Optimal Reconfiguration of Complex Production Lines for Profit Maximization via Simulation Modeling

by

Mustafa Fatih Yegul

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Mechanical Engineering

Waterloo, Ontario, Canada, 2014

AUTHOR'S DECLARATION

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

With the recent trend of re-shoring, transferring manufacturing systems from a workforce-intensive to a capital-intensive production environment becomes more common. One challenge multinational manufacturing companies may face in such an endeavor is reconfiguration of the transferred manufacturing system according to the availability of better machinery in the capital-intensive environment. In this dissertation, based on a real-life problem, I develop several simulation optimization methods for the problem of production line reconfiguration. The case is a reverse transfer of manufacturing system/technology, i.e. transfer from a workforce-intensive environment to a capital-intensive one. I investigate the performances of nine different simulation optimization approaches based on the real-life case in automotive industry to illustrate their relative strengths under different parameter scenarios. I also create a test-bed problem to determine the specifications of these methods, and further analyze their performances. Numerical results may guide the practitioners facing similar challenges in choosing a suitable solution approach depending on the problem size and solution time availability.

Acknowledgements

I would like to acknowledge the contributions of the following individuals and organizations to the development of this dissertation:

First of all, I would like to thank my Supervisor, Prof. Mustafa Yavuz, who believed in me and patiently supported my efforts for many years even though there were times I felt desperate about completing my studies.

I would like to thank Dr. Fatih Safa Erenay and Dr. Soeren Striepe, who gave me countless recommendations and insight in developing the strategy for finalizing my research findings.

I also would like express my gratitude to the members of my comprehensive and defense examination committee: Dr. Ibrahim Dincer, Dr. Mehmet Begen, Dr. Levent Kandiller, Dr. Ozlem Turker Bayrak.

I would like to thank Dr. Gary Polonsky, who helped me find the research position at the company mentioned in the thesis.

I thank Simul8 Corporation, who provided me with a complimentary research license for their professional edition.

Finally, I want to express my gratitude to my wife, who relentlessly supported my studies for years, taking the burden of looking after our three sons.

.

Table of Contents

List of Figures

List of Tables

.

List of Abbreviations

AAD:	Approximation via Analytical Decomposition Method
ABC:	Alias for the company (real name not revealed due to confidentiality agreement)
ACO:	Ant Colony Optimization
ANN:	Artificial Neural Network
AOD:	Approximation through Overlapping Decomposition Method
BEP:	Best Expected Profit
CONWIP:	Constant Work In Process
FGS:	Fast Greedy Search
GA:	Genetic Algorithm
GS:	Greedy Search
LP:	Linear Programming
MIP:	Mixed Integer Programming
MSC:	Manufacturing System Configuration
MTTF:	Mean Time to Failure
MTTR:	Mean Time to Repair
NLP:	Nonlinear Programming
NMIP:	Nonlinear Mixed Integer Programming
OR:	Operations Research
RSM:	Response Surface Methodology
RTMS:	Reverse Transfer of Manufacturing Systems
SA:	Simulated Annealing
TS:	Tabu Search
WIP:	Work in Process
ACO_GS:	Ant Colony and Greedy Search Hyprid Algorithm
ACO_OptQ:	Ant Colony and OptQuest Hybrid Algorithm
MidPnt_GS:	Greedy Search with Mid-Point as the initial solution

Nomenclature

Variable	Explanation
i:	workstation index, $i = 1,2,\cdots,n$.
j:	buffer zone index, $j = 1,2,\cdots,m$.
$i^*(j)$:	workstation that feeds buffer zone j.
Φ:	vector of decision variables.
$\mathbb{R}_i$:	set of labor types required at Workstation i.
$\mathbb{V}_j$:	set of buffer zones that precede Buffer Zone j along the production flow.
p:	sales price per product.
r^{inv}:	rate of inventory carrying cost.
r^{lst}:	rate of lost sale cost.
r^{int}:	annual interest rate.
π:	lost sale cost per product.
T:	time period for which the problem has been solved ($T = 1$ year for this case).
δ:	time interval for the shipment of a fixed amount of product.
ε:	smaller time interval used by the simulation to estimate the average inventory in buffer zones.
t:	simulation time in terms of $\delta, t = 0,1,2,\cdots,T/\delta$.
τ:	simulation time in terms of ε, $\tau = 0,1,2,\cdots,T/\varepsilon$.
D:	annual demand.
D_t:	amount of demand between $(t-1)$ and t.
$S_t(\Phi)$:	amount of sales between $(t-1)$ and t.
$Pr_t(\Phi)$:	amount of production between $(t-1)$ and t.
$I_{j\tau}(\Phi)$:	amount of WIP inventory at Buffer Zone j at time .
$I_{mt}(\Phi)$:	amount of finished goods inventory at time t.
$\bar{I}_j(\Phi)$:	annual average amount of inventory held at Buffer Zone j.
$\bar{I}_m(\Phi)$:	annual average amount of inventory held at finished goods inventory.
C_i^M:	investment cost per identical parallel machine at Workstation i.
C_i^A:	annual amortized cost per identical parallel machine at Workstation i.
M_i:	number of identical machines at Workstation i.
W_{li}:	number of type l workers employed at Workstation i.
C_l^W:	annual cost of a type l worker.
h_j:	inventory carrying cost for Buffer Zone j (e.g., cost of having 1 unit inventory for the whole year).

C_{mtr}: total material costs per product.

C_j^{lb}: isolated cost of labor per item (part) processed at the workstation that feeds Buffer Zone j.

C_j^{eq}: isolated cost of equipment per item processed at the workstation that feeds Buffer Zone j.

C_j^{mt}: isolated cost of raw material used for per item processed at the workstation that feeds Buffer Zone j.

PT_i: processing time at workstation i (expressed in years).

C_j^{va}: cumulative value-added cost of per item at the Buffer Zone j.

$L_t(\Phi)$: amount of lost sales between $(t-1)$ and t.

Per_s The percentage of the expected profit for the test-bed problem.

$\overline{Per_s}$ The percentage of the expected profit for the real life problem.

Chapter 1

Introduction

1.1 Overview

One of the key challenges investors face when building a new manufacturing facility is determining the best manufacturing system configuration (MSC), which is a complex task (Wang & Chatwin, 2005). For example, such a configuration for a production-line-based facility needs to specify many factors including facility layout, production flow, station structures, configuration of the machinery (machinery type and process capacity, buffer size, job allocation mechanism, etc.), number of workers, and skill sets. Koren et al. (1998) report that improving MSC significantly affects reliability, product quality, capacity scalability, and costs; thus, may lead to up to 22% savings in costs and up to 100% improvement in productivity. Therefore, optimizing MSC is a crucial task for profitability. However, this task may become rather critical and complex when the manufacturing system/technology used to build this facility is in its early stages of maturity (e.g., electric car components), because there could be many alternative configurations for each factor mentioned above.

Transferring an existing manufacturing system from another company rather than building one anew can reduce the complexity of optimizing MSC, especially if the transfer is done between two companies with similar level of automation or from an automation-oriented to a workforce-oriented company where implementing the targeted manufacturing system as-it-is is a feasible option. Note that, most manufacturing system transfers, generally from automation-oriented production cultures (e.g., Western countries) to workforce-oriented ones (e.g., Asian countries), fall into this category (Saggi, 2002). On the other hand, a manufacturing system transfer from a workforce-oriented to an automation-oriented company is more challenging. Such a transfer is less complex than building the system anew, because the transferred manufacturing system specifies the production steps, stations, and process flow. However, this manufacturing system may need to be altered based on the machinery options of the automation-oriented company (e.g., replacing the workers of a particular station with machines to improve profitability and reliability).

I refer to this third type of transfer as “reverse transfer of manufacturing system/technology” (RTMS) being motivated from the “reverse transfer of technology” concept of Elshout (1995).

Reverse transfer of technology is defined as transferring knowledge from a relatively labor-intensive environment to a capital-intensive environment. RTMS is especially possible for high-value-added products such as electric car components that are prototyped in labor-intensive production cultures.

Such transfers are viable as several multinational manufacturers consider re-establishing their previously outsourced or offshored domestic production services due to the several factors listed by Ellram (2013) such as: 1) Rising wages in the developing countries as well as increasing global commercial transportation fares on par with surging fuel prices. 2) The improving ratio of [labor output]/[productivity per labor dollar] in some developed countries. 3) Real and anticipated volatility in currency valuation. Moreover, with decades of offshoring and outsourcing, the manufacturing industry shifted to the workforce-oriented countries resulted in technology spillovers (Liu, 2002). Therefore, it is reasonable to expect more advances in manufacturing technologies and development of new manufacturing systems from the workforce-oriented Asian companies. Actually, the three reasons given have already led to a new trend of re-shoring (Gray et al., 2013) with several recent examples (Economist, 2013; Fishman, 2012; Foroohar & Saporito, 2013). In this context, more RTMS cases can be expected in the future at least for the products with high profitability/volume ratio. Therefore, developing novel Operations Research (OR) approaches to optimize MSC in an RTMS environment is desirable.

1.2 Motivation

This research is motivated by a real-life case: A capital-intense multinational corporation in a western country (referred to as "ABC" – actual name not to be mentioned as per the confidentiality agreement) plans to build a new manufacturing plant in the same country to produce electric car components. The manufacturing system, in the early stages of its life cycle, is designed by a labor-intensive Asian company. ABC will build a new manufacturing plant rather than outsourcing these components because: 1) the labor-intensive company has diversified product portfolio with smaller production volumes compared to larger potential demand from ABC. Having more capital, ABC can better manage the potential demand for its end-products by producing these components. 2) Having access to the better automated-machinery options, ABC may benefit from better product quality and more reliable supply chain by producing these components. 3) ABC may reduce the transportation costs if these components are produced in the same continent as its major customers. The existing manufacturing system is a flow-type production line. ABC needs to reconfigure it incorporating newly designed automated machinery to improve efficiency.

1.3 Contribution

Motivated by a real-life case, this thesis proposes an approach to the RTMS problem where a production line for a high-value-added product will be built in a capital-intensive country based on a similar existing workforce-oriented manufacturing system. Based on the information provided by the ABC engineering team, I assume that the demand for the end-product is constant and known for the period of interest. I also assume that the stations and process flow in the existing system will be preserved during the transfer. However, the structure of the stations (machinery, workforce, or hybrid) and their configurations (number of machines, process rate, buffer sizes etc.) are decision variables.

My approach for this problem consists of three steps: 1) Determination of the production system requirements, performance measures, and associated data, 2) Determination of the alternative station structures and values of the configuration variables for each station structure as well as their evaluation via OR tools. 3) Determination of the optimal station structure and the corresponding configuration variables. A systematic approach covering these steps (based on the experience with the real-life case) is provided in Section 1.4. In addition, a discrete-event simulation model is developed to evaluate the performance of any given station structure and corresponding configuration variable combination. Several simulation optimization algorithms are proposed to find the optimal production line configuration.

The contributions of this thesis are four-folds: 1) to the best of my knowledge, this is the first study that considers optimizing production line configuration in an RTMS case. A systematic approach for the problem is provided as a guideline for the companies facing similar challenges. 2) A novel simulation model is proposed to evaluate the performances of different production line configurations incorporating several configuration variables including, number of machines, machine speeds (cycle time), min/max buffer allocations, and number of workers. 3) Several search-algorithm-based simulation optimization methods are proposed and tested to illustrate their relative strengths under different parameter scenarios in terms of solution quality and speed. 4) As being motivated from a real-life case, real-data is used for the numerical experiments.

1.4 The Real-life Case and the Systematic Approach to the Problem

Figure 1.1 illustrates a six-step generic systematic approach that represents how ABC managed this task of RTSM combined with the optimization of production line configuration. Understanding the features of the real-life production line and how ABC followed the steps in Figure 1.1 is paramount to appreciate the contribution of the thesis.

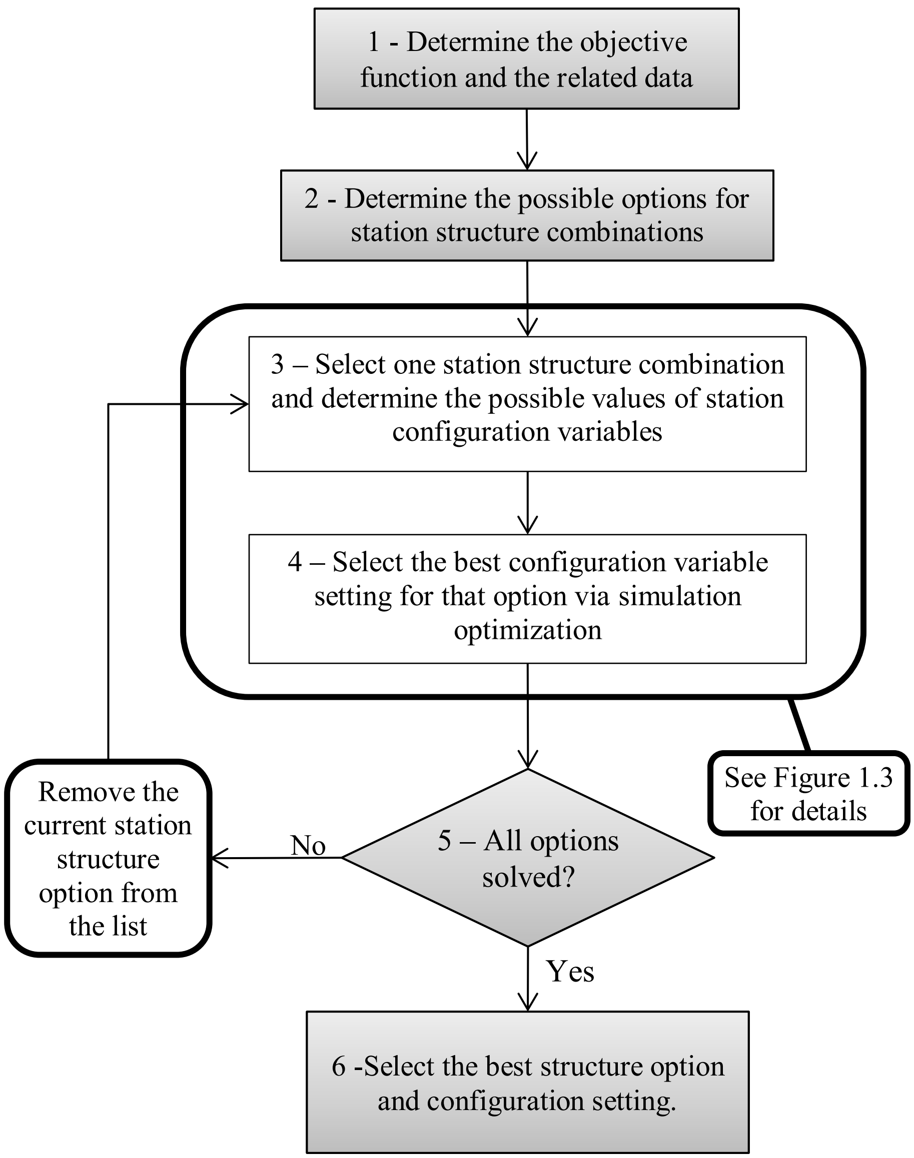


Figure 1.1 Systematic Approach for Practitioners.

As the first step, decision makers need to determine the objective function, decision variables, and the related data. ABC sees it strategically very important to reach a throughput level equal to annual estimated demand. Besides there are cost factors involved such as investment cost for acquiring equipment, costs related to work-in-process (WIP) and finished product inventories, labor cost, and cost of lost sales. Therefore, maximization of annual profit has been taken as the objective function, which reflects all the conflicting goals of ABC. The configuration variables includes number of machines, machine speeds (cycle time), min/max buffer allocations, and number of workers. ABC assigned an analyst who contacted potential customers to estimate the future demand. A cost analyst estimated cost of labor, equipment, and raw materials. Section 4.2.1 presents more information about the input data.

The second step involves determining the possible structure options for each workstation specified by the existing labor-intensive production line and preparing a list of station structure combinations for the whole production line. ABC established a multi-disciplinary team including engineers, production planners, and managers to decide which options are available for each station (e.g., selection of machine types, selecting manpower over machines, or a combination of both). The number of possible station structures could be high; therefore, the practitioners may need to eliminate un-preferable structures to reduce the complexity of the problem. ABC team achieved to reduce the possible structure combinations to a single one by eliminating the other options based on their performances and costs. Note that ABC is satisfied with considering only a single station structure combination because they aim to build a prototype manufacturing system. However, they are aware that upon successful introduction of the product in the market, they will need to consider many alternative structure combinations for the full implementation of the prototyped manufacturing system.

At the third step, a station structure option should be selected from the list generated in the previous step. Then, the possible configuration variable values for each station should be specified (number of machines, machine speed, number of workers, buffer sizes, etc.). To prevent confusion about the concepts of station structure option and station structure configuration, I provide a simple and self-explanatory example in Figure 1.2.

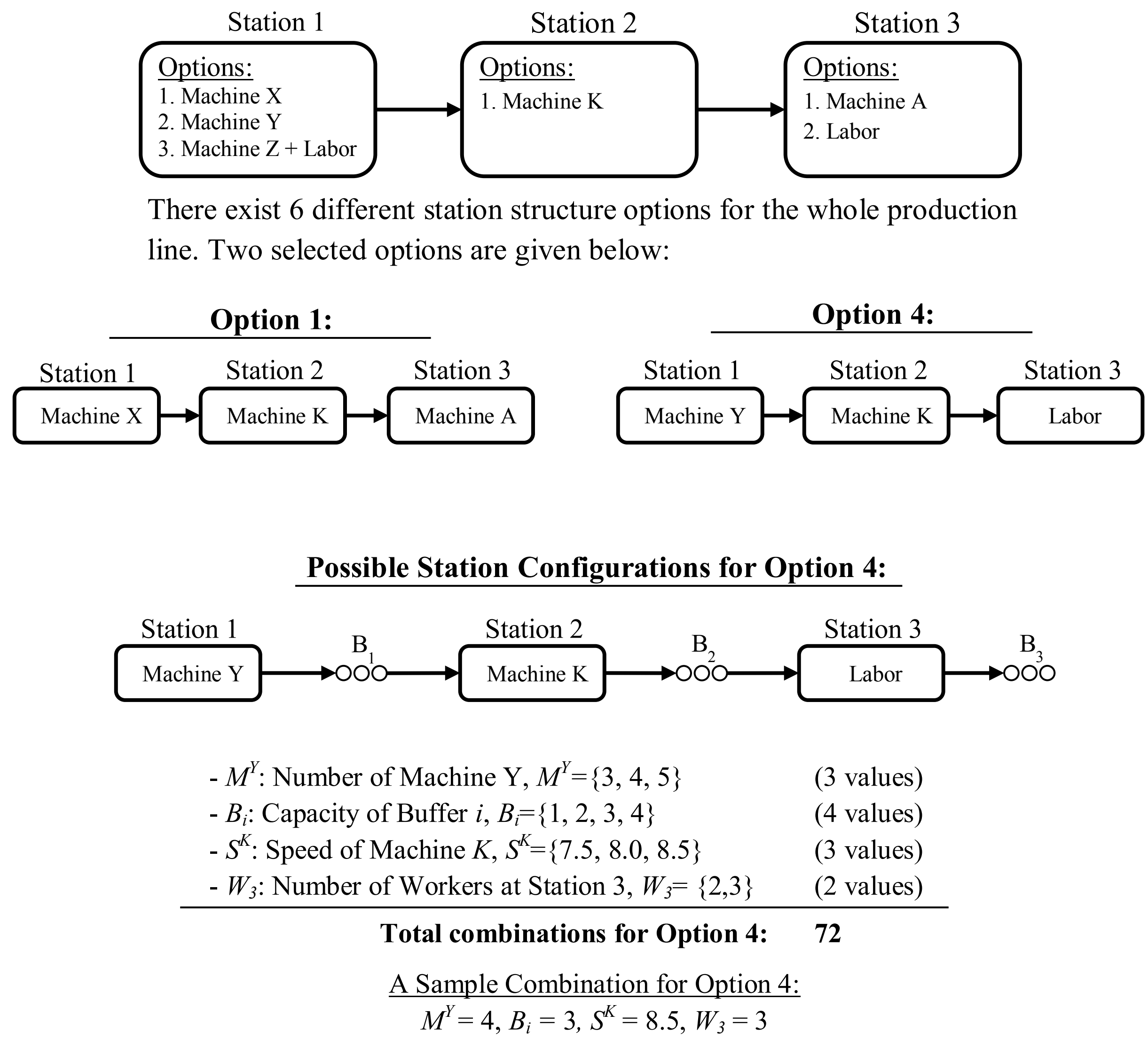


Figure 1.2 Illustrative Example for Station Structure Options and Configurations.

