibits the average productivity, average efficiency change and average technical change components for each site over the entire FY 2010 – FY 2015 time period. The results tell another story about productivity growth. At some sites, changes in productivity were due to efficiency change (EC) rather than technology shift (TC), but the pattern was different at other sites. For instance, the productivity changes of sites such as Site-02, Site-12 and Site-15 were largely due to the innovation (TC); at other sites improvements in both efficiency and technology shift made an impact on the productivity. Site-12, Site-14, and Site-15 were the three top performers in terms of productivity, but not all of them were good at shifting the frontier. In fact, the rate of technology change of Site-14 was second highest, but technology change regressed (TC < 1) at the other two sites.

Overall, the technical change (TC) component given in Table 3.5 signifies a shift in frontier technology for each country, but the average results do not fully identify which site actually causes the technical frontier to shift between period t and t + 1. Färe et al. (1994) suggested a set of criteria to identify the “innovators” that contribute to the frontier shift over the pair of periods. In addition to TC greater than 1, the criteria include two distance functions for period t + 1. The innovators classified by these conditions are shown in the second column of Table 3.6.
<table>
<thead>
<tr>
<th>Fiscal Year</th>
<th>Sites</th>
<th>Site 03</th>
<th>Site 05</th>
</tr>
</thead>
<tbody>
<tr>
<td>FY10 Q1-FY10 Q2</td>
<td>Site-03, Site-05</td>
<td>Site-03</td>
<td>Site-05</td>
</tr>
<tr>
<td>FY10 Q2-FY10 Q3</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>FY10 Q3-FY10 Q4</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>FY10 Q4-FY11 Q1</td>
<td>Site-07, Site-04</td>
<td>Site-07, Site-04</td>
<td>-</td>
</tr>
<tr>
<td>FY11 Q1-FY11 Q2</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>FY11 Q2-FY11 Q3</td>
<td>Site-10</td>
<td>Site-10</td>
<td>-</td>
</tr>
<tr>
<td>FY11 Q3-FY11 Q4</td>
<td>Site-10</td>
<td>Site-10</td>
<td>-</td>
</tr>
<tr>
<td>FY11 Q4-FY12 Q1</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>FY12 Q1-FY12 Q2</td>
<td>Site-14</td>
<td>Site-14</td>
<td>-</td>
</tr>
<tr>
<td>FY12 Q2-FY12 Q3</td>
<td>Site-09, Site-10</td>
<td>Site-09, Site-10</td>
<td>-</td>
</tr>
<tr>
<td>FY12 Q3-FY12 Q4</td>
<td>Site-06, Site-14</td>
<td>Site-06, Site-14</td>
<td>-</td>
</tr>
<tr>
<td>FY12 Q4-FY13 Q1</td>
<td>Site-02</td>
<td>Site-02</td>
<td>-</td>
</tr>
<tr>
<td>FY13 Q1-FY13 Q2</td>
<td>Site-11</td>
<td>-</td>
<td>Site-11</td>
</tr>
<tr>
<td>FY13 Q2-FY13 Q3</td>
<td>Site-14</td>
<td>Site-14</td>
<td>-</td>
</tr>
<tr>
<td>FY13 Q3-FY13 Q4</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>FY13 Q4-FY15 Q1</td>
<td>Site-11, Site-16</td>
<td>Site-16</td>
<td>Site-11</td>
</tr>
<tr>
<td>FY14 Q1-FY14 Q2</td>
<td>Site-14</td>
<td>Site-14</td>
<td>-</td>
</tr>
<tr>
<td>FY14 Q2-FY14 Q3</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>FY14 Q3-FY14 Q4</td>
<td>Site-16</td>
<td>Site-16</td>
<td>-</td>
</tr>
<tr>
<td>FY14 Q4-FY15 Q1</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>FY15 Q1-FY15 Q2</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>FY15 Q2-FY15 Q3</td>
<td>Site-01, Site-16</td>
<td>Site-01</td>
<td>Site-16</td>
</tr>
<tr>
<td>FY15 Q3-FY15 Q4</td>
<td>Site-03</td>
<td>Site-03</td>
<td>-</td>
</tr>
</tbody>
</table>

To delve deeper into the shift in the technology frontier over time, we further look at the robust productivity bound. For each site, we construct upper and lower limits of the robust productivity bound based on the most and least favorable estimates of period $t + 1$, respectively. Using the chart, we can monitor the quarterly trend of average performance over time and control the quality of prevention service.
delivery. Assuming proportional changes, we estimate the upper limit for period $t+1$ where the input decreases by $\alpha$ and output increases by $\alpha$ from period $t$. Namely,

\[
\begin{align*}
  x_{k'n}^{t+1} &= (1 - \alpha)x_{k'n}^t \forall n = 1, ..., N, \\
  y_{k'm}^{t+1} &= (1 + \alpha)y_{k'm}^t \forall m = 1, ..., M,
\end{align*}
\]  

(3.22)

(3.23)

and

\[
\begin{align*}
  x_{kn}^{t+1} &\in [(1 - \alpha)x_{kn}^t, (1 + \alpha)x_{kn}^t] \forall k \in \{1, ..., K\} \setminus \{k'\}, \forall n = 1, ..., N, \\
  y_{km}^{t+1} &\in [(1 - \alpha)y_{km}^t, (1 + \alpha)y_{km}^t] \forall k \in \{1, ..., K\} \setminus \{k'\}, \forall m = 1, ..., M.
\end{align*}
\]  

(3.24)

(3.25)

The lower limit is estimated using a similar (opposite) method. As explained in the previous section, the lower productivity bound can be achieved when a site has fewer discharge cases with more active participants and new enrollments, and the upper bound can be achieved in the opposite case.
Figure 3.8 Robust productivity bound and productivity index

Figure 3.8 depicts the average productivity index and its bounds with $\alpha = 0.1$. While most of the productivity index points fall within the bounded interval, there are some data points that are “out-of-control.” To interpret such nonrandom patterns, we identify “disruptive innovators” based on the component distance function and technology change index exceeding their upper bound. Specifically, if

$$\text{TC} > \max\{\overline{\text{TC}}, 1\}, \quad (3.26)$$

$$D^t_0(x_{k'}^{t+1}, y_{k'}^{t+1}) > \max\{D^t_0(x_{k'}^{t+1}, y_{k'}^{t+1}), 1\} \quad (3.27)$$

and

$$D^{t+1}_0(x_{k'}^{t+1}, y_{k'}^{t+1}) = 1, \quad (3.28)$$

then site $k'$ makes a disruptive shift in the technology frontier between periods $t$ and $t + 1$. As innovations from such sites are far larger than those of other sites in the program, they may cause disruption for the rest of the sites. This tends to happen when a new site enters the system and preserves its advantage in the early stage. This is because an established site can serve more participants and possibly release more, but the
The number of discharges tends to be diminished as program implementation becomes mature and finally winds up due to contract termination. We distinguish innovators that are not disruptive and make a frontier shift by sustaining productivity. The third and fourth columns in Table 3.6 represent a list of sites that make disruptive and sustaining innovations, respectively.
<table>
<thead>
<tr>
<th>Site</th>
<th>Index</th>
<th>FY10 Q1</th>
<th>FY10 Q2</th>
<th>FY10 Q3</th>
<th>FY10 Q4</th>
<th>FY11 Q1</th>
<th>FY11 Q2</th>
<th>FY11 Q3</th>
<th>FY11 Q4</th>
<th>FY12 Q1</th>
<th>FY12 Q2</th>
<th>FY12 Q3</th>
<th>FY12 Q4</th>
<th>FY13 Q1</th>
<th>FY13 Q2</th>
<th>FY13 Q3</th>
<th>FY13 Q4</th>
<th>FY14 Q1</th>
<th>FY14 Q2</th>
<th>FY14 Q3</th>
<th>FY14 Q4</th>
<th>FY15 Q1</th>
<th>FY15 Q2</th>
<th>FY15 Q3</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>MPI</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
</tr>
<tr>
<td></td>
<td>EC</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>-</td>
<td>-</td>
<td>&lt;</td>
<td>-</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>-</td>
<td>-</td>
<td>&lt;</td>
<td>&lt;</td>
<td>-</td>
<td>&lt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TC</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
</tr>
<tr>
<td>02</td>
<td>MPI</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
</tr>
<tr>
<td></td>
<td>EC</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TC</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
</tr>
<tr>
<td>03</td>
<td>MPI</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&gt;</td>
</tr>
<tr>
<td></td>
<td>EC</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&gt;</td>
</tr>
<tr>
<td></td>
<td>TC</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
</tr>
<tr>
<td>04</td>
<td>MPI</td>
<td>-</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td>EC</td>
<td>&lt;</td>
<td>-</td>
<td>&lt;</td>
<td>&lt;</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td>TC</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>05</td>
<td>MPI</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td>EC</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td>TC</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>06</td>
<td>MPI</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>&gt;</td>
<td>&lt;</td>
<td>-</td>
<td>-</td>
<td>&gt;</td>
<td>&lt;</td>
<td>-</td>
<td>-</td>
<td>&gt;</td>
<td>&lt;</td>
<td>-</td>
<td>-</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>-</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&lt;</td>
</tr>
<tr>
<td></td>
<td>EC</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>-</td>
<td>-</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TC</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&lt;</td>
</tr>
<tr>
<td>07</td>
<td>MPI</td>
<td>-</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
</tr>
<tr>
<td></td>
<td>EC</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
</tr>
<tr>
<td></td>
<td>TC</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&lt;</td>
</tr>
<tr>
<td>08</td>
<td>MPI</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td>EC</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td>TC</td>
<td>&gt;</td>
<td>&lt;</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>09</td>
<td>MPI</td>
<td>-</td>
<td>-</td>
<td>&lt;</td>
<td>&lt;</td>
<td>-</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>-</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>-</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
</tr>
<tr>
<td></td>
<td>EC</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
</tr>
<tr>
<td></td>
<td>TC</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
</tr>
<tr>
<td>10</td>
<td>MPI</td>
<td>-</td>
<td>-</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td></td>
<td>EC</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>TC</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>MPI</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td>EC</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TC</td>
<td>-</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
</tr>
<tr>
<td></td>
<td>MPI</td>
<td>EC</td>
<td>TC</td>
<td></td>
<td>MPI</td>
<td>EC</td>
<td>TC</td>
<td></td>
<td>MPI</td>
<td>EC</td>
<td>TC</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>-----</td>
<td>----</td>
<td>----</td>
<td>---</td>
<td>-----</td>
<td>----</td>
<td>----</td>
<td>---</td>
<td>-----</td>
<td>----</td>
<td>----</td>
<td>---</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>&gt;</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>&gt;</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&gt;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>M/A</td>
<td>M/A</td>
<td>M/A</td>
<td>M/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>&gt;</td>
<td>&lt;</td>
<td>&gt;</td>
<td>&gt;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>M/A</td>
<td>M/A</td>
<td>M/A</td>
<td>M/A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>M/A</td>
<td>M/A</td>
<td>M/A</td>
<td>M/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>M/A</td>
<td>M/A</td>
<td>M/A</td>
<td>M/A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>M/A</td>
<td>M/A</td>
<td>M/A</td>
<td>M/A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>M/A</td>
<td>M/A</td>
<td>M/A</td>
<td>M/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>M/A</td>
<td>M/A</td>
<td>M/A</td>
<td>M/A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>M/A</td>
<td>M/A</td>
<td>M/A</td>
<td>M/A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>M/A</td>
<td>M/A</td>
<td>M/A</td>
<td>M/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>M/A</td>
<td>M/A</td>
<td>M/A</td>
<td>M/A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

73
Table 3.7 reports the detailed component information associated with its bounds. The greater than sign (>) shows where the actual index is greater than both 1 and the estimated upper limit, while the less than sign (<) implies that the actual index is less than the estimated lower limit. This approach helps to keep the system within an acceptable level of productivity by identifying repeated patterns. If a sequence of data points falls below the lower limit (or above the upper limit), we can detect a significant deterioration (or suspicious false-positivity) in productivity. For instance, component TC of Site-02 exceeds its upper bound in three consecutive quarters during FY12 Q2 – FY12 Q4, and Site-16 does the same from FY13 Q3 to FY14 Q1. Both exceptions show disruptive innovations in the following periods, as shown in Table 3.6. Ideally, we can avoid harmful situations by early detection and notification of an exception.