The fourth step is to find the best configuration variable values for a given station structure. ABC agreed with the author of this thesis to develop a simulation model to evaluate the performances of possible production line configurations. Several simulation optimization methods were also developed to search among the possible configurations and find an approximately optimal solution. When the approximate optimization of the configuration variables for the selected station structure option is complete, its performance is recorded and another station structure option is selected from the list in Step 2. Once all options are evaluated then the best one is selected among them. Simulation optimization process that is explained in Steps 3 and 4 of the general approach (Figure 1.2) is depicted in Figure 1.3.

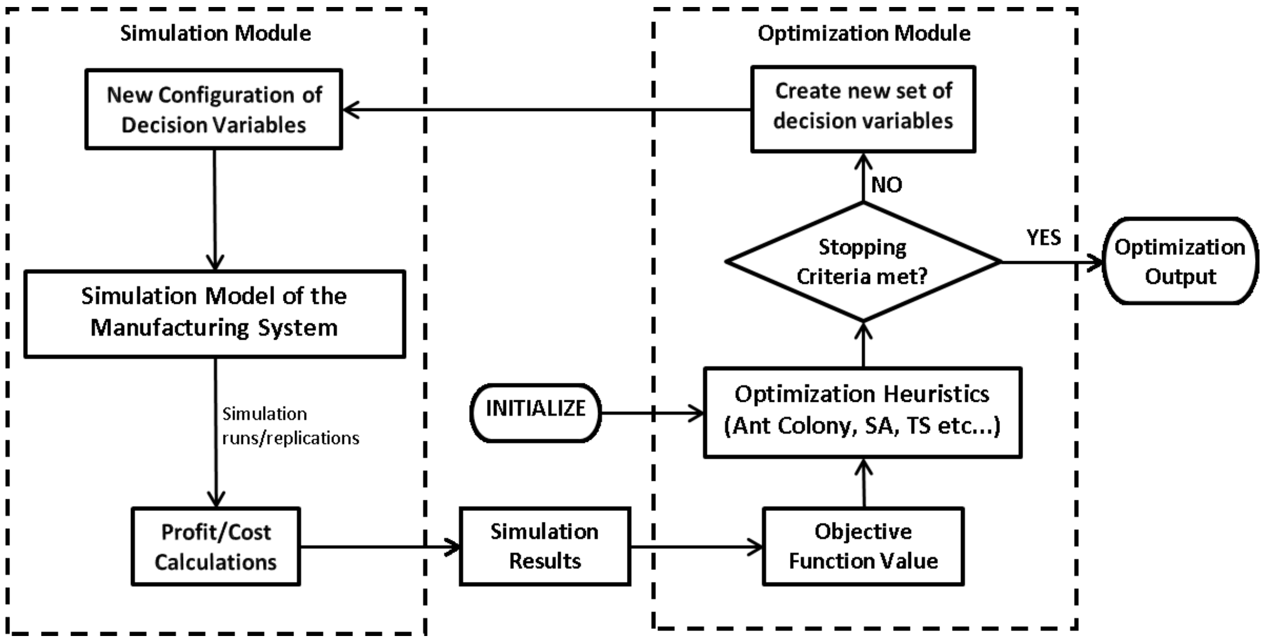


Figure 1.3 Simulation Optimization Process.

Optimization algorithms are coded in Visual Basic, which controls the simulation module (Simul8®) and the profit calculation worksheet file (MS Excel®) simultaneously. Screenshots from the optimization module is given in Figure 1.4.

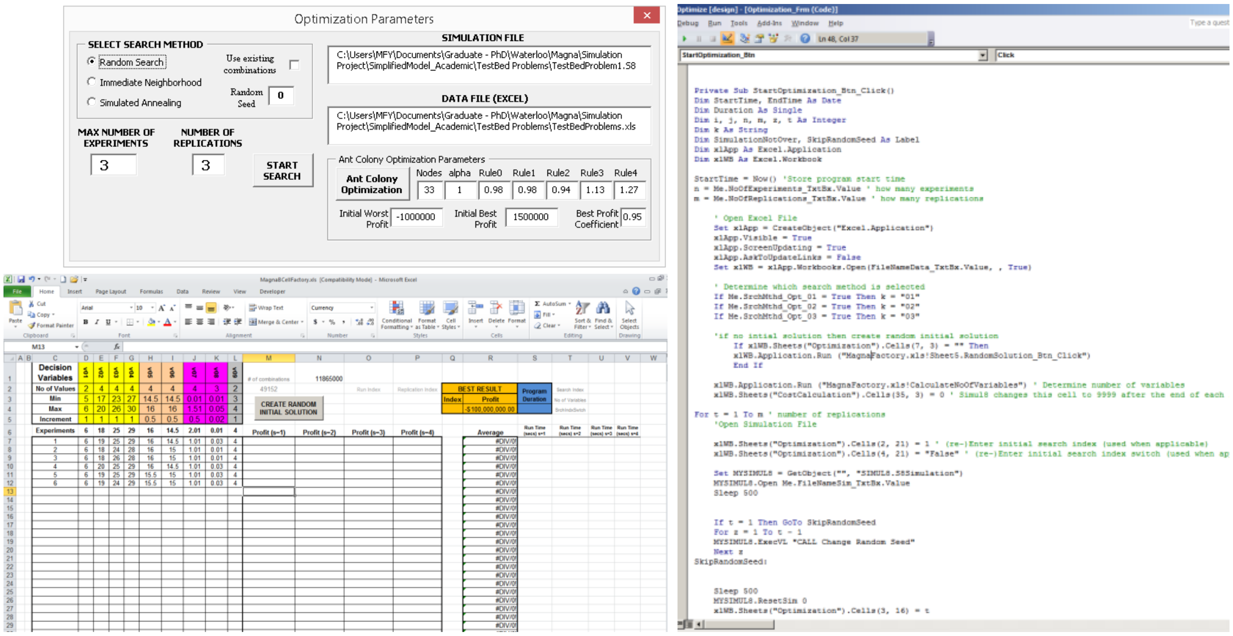


Figure 1.4 Optimization module screenshot.

1.5 Thesis Outline

Next chapter gives background information for the case and reviews the related literature. Chapter 3 provides detailed description of the real-life problem. Chapter 4 describes the simulation model in details, and also provides information on the test-bed problem. Chapter 5 describes the simulation optimization methods proposed to solve this problem. Chapter 6 presents the results of the numerical experiments on a test-bed problem and the real-life case. Finally, conclusion remarks are given in Chapter 7.

Chapter 2
Background and Literature Review

2.1 Reverse Transfer of Manufacturing System/Technology

Because most technology transfers are from capital-intensive to labor-intensive markets, there are only a few studies directly covering RTMS. In one of those studies Elshout (1995) defines reverse transfer of technology as "all forms of technological services which are exported by a developing country in exchange for financial benefits". However, more studies are likely to appear on this concept as several sources indicated that a number of capital-intensive manufacturers in countries such as US, Canada, and Japan consider re-opening some of their domestic manufacturing operations. The Economist recently reported that a growing number of American companies are moving their manufacturing back to the United States (Economist, 2013). An article published in Nikkei (2013) reports that Panasonic is considering bringing back its white goods production back to Japan. Globe and Mail daily newspaper narrates that reshoring could be a good opportunity for Canadian economy (Carmichael, 2012).

In addition, the recent trend of re-shoring, i.e., bringing manufacturing part of the supply chain back to or closer to the western countries for profitability, is reported to receive increasing interest (Ellram, 2013). Existing related studies in the re-shoring literature focus on the redesign of the supply chain or logistics networks under the option of domestically re-establishing the previously outsourced manufacturing operations (Gray et al., 2013). However, unlike this thesis, those studies do not consider the details of the manufacturing system re-configuration in an RTMS case.

2.2 Production Line Configuration

There is a vast amount of research devoted to the development of approaches for the performance analysis and optimization of flow-type manufacturing systems with finite buffers between workstations. Majority of these studies propose methods towards optimal buffer allocation, while others concentrate on performance measures such as machine selection or determining the number of parallel identical machines. When production lines include unreliable machines with stochastic break-down and repair times, planners use buffers as a means of increasing throughput. In a flow line, if a station stops working due to maintenance or break-down, the succeeding station may continue working provided that the buffer before that station is not empty; if it is not starved in other words. In contrast, a station will be blocked if the buffer that comes right after it is full. So, determining the optimal buffer capacities has been subject to many studies in production management field.

Production lines may either be of continuous (fluid transfer) or discrete nature. Focus in this study is given to discrete production lines. Unless otherwise stated all production lines mentioned in this study imply discrete flow of materials.

2.2.1 Classification of Production Line Problems

Production lines problems can be classified based on different viewpoints such as line assumptions, decision variables, objectives and solution methodology.

2.2.1.1 Line Features

A basic linear production line is shown in Figure 2.1, where workstations are denoted by P_i and buffers by B_i. This is the simplest form of a flow line with n workstations and (n–1) buffers.

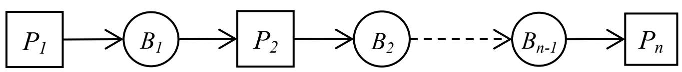


Figure 2.1 Basic production line.

Commonly used classifications regarding flow lines are as follows (Dallery & Gershwin, 1992):

Synchronous/asynchronous lines: Most real systems are unsynchronized. That is, the machines are not constrained to start or stop their operations at the same instant. Another term used is paced/unpaced lines.

Homogeneous/non-homogeneous lines: Lines with all workstations having equal processing rates are homogeneous lines.

Saturated/non-saturated lines: In saturated models, the first machine is never starved and the last is never blocked, which assumes that there are always enough raw materials available when necessary and finished goods inventory is unlimited. In literature, most lines are assumed be saturated.

Processing times: Processing times of workstations may either be deterministic or stochastic. Usually, if the machines in workstations are automated with minor human input, processing times are assumed to be deterministic. In other cases where human input is in considerable amount, processing times are taken as random variables of a probabilistic distribution.

Process reliability: In contrast to process times, if the process is performed mainly by human resources, then there are no considerable machine failures that can halt processing. Such processes are assumed to be reliable. If the process depends on machine performance, then the process stops from time to time due to machine breakdowns. This type of processes is assumed to be unreliable and treated with probabilistic down times and repair times.

Mean time to failure (MTTF) can be calculated in two different ways. First one accounts for all the times including those when the machine is not working due to blockage or starvation. Second one only takes into consideration the busy time of the machine. Mean time to repair (MTTR) is used to describe the mean repair time when a machine is down.

Not all the production lines in real world are linear. Other forms and shapes a flow line can take are explained below.

Series-parallel lines (parallel machines)

In series-parallel lines, workstations may include more than one identical/non-identical machines performing the same operation (Figure 2.2). The idea is to reduce the cycle time of the workstation by adding more machines. Number of machines for any workstation (m_i) can be a decision variable for series-parallel flow lines.

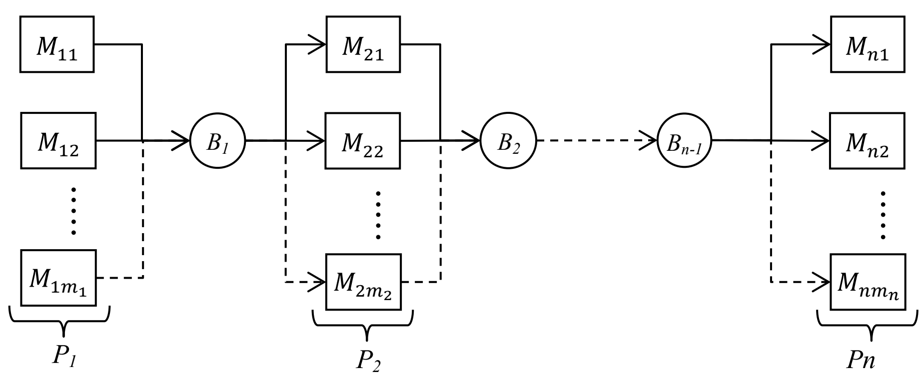


Figure 2.2 Series-parallel production line.

Split / merge lines

In such production lines, at some point the line forks into m number of parallel lines, and then merges back into a single line (Figure 2.3). There is no disassembly at the station where splitting occurs, or there is no assembly at the station where merging takes place. Instead, parts produced are classified and fed into one of m parallel lines the based on a probability. This classification can be due to quality issues or some other design related specification.

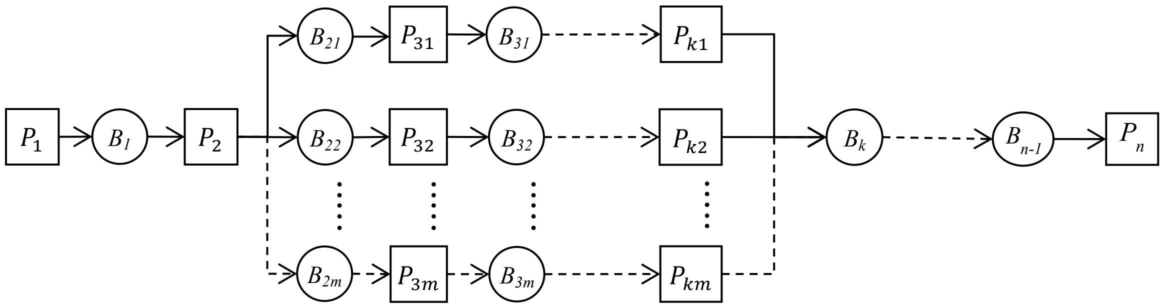


Figure 2.3 Split / merge production line.

Lines with rejection

Rejection occurs when there are inspections at certain workstations. Parts are inspected and those parts that are not acceptable are separated as rejections as seen in Figure 2.4 (R_i denotes rejection zone for workstation P_i).

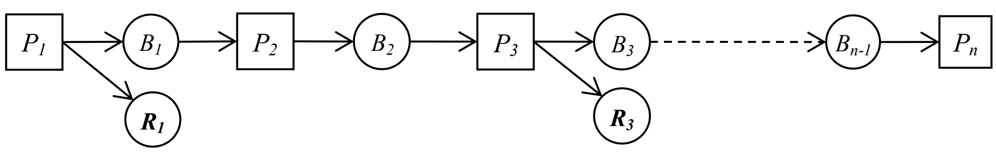


Figure 2.4 Production line with rejection.

Reentrant lines (lines with rework)

In some production lines, parts need to be processed by the same workstations more than once (Figure 2.5). This could be either due to a quality problem that needs to be fixed, or due to certain design specifications that requires re-work on the parts at certain stations. For example, in a car factory, cars can be processed by the painting station more than once. Thus, there is more than one buffer zone before the workstation where the parts reenter the system (B_i'). Reentry workstation receives parts from buffers based on a priority rule specific to the production system itself. For example, if the reentry is due to quality problems, priority could be given to reentering parts.

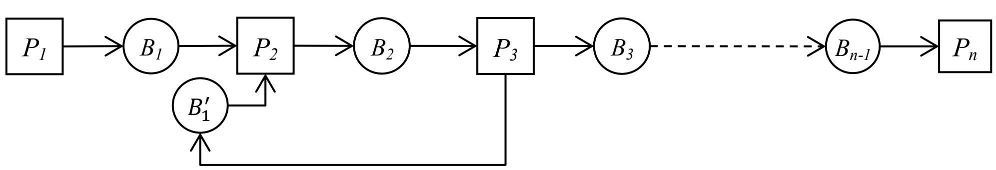
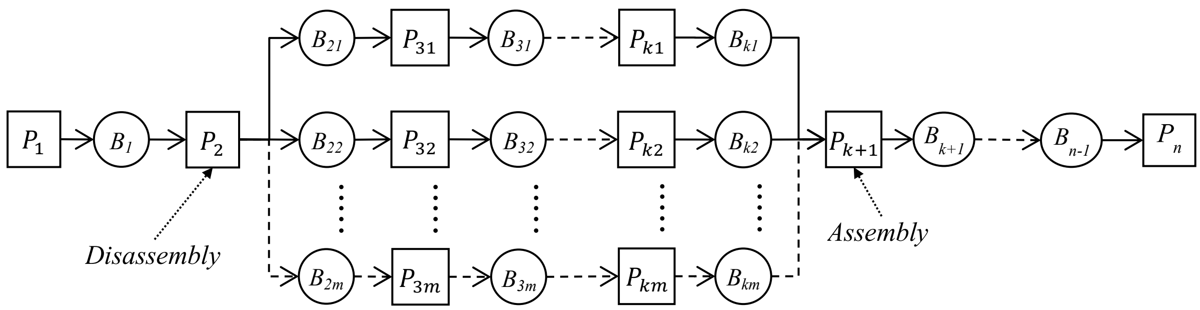


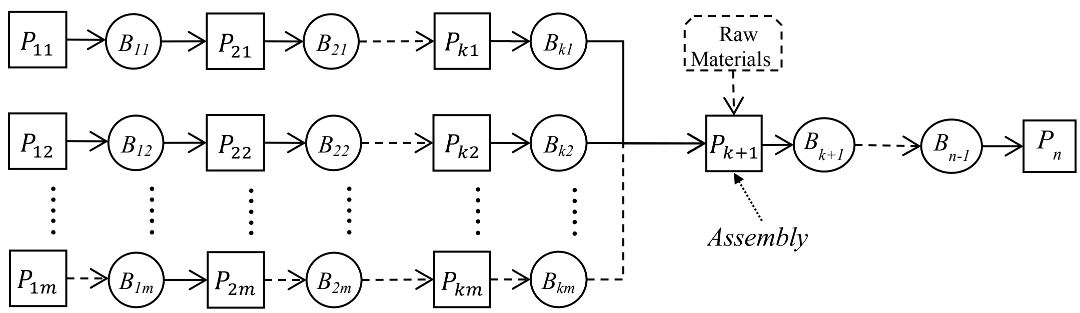
Figure 2.5 Production line with rework (reentrant lines).

Assembly/disassembly lines

In some flow lines, parts may be disassembled and/or assembled at certain workstations (Figure 2.6.a). Parts are divided into several subparts. Subparts are then processed in parallel lines before they are joined together with an assembly operation. It is also possible to have an assembly operation without a disassembly process, where processed / unprocessed raw materials are assembled (Figure 2.6.b).



(a) Disassembly/assembly line.