3.5 Implementation and Integration of Monitoring Engine

The proposed approach can be incorporated as a key element of an enhanced data-driven system. With the current INSPIRE database we described earlier as a foundation, the approach we proposed earlier can be integrated into the new data-driven system. First, it will provide an automatic end-of-period outcome analysis and performance evaluation based on it. The monitoring engine will detect deviations based on the productivity index and generate alerts for appropriate prevention personnel accordingly. The event will be driven by the significance of deviation from the estimation of the implementation plan. For instance, when the negative productivity patterns are repeated in three consecutive periods, we can detect a significant deterioration in productivity. The system issues an alert message to the prevention service providers, such as practitioners or local site administrators, so that they can recognize explicitly where the exception occurs and which outcome they need to promote. This will remove an ad hoc nature in prevention programs that are being executed.

Furthermore, the monitoring engine can be aligned with the early waning functionality. Based on the system warning for the possible abnormality of the productivity in the next period, program providers can accommodate its resources in advance. They can concentrate their efforts on enhancing specific
outcome measure to avoid such situations. The productivity bound will be presented as a target value for the goal they need to achieve in the following period.

Overall, this new environment can fundamentally transform the implementation of prevention programs and eventually enhance the efficiency of prevention service delivery. Our approach can further motivate an adoption of engineering tools and techniques in the prevention research, practice, and policy.

3.6 Conclusions

Stakeholders from community youth prevention programs call for appropriate measures of statewide growth productivity of the programs as well as the efficiency of individual sites. There is a variety of outcome variables for youth at discharge, but no common consent on measures of the efficiency of program providers. In the present study, we introduce a robust productivity index approach that enables periodic monitoring and early detection of exception. The upper and lower bounds can be interpreted as the maximum and minimum achievable productivity of estimated performance for the following period. The derivation of both limits is done by solving the robust version of the Malmquist productivity index.

We apply the proposed approach to performance data for a community youth program from FY 2010 to FY 2015, including data from both a national database and local practitioners. Analysis of MPI, EC and TC shows that the total factor productivity of all sites increased by 120.62% from FY 2010 to FY 2015, and mainly driven by efficiency improvement rather than technological progress. Innovators that caused the frontier shift are identified, and differentiated into two separate classes by investigating the robust bound. The results from the robust productivity analysis also show that random occurrences of an exception and significant patterns of productivity change can be detected and monitored. Ultimately, the approach has the potential to become a decision support tool for prevention service providers, and permits early identification (and thus elimination) of sources of exception.

It should be noted that the results of this empirical study are dependent upon the limited number of input and output variables available, given the small number of sites running the program as compared to
the set of data available in the system. This study could be extended by covering a prevention program with
a larger number of sites, and including more relevant variables for better solution quality.
4 REAL-TIME SUPPLY CHAIN CONTROL USING ADAPTIVE WIP ALGORITHM

4.1 Introduction

In determining inventory levels of multistage supply chain and production system problems, it is hard or even not feasible to obtain closed-form, analytic solutions, and hence computationally demanding heuristic algorithms are often used to find the best inventory level. Moreover, less is known about the exact effect of the inventory level of a single stage or subsystem, to the overall system performance and service level of the supply chain. In this chapter, we provide a feedback algorithm that efficiently finds the best inventory solution of the system, considering throughput, individual inventory level, and blocking parts. The algorithm dynamically controls the three performance measures by adjusting the maximum level of inventory in the subsystem.

The focus of this chapter is to show the representation of the adaptive feedback algorithm and study the convergence and robustness of the algorithm. Feedback control is an efficient methodology to manage dynamics of complex systems and has been applied to various production systems and inventory management problems (Ortega and Lin 2004, Sarimveis et al., 2008). Target performance measure, which is called a ‘command’, indicates a desired level and its value is compared to the one from simulation runs. Based on the difference between the command and feedback value, a ‘controller’ makes corrective actions to the system and manipulation is adjusted through an ‘actuator’. Given the target level of throughput of the system, we choose two additional commands, Work-In-Process (WIP) levels and blocking time. As the iteration runs, the algorithm dynamically changes the inventory levels and finds the inventory level in each of the subsystem.

In pull manufacturing systems, such as Kanban and CONWIP, production timing and release of raw materials are synchronized and controlled by the ‘cards’. Upon the departure of a job at the end of the

---

2 The AWIP algorithm is originally suggested by Masin and Prabhu (2005)
subsystem, the card is released and sent back to the initial point, so that it indicates there is available space for the new arrival of the next job. Therefore, the number of cards in the subsystem governs production activities and finding the optimal number of cards makes a significant impact on performances of the overall system. If the number of cards is too low, then not enough parts can be processed in the subsystem and the throughput level may not be desirable. On the other hand, if too many parts are allowed in the card loop, then one may suffer from large inventory costs. Therefore, we need to find the best number of cards that tradeoff between WIP level and throughput.

The rest of this chapter is organized as follows. In Section 4.2, we provide a review of the literature of relating to dynamic card controlling scheme of pull systems, and basic concepts of self-regulated production systems. The Adaptive WIP (AWIP) algorithm design is illustrated in Section 4.3. In Section 4.4, experimental results are presented to show the dynamics of the algorithm, and compared to the results of the mathematical problem. Finally, we discuss the conclusions and future works in Section 4.5.

4.2 Literature Review of Card Controlling System

4.2.1 Kanban and CONWIP system

With the adoption of Just-In-Time (JIT) principles, the Kanban system is one of the most well-known pull manufacturing systems. As a medium of production information, Kanban cards authorize both order quantity and production timing. Operation starts processing when both raw materials and Kanban cards become available as the cards are released upon the departure of parts at the end of the production stage. Therefore, the number of Kanban cards in each stage corresponds to the inventory level of the stage because it dominates the work-in-process (WIP) level by capping the number of parts in that stage. One can find the optimal number of cards in Kanban system, so as to minimize total costs including inventory holding cost and backorders.

The Kanban system has its advantages because it is easy to optimize and runs effectively when demand is stable. However, when a company experiences variabilities in demand or processing times, the system needs to be redesigned by readjusting the parameter of the system. Rees et al. (1987) consider the
dynamic card control methods, where the number of cards matches current environmental conditions. Their approaches are based on the demand forecast and statistical estimation of the lead time, but once the Kanban level is readjusted, the system runs during the planning period. Gupta and Al-Turki (1997) suggest a flexible Kanban system, a simple procedure to decide the number of additional cards needed. However, their algorithm also requires the known demand and system capacity for the planning period. Therefore, both methods are more of medium run planning, lacking the real-time reactive adjustment.

Takahashi and Nakamura (1996) and Takahashi and Nakamura (1999) explore the reactive JIT control mechanism of the size of buffer allocation. The unstable changes in demand and production time is absorbed in the buffers allocated to each inventory point. By monitoring time series data, the performances of systems are analyzed based on the simulation results. The number of Kanban cards is intelligently reallocated according to the previous simulation run, revealing the trade-off relationship of two performance measures, the average waiting time of production demand and the total average WIP level. Tardif and Maaseidvaag (2001) consider an adaptive method for a single stage Kanban system. Their algorithm finds a set of optimal parameters for the Kanban cards and release and capture thresholds of inventory level, and determines release and reorder points based on the current demand. It should be noted that the single stage, one card Kanban system is equivalent to the CONWIP production system, stated below.

CONWIP production system is another pull manufacturing system that limits the total number of jobs in the production line. The number of jobs in the system is usually governed by cards attached to the jobs. When a job is processed in the last station, the card is detached and sent back to the beginning of the system to be attached to the new job queueing at the first station. Therefore, the overall number of WIP in the production line remains at constant (Spearman et al. 1990). In that sense, the card controlling mechanism of the CONWIP system can be viewed as that of a single-stage Kanban system. The noticeable contribution about the card controlling in CONWIP system can be seen in Hopp and Roof (1998) and Framinan et al. (2006).
Hopp and Roof (1998) present a statistical control chart-based mechanism, where the number of cards in the CONWIP line is dynamically adjusted based on the current system throughput. Intermediate outputs of the current production line are evaluated and compared to the target throughput level or the critical WIP level. When it falls below (or increase) more than three times of the standard deviation, just like the case of statistical process control (SPC), the card count is increased (or decreased) by one in each step. The simulation results show the dynamic step size method expedites a fast convergence of the algorithm. Framinan et al. (2006) suggest a card controlling mechanism of the CONWIP system based on the constant monitoring of the throughput rate. They take into account the number of extra cards as in Tardif and Maaseidvaag (2001), but it is bounded by upper and lower limits to avoid uncontrollable behaviors of the system. The numerical experiments demonstrate that their method reaches the target throughput level faster and closer than the statistical data-based algorithm by Hopp and Roof (1998) under various scenarios.

4.2.2 Self-regulated WIP (SWIP)

Masin et al. (1999) propose the self-regulating WIP (SWIP) approach, which unifies multiple production control systems, such as Kanban, CONWIP, Drum-Buffer-Rope (DBR), buffered line, base-stock control, etc. As shown in Figure 4.1, the key concept of SWIP is to limit the number of parts in a contiguous portion of the production system that represents the above-mentioned production control systems. A group of workstations are bound into a single subsystem that shares the same number of container (cards). For instance, the Kanban system can be viewed as a system with one or two containers to regulate the transfer of parts between two adjacent workstations. In the CONWIP system, however, the total number of parts in the entire system remains the same. This can be translated into the self-regulating production control system, where a fixed number of containers traverses the entire production line. When a part reaches at the end of production line and leaves the system, the ‘virtual’ container is being emptied and sent back to the beginning of the line so that the system can accept the new arriving part into the production line.
It should be noted that SWIP incorporates a very unique type of production control system, called the inverse base stock (IBS) (Masin 1999). One of the key concepts of the production control system is to delay the release of the raw material. Since flow time is defined as the difference between departure time and arrival or release time of the raw material into the system, IBS can decrease the flow time by releasing the raw material as late as possible in the beginning of the first workstation. Once the job is being released, however, it is allowed to be processed until the end point as quickly as possible. Therefore, as shown in Figure 4.2, this approach is implemented by having a card loop with the starting workstation to each workstation in the downstream, and the card loops of the production system are completely opposite to the base stock model.
In the SWIP system, admission of a part in the specific ‘card loop’ is determined by the number of cards assigned to the loop. The total number of assigned cards works as a blocking mechanism of the system: parts can either be transferred into the loop if there exist available spaces in the number of cards. Otherwise, they remain blocked and wait for the release of cards at the end of the card loop. Masin et al. (2005) use a simulation-based heuristic technique, Tradeoffs Programming (TOP), in designing SWIP systems. Given the overall system performance measure, they decompose the system into separable subsystems, where the performance measures can be correctly defined. One of the benefits of the model is it provides a flexible framework that can encompass almost all well-known production control systems. Also, the system can be self-regulated, being run by themselves without human interventions.

A scalable design of the unified production control model is further considered by Masin and Prabhu (2009). The number of containers in every contiguous portion of the system can be controlled adaptively, using a simulation-based feedback control. The algorithm efficiently finds the best feasible solution to minimize the average WIP of the system with a given target throughput level. It estimates three performance measures, average WIP, average throughput, and relative blocking time in the loop, for both subsystems and adjusts the number of cards in the loop based on the feedback obtained from every simulation run. The computational time complexity of the algorithm is polynomial in the size of the production system. Depending on various parameter setting, they extensively investigate the results of
computational experiments of the algorithm and find out the solution is near optimal compared to other schemes commonly used in supply chain. In the following sections, we will further review the key aspects of the AWIP algorithm and illustrate robustness and dynamics of the algorithm.