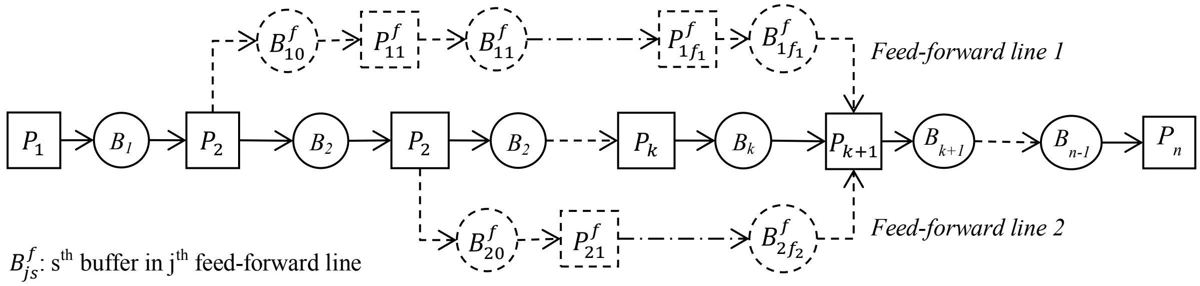


(b) Assembly-only line.

Figure 2.6 Assembly/disassembly lines.

Feed-forward / bypass lines

In such lines, some part may leave the main flow line to bypass some of the stations and then join the main flow line again (Figure 2.7). During the bypass, parts may or may not be subject to additional processing. This might be the case when the product manufactured has different versions, which require different processing at certain points in the system.



B_{js}^f: s[th] buffer in j[th] feed-forward line

P_{js}^f: s[th] workstation in j[th] feed-forward line

f_j: number of workstations in j[th] feed-forward line, $f_j \geq 0$.

Figure 2.7 Feed-forward / bypass flow line.

2.2.1.2 Decision Variables

The most common decision variable used in related research is buffer capacity (max buffer size). Allocating space for Work-In-Process (WIP) inventory is costly, but achieving the desired output rate is not possible with zero buffers in most of the cases. Machine failures and variability in processing times may cause workstations becoming either blocked or starved, which negatively affects the overall throughput. That's why determining the optimal buffer capacity between workstations is an important decision variable.

For production lines where CONstant Work In Process (CONWIP) approach (Spearman et al., 1990) is applied, another decision variable would be the optimal allocation of fixed amount of total buffer spaces to available buffer zones.

For series-parallel lines, estimating the number of machines for parallel stations can be significant. While increasing the number of machines will reduce the cycle time of the workstation, it will have a negative effect on the investment and operating costs.

For unbalanced lines, processing speeds (cycle time) of certain machines could also be used as a decision variable. For some workstations that share the same type of labor resources, optimal number of workers can become a decision variable as well.

2.2.1.3 Objectives

The most common objective of production line problems is maximizing the throughput. Planners try to find the best configuration for the production line that will maximize the overall output rate of the line.

Although less used by the researchers, other objectives may be minimization of WIP, minimization of cost, or maximization of profit (as used in this thesis).

2.2.2 Methodologies for Solving Production Line Problems

Production line problems mostly involve finding the optimal buffer sizes or optimal buffer locations (distribution of a specified total number of WIP inventory to a specified number of buffer zones). Solution methods try to find the best configuration of buffers and resources that would optimize one or more performance criteria. These methods mainly incorporate a search algorithm and a performance evaluation technique. Heuristics, dynamic programming or non-linear programming based methods have been used by researchers. As for performance evaluation, exact analytical methods (for shorter flow lines with 2 or 3 workstations), analytical decomposition approximation, aggregation or simulation have been employed by researchers.

2.2.3 Related Literature

Table 2.1 gives a summary of recent research on production line design (last row describes the assumptions of the model analyzed in this study). Review papers of Bergeron et al. (2010), Papadopoulos & Heavey (1996), and Dallery & Gershwin (1992) also give valuable insight on production line research. Besides, Tempelmeier (2003) analyses the problems in the design of real-life asynchronous production lines under stochastic conditions that may be due to breakdowns or random processing times.

Use of simulation based optimization for buffer allocation problems go back to 60s. Researchers either use meta-heuristics (e.g. Tabu Search (TS), Genetic Algorithm (GA)) or meta-models (e.g. regression models, Artificial Neural Network (ANN)) for optimization of their simulation model. Anderson & Moodie (1969) generate a meta-model based on regression using simulation output data.

Ho et al. (1979) produced a gradient technique based on perturbation analysis to determine buffer sizes. Powell (1994) investigates the allocation of buffers in a three-station unbalanced flow line. In their study, Bulgak et al. (1995) use GA for buffer size optimization. Lutz et al. (1998) presents a TS based simulation optimization algorithm in order to calculate near-optimal buffer allocations in a flow line. Vouros & Papadopoulos (1998) developed a knowledge based system for optimal buffer allocation. Harris & Powell (1999) offered a simplex search method to maximize throughput by optimized buffer placement in reliable serial lines. In a study by Spieckermann et al. (2000), GA and Simulated Annealing (SA) was used to find the optimal combination of buffers and cycle times to achieve the desired production rate. Alabas et al., (2002) compare GA, SA and TS based heuristic techniques to find the optimum number of kanbans in a Just in Time system. Dengiz & Belgin (2007) propose using response surface methodology to find the optimum levels of considered factors. Unlike many researchers, who studied the buffer size alone, Qudeiri et al. (2008) proposes a simulation based GA that maximizes production efficiency of a serial-parallel production line by finding the optimal buffer sizes, number of machines, selecting the best machine types. Vitanov et al. (2009) generated an ant-colony based optimization algorithm for near-optimal allocation of buffers in an assembly line.

Some authors proposed expert systems for optimization of production systems via simulation. Mebrahtu et al. (2009) developed an expert mechanism that interprets the simulation results to achieve gradually the optimized manufacturing performance. Masmoudi (2006) presents an approach that blends an expert system with simulation for optimally sizing manufacturing cells. Masmoudi et al. (2007) solved the machine and labor sizing problem in manufacturing systems with an expert mechanism based simulation optimization.

There are also studies that investigate the use of meta-models based on simulation results to achieve optimized production system design. Lin et al. (1994) use regression meta-model to maximize the throughput of an automated flow line by optimizing system parameters such as the number of machines, machine processing times, and capacity of the buffer. Altiparmak et al. (2002) generated an ANN based simulation meta-model to optimize buffer sizes in an asynchronous assembly system. Durieux & Pierreval (2004) use regression based meta-modeling to assess the influence of the material handling system in a flow line. Feyzioglu et al. (2005) also utilize regression meta-modeling for optimal sizing of manufacturing systems. In another study, Dengiz et al. (2006) employ a regression meta-model for generating a decision support system to predict the number of machines and workers necessary to achieve the desired manufacturing output level. Baykasoglu (2008) proposes gene expression programming technique for meta-modeling simulation outputs so as to optimize production line design. Dengiz et al. (2009) make use of ANN trained via TS to optimize two different manufacturing systems.

Table 2.1 Classification of Production Line Problems

Reference	Line Features														Decision Variables						Objective(s)				Methodology
	A		B								C		D		Buffer										
	Synchronous	Asynchronous	Homogenous	Inhomogeneous	Rejection	Re-entrant / rework	Split / merge	Parallel machines	Assembly / disassembly	Feed forward / by-pass	Deterministic times	Stochastic times	Reliable processes	Unreliable processes	Maximum	Location / allocation	Numb. of machines	Machine selection	Numb. of workers	Machine speeds	Cost minimization	WIP minimization	Profit maximization	Maximum throughput	
(Bonvik et al., 2000)	*		*						*		*			*	*							*		*	Approximation via analytical decomposition (AAD) method.
(Gershwin & Schor, 2000)	*		*								*			*	*								*	*	AAD method.
(Gershwin & Burman, 2000)		*		*					*		*			*	*									*	AAD method.
(Helber, 2000)	*		*		*		*				*			*	*									*	AAD method.
(Jeong & Kim, 2000)		*		*					*			*		*	*			*			*			*	Heuristic methods finding the minimum cost for a desired throughput.
(Kouikoglou, 2000)		*		*					*			*		*	*									*	Steepest descent alg. & simulation for perform evaluation & gradient estim.
(Chan, 2001)		*		*							*			*	*									*	Simulation, multi-product case with focus on preventive maintenance.
(Chan & Ng, 2002)		*		*								*	*		*	*								*	Dynamic programming, allocating a given number of buffer spaces.
(Kouikoglou, 2002)		*		*					*		*			*	*									*	Hybrid simulation/analytic model.
(Sabuncuoglu et al., 2002)		*		*				*	*			*	*		*	*	*							*	Simulation modeling.
(Hemachandra & Eedupuganti, 2003)		*		*					*			*	*		*							*		*	A search algorithm with performance evaluation via Markov Chain.
(De Vericourt & Gershwin, 2004)		*		*								*		*	*									*	AAD Method.
(Li, 2004)	*		*			*					*			*	*									*	Approximation through overlapping decomposition (AOD) method.
(Sadr & Malhame, 2004)		*		*							*			*	*									*	Decomposition/aggregation method (Kanban controlled line).
(Li, 2005)	*		*		*	*	*		*	*	*			*	*									*	AOD method.
(Blumenfeld & Li, 2005)	*		*								*			*	*									*	Approximate analytical formulation (all buffers are identical).
(Enginarlar et al., 2005)	*		*								*			*	*									*	Monte-Carlo simulation with focus on lean buffering.
(Nourelfath et al., 2005)	*		*								*			*	*			*			*			*	Ant system meta-heuristic with AAD method.
(Bulgak, 2006)		*	*						*		*			*	*									*	GA-based simulation optim., meta-modeling using simulation & ANN
(Hu & Meerkov, 2006)	*		*								*			*	*									*	Aggregation method (with emphasis on lean buffering).
(Kim et al., 2006)		*		*	*	*						*		*	*									*	Approximation-based on mean value analysis technique.

Table 2.1 Classification of Production Line Problems – continued.

Reference	Line Features														Decision Variables						Objective(s)				Methodology
	A		B								C		D		Buffer										
	Synchronous	Asynchronous	Homogenous	Inhomogeneous	Rejection	Re-entrant / rework	Split / merge	Parallel machines	Assembly / disassembly	Feed forward / by-pass	Deterministic times	Stochastic times	Reliable processes	Unreliable processes	Maximum	Location / allocation	Numb. of machines	Machine selection	Numb. of workers	Machine speeds	Cost minimization	WIP minimization	Profit maximization	Maximum throughput	
(Altiparmak et al., 2007)		*	*								*			*	*									*	ANN meta-modeling & simulation.
(Chiang et al., 2008)	*		*								*			*	*									*	Aggregation & bottleneck identification with focus on lean buffering.
(Colledani et al., 2008)	*		*				*				*			*	*									*	Aggregation & decomposition methods, focus on multiproduct production lines.
(Manitz, 2008)		*		*					*			*		*	*									*	Decomposition approach.
(Nahas et al., 2008)		*		*							*			*	*		*	*			*			*	AAD method and Ant Colony Optimization (ACO).
(Qudeiri et al., 2008)		*		*							*			*	*		*	*						*	Aggregation method and GA.
(Yamamoto et al., 2008)		*		*						*	*			*	*									*	Simulation modeling and GA.
(Battini et al., 2009)		*	*								*			*	*	*								*	Simulation modeling (with focus of micro-breakdowns).
(Nahas et al., 2009)		*		*				*			*			*	*			*			*			*	Analytical decomposition approximation with ACO & SA.
(Shaaban & McNamara, 2009)		*		*								*	*		*									*	Simulation modeling
(Vergara & Kim, 2009)		*		*								*		*	*	*								*	Simulation modeling.
(Xu et al., 2009)		*		*							*			*	*									*	Fuzzy linear programming and aggregation techniques
(Qudeiri et al., 2009)	*		*			*					*			*	*									*	Analytical decomposition approximation and GA.
(Shi & Gershwin, 2009)	*		*								*			*	*								*	*	Non-linear programming based algorithm.
(Aziz et al., 2010)		*		*				*			*			*	*									*	Approximation method based on discrete state Markov chain.
(Liu & Li, 2010)	*			*			*				*			*	*									*	Approximation based on overlapping decomposition method.
(Xia et al., 2010)	*		*								*			*	*									*	AAD Method.
(Demir et al., 2011)	*		*								*			*	*							*		*	AAD method and tabu search.
(Helber et al., 2011)		*		*								*	*		*			*						*	Simulation incorporated into a linear programming framework.
(Papadopoulos et al., 2013)	*											*	*		*	*						*		*	Markovian & decomposition with GA, TS, SA & Myopic algorithms
This Study		*		*	*	*		*	*			*		*	*		*		*	*			*		Simulation optimization.

The studies proposing mathematical modeling approaches generally use stochastic models, linear programming (LP), nonlinear programming (NLP), mixed integer programming (MIP), or nonlinear mixed integer programming (NMIP) to optimize the production line configuration under simplifying assumptions or/and by decomposing the production line into manageable sections (Li, 2005; Liu & Li, 2010; Nahas et al., 2009; Sadr & Malhame, 2004). For tractability, these studies generally limit themselves into production line systems that consist of only a few stations and up to 3 different configuration variable types. Because this research is based on a real-life case, the problem involves 24 stations and 5 types of configuration variables, for which developing an analytical model is not practical. However, a conceptual model capturing the objective function of profitability is given in Section 3.5.

2.3 Simulation Optimization

There is a huge number of methods for simulation optimization. To date, a number of comprehensive reviews about simulation optimization techniques have been written by several authors including Meketon (1987), Jacobson & Schruben (1989), Safizadeh (1990), Azadivar (1992), Fu (1994), Fu et al. (2005), Andradottir (1998), Swisher et al. (2000, 2003, 2004). Tekin & Sabuncuoglu (2004) classifies simulation optimization problems into two main categories; local optimization and global optimization as seen in Figure 2.8.

Some recent studies in the field are as follows. Pichitlamken et al. (2006) propose a sequential indifference-zone selection procedure for optimization of expensive simulations. Kim (2006) presents a review of gradient-based techniques for continuous optimization. Ghiani et al. (2007) developed an iterative method based on SA framework for solving discrete optimization problems. Carlos et al. (2008) use reinforcement learning algorithms with ANN for optimization of simulation models. Horng & Lin (2009) propose an ordinal optimization theory-based two-stage algorithm for a good enough solution of the stochastic simulation optimization problem with huge input-variable space.

Another classification of simulation optimization is based on whether the problem is constrained or not. If the problem is constrained, it means that some simulation output parameters are required to stay within specified feasible region. Existence of constraints regarding input variables is irrelevant, because almost all models have limits on input variables. Most of the related research deals with solving unconstrained optimization problems. Study of Hong & Nelson (2006) on the other hand, present an algorithm called COMPASS that can solve constrained simulation optimization problems.

Angun et al. (2009) and Kleijnen (2008b) propose a methodology that generalizes classic RSM to account for stochastic output constraints.

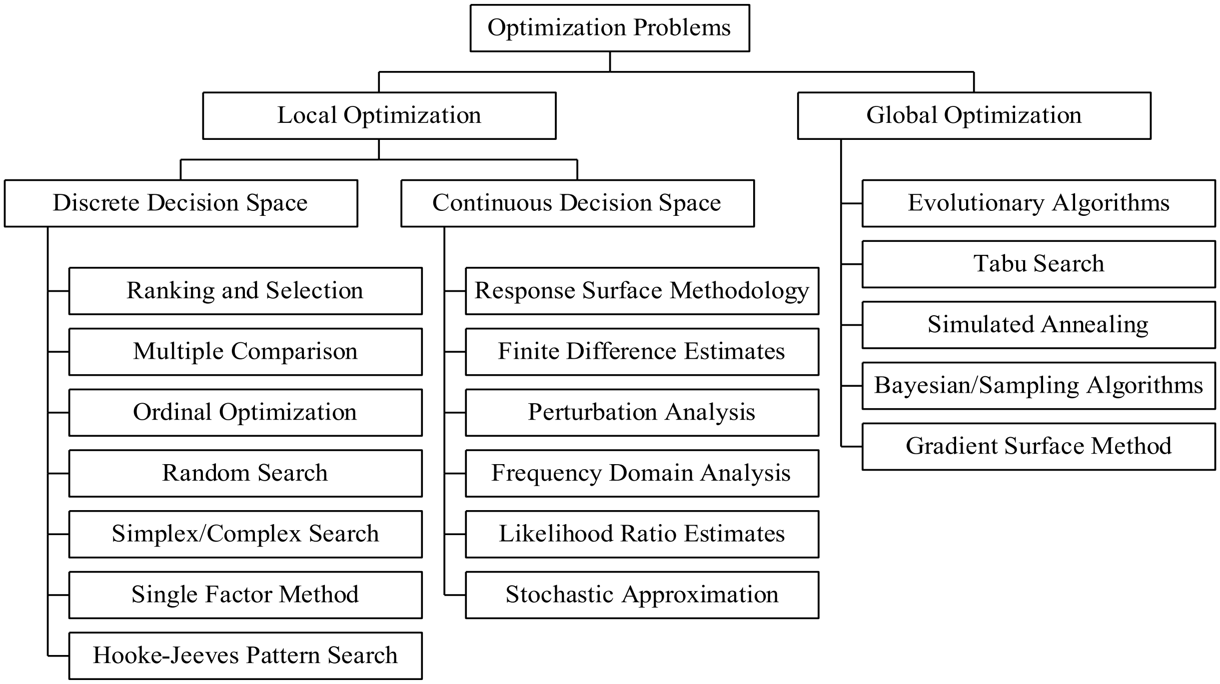


Figure 2.8 Simulation Optimization Classification Scheme (Tekin & Sabuncuoglu, 2004)

While several different simulation optimization techniques are proposed in the literature, no more than a few search mechanisms are tested in most simulation optimization studies (Kleijnen, 2008a). In this paper, several different simulation optimization methods are proposed rather than focusing on a few of them. Then their relative performances in the solution quality and speed spectrum are reported.

Chapter 3
Problem Description

The problem this thesis deals with is a real life case, where ABC Company in a capital-intensive environment transfers a new manufacturing technology/system from a labor-intensive environment. ABC needs to reconfigure the production line in order to achieve higher capacity with less amount of workforce. ABC also wants to increase the overall product quality by introducing more automatized stations to the lines. The production line comprises complexities, which are not desired to be simplified for the sake of calculation simplicity. ABC wants to get a clear picture of the system performance by also incorporating probabilistic nature of the system. Complexity of the system is described in next section.

3.1 Assumptions and Line Complexity

This thesis tries to maximize the profit for a complex production line. Complexity of system stems from the several factors listed below (see Section 2.2.1 for explanations of terms). The system;

- is asynchronous and inhomogeneous.
- incorporates series-parallel flow .
- has one assembly station synchronizing two separate sub-lines (assembly system).
- involves parts returning to the same workstation for reprocessing, i.e. reentrant-flow (Burman, 1995).
- contains unreliable workstations with stochastic downtimes and repair times.
- contains both deterministic and stochastic processing times.
- contains workstations with random setup (change over) times.
- has processes where rejection or scrapping occurs.
- is designed to work non-stop in three shifts.
- has stations that work 7 days a week, while others working 6 days a week.
- involves minimum wait times at certain buffers (parts need to wait for a certain amount of time before they can be processed by the next station).
- contains processing of different parts at certain stations in a cyclic manner (i.e. 'station k' is scheduled to process 'part y' after processing a specified number of 'part x').

Other assumptions about the system are as follows;

- Production flow is saturated.
- All machines run with 85% efficiency (based on the design specification of the producer).

  $$e_i = \frac{MTTR_i}{MTTR_i + MTTF_i}$$

  where

  e_i is the isolated efficiency of workstation i, $MTTR_i$ is the mean time to repair for workstation i, and $MTTF_i$ is the mean time to failure for workstation i.

- Some workstations are automated, others are non-automated.
- Production rate is calculated based on good product output only (rejects are not counted).
- System produces single type of product as specified by ABC.
- The annual demand for the product is fixed and known.
- Buffers between workstations cause inventory holding costs calculated using their accrued values (value-added cost in each buffer zone is used to estimate holding costs).
- Buffers are not finite in the simulation model, which means workstations are never blocked. Optimal buffer capacities are calculated through profit maximization as described in Section 3.4.
- All machines are set to run up to their full capacity as long as they are up and not starved.