4.3 Adaptive WIP (AWIP) Design

4.3.1 AWIP integral controller

We first consider the definition of a card loop and performance measures associated with it. Figure 4.3 illustrates the card loop \((i, j)\) in the production line with total \(M\) number of workstations. In designing the AWIP production system, the following performance measures are estimated (Masin and Prabhu 2009):

- Inventory in card loop \((0, i)\);
- Inventory in card loop \((i, j)\);
- Blocking at workstation \(i\);
- Blocking at the beginning, workstation \(0\);
- Variation of flow time of card loop \((0, j)\);
- Variation of time between departures at workstation \(j\) weighted by the maximum inventory in card loop \((0, j)\);

![Figure 4.3 Release point 0, and inventory loop \((i, j)\) in \(M\) stage production system.](image)

Notations of measures used in the AWIP algorithm are summarized in Table 4.1.
Table 4.1 Notations

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(m)</td>
<td>station, (\forall m = 1, \ldots, M)</td>
</tr>
<tr>
<td>(n)</td>
<td>part, (\forall n = 1, \ldots, N)</td>
</tr>
<tr>
<td>((i, j))</td>
<td>pair of card loops from station (i) to (j), (\forall i = 0, \ldots, M - 1, \forall j = 1, \ldots, M)</td>
</tr>
<tr>
<td>(u_{ij})</td>
<td>maximum number of parts in card loop ((i, j))</td>
</tr>
<tr>
<td>(\mu)</td>
<td>average TBD from the production system ((\mu = T_M))</td>
</tr>
<tr>
<td>(\mu^*)</td>
<td>required TBD from the production system</td>
</tr>
<tr>
<td>(F_{ij})</td>
<td>average flowtime in card loop ((i, j))</td>
</tr>
<tr>
<td>(T_i)</td>
<td>average TBD from point (i)</td>
</tr>
<tr>
<td>(W_{ij})</td>
<td>average WIP in card loop ((i, j))</td>
</tr>
<tr>
<td>(W)</td>
<td>average WIP in production system</td>
</tr>
<tr>
<td>(w_{ij})</td>
<td>a measure of WIP level in card loop ((i, j))</td>
</tr>
<tr>
<td>(B_{ij})</td>
<td>proportion of time when a part cannot pass point (i) because card loop ((i, j)) is blocked</td>
</tr>
<tr>
<td>(\overline{B}<em>{0i} = \sum</em>{k=0}^{i-1} B_{0k} B_{ki})</td>
<td>a measure of total blocking of point 0 from point (i), ((\overline{B}_{00} \equiv 1))</td>
</tr>
<tr>
<td>(B_{0ij} = \overline{B}<em>{0i} B</em>{ij})</td>
<td>a measure of blocking of point 0 by card loop ((i, j))</td>
</tr>
</tbody>
</table>

As mentioned earlier, the objectives of the AWIP model are to defer the release of raw material into the production system as late as possible, to minimize the blocking of parts within the system, and to minimize variances in release procedure of the system. To achieve these objectives at the same time, the
integral controller of the AWIP system has the following three commands: (1) target throughput level, (2) proportional WIP level of the card loop, and (3) no blocking of parts in the card loop. The first command is the target throughput level we desire to achieve. The second command for the inventory level implies the even distribution of the inventory, where the WIP level is well-allocated throughout the entire system. This can be accomplished by comparing flat allocation of average WIP before the loop (uniformly allocated WIP in loop \((0, i)\) of the production line) to the actual WIP. Lastly, the algorithm targets no blocking of parts in the card loop.

The last two measures are closely related in that the card loop \((i, j)\) is likely to block parts in loop \((0, i)\) when the actual WIP is greater than the evenly-distributed level. The algorithm tries to increase overall flow time, rather than reducing the throughput. In this case, the number of cards in the loop should be increased in order to reduce blocking. On the other hand, if the actual WIP is less than the flat allocation, the number of cards in the loop should be decreased. The average WIP of the system can be reduced by decreasing throughput, rather than by increasing the flow time. If the card loop \((i, j)\) does not block parts, we do not have to increase the maximum number of cards in the loop.

Equation (4.1) through (4.5) describe the performance measures used in the AWIP integral controller in Figure 4.4.

\[
\begin{align*}
\dot{u}_{ij}(t) &= \int_0^t k_{ij}^*(\tau) \xi \left( \mu^* - \mu(\tau), \frac{1}{i} - w_0i(\tau) \right) d\tau + u_{ij}(0), \\
\xi \left( \mu^* - \mu(\tau), \frac{1}{i} - w_0i(\tau), 0 - B_{ij} \right) &= \begin{cases} 
0, & \text{if } \mu^* < \mu(t) \text{ and } B_{ij}(t) = 0, \\
1, & \text{if } \left( \mu^* < \mu(t) \text{ or } \frac{1}{i} < w_0i(t) \right) \text{ and } B_{ij} > 0, \\
-1, & \text{otherwise}
\end{cases} \\
k_{ij}(t) &= \begin{cases} 
\left( k(1 - \gamma_{ij}(t)), \right. & \text{if } \xi \left( \mu^* - \mu(\tau), \frac{1}{i} - w_0i(\tau), 0 - B_{ij} \right) = -1, \\
k \gamma_{ij}(t), & \text{otherwise}
\end{cases}
\end{align*}
\]

where
\[ y_{ij}(t) = \frac{B_{0ij}(t) + v_{ij}^f(t) + v_{ij}(t) + w_{oi}(t) + (1 - w_{ij}(t))}{3 + \delta(B_{ij}(t)) + \delta(w_{oi}(t))}, \]  
(4.4)

\[ v_{ij}^f(t) = \frac{\Bar{B}_{oi}(t)Var(F_{0j}(t))}{\sum_{k,l \in \text{loop}(k,l)} \Bar{B}_{0k}(t)Var(F_{0j}(t))}, \]  
(4.5)

\[ v_{ij}(t) = \frac{\Bar{B}_{oi}(t)(u_{oj}(t) - 1)Var(T_{j}(t))}{\sum_{k,l \in \text{loop}(k,l)} \Bar{B}_{0k}(t)(u_{ol}(t) - 1)Var(T_{l}(t))}, \]  
(4.6)

and

\[ \delta(X) = \begin{cases} 1, & \text{if } X > 0 \\ 0, & \text{otherwise} \end{cases} \]  
(4.7)

4.3.2 AWIP algorithm

The AWIP algorithm iteratively minimizes the average inventory in the production line and supply chain for the target throughput. In each iteration, performance measures are estimated and compared to the commands to determine the number of cards in every card loop. The target throughput, the initial number of card in each loop, and the maximum number of iterations are selected in the initial phase. As the algorithm runs, it calculates the performance of the production line subject to the given maximum number of cards in each loop. Performance measures include the average flow time, the average time between departures, the average WIP, variations in flow time, variations in time between departures, relative

Figure 4.4 AWIP Integral Controller
blocking time in each loop, etc. The discrete computation of the algorithm is conducted based on the following implementation:

\[ u_{ij}(t) = u_{ij}(t - 1) + k_{ij}(t - 1) \xi \left( \mu^* - \mu(\tau), \frac{1}{\tau} - w_{0i}(\tau), 0 - B_{ij} \right), \]  \hspace{1cm} (4.8)

The integral controller adjusts the maximum number of cards by using aforementioned formulae from (4.1) to (4.7). When the system throughput is below the target throughput, we increase the maximum number of cards so that the production line can take more parts in the system. When the throughput level satisfies the requirement, we need to consider two possibilities. If parts are blocked due to the specific downstream card loop and the proportional WIP of the card loop is greater than the system average, then we again increase the number of card. On the other hand, if parts are blocked, but the proportional WIP of the card loop is less than the system average, then we decrease the card loop. Lastly, when the throughput level is satisfied and there is no blocking, we keep the current number of cards. The logic flow under different conditions is illustrated in

Figure 4.5.

![Flow chart of AWIP algorithm](image)

Figure 4.5 Flow chart of AWIP algorithm
Note that the controller determines the proper level for the number of cards in the loop regardless of interactions with other controllers. In addition to the perturbation in the processing time, impacts from the interactions are reflected and then compensated in the next iteration. The algorithm runs until it reaches the maximum number of iterations, or the objective value converges and no longer improves significantly. The algorithm is summarized in Table 4.2:

Table 4.2 AWIP Algorithm

<table>
<thead>
<tr>
<th>Step 0.</th>
<th>Set initial solutions, target throughput level, controller gain, and maximum number of iterations. $(u_{ij}(0), \mu^<em>, k, I^</em>)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1.</td>
<td>Run AWIP integral controller.</td>
</tr>
<tr>
<td>(a)</td>
<td>Evaluate WIP, blocking, flow time and throughput based on the current solution. $(W_{ij}, w_{ij}, B_{ij}, \bar{B}<em>{ij}, F</em>{ij}, \mu, T_I)$</td>
</tr>
<tr>
<td>(b)</td>
<td>Adjust card loop based on the evaluation. $(u_{ij}(t))$</td>
</tr>
<tr>
<td>Step 2.</td>
<td>Find the best feasible system design that minimizes the average WIP of the system.</td>
</tr>
<tr>
<td>Step 3.</td>
<td>IF stopping conditions are met; Go to step 1.</td>
</tr>
</tbody>
</table>

### 4.4 Performance of AWIP Algorithm

#### 4.4.1 Solution quality

We first test the convergence of the AWIP algorithm using the instance with 6 workstations and 400 parts. The experimental results are compared to the first-come-first-served-based initial solution. The numerical results indicate that the algorithm not only efficiently finds the best solution in less than 50 iterations, but the final solution significantly improves each performance measure of the instance. As shown in Figure 4.6, the system throughput level is improved by 7.57% and mean flow time is decreased by 6.51% compared to the FCFS initial solution, while the average WIP of the system remains the same level, increased only by 0.57%. The algorithm also seeks small variabilities in flow time and the time between
departures. The squared coefficient of variation (SCV) values for both flow time and time between departures are stabilized very quickly. The result illustrates that SCV of flow time drops by 3.62%, while the SCV of time between departures increases by 9.31%. The results are summarized in Table 4.3.
Figure 4.6 Experimental result of medium size supply chain problem

Table 4.3 Performance measures and numerical results

<table>
<thead>
<tr>
<th>Performance measures</th>
<th>Initial solution (FCFS)</th>
<th>Final solution</th>
<th>Increase (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average WIP</td>
<td>205.041</td>
<td>206.215</td>
<td>0.57%</td>
</tr>
<tr>
<td>Average throughput</td>
<td>0.430</td>
<td>0.463</td>
<td>7.57%</td>
</tr>
<tr>
<td>SCV of TBD</td>
<td>0.817</td>
<td>0.893</td>
<td>9.31%</td>
</tr>
<tr>
<td>Average flow time</td>
<td>476.877</td>
<td>445.848</td>
<td>-6.51%</td>
</tr>
<tr>
<td>SCV of flow time</td>
<td>0.316</td>
<td>0.305</td>
<td>-3.62%</td>
</tr>
</tbody>
</table>

As the AWIP algorithm runs, the maximum number of cards in each loop has been adjusted based on the simulation result. Figure 4.7 illustrates how the number of cards changes in each card loop \((i, j)\). The dynamics of the number of cards in subsystems are tested using the same instance with 400 parts and 6 workstations. The horizontal axis of the figure represents the starting machine \((i)\) of the contiguous card loop \((i, j)\), and the vertical one is for the last machine \((j)\) of the loop. The yellow bar in the chart indicates the data point of the card loop whose value is changed from the previous run. In this specific instance, the initial number of cards is set to be the number of machines in the card loop because ideally one job can be processed in a single machine at a time. So, the initial state of the system is represented by the system with
no extra room for inventory in each and every machine in the production line. The maximum number of jobs in card loops of the system tends to increase in order to optimize average WIP, flow time, and throughput. When the algorithm satisfies the stopping conditions, the algorithm is stabilized in iteration 12 where all performance measures are no longer updated any further in the following iterations.
4.4.2 Fidelity of the AWIP algorithm

Due to the limitations of control-theoretic heuristic algorithm, the fidelity of the model has to be assessed by considering factors that affect the performance of the algorithm. Here, we test the fidelity of the AWIP algorithm by comparing simulation results of three different initial solutions. In the first instance, the number of cards is initially set to the minimum level, where a single card is allocated to every single card loop in the system. Conversely, we consider the maximum level of initial solution in which the initial number of cards in the loop is set to be total number of machines in the production line. In this case, the initial solution vector is composed of $n$-tuples of element of $M$. The last instance of the initial solution is medium level, where the number of cards in the subsystem is equal to the number of workstations in the card loop. Since the card loops are inclusive, it may be intuitive to determine the initial solution as the number of workstations contained in the card loop. The initial number of cards in the card loop $(i, j)$ is $(j-i+1)$ in this case. It implies that there exists no extra room for inventory in each machine, and each workstation can hold of only a single part initially.