3.2 Workstation and Buffer Zone Parameters

In Figure 3.1 and Figure 3.2 parameters related to buffer zones and workstations are given. Not every workstation/buffer zone need to incorporate all of these parameters. Depending on the nature of the problem, some or all of them may become relevant.

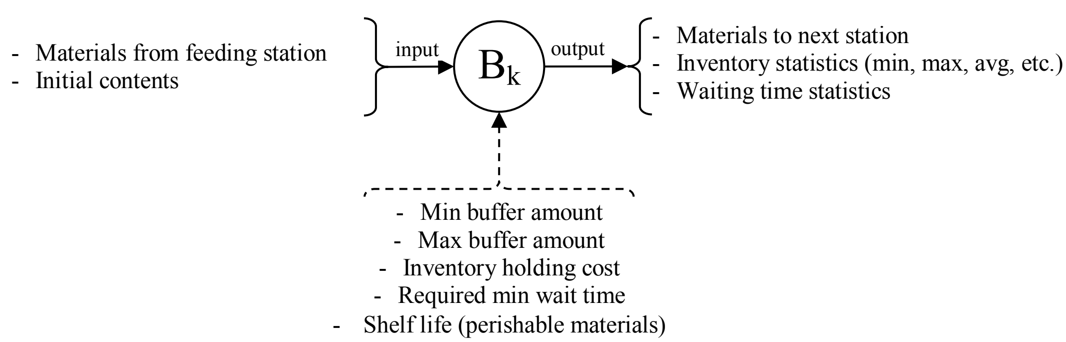


Figure 3.1 Buffer zone inputs, outputs and parameters.

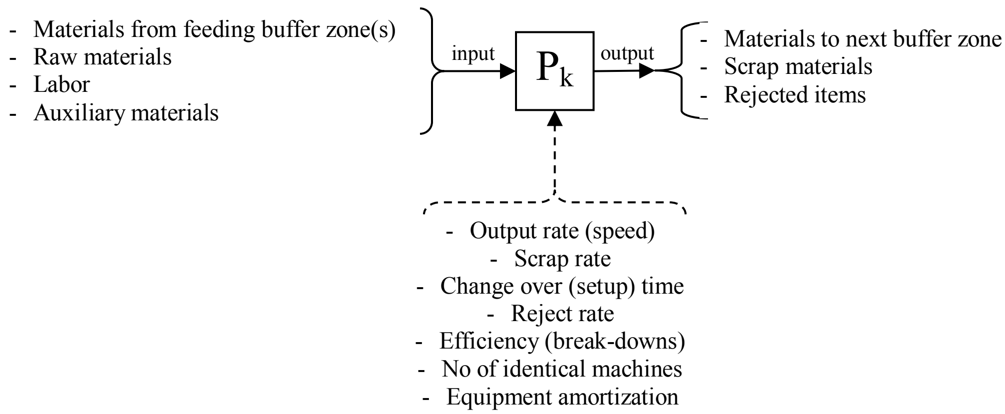


Figure 3.2 Workstation inputs, outputs and parameters.

3.3 Process Flow of the Real-life Case

Process flow of the prospective production line is given in Figure 3.3. The series-parallel flow is asynchronous and inhomogeneous. Production flow is saturated and produces a single product. Initially, two sub-lines process subparts independently, one of which breaking the subparts into three smaller subparts (Station 5). Then all subparts from both lines are assembled together at Station 13. The subparts need to cool down for a specified period of time at Buffer Zones 8, 9, 10, and 16 before they can be assembled.

Raw materials enter the system at Stations 1, 2, 7, 8, 13, 14, 15 and 24. Stations 2, 4, 5, 8, 10, and 11 produce significant amounts of scrap product, while Stations 14, 15, and 24 are the quality control points where rejections occur. Parts visit Station 17 three times (16→17→18→19→20→ 21→17→ 22→17→23, this flow is deterministic) to be reprocessed before finally leaving for Buffer Zone 26 (reentrant-flow). All machines are associated with stochastic downtimes and repair times. In addition, Stations 2, 3, 4, 5, 8, 9, 10, 11, and 13 involve probabilistic setup (change-over) times. Furthermore, Stations 14, 15, 16, 17, 20, and 24 require direct labor input and have probabilistic processing times. System works 6 days a week, 3 shifts a day, and 8 hours a shift with the exception of Stations 17 to 24 that run 7 days a week. Stations 17 to 24 share a single type of worker. Labor requirements for other stations are predetermined.

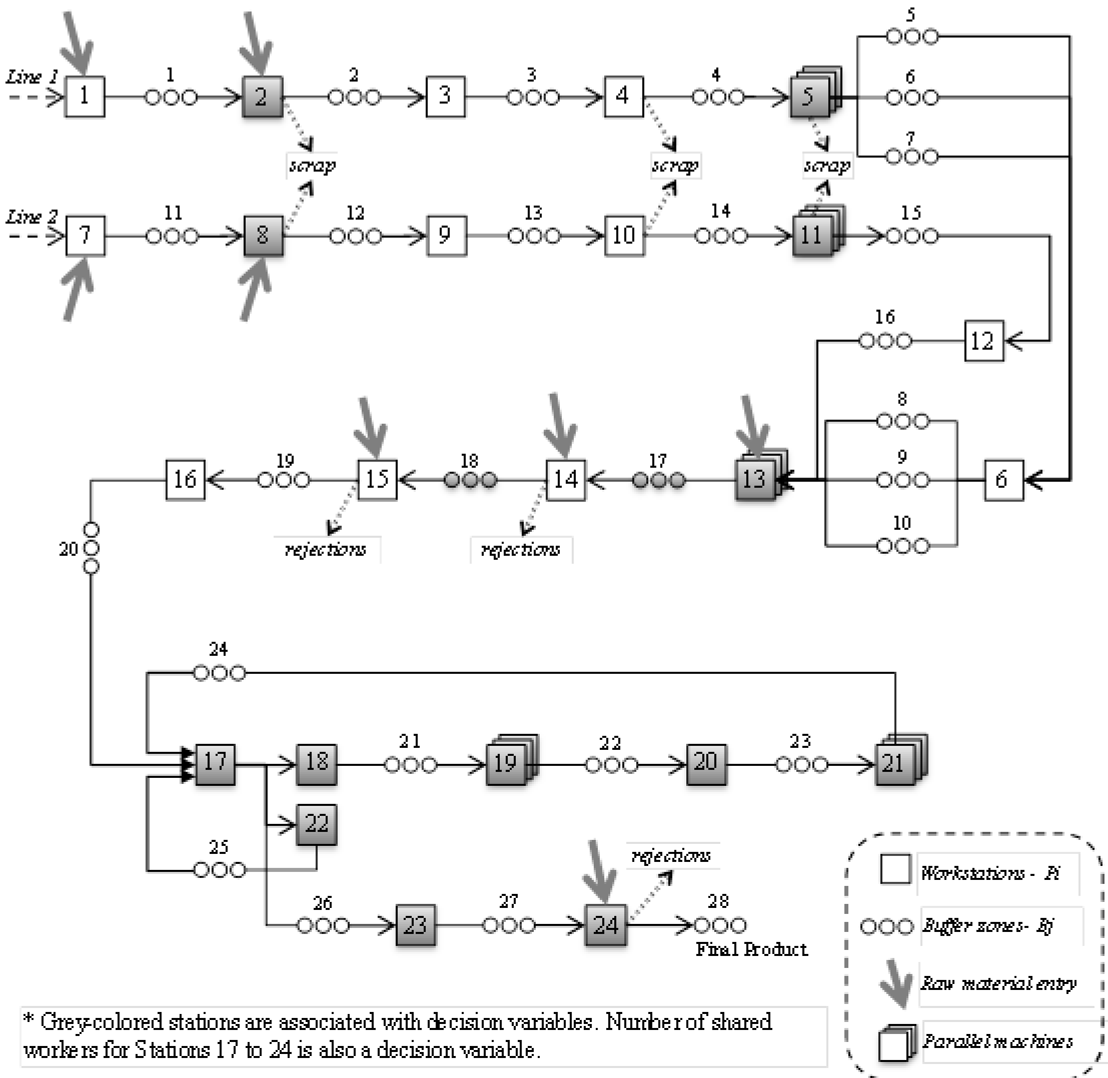


Figure 3.3 Process flow of ABC production line

3.4 Decision Variables of the Real-life Case

Nine decision variables leading to a total of 49,152 possible station configurations are described in Table 3.1. Figure 3.3 shows the stations and buffer zones associated with these decision variables. The minimum, maximum, and increment values of decision variables are determined based on the machine specifications provided by the producers, and through consultations with the ABC engineering team.

Table 3.1 Decision variables and range of feasible values.

Variable ID	Decision Variables	Min	Max	Number of Values	Increment
V_1	Speed of Machine at P_2 (parts/minute)	14.5	16	4	0.5
V_2	Speed of Machine at P_8 (parts/minute)	14.5	16	4	0.5
V_3	Number of Machines at P_5 & P_{11}	5	6	2	1
V_4	Number of Machines at P_{13}	17	20	4	1
V_5	Min. Buffer Amount at B_{17} (unit: days of production by P_{14})*	0.01	1.51	4	0.5
V_6	Min. Buffer Amount at B_{18} (unit: days of production by P_{15})	0.01	0.05	3	0.02
V_7	Number of Machines at P_{19}	23	26	4	1
V_8	Number of Machines at P_{21}	27	30	4	1
V_9	Number of Workers/shift at P_{17} to P_{24}	3	4	2	1
Number of Combinations				**49,152**	

* For example, daily production of P_{15} is 4128 parts, so 1.51 equals 6234 parts. Value 0.01 used as the smallest value, because the program generates an error when 0 is entered.

There are two variables associated with the buffer sizes, V_5 and V_6, which denote the minimum buffers for B_{17} and B_{18}. The minimum buffer amounts need to be considered for unbalanced lines where there exists a preceding station operating slightly slower on average than the latter one. For example, P_{13} consists of at least 17 identical machines with deterministic processing times and variable setup times. Assuming that all workstations are set to work at full capacity, the variability in the setup and processing times of P_{13} and P_{14} may create an imbalance in favor of P_{14} for some configurations. A minimum buffer amount is defined for B_{17} to diminish the effect of this imbalance on the overall profit. If ever the amount of parts in B_{17} becomes zero, P_{14} stops and waits until the amount reaches the minimum level. In this way, P_{14} works continuously without stopping repeatedly in short time periods. The bulk time during which P_{14} waits for B_{17} can be utilized for maintenance. The inclusion of minimum buffers creates a profit improvement of 1.6% in the optimal configuration.

ABC also needs to specify the space allocated for each buffer zone. In order to reduce the complexity of the problem, the method proposed by Battini et al. (2009) is used in determining the maximum buffer capacities, i.e., the buffer capacities are set to the maximum accumulation level of sub-products in the corresponding buffer zones during the simulation runs. This approach is reasonable because the objective of this study is profit maximization and inventory holding costs penalize high WIP inventory. Therefore, the optimal solution is supposed to have reasonable buffer capacities.

3.5 Objective Function

The objective function that inexplicitly defines the profit associated with a particular set of station configuration variables is given below. All decision variables are denoted by the vector $\Phi = [V_1, V_2, \dots, V_9]$. The objective function is composed of variables (such as annual sales, annual production, etc.) which are functions of Φ and calculated using the simulation outputs. Nomenclature can be visited for the notation used in the objective function.

Objective function, which is maximization of the annual profit, is given below:

$$\max_{\Phi} \; p\sum_{t=1}^{T/\delta} S_t(\Phi) \; - \sum_{j=1}^{m} h_j \bar{I}_j(\Phi) \; - \sum_{t=1}^{T/\delta} \pi L_t(\Phi) - C_{mtr}\sum_{t=1}^{T/\delta} Pr_t(\Phi) \; - \sum_{i=1}^{n} C_i^A M_i$$
$$- \sum_{i=1}^{n}\sum_{l\in\mathbb{R}i} C_l^w W_{li} - C_{other}$$

where

$p\sum_{t=1}^{T/\delta} S_t(\Phi)$: Expected sales income.

$\sum_{t=1}^{T/\delta} \pi L_t(\Phi)$: Expected lost sales cost.

$C_{mtr}\sum_{t=1}^{T/\delta} Pr_t(\Phi)$: Expected material costs.

$\sum_{i=1}^{n} C_i^A M_i$: Expected equipment costs.

$\sum_{i=1}^{n}\sum_{l\in\mathbb{R}_i} C_l^w W_{li}$: Expected labor costs.

$\sum_{j=1}^{m} h_j \bar{I}_j(\Phi)$: Expected inventory holding costs for buffer zones.

C_{other} refers to the fixed costs independent from Φ such as costs of overhead, sales/marketing, and other amortization. Equations (3.1) to (3.12) explain the details of the objective function.

It can be said that M_i, W_{li}, h_j, C_j^{lb}, C_j^{eq}, and C_j^{va} are also functions of (Φ), because number of workers and machines are decision variables for some workstations. However, I do not give this dependency in the notation for the sake of simplicity.

$$D_t = D * \delta/T, \tag{3.1}$$

$$I_{mt}(\Phi) = max\left[\left(I_{m(t-1)}(\Phi) + Pr_t(\Phi) - D_t\right), 0\right], \; I_{m0}(\Phi) = 0 \,, \tag{3.2}$$

$$S_t(\Phi) = min\left[\left(I_{m(t-1)}(\Phi) + Pr_t(\Phi)\right), D_t\right], \tag{3.3}$$

$$C_i^A = C_i^M \Big/ \left[\frac{1-\left(1+r^{int}\right)^{-1}}{r^{int}}\right] \text{ (derived from financial annuity formulation),} \tag{3.4}$$

$$C_j^{lb} = \sum_{l \in \mathbb{R}_{i^*(j)}} PT_{i^*(j)} C_l^w W_{li^*(j)} \,, \tag{3.5}$$

$$C_j^{eq} = PT_{i^*(j)} C_{i^*(j)}^A \,, \tag{3.6}$$

$$C_v^{va} = \sum_{j \in \mathbb{V}_v} C_j^{lb} + \sum_{j \in \mathbb{V}_v} C_j^{eq} + \sum_{j \in \mathbb{V}_v} C_j^{mt} \,, \;\; 1 \le v \le m, \tag{3.7}$$

$$h_j = r^{inv} C_j^{va}, \tag{3.8}$$

$$\pi = r^{lst} p \,, \tag{3.9}$$

$$L_t(\Phi) = max \left[\left(D_t - \left(I_{m(t-1)}(\Phi) + Pr_t(\Phi)\right)\right), 0\right], \tag{3.10}$$

$$\bar{I}_m(\Phi) = \left(\sum_{t=0}^{((T/\delta)-1)} \left(I_{mt}(\Phi) + \frac{Pr_{(t+1)}(\Phi)}{2}\right)\right) \Big/ (T/\delta), \tag{3.11}$$

$$\bar{I}_j(\Phi) = \sum_{\tau=1}^{T/\varepsilon} I_{j\tau}(\Phi) \Big/ (T/\varepsilon) \,, \; j = 1,2,\cdots,m-1 \,, \tag{3.12}$$

Equation (3.7) calculates the cost of a sub-product at any Buffer Zone v by adding up the labor, equipment, and material cost accrued so far. Equation (3.8) computes the cost of holding one unit of sub-product at Buffer Zone *j* for one year. Equation (3.2) calculates the amount of finished goods inventory at every *t*, which is then used in Equation (3.11) to calculate the annual average inventory. Unlike Equation (3.11), Equation (3.12) does not take the next term's production into account, because interval ε is small enough to ignore its effect.

Figure 3.4 illustrates possible movements in the finished goods inventory, $I_{mt}(\Phi)$, while Figure 3.5 shows those in WIP inventory of Buffer Zone *j*, $I_{j\tau}(\Phi)$. The simulation model updates the inventory levels discretely at each time period ε. In Figure 3.4, sharp declines at every δ represent the product shipments.

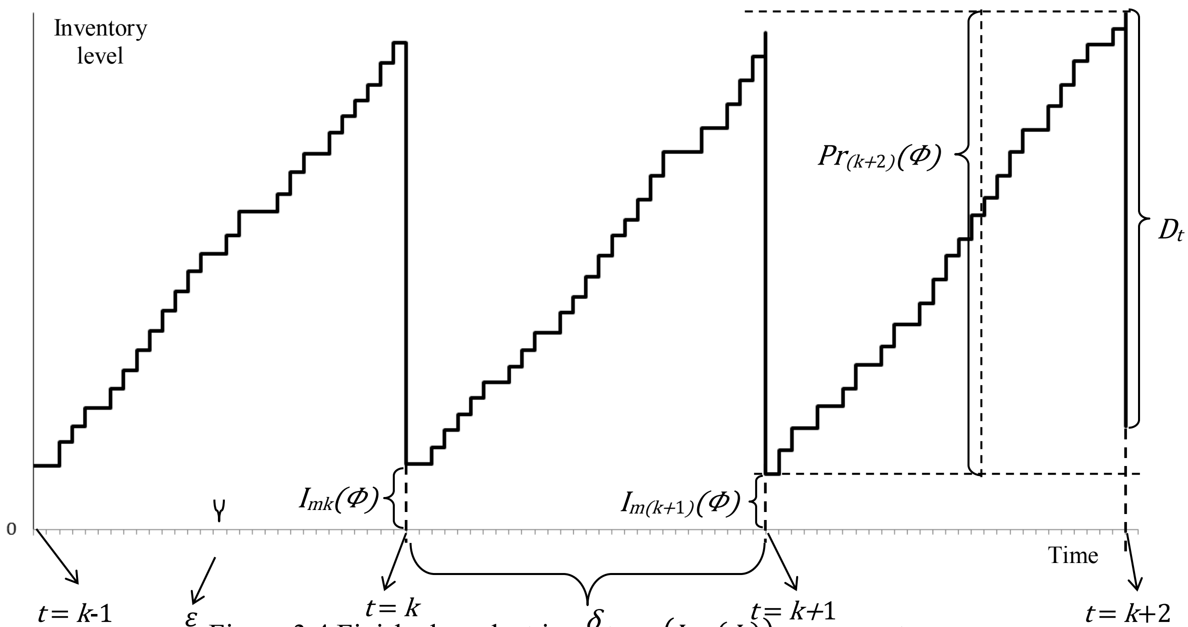


Figure 3.4 Finished product inventory $\left(I_{mt}(\Phi)\right)$ movements.

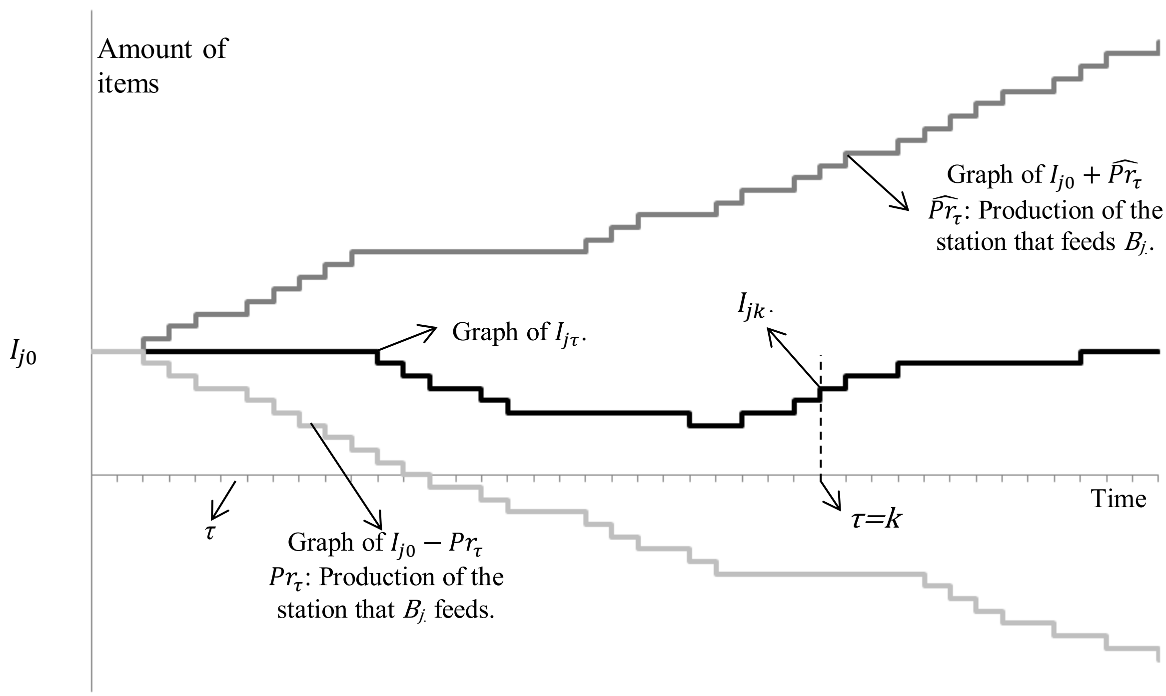


Figure 3.5 WIP inventory $\left(I_{j\tau}(\Phi)\right)$ movements.

Chapter 4
Simulation Model

Most complex, real-world systems with stochastic elements cannot be accurately described by a mathematical model that can be evaluated analytically. Thus, a simulation is often the only type of investigation possible (Law, 2007). Through simulation modeling of investments, decision-makers can make informed decisions and evaluate potential alternatives. Simulation is one of the most widely used operations research and management science techniques. One indication of this is the Winter Simulation Conference, which attracts 600 to 800 people every year. According to a study related to the use of simulation methods, simulation is consistently ranked as one of the three most important "operations research techniques" (Lane et al., 1993). One other study shows that simulation is second only to "mathematical programming" among 13 techniques considered (Gupta, 1997). In this study, simulation is used to model and analyze a complex production line. The simulation model also serves as the black-box type objective function.

The seven-step approach of Law (2003) as given in Figure 4.1 was used to build the simulation model in this research.

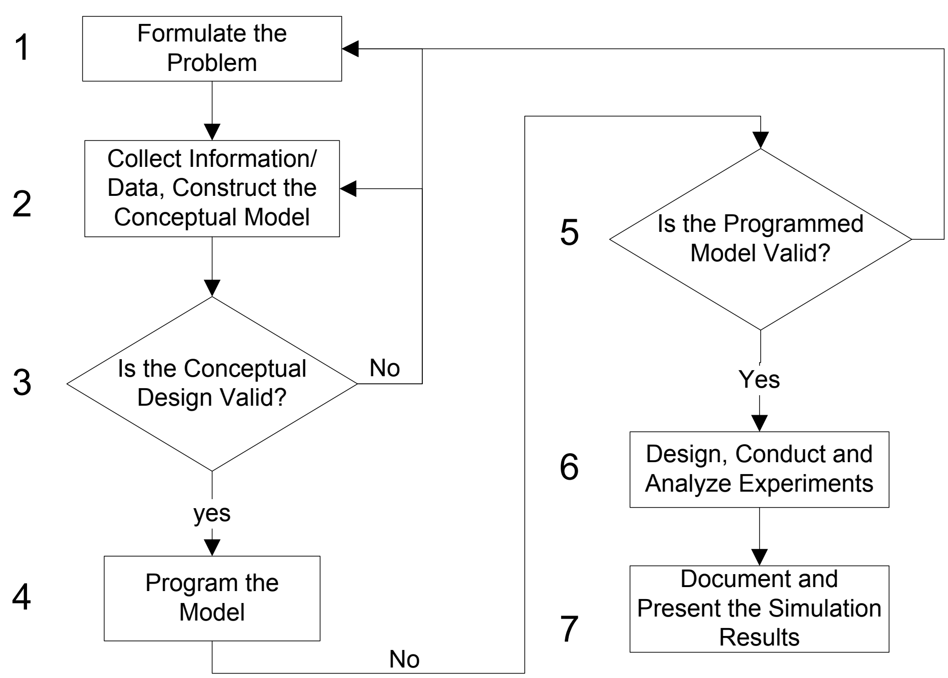


Figure 4.1 Seven-step approach for conducting a successful simulation study (Law, 2003).

4.1 Simulation Software Selection

An important resource to develop discrete-event simulation models is either a general purpose programming language or a simulation software package. In spite of the high flexibility they offer, building such a complex model using a general purpose language such as C or Pascal requires considerable amount of time and energy. Therefore, an off-the-shelf simulation software package has been used. Selection of the appropriate software is a crucial issue in such cases. Selection process for the most appropriate simulation software for ABC's case is described in this section.

There are more than 15 discrete-event simulation software packages in the market. Regarding selection of the software, there are methodologies proposed by several authors in related research literature. Furthermore, OR/MS Today magazine publishes a survey of simulation software every two years. The most recent survey at the time of the selection was published in October, 2009.

For ABC's case, the following references and inputs have been taken as the basis to develop a weighted criteria selection matrix (Table 4.2).

- Azadeh et al. (2010)
- OR/MS Today's Survey (Swain, 2009)
- Querying team members to identify ABC's own expectations (used for weighing)
- One-on-one meetings with software vendors (At the Winter Simulation Conference - Dec 2009, Austin TX).
- Web meetings with software vendors.
- Webinars and seminars of software vendors.
- Building sample models on demo versions of short-listed software.
- Second opinions from other users of the software (At the Winter Simulation Conference - Dec 2009, Austin TX)
- Software vendors in the final short-list invited to ABC's office for demonstration to all team members.

The following 5-step approach has been implemented for selection of the software:

1. Generate an extensive list of available software.

2. Create short list by elimination of some software based on some obvious reasons (Table 4.1).
3. Generate the weighted criteria selection matrix for the short-listed software (Table 4.2).
4. Invite representatives of top two software vendors for demonstration.
5. Select the software.

Short listed products are given in Table 4.1. Several other software products that were in the extensive list have been eliminated due to observable reasons such as being specialized in certain areas, being relatively new in market, being unnecessarily complex, or not having enough market presence.

Table 4.1 Short list of available simulation software products.

Company	**Product**
1.Rockwell Automation	Arena
2.Promodel Corporation	Promodel
3.Flexsim Software Prod.	Flexsim
4.Lanner Group	Witness
5.Visual8 Simulation Solutions	Simul8
6.Imagine That Inc.	ExtendSim

Weighted criteria selection matrix was developed for the remaining short-listed 6 products (step 3), which is given in Table 4.2. Cells highlighted in green show the criteria for which all the software received same ratings. It means these criteria do not have any actual effect on the overall score of the software products. The cell highlighted in red represents missing data (it is assumed to be 0.5).

As can be seen in

Figure 4.2, top two candidates are FlexSim and Simul8. Representatives of these two software vendors have been invited to ABC's office for a demonstration of their product. After these two demonstrations, ABC has decided to go with Simul8, which offered all the necessary requirements at the best benefit/cost ratio. Commercial licence for the professional edition of Simul8 was acquired by ABC, and another licence was granted by Simul8 to the author of this thesis for research purposes.

The comparison results should be considered case-specific rather than a general metric for comparing the performances of the companies. Another company can outperform others in other circumstances.

4.2 Simulation Model of the Real-life Case

A simulation model mimicking the processes in the prospective ABC production line was developed using SIMUL8 software package. Simulation model has been built in a way that most of the input and output data exchange is being done from and to a MS Excel workbook. A sample screenshot is given in Figure 4.3. A screenshot from the simulation model on Simul8 is given in Figure 4.4.

Table 4.2 Weighted criteria selection matrix for the short-listed six simulation software products.

		Criteria	Weight	Arena		ProModel		FlexSim		Witness		Simul8		ExtendSim	
				Rt.	W. Rt.	Rt.	W. Rt.	Rt.	W. Rt.	Rt.	W. Rt.	Rt.	W. Rt.	Rt.	W. Rt.
User / support	1	User Discussion Groups	0.010	1.00	0.01	1.00	0.01	1.00	0.01	1.00	0.01	1.00	0.01	1.00	0.01
	2	Ease of Use	0.132	0.60	0.08	0.65	0.09	0.70	0.09	0.50	0.07	0.70	0.09	0.50	0.07
	3	Training	0.020	0.60	0.01	0.50	0.01	1.00	0.02	0.50	0.01	0.80	0.02	0.40	0.01
	4	Documentation	0.022	1.00	0.02	1.00	0.02	1.00	0.02	1.00	0.02	1.00	0.02	1.00	0.02
	5	Support	0.080	0.50	0.04	0.50	0.04	1.00	0.08	1.00	0.08	1.00	0.08	0.50	0.04
Data analysis	7	Built-in Input Analyzer	0.054	0.00	0.00	1.00	0.05	1.00	0.05	0.00	0.00	1.00	0.05	1.00	0.05
	8	Built-in Output Analyzer	0.081	1.00	0.08	1.00	0.08	1.00	0.08	1.00	0.08	1.00	0.08	1.00	0.08
	9	Built-in Optimizer	0.040	0.00	0.00	1.00	0.04	1.00	0.04	0.00	0.00	1.00	0.04	1.00	0.04
	10	Warm-up Period	0.060	1.00	0.06	1.00	0.06	1.00	0.06	1.00	0.06	1.00	0.06	1.00	0.06
	11	Scenario Manager	0.050	1.00	0.05	1.00	0.05	1.00	0.05	1.00	0.05	1.00	0.05	1.00	0.05
	12	Multiple Runs	0.049	1.00	0.05	1.00	0.05	1.00	0.05	1.00	0.05	1.00	0.05	1.00	0.05
Animation	13	Icons	0.020	0.65	0.01	0.70	0.01	0.80	0.02	0.70	0.01	0.50	0.01	0.50	0.01
	14	2D Animation	0.030	0.50	0.02	0.65	0.02	0.75	0.02	0.65	0.02	0.55	0.02	0.55	0.02
	15	3D Animation	0.030	0.60	0.02	0.60	0.02	0.75	0.02	0.65	0.02	0.00	0.00	0.65	0.02
	16	Integration of Animation	0.026	0.40	0.01	0.40	0.01	0.90	0.02	0.70	0.02	0.00	0.00	0.70	0.02
Other	17	Cost	0.114	0.43	0.05	0.43	0.05	0.83	0.09	0.30	0.03	0.93	0.11	0.93	0.11
	18	Talk to other applications	0.071	1.00	0.07	1.00	0.07	0.75	0.05	1.00	0.07	1.00	0.07	1.00	0.07
	19	Pedigree	0.044	0.88	0.04	0.91	0.04	0.26	0.01	1.00	0.04	0.61	0.03	0.91	0.04
	20	Breakpoints	0.018	1.00	0.02	0.00	0.00	0.00	0.00	1.00	0.02	1.00	0.02	0.00	0.00
	21	Runtime Application	0.030	1.00	0.03	0.80	0.02	0.80	0.02	0.50	0.02	1.00	0.03	0.60	0.02
	22	Future Use	0.020	0.40	0.01	0.20	0.00	0.15	0.00	0.15	0.00	0.15	0.00	0.15	0.00
		Sum	1.000	0.673		0.750		0.828		0.683		0.834		0.781	

Rt.: Rating, W.Rt.: Weighted Rating

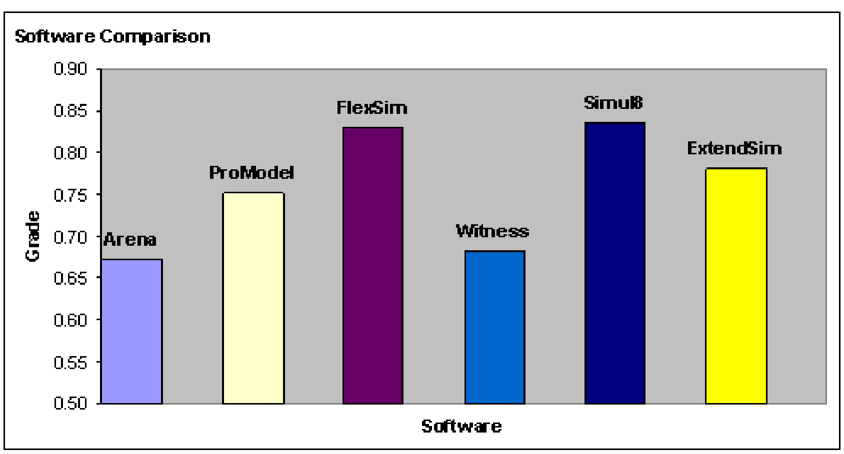


Figure 4.2 Final scores of the short-listed simulation software products.

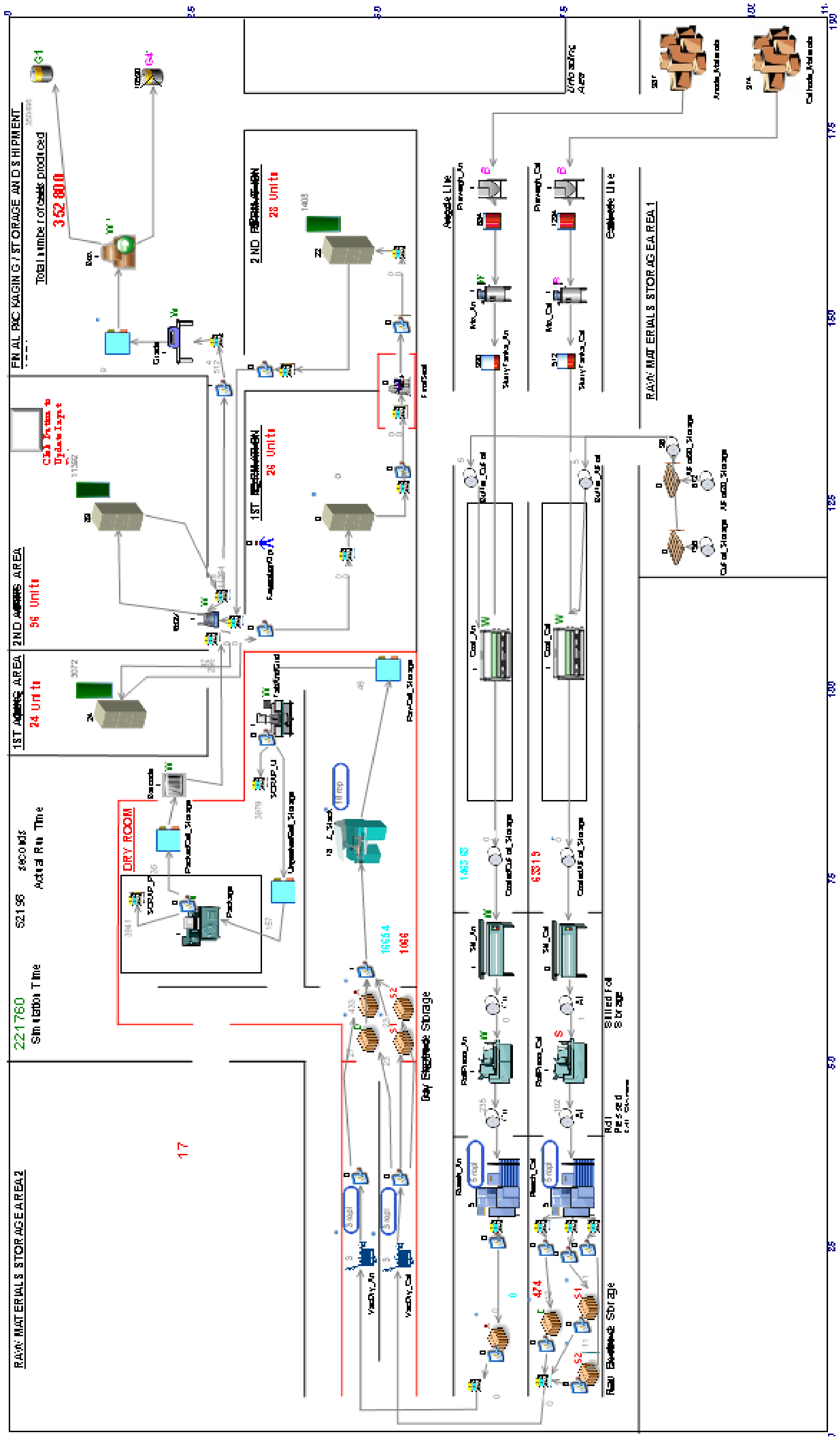


Figure 4.3 Screenshot from the Simul8 model of the real-life problem

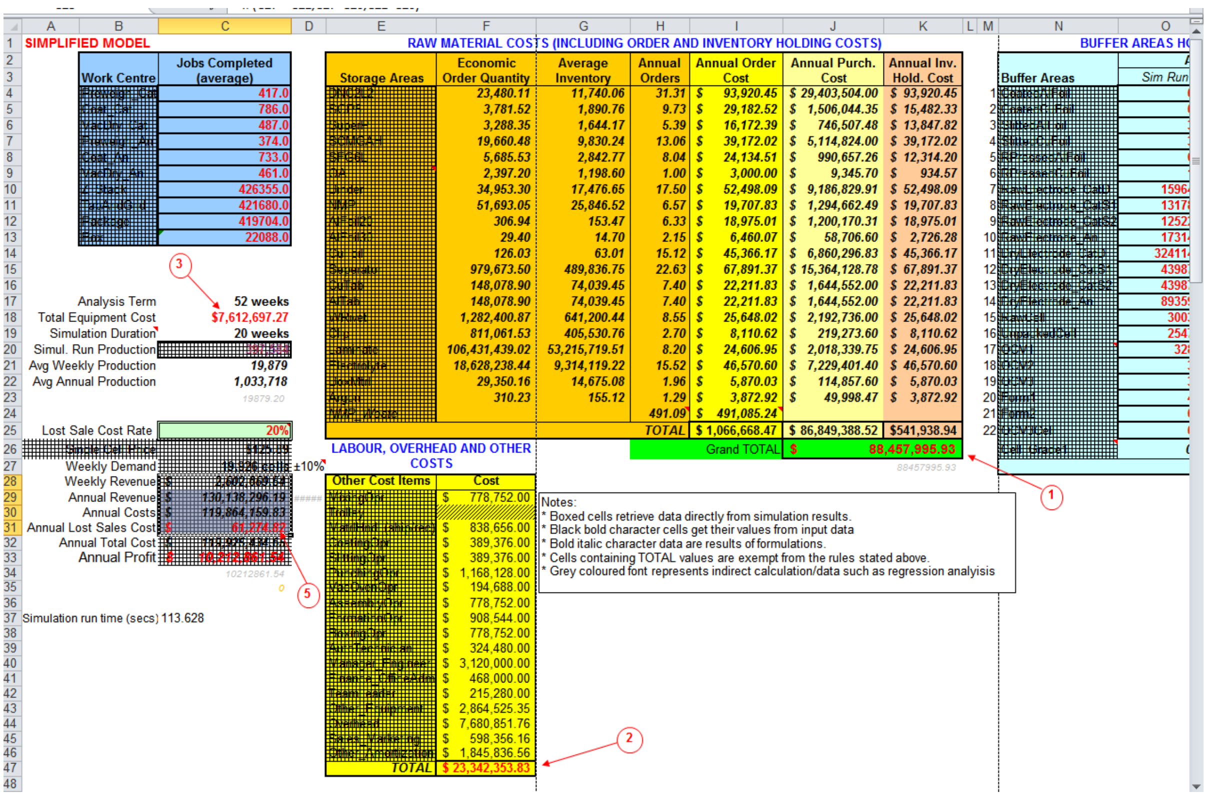


Figure 4.4 Partial screenshot from cost calculation worksheet (confidential data shaded).