Figure 4.8 shows the comparison of the simulation results associated with different initial solution. It depicts that regardless of the initial solution the algorithm provides a consistent final solution with different number of iterations depending upon initial solutions. The minimum level of initial solution converges to the final solution much slower than the other two because the simulation results of the
performance measures tend to be far from the final solution. While the other two instance seem to have similar convergence pattern, the maximum level initial solution slightly outperforms in all metrics. This is because the algorithm finds there are less blockings in the maximum level instance, thereby arriving at the final value of the maximum number of cards faster than other instances.
4.4.3 The AWIP algorithm versus the mathematical model

In this section, we compare the solution quality of the AWIP algorithm to the mathematical model. We consider a flow shop type of mathematical problem that minimizes the makespan of the system, assuming that the sequence of jobs is identical across all workstations. It should be noted that the sequencing problem of the static flow shop is a famous combinatorial problem, involving $n!$ different solutions to the problem. As the complexity of finding a permutation of $n$ jobs prohibits its usage, researches have been studied only for the small size problem to find the optimal solution using branch and bound and lower bounding. Instead, sequencing problems in practical application are mostly depend upon heuristic procedures are widely used for practical applications. More comprehensive reviews of approaches for flow shop scheduling problems are shown in Gupta and Stafford (2006) and Ruiz and Maroto (2005).

In our mathematical formulation, the progression of parts to the next station is governed by the target WIP level. The total number of jobs over the set of workstations at a time does not exceed the target WIP level ($W_0$). Assuming the permutation of jobs are kept throughout the entire procedure, we have the following mixed integer programming problem:
\( P_{mn} \)  Processing time of job \( n \) on station \( m \)

\( X_{nk} = \begin{cases} 1, & \text{if job } n \text{ is } k\text{th job in the sequence} \\ 0, & \text{otherwise} \end{cases} \)

\( I_{nk} \)  Idle time on machine \( n \) between \( k\)th job and \((k + 1)\)th job

\( B_{nk} \)  Waiting time of \( k\)th job in between machine \( n \) and \( n + 1 \)

\( W_0 \)  Target WIP level to be maintained

\[
\begin{align*}
\min & \quad \sum_{m=1}^{M-1} \sum_{n=1}^{N} P_{mn}X_{n1} + \sum_{k=1}^{N-1} I_{Mk} \\
\text{s.t. } & \quad \sum_{n=1}^{N} X_{nk} = 1, \forall k = 1, ..., N \\
& \quad \sum_{k=1}^{N} X_{nk} = 1, \forall n = 1, ..., N \\
& \quad \sum_{n=1}^{N} P_{mn}X_{nk} + I_{nk} + B_{n,k+1} = \sum_{n=1}^{N} P_{m+1,n}X_{nk} + I_{m+1,k} + B_{mk}, \\
& \quad \forall m = 1, ..., M - 1, \forall k = 1, ..., N - 1 \\
& \quad \sum_{m=1}^{M-1} \sum_{n=1}^{N} P_{mn}X_{m1} + \sum_{k=1}^{l-1} I_{Mk} + \sum_{k=1}^{l} \sum_{n=1}^{N} P_{mn}X_{nk} \\
& \quad = \sum_{k=1}^{l+W_0} \sum_{n=1}^{N} P_{1n}X_{nk} + \sum_{k=1}^{l+W_0} I_{1k} + \sum_{k=1}^{l+W_0} B_{1k}, \forall l = 2, ..., N - W_0 \\
X_{nk} & \in \{1, 0\}, \forall n = 1, ..., N, \forall k = 1, ..., N
\end{align*}
\]
\[ I_{mk}, B_{mk} \geq 0, \forall m = 1, \ldots, M, \forall k = 1, \ldots, N \]  

(4.15)

Note that minimizing the makespan is equivalent to minimization of the total idle time on the last machine. The objective function (4.9) is further divided into two parts: idle time that the first job arrives the final station and total idle time of the last station. Constraints (4.10) and (4.11) is for the assignment of jobs in the sequence and across all stations, respectively. Constraint (4.12) explains the balance of jobs in kth slot in station m, as described in Figure 4.9. The CONWIP-type constraint (4.13) ensures the constant WIP level in the card loops. In the constraint, the first three terms represent the time spent in the last machine to process the first l jobs, calculated by the summation of (1) time of the first job spends from the first to the second last \((M - 1)\) station, (2) idle time of the first \((l - 1)\) jobs in the final stage, and (3) processing time of the first l jobs in the last station. The value should correspond with the time in the right-hand side, which consists of (1) time of the first \((l + W_0)\) jobs in the first station, (2) idle time of the first \((l + W_0 - 1)\) jobs in the first machine, and (3) waiting time of the first \((l + W_0)\) jobs in the first machine. This enforces that the new job can start processing in the first station only after the first \(l\)th jobs finish processing in the last machine and leave the system, maintaining the target WIP level in the card loop \((1, l)\).

![Diagram](image)

Figure 4.9 Balance condition for kth job on machine m in the part of Gantt chart

We solve the problem with 400 parts and 6 stations using CPLEX 12.2 MIP solver. To improve the computational time and reduce the complexity of the problem, we assume the FIFO processing of the jobs, just as we did in the previous experiment. Given the target WIP level, the optimal scheduling solution of
the problem is obtained, so as to minimize the makespan. Based on the optimal solution, starting and ending time of the parts are calculated, followed by performance measures. We used the fixed number of warm-up parts in evaluating the key performance measures, the average throughput, the average flow time, and average WIP.