It is statistically enough to run the model for 20 weeks so as to estimate approximate results for annual performance. Hypothesis tests for two randomly selected configurations to compare the mean of the first 20 weeks' production to annual (52 weeks') production are given in Table 4.3. Production values in the table have been divided to an undeclared real number to ensure confidentiality.

4.2.1 Input Data

Production system simulation models in general require the following data:

a) Sufficient statistical data about the machines including but not limited to output rate, maintenance requirements, breakdown statistics and reject ratios.

b) Annual/Monthly/Weekly/Daily production plans/targets.

c) Draft physical layout of the plant.

d) Bill of materials.

e) Detailed cost analysis for profit calculation.

f) Breakdown of all tasks, task completion times, precedence relations and other task requirements (e.g. *task b* must be performed by *resource x*).

g) Planned shifts regime for the factory.

The expected annual demand, detailed cost values, interest rate, and product prices were provided by the ABC engineering team. The data about the processing times, setup times, and efficiencies of the machinery are calculated using the design specifications provided by the suppliers. The production rates are assumed to be deterministic for the completely automated machines. As suggested by Law (2007), for the cases where historical data is absent, triangular distribution is used for the processing times of the non- or semi-automated tasks. Some of the automated machines require labor based setup time, whose length is also assumed to be triangularly distributed. Based on the design specifications, machines work with 85% efficiency. A built-in function of SIMUL8 handles the efficiency calculation for which the time periods between breakdowns and repair times are distributed with negative exponential and Erlang distributions, respectively.

The production rate becomes stable after about 12 days, thus, a warm-up period of two weeks is used. Each simulation run takes 56.4 seconds (on a PC with Intel® Core™ i7-3770 3.4GHz CPU) on average.

4.2.2 Number of Replications

The simulation optimization approaches described in this paper take three replications of the simulation model to evaluate each configuration. Tests conducted on 100 random configurations show that for a big majority of the cases there is not enough evidence to reject the hypothesis that mean profit from 3 replications is same as that of 30 replications. Figure 4.5 shows the chart of test results for the mentioned 100 random configurations. Blue line represents an undeclared fraction of the expected annual profit based on 30 replications, in descending order. Green line is the results of the hypothesis tests. Parts of green line with value 1 show for which configurations H_0 is accepted; where H_0: there is not enough evidence to prove that the mean from 3 replications is different than that from 30 replications. H_0 is accepted for 80 of the 100 random configurations. H_0 is only rejected for cases where expected profit is at a certain level. This is due to the increased marginal effect of lost sale costs at this level, which causes more variability on the profit even for small changes in the production level.

Table 4.3 Hypothesis tests to compare 20 weeks to annual production

Configuration 1: 6 18 24 28 16.0 15.0 1.01 0.01 4

Week	Weekly Prod.	Week	Weekly Prod.
1	1530	27	1578
2	1523	28	1565
3	1601	29	1551
4	1530	30	1556
5	1590	31	1562
6	1534	32	1541
7	1559	33	1541
8	1549	34	1610
9	1547	35	1487
10	1541	36	1590
11	1598	37	1570
12	1580	38	1560
13	1563	39	1590
14	1526	40	1565
15	1458	41	1484
16	1739	42	1567
17	1421	43	1610
18	1650	44	1556
19	1524	45	1542
20	1588	46	1565
21	1692	47	1559
22	1538	48	1513
23	1588	49	1594
24	1518	50	1529
25	1591	51	1590
26	1519	52	1533

Hypothesis

H_0: There is no significant difference between the sample mean (n) and the population mean (N).

H_1: There is a significant difference between the sample mean (n) and the population mean (N).

$$t = \frac{\bar{X} - \mu}{S_{\bar{X}}} \qquad S_{\bar{X}} = \frac{S}{\sqrt{n-1}}$$

N = 52

μ = 1,559.72

α = 0.1 Level of significance

compare to $t_{n-1,\alpha/2}$

Test for n = 20 Sample includes the first 20 weeks.

$\bar{X}$ = 1,557.55

S = 65.71

$t_{n-1,\alpha/2}$ = 1.7291328

$S_{\bar{X}}$ = 15.0738

t = **-0.143435** **H_0 is not rejected**

4.2.3 Number of Replications

The simulation optimization approaches described in this paper take three replications of the simulation model to evaluate each configuration. Tests conducted on 100 random configurations show that for a big majority of the cases there is not enough evidence to reject the hypothesis that mean profit from 3 replications is same as that of 30 replications. Figure 4.5 shows the chart of test results for the mentioned 100 random configurations. Blue line represents an undeclared fraction of the expected annual profit based on 30 replications, in descending order. Green line is the results of the hypothesis tests. Parts of green line with value 1 show for which configurations H_0 is accepted; where H_0: there is not enough evidence to prove that the mean from 3 replications is different than that from 30 replications. H_0 is accepted for 80 of the 100 random configurations. H_0 is only rejected for cases where expected profit is at a certain level. This is due to the increased marginal effect of lost sale costs at this level, which causes more variability on the profit even for small changes in the production level.

Table 4.3 Hypothesis tests to compare 20 weeks to annual production - continued

Configuration 2: 5 18 25 28 15.0 15.0 0.51 0.01 4

Week	Weekly Prod.	Week	Weekly Prod.
1	1405	27	1401
2	1357	28	1320
3	1477	29	1290
4	1303	30	1234
5	1241	31	1314
6	1430	32	1375
7	1259	33	1540
8	1451	34	1243
9	1389	35	1187
10	1491	36	1408
11	1324	37	1376
12	1232	38	1439
13	1461	39	1117
14	1440	40	1153
15	1375	41	1310
16	1333	42	1410
17	1282	43	1320
18	1390	44	1354
19	1411	45	1342
20	1378	46	1250
21	1475	47	1300
22	1333	48	1466
23	1493	49	1390
24	1310	50	1326
25	1313	51	1225
26	1383	52	1380

Hypothesis

H_0: There is no significant difference between the sample mean (n) and the population mean (N).

H_1: There is a significant difference between the sample mean (n) and the population mean (N).

N= 52

μ = 1,350.13

α = 0.1 → Level of significance

$$S_{\bar{X}} = \frac{S}{\sqrt{n-1}} \qquad t = \frac{\bar{X} - \mu}{S_{\bar{X}}}$$

compare to $t_{n-1,\alpha/2}$

Test for n = 20 Sample includes the first 20 weeks.

$\bar{X}$ = 1,371.46

S = 78.57

$t_{n-1,\alpha/2}$ = 1.7291328

$S_{\bar{X}}$ = 18.0244

t = **1.1833025** **H_0 is not rejected**

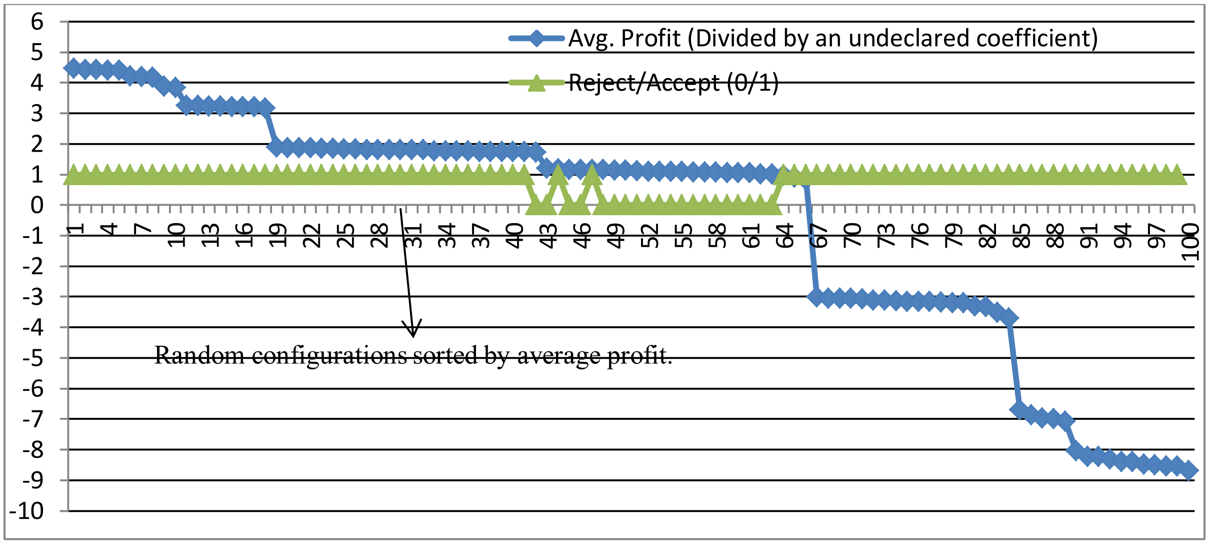


Figure 4.5 Statistical tests to analyze the number of replications (real-life problem)

4.2.4 Validation of the Simulation Model

A simulation model can only be trusted if it is credible for the decision makers. Although validation is not always enough to make the model credible, it sure is necessary. Decision makers need to know whether the results of the model are comparable to the actual system, before they can use the simulation model for critical decision making. The easiest technique to validate a simulation model is to compare its output with the output of the actual system. This is only possible when the simulation is a model of an existing system.

As discussed in Sargent (2010), there are a number of approaches and techniques used for validation and verification of the simulation models. Most widely used three approaches are; (1) validation by simulation engineer/team, (2) validation by involving simulation model users, (3) validation by a professional third party. Some techniques that are applicable to modeling of non-existing systems are Animation, Face Validity, Internal Validity, and Nominal Comparison.

The model in this study was validated based on the second approach by employing the techniques mentioned above. The following steps were carried out for validation of the model:

1. From the very start of the project, regular meetings were held with participation of the engineering team and the simulation engineer (author of this thesis). Meetings served two purposes; (1) at the early stages of the project, validation of the conceptual design prepared by the simulation engineer, and (2) validation of the simulation model at the later stages.

Simulation engineer had the chance to share his findings with the team and receive feedback from the team steadily throughout the project. Remaining steps below explains the techniques used for validation of the simulation model.

2. *Animation*: Graphical interface of Simul8® was a good visual tool for seeing on computer screen some of the entity movements, breakdowns, buffer amounts through time as the simulation runs. This gave the team to pinpoint and correct some obvious issues.
3. *Face Validity*: The behavior (input-output relations) of the model was discussed at the meetings. Engineering team members, who had seen a similar system before at the site of the technology provider, were able to make judgments about the performance of the model.
4. *Internal Validity*: Several replications were run to detect whether abnormal variability existed for replication results.
5. *Nominal Comparison*: The engineering team had already come up with predictions about the output of the system using nominal calculations without considering variability (probabilistic elements). Outputs of the simulation model were compared with these predictions. As the simulation model included variability and more details than the nominal calculations did, significant differences were noted. Reasons of these differences were traced carefully within the simulation and rational explanations were provided satisfying all members of the engineering team. One example to this is the number of machines at a certain workstation. Engineering team estimated that four machines would be enough to achieve the desired output rate; however simulation results indicated that at least five or maybe six machines might be needed. A thorough analysis revealed that the engineering team dismissed in their calculations the setup times that is necessary before loading each new item to the machines.

4.3 Test-bed Problem

4.3.1 Motivation

Because running the simulation model for the real-life case is time-consuming, a smaller size fictitious test-bed problem was generated and modeled. Test-bed problem is used to 1) determine specifications of the proposed simulation optimization methods, 2) compare the performances of these methods with the best solution obtained by total enumeration. I refer to this best solution for the test-bed problem as the optimal solution for evaluating all meaningful configuration variable combinations.

4.3.2 Process Flow and Input Data

The test-bed problem reflects the features of the real-life case on a smaller scale, e.g. being asynchronous and inhomogeneous, and having assembly stations, re-entrant flow, stochastic times, and machines subject to probabilistic failure times. The test-bed problem has 8 decision variables with 46080 possible configurations. Each simulation replication takes 8.5 seconds, much smaller compared to the real-life case (56.4 seconds). The process flow of the test-bed problem is given in Figure 4.6. A screenshot from the Simul8 model can be seen in Figure 4.7.

The input data for the test-bed problem is given in Table 4.5 and a copy of the corresponding SIMUL8 model can be obtained from the author of this thesis.

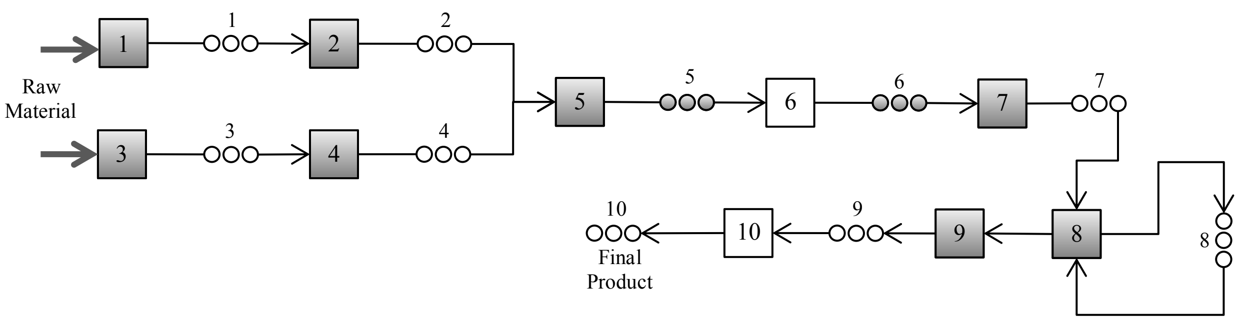


Figure 4.6 Process flow of test-bed problem

Decision variables of the test-bed problem can be seen in Table 4.4 along with their feasible ranges.

In Table 4.5, framed shaded boxes represent decision variables. B_8 is a special storage with limited capacity. Adding one unit of capacity has a cost and its capacity is a decision variable. Each part has to wait there for a specific period of time. Number of machines for P_2 and P_4 are identical, and thus constitute a single decision variable for both stations.

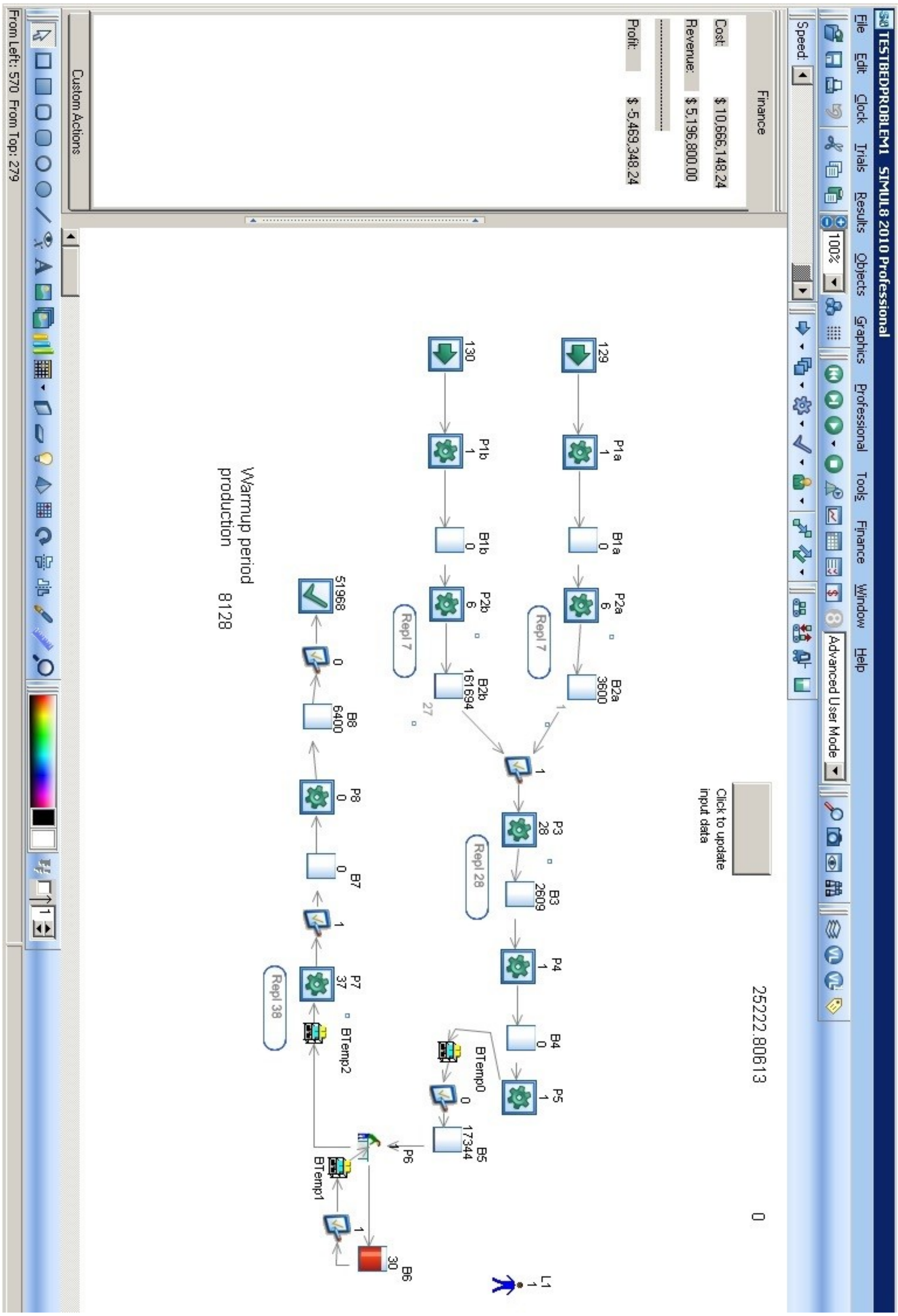


Figure 4.7 Screenshot from test-bed problem Simul8 model

In Table 4.5, Parts In/Out explains whether the processes break down parts into sub-parts or assemble sub-parts. For example B_2 routes in 6250 sub-parts each batch, which means P_2 breaks down a part into 6250 sub-parts. B_2 then routes out 11 sub-parts in each batch to P_3 for assembling. There are similar processes in the real-life case, details of which are not explained in the paper due to the confidentiality agreement signed with ABC.

P_7, P_8 and P_9 shares same type of workforce, cost of which is given in Table 4.5. All other labor costs are predetermined and embedded into overhead cost. To simplify the model, scrap and rejections are not considered in test-bed problem. Also equipment amortization costs, raw material costs, and inventory holding costs ($/min) are assumed to be pre-calculated constants. End products are shipped to buyers weekly.

Table 4.4 Decision variables and range of feasible values (test-bed problem).

Variable ID	Decision Variables	Min	Max	Number of Values	Increment
V_1	Speed of Machine at P_1 (meters/minute)	23	26	4	1
V_2	Speed of Machine at P_3 (meters/minute)	23	26	4	1
V_3	Number of Machines at P_2 & P_4	4	7	4	1
V_4	Number of Machines at P_5	25	28	4	1
V_5	Min. Buffer Amount at B_5 (days of production by P_6)	0.01	1.01	3	0.5
V_6	Number of Workers at P_7, P_8 & P_9	2	4	3	1
V_7	Capacity of B_8 (cartridges, 64 parts each)	28	32	5	1
V_8	Number of Machines at P_9	35	38	4	1
Number of Combinations				**45,080**	

4.3.3 Number of Replications

Similar to real-life case, test-bed problem results were also analyzed over three replications. Tests conducted on 100 random configurations show that almost for all of the cases there is not enough evidence to reject the hypothesis that mean profit from 3 replications is no different than that of 30 replications.

Figure 4.8 shows the chart of test results for the mentioned 100 random configurations. H_0 is accepted for 99 of the 100 random configurations.

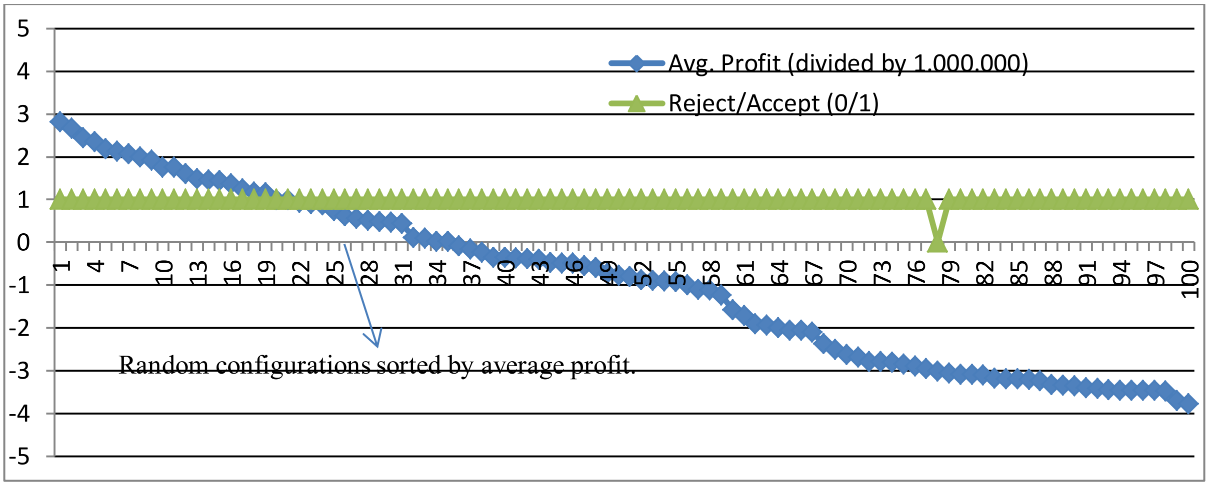


Figure 4.8 Statistical tests to analyze the number of replications (test-bed problem)

Table 4.5 Test-bed problem input data

Warm-up Period = 2 weeks (5760 minutes)
Simulation Duration = 25 weeks (72000 mins)
Demand = 225,000 pieces (25 weeks)
Sale Price = $100.00/piece

Cost of Raw Material = $12,500.00
Overhead Cost = $3,846,154.00 (for 25 weeks)
Lost Sale Cost Rate = 20% (over sale price)
Inventory Carrying Cost Rate = 20%
Efficiency Rate for all machines = 90%

Factory Working Schedule:
6 days/week, 312 days/year, 1 shifts/day, 8 hours/shift

PROCESS DATA (P_i)

i	NM	Cost of equipment (25 weeks)	Processing Times (minutes)				Setup Times (minutes)				
			PD	Min	Mode	Max	PD	Min	Mode	Max	Trigger
1	1	$168,269.23	Det.		130.43		Trng.	19	20	21	1
2	6	$ 28,846.15	Det.		781.25		Trng.	7.6	8	8.4	1
3	1	$168,269.23	Det.		130.43		Trng.	19	20	21	1
4	6	$ 28,846.15	Det.		757.75		Trng.	7.6	8	8.4	1
5	25	$ 16,826.92	Det.		6		Trng.	20.79	21	21.21	50
6	1	$141,826.92	Det.		15.36						
7	1	$247,596.15	Det.		15.36						
8	1	$31,250.00	Trng.	6.08	6.4	6.72					
9	38	$ 8,894.23	Trng.	712.8	720	727.2					
10	1	$15,625.00	Trng.	14.59	15.36	16.13					

BUFFER DATA (B_i)

j	Capacity	Min Wait (mins)	Parts		Capital Cost	Inventory Cost ($/min)	Minimum Buffer		
			In	Out			Amount	Days	Buffer/day
1			1	1		0.001			
2			6250	11		0.00000024			
3			1	1		0.001			
4			6062	9		0.000000247			
5			1	1		0.0000007	19	0.01	1869
6			1	1		0.0000012			
7			1	64		0.0000017			
8	32	630	64	64	$31,250.00	0.0006017			
9			64	64		0.0012017			
10			360	1		0.0018017			

Labor Cost	Quantity per shift	Hourly Wage	Total Cost (25 weeks)
	3	$25.00	$30,000.00/worker

NM: Number of Machines, **PD:** Probability Distribution, **Det.:** Deterministic, **Trng.:** Triangular,

Trigger: Setup time occurs after every trigger amount of parts were processed.

Chapter 5
Proposed Simulation Optimization Approaches

I propose a number of methods to approximately optimize the station configuration variables for the transferred production line. The specifications of the methods are developed and tested on the smaller size test-bed problem and then implemented on the real-life case.

5.1 OptQuest

OptQuest is a commercial simulation optimization tool readily available in simulation software packages such as SIMUL8, Arena, and FlexSim. OptQuest is used as a solution method to determine whether an easy-to-implement commercial simulation optimization tool can satisfy the requirements of the decision makers in terms of solution quality and speed. OptQuest is a meta-heuristic that moves from solution to solution following a combination of TS, ANN, and Scatter Search logics till the desired number of iterations is reached (Kleijnen & Wan, 2007). Most configurations of OptQuest are pre-specified by default except the initial solution and number of iterations which determine the solution quality.

I let OptQuest to pitch its default initial solution, i.e., the mid-point for each decision variables. To be fair in my comparison, I set the number of iterations for OptQuest as 500 based on experiments conducted on the test-bed problem, which show that OptQuest's performance becomes stable after 400 iterations.

5.2 Simulated Annealing (SA)

I develop an SA approach for benchmarking because it is a commonly used method for simulation optimization (Alrefaei & Diabat, 2009; Ghiani et al., 2007; Haddock & Mittenthal, 1992; Prudius & Andradottir, 2012; Rosen & Harmonosky, 2005). The proposed SA algorithm is as follows:

BEGIN

BestSolution = *InitialSolution*

$Temp$ = *InitialTemparature*

UNTIL (No better solution can be found) DO

Search immediate neighborhood of *BestSolution* for *NewSolution(s)*

Apply acceptance test $P_i = exp\left(\frac{-\Delta P}{Temp}\right)$

IF *NewSolution* is accepted THEN *BestSolution* = *NewSolution*

IF Temparature decrease criteria met THEN $Temp$= $(1 - d) \cdot Temp$

END UNTIL

END

where P_i is the probability of accepting solution i. ΔP is the difference between the profit values of the *NewSolution* and that of the current *BestSolution*. $Temp$ stands for the current temperature and d is the rate of reduction in the temperature. The best among 10 random solutions is taken as the initial solution. Tests with more random solutions failed to produce a significant change in the solution quality in most cases. The initial temperature is set as 20000 with d = 0.10, and decreased every 10 iterations. The immediate neighborhood of a solution is formed using two adjacent values of each variable; an example is given in Table 5.1. The algorithm stops when a neighborhood fails to produce an accepted solution.

Table 5.1 Sample neighborhood

V01	V02	V03	V04	V05	V06	V07	V08	V09	Neighborhood of
16	*15*	*6*	*18*	*0.51*	*0.03*	*4*	*25*	*28*	*Base Solution*
15.5	15	6	18	0.51	0.03	4	25	28	V01
15	15	6	18	0.51	0.03	4	25	28	V01
16	15.5	6	18	0.51	0.03	4	25	28	V02
16	14.5	6	18	0.51	0.03	4	25	28	V02
16	15	5	18	0.51	0.03	4	25	28	V03
16	15	6	19	0.51	0.03	4	25	28	V04
16	15	6	17	0.51	0.03	4	25	28	V04
16	15	6	18	1.01	0.03	4	25	28	V05
16	15	6	18	0.01	0.03	4	25	28	V05
16	15	6	18	0.51	0.05	4	25	28	V06
16	15	6	18	0.51	0.01	4	25	28	V06
16	15	6	18	0.51	0.03	3	25	28	V07
16	15	6	18	0.51	0.03	4	26	28	V08
16	15	6	18	0.51	0.03	4	24	28	V08
16	15	6	18	0.51	0.03	4	25	29	V09
16	15	6	18	0.51	0.03	4	25	27	V09

5.3 Ant Colony Optimization (ACO)

Although ant colony optimization (ACO) has been applied to several types of optimization problems (Mohan & Baskaran, 2012), its use for simulation optimization is limited. ACO logic considers combinatorial discrete optimization problems as networks. It mimics the attitude of agents of an artificial ant colony who try to reach a target node (T) from a source node (S) following multiple randomly-selected paths. Each path represents a solution to the corresponding problem. Each agent leaves a pheromone trail behind, and trails on paths that are associated with better solutions are updated more often which eventually leads to convergence to an approximately optimal solution.

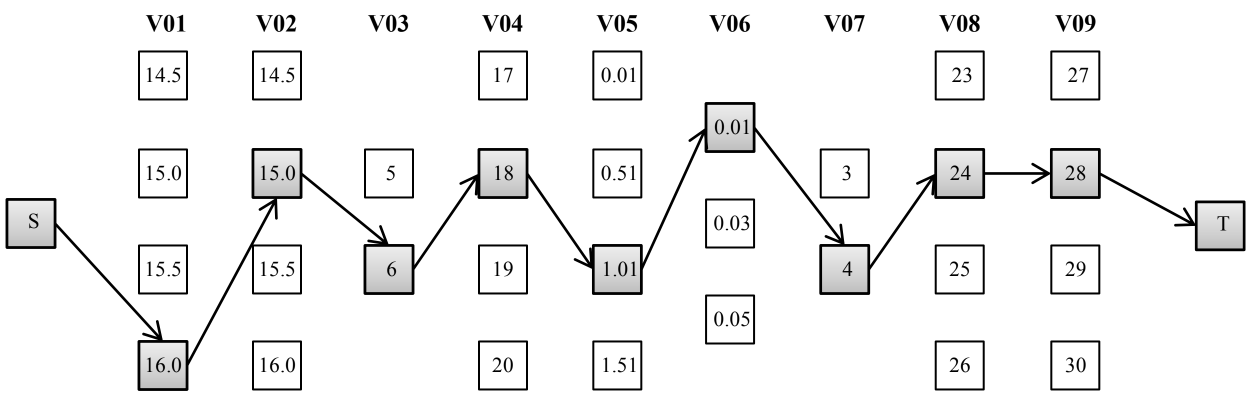


Figure 5.1 Possible values of decision variables structured as a network.

Because ACO requires a network structure, I formulate the problem as a network using the possible values for each decision variable as shown in Figure 5.1. Each column represents a decision variable, and is listed in the order they appear in the production line. The path made of the shaded nodes represents a feasible solution. For each path/solution, the simulation model is run and the corresponding profit value is reported as a measure of solution quality. The following pseudo-algorithm explains the steps of the proposed ACO algorithm.

BEGIN

Define initial *CurrentBest* = $e \cdot D \cdot p$

UNTIL (convergence criteria met or max # of iterations reached) DO

Construct Solution with Agent *k with*

$$P_{ij}^{k} = \begin{cases} \frac{\tau_{ij}^{\alpha}}{\sum_{l \in N_i^k} \tau_{ij}^{\alpha}}, & if\ j \in N_i^k \\ 0, & if j \notin\ N_i^k \end{cases}$$

Update Pheromone Trail

Evaporate pheromone from all arcs

$\tau_{ij} \leftarrow (1 - \rho)\tau_{ij}\ \forall (i, j) \in A$

IF Solution found by Agent k is better than *CurrentBest* then

update pheromone levels on the arcs of the solution path

$\tau_{ij} \leftarrow (1 + \gamma)\tau_{ij}\ \forall (i, j) \in\ S^k$

CurrentBest = Solution found by Agent *k*

END IF

NEXT *k*

END UNTIL

END

In this algorithm, CurrentBest represents the maximum profit value found so far whose initial value is a fraction (e) of the expected annual income ($p.D$). α is a constant. Convergence speed increases with bigger values of α.

The value of e is taken as 0.25, which facilitates faster convergence as shown by several preliminary tests. A is the set of nodes and $N_i^k$is the neighborhood of agent k when in node i. τ_{ij} is the amount of pheromone on the arc between nodes i and j. In each iteration, an agent k follows a path from the source to the target randomly using the probabilityP_{ij}^k, which is the probability of agent k at node i choosing to move to node j. An agent is more likely to choose a path that holds more pheromone than the others. After each iteration, a certain amount of pheromone evaporates with rate ρ while γ is the rate of increase in the pheromone level if a good solution is found. S^k is the set of nodes in the solution path found by agent k. Preliminary tests show that the combination of α =3, e =0.25, $\rho = 0.10$, and $\gamma = 0.35$ enables a quick convergence to a good solution in most cases. The ACO algorithm starts with a random solution where the initial pheromone amount (τ_{ij}) is the same (i.e. 1) for each trail. The max number of iterations is defined as 300, by which the algorithm converges to a good solution for majority of the cases.

5.4 Response Surface Methodology (RSM)

RSM is a method used in statistics, which analyzes the association between a number of input variables and one or more response variables. It has been applied as a simulation optimization mechanism (Kleijnen, 2008b). Although RSM is intended for solving problems with continuous variables, it is also applicable for problems with discrete variables. The RSM-based simulation optimization mechanism is described in detail by Kleijnen, (2008a).

The method starts with a given initial solution (same as SA approach), and calculates the solutions in the immediate neighborhood. A linear regression model is developed based on the neighborhood solution space. The value of the decision variable with the highest regression coefficient (absolute value) is increased (if coefficient is positive) or decreased (if negative) one step to generate the next solution. The algorithm ends when there is no room left for improvement.

5.5 Greedy Search (GS) & Fast Greedy Search (FGS)

GS uses the same initial solution as RSM. It differs from RSM in the selection of the next solution. GS simply selects the solution with the best profit in the neighborhood solution space and continues the search in the neighborhood of the selected solution. FGS selects the first better solution while scanning the immediate neighborhood of the current solution instead of enumerating the whole neighborhood. Search ends when a neighborhood fails to produce an improved solution.

5.6 Hybrid Methods

To determine whether combinations of any two of these methods can produce better results, I propose two hybrid methods: (1) ACO_GS takes the best combination from ACO as the initial solution for GS; (2) ACO_OptQ uses the result from ACO as the initial solution for OptQuest.

By default OptQuest uses the mid-points of the intervals of the values any decision variable can take as the initial solution. To test whether such an initial solution would produce good results for other methods, I also introduce MidPnt_GS method, which takes the mid-point of decision variables as the initial solution for GS.

Chapter 6
Results

6.1 Numerical Experiments

I solved both the real-life case and test-bed problem using the methods described above. I also performed sensitivity analyses under different demand (D) and rate of inventory carrying cost (r^{inv}) scenarios as these parameters are associated with variation. Engineering team at ABC expect a fluctuation between + 10% and -15% in the annual demand and a ±25% change in r^{inv}. Table 6.1 and Table 6.2 display the solutions found by each simulation optimization method for the base case (1 x D and 1 x r^{inv}) and the sensitivity scenarios (e.g., 1.1 x D and 1.25 x r^{inv}) in terms of the percentage of the expected profit with respect to the best solution found.

The best solution refers to the optimal solution for the cases of the test-bed problem, as the solution spaces were made available via total enumeration. The percentages of the expected profit for the test-bed and the real-life cases are denoted by Per_s and $\overline{Per_s}$, respectively. These percentages are calculated according to Equations (6.1) and (6.2), where $\hat{P}_s, s \in (1,2,\cdots,9)$ denotes the highest expected profit found by the method *s* and P^* refers to the maximum expected profit found by total enumeration. The profit values for the real-life case are divided by an undeclared real number to respect the confidentiality agreement with ABC.

$$Per_s = 100 \;.\; \hat{P}_s / P^* \tag{6.1}$$

$$\overline{Per_s} = 100 \;.\; \hat{P}_s / max\{\hat{P}_1, \hat{P}_2, \cdots, \hat{P}_9\} \tag{6.2}$$

The number of necessary iterations for each method (**Iterations**) and at which iteration the best solution is attained (**Max @**) are also given in Table 6.1 and Table 6.2. "Max. Expected Profit" column shows the best profit found ($P^*\ or\ max\{\hat{P}_1, \hat{P}_2, \cdots, \hat{P}_9\}$) for each scenario. In this experiment setting, the percentage of the expected profit shows the optimality gap as a measure solution quality. While **Iterations** provides a measure of the required computation time, **Max @** reflects the convergence rate of the proposed methods.

6.2 Results for the Test-bed Problem

Table 6.1 is summary of results for test-bed problem. It illustrates the optimality gap associated with the proposed simulation optimization approaches by comparing them with total enumeration. The comparison implies that ACO and RSM significantly deviate from the best solution while FGS finds the best configurations in most cases and it is significantly faster than other algorithms. Although it is myopic FGS mostly reaches the optimal point in most cases while ACO and RSM often stuck at local maxima as shown in Figure 6.1. However, in one case (0.85 x D and 1 x r^{inv}), FGS proposes a significantly inferior solution. As expected, GS is slower than FGS, whereas, associated with better solution quality. OptQuest, the slowest method, also either finds the best solutions in most cases or do not significantly deviate from the best solution. SA and ACO_GS are faster than OptQuest while their solution qualities are comparable to that of OptQuest on the test-bed problem.

Table 6.1 implies that OptQuest is a reliable solution technique for small size problems in terms of solution quality. Therefore, practitioners may prefer to use the easy-to-implement commercial tool for such problem instances. However, OptQuest is very slow; thus, it may not be an effective method for problems with a large number of station structure options where a separate OptQuest run is needed to find the approximately-optimal station configuration for each station structure option. For small size problems, FGS provides a fast and almost reliable alternative. However, the practitioners may prefer to use GS, SA, or ACO_GS instead by sacrificing some computation time in order to avoid a potentially significant deviation from the optimal solution.

Figure 6.1 shows the path taken by each optimization method on the total enumeration surface. Horizontal axis shows the combination ID, and the vertical axis denotes the profit. Local minima and maxima can clearly be seen on the enumeration graph. The dots on the graph are the solution points that correspond to individual combinations. Although there are some random scattering, all methods except for OptQuest, follows a certain path on the enumeration graph. Because the exact solution algorithm for OptQuest is unknown, it is hard to comment on how it moves on the graph.

Figure 6.2 displays the convergence graphs related with the methods applied for the test-bed problem and the cases of the sensitivity analysis. Graphs trace the best value found for every 20 iterations; i.e. the best value for the first 20 iterations followed by the best value found for the first 40 iterations and so on. These graphs also show that OptQuest is the slowest converging method among all, then comes the ACO. FGS is the fastest converging algorithm, while it may get stuck in a local minimum for cases, where annual demand is less than expected, which might create shortage of effective system solutions.