We first see the solution quality of the mathematical problem (MP) to the AWIP algorithm. As shown in Table 4.4, the quality of the AWIP solution are not uniformly superior, but they are comparable to the ones from the optimal solution to the mathematical model. With smaller target WIP levels ($W_0 \leq 10$), the average throughput from AWIP is 5.57% to 8.92%, while the gap becomes greater in larger $W_0$’s. The average flow time is fractionally better in AWIP, especially in the smaller $W_0$’s, but as the target level increases, the solution of MP outperforms the AWIP solutions. Note that here we can observe that increasing the target WIP will increase both average throughput level and flow time, corresponding to the insight of Little’s law (Hopp and Spearman 2011).

Table 4.4 Relative results of optimization model compared to the AWIP results

<table>
<thead>
<tr>
<th>$W_0$</th>
<th>average throughput</th>
<th>average flow time</th>
<th>average WIP</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>1.0593</td>
<td>0.9890</td>
<td>1.0474</td>
</tr>
<tr>
<td>6</td>
<td>1.0848</td>
<td>0.9703</td>
<td>1.0525</td>
</tr>
<tr>
<td>7</td>
<td>1.0557</td>
<td>0.9897</td>
<td>1.0447</td>
</tr>
<tr>
<td>8</td>
<td>1.0892</td>
<td>0.9608</td>
<td>1.0468</td>
</tr>
<tr>
<td>9</td>
<td>1.0640</td>
<td>0.9857</td>
<td>1.0487</td>
</tr>
<tr>
<td>10</td>
<td>1.0567</td>
<td>0.9962</td>
<td>1.0525</td>
</tr>
<tr>
<td>50</td>
<td>1.1162</td>
<td>1.1066</td>
<td>1.2353</td>
</tr>
<tr>
<td>100</td>
<td>1.1519</td>
<td>1.2569</td>
<td>1.4479</td>
</tr>
</tbody>
</table>
Next, we illustrate more clear indication of AWIP solutions based on the 10 different instances of the random processing time. Given the target throughput rate that is large enough to avoid initial blocking, we first obtain the AWIP solutions and its performance measures. We estimate the maximum WIP level system from the solution and applied the value in the mathematical problem. Depending on the instance, $W_0$ values range from 10 to 18. Given each $W_0$ value, we solve the MP and compare the solution to the AWIP approach. Figure 4.10 compares the solutions of the AWIP approach to the MP problem. Given the throughput level of AWIP algorithm, we obtain the corresponding $W_0$ level to the MP problem and solve the MP problem. It is shown that the AWIP solutions have less WIP level with less throughput rate, compared to the MP problem.

<table>
<thead>
<tr>
<th></th>
<th>1.2621</th>
<th>1.5369</th>
<th>1.9402</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>300</td>
<td>1.4491</td>
<td>1.7515</td>
<td>2.5375</td>
</tr>
</tbody>
</table>
Figure 4.10 Experimental results of AWIP vs optimization model

4.5 Concluding Remarks

The main contribution of this work is to successfully design a model of generic supply chain and production line that is robust and efficient with respect to WIP. We combine the concept of the production control system model with feedback control approach for discrete-event system to construct a unified integral controller. The AWIP integral controller manages three desirable characteristics of the production distribution system. The target throughput rate, the average WIP level, and blocking are estimated in every run, and the algorithm dynamically adjust the number of cards in the subsystems based on the feedback algorithm. The overall inventory level of the system is controlled so as to find the best solution that balances among the performance measures.

The effectiveness of the AWIP algorithm is presented in various ways. We first show that the convergence rate of the AWIP solution and compare the quality of it with the FCFS-based solution. The fidelity of the algorithm is presented with different set of initial solutions. Finally, comparison with the mathematical model illustrates that the AWIP solution produces a production line design with smaller WIP level.
5 CONCLUSION AND FUTURE RESEARCH

5.1 Conclusion

In this dissertation, we discuss the robust planning and execution method applied to various applications. In Chapter 1, the motivation and introduction of the dissertation research is provided. In Chapter 2, we begin with the robust surgery planning and execution based on the robust optimization approach. The model consider both underutilization of resources and overtime costs under uncertain surgery processing time. Case studies using sample data of small and large hospital illustrate the benefits of our approach. Compared to the practical fixed gap planning model, the proposed method reduces not only the total cost of surgery scheduling, but also the probability of schedule disruption significantly. Chapter 3 deals with evaluating and monitoring the longitudinal efficiency of the organization. We present the robust productivity index to obtain the productivity bounds for detect the exceptions in performance. Panel data of the evidence-based prevention program are used to demonstrate the suggested approach. Progression/regression patterns of the productivity growth are detected, and innovators who make technical shift are identified and classified into two groups. In Chapter 4, we study the dynamic card controlling algorithm in supply chain management. The number of cards governs the inventory level of the system, and the adaptive WIP algorithm efficiently find the best design of the system using feedback controller. Real time WIP level, target throughput, and blocking are continuously evaluated and card loops are updated accordingly to reach the best possible performance. The algorithm is not only efficient, but also robust in convergence. In each chapter, we also demonstrate how we apply the method to validate the benefits of the proposed approach.
5.2 Future Research

This dissertation builds the analytic models of the robust planning and execution and illustrates applications to the real-world problems. In our future research, we can expand our model to other various domains, such as energy, power supply, smart manufacturing, etc. Also, while this study mainly deals with the dynamics of the robust model between two planning stages, we consider the more holistic approach that covers the entire planning hierarchy. This will entail constructing a proper information technology infrastructure to collect and analyze the data.
REFERENCES


Gupta, S.M., Al-Turki, Y.A., 1997. An algorithm to dynamically adjust the number of kanbans in stochastic processing times and variable demand environment. Production Planning & Control 8, 133–141.


VITA

Yooneun Lee is a Ph.D. candidate in the Department of Industrial and Manufacturing Engineering at the Pennsylvania State University. He received his Master degree in the Industrial and Operations Engineering from the University of Michigan in 2010. He received Bachelor’s degree in the Department of Digital Media at Ajou University in 2005. Since February 2005, he had been working as a software engineer for Samsung Electronics at Mobile Communication Division. His main research interest is in the robust scheduling and feedback algorithm in healthcare and socio-economic planning domain.