Table 6.1 Results for the test-bed problem

PROBLEM TYPE		METHODS									Max. Expected Profit
		OptQuest	**SA**	**ACO**	**RSM**	**GS**	**FGS**	**ACO_GS**	**ACO_OptQ**	**MidPnt_GS**	(Optimal)
Base Case	**BEP***	**100%**	**100%**	86.18%	97.62%	**100%**	**100%**	**100%**	**100%**	**100%**	$2,966,279
	Max @	357	73	**50**	67	90	51	97	310	119	
	Iterations	500	89	**60**	88	112	65	113	500	142	
SENSITIVITY ANALYSIS											(Optimal)
0.85xD r^{inv}=20%	**BEP**	**100%**	84.74%	89.34%	63.19%	92.28%	63.19%	89.34%	99.48%	93.30%	$880,120
	Max @	189	**38**	212	134	83	50	215	323	99	
	Iterations	500	**51**	215	138	96	88	231	500	114	
0.9xD r^{inv}=20%	**BEP**	92.23%	**100%**	84.22%	92.47%	**100%**	**100%**	92.98%	**100%**	82.96%	$2,036,146
	Max @	364	103	37	**36**	65	73	78	281	71	
	Iterations	500	107	**39**	68	82	88	92	500	87	
1.1xD r^{inv}=20%	**BEP**	**100%**	**100%**	97.71%	97.19%	**100%**	**100%**	**100%**	**100%**	**100%**	$2,516,279
	Max @	377	77	71	66	90	**51**	79	194	119	
	Iterations	500	85	75	81	108	**60**	101	500	140	
1xD r^{inv}=15%	**BEP**	**100%**	**100%**	98.82%	97.72%	**100%**	**100%**	**100%**	**100%**	**100%**	$3,107,713
	Max @	465	73	126	66	89	51	147	218	119	
	Iterations	500	87	134	94	107	65	165	500	140	
1xD r^{inv}=25%	**BEP**	**100%**	**100%**	98.69%	97.50%	**100%**	**100%**	**100%**	**100%**	**100%**	$2,824,829
	Max @	465	73	87	66	89	**51**	166	241	119	
	Iterations	500	87	159	94	107	**65**	182	500	140	

*BEP: Best Expected Profit

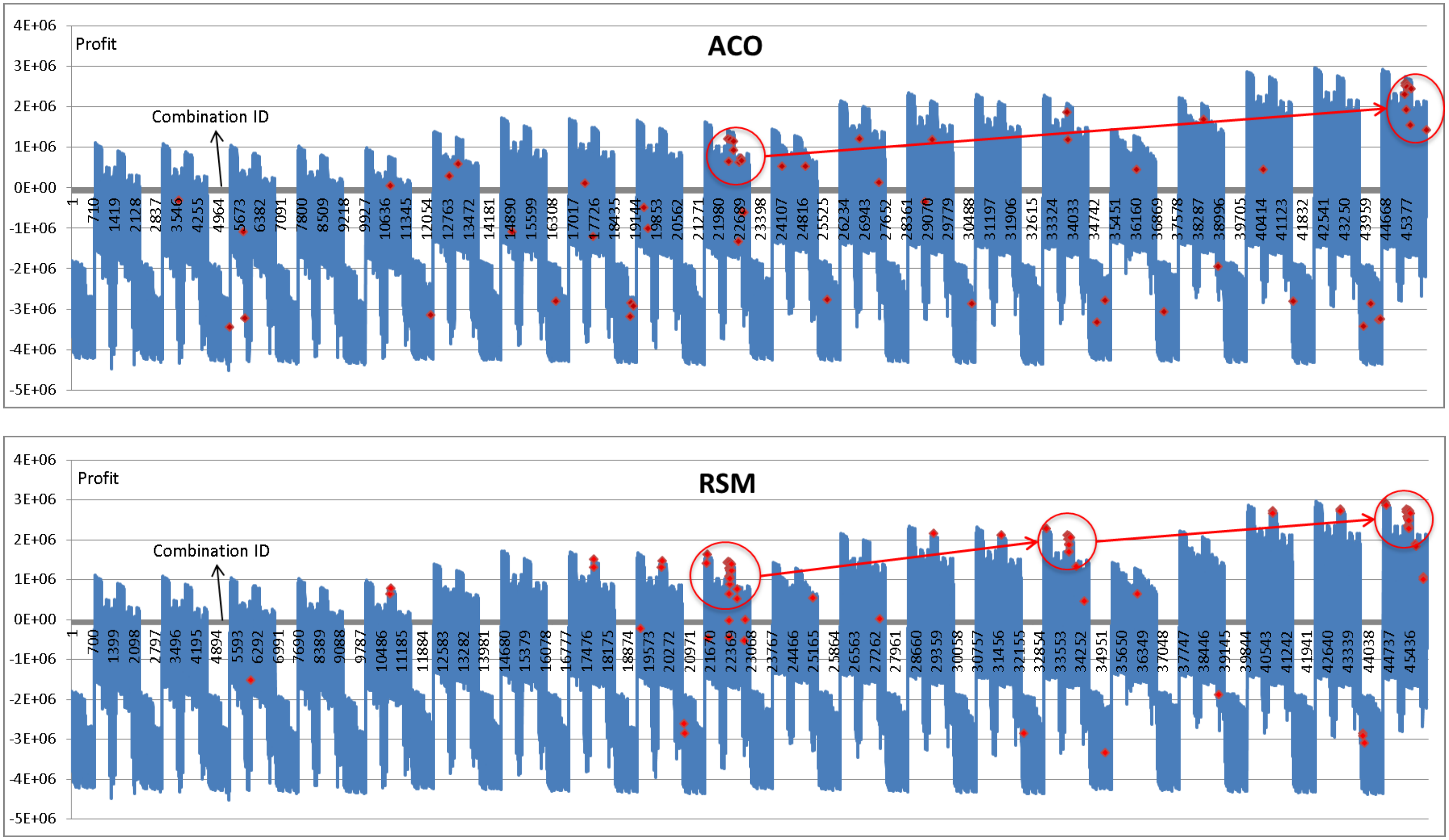


Figure 6.1 Solution path followed by each method on the total enumeration graph (test-bed problem)

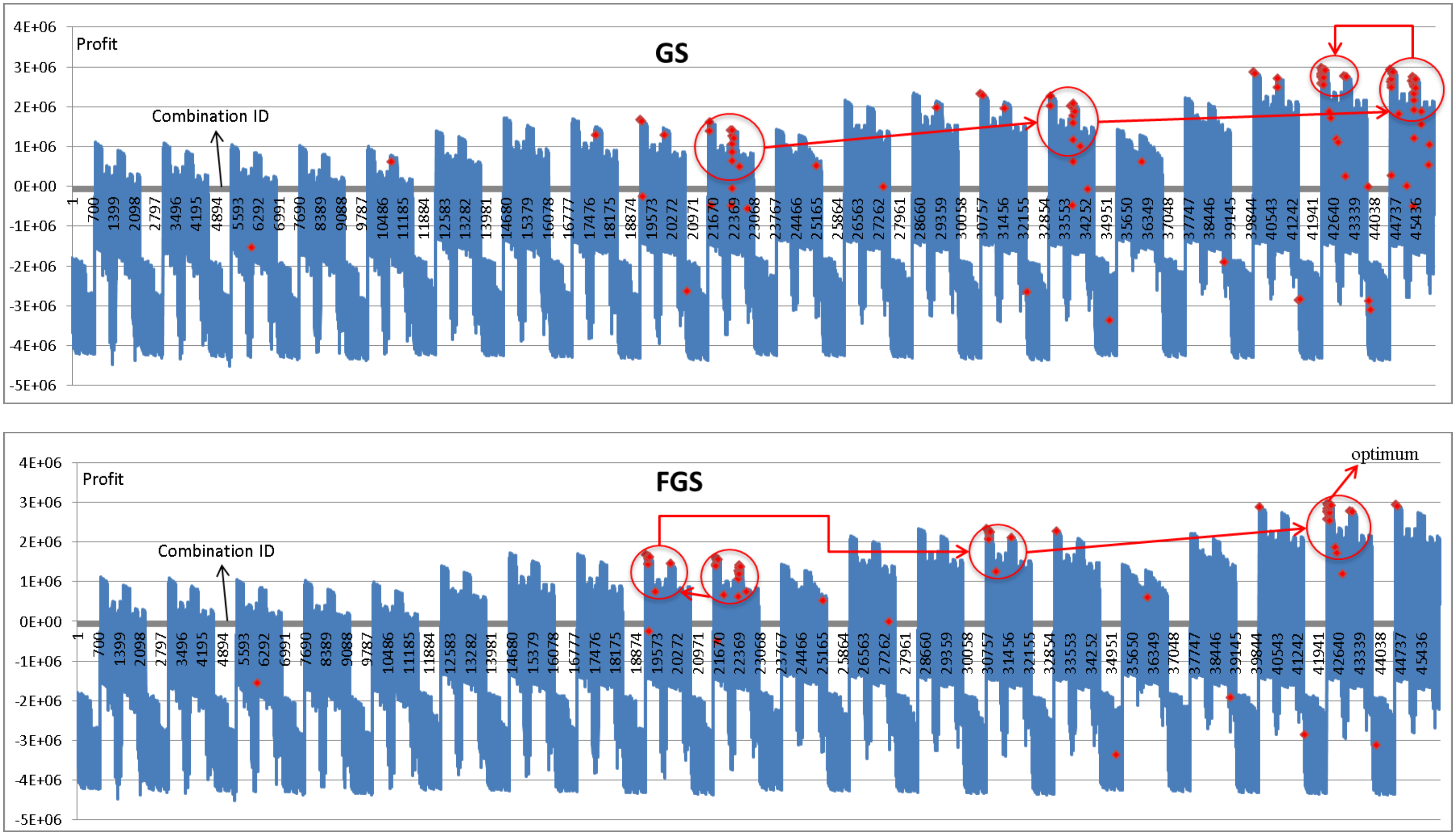
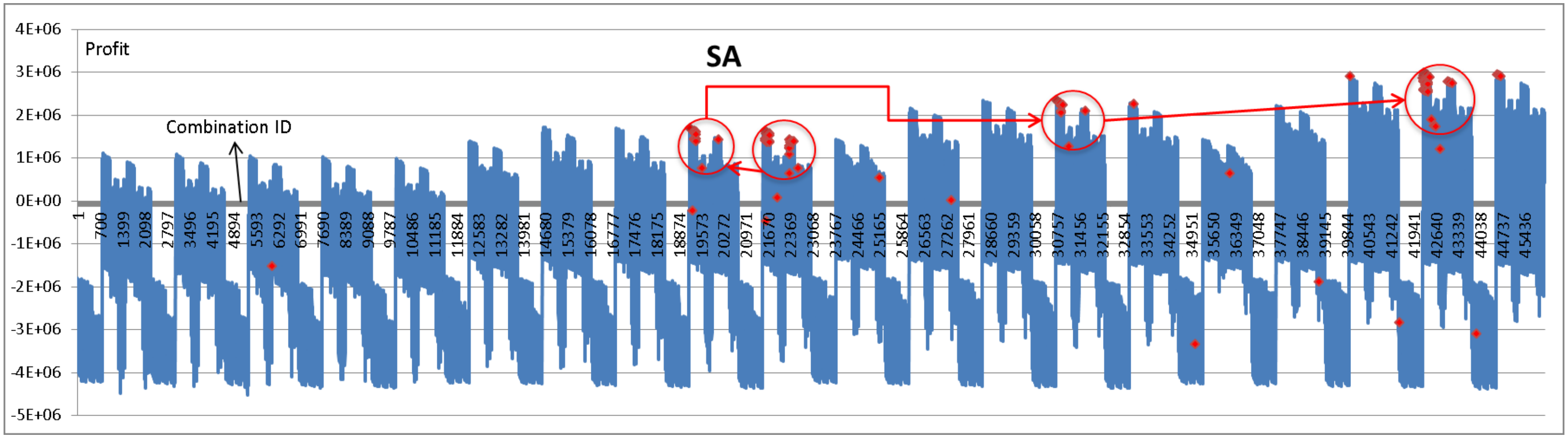


Figure 6.1 Solution path followed by each method on the total enumeration graph (test-bed problem) – Continued



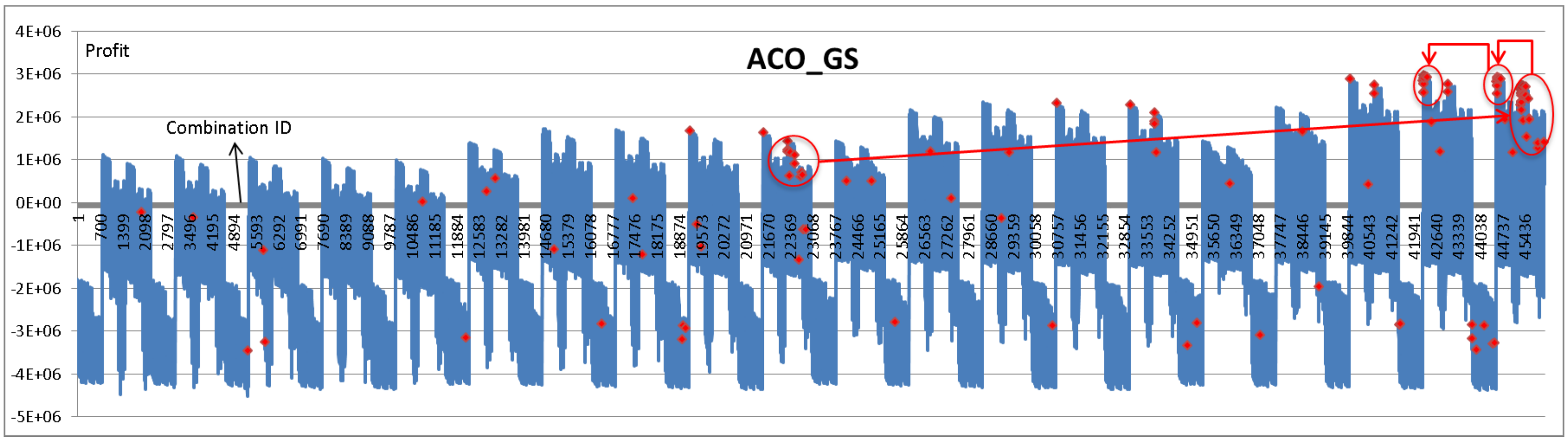


Figure 6.1 Solution path followed by each method on the total enumeration graph (test-bed problem) – Continued

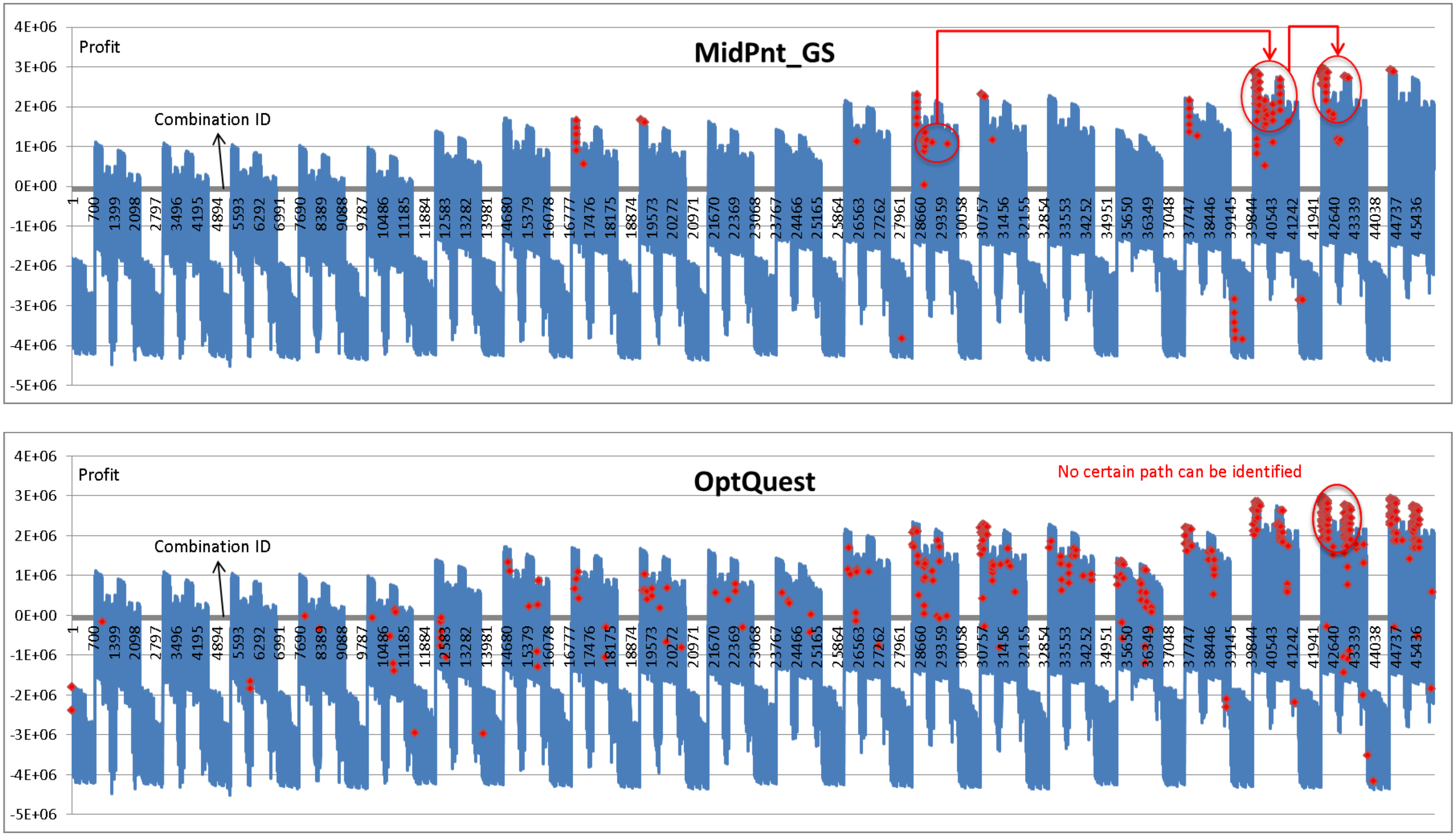


Figure 6.1 Solution path followed by each method on the total enumeration graph (test-bed problem) – Continued

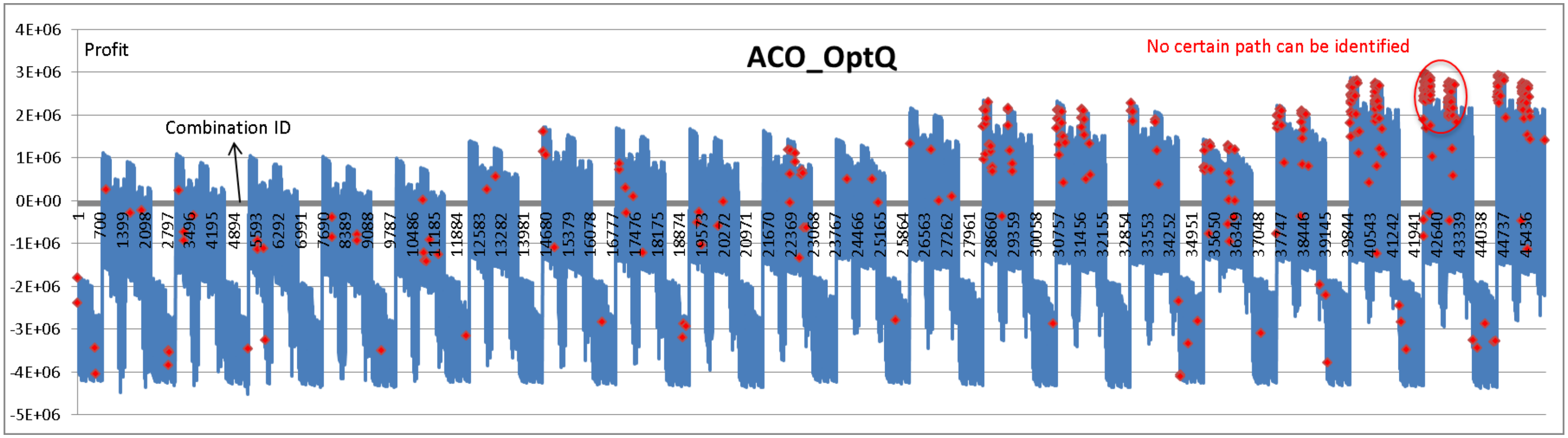


Figure 6.1 Solution path followed by each method on the total enumeration graph (test-bed problem) – Continued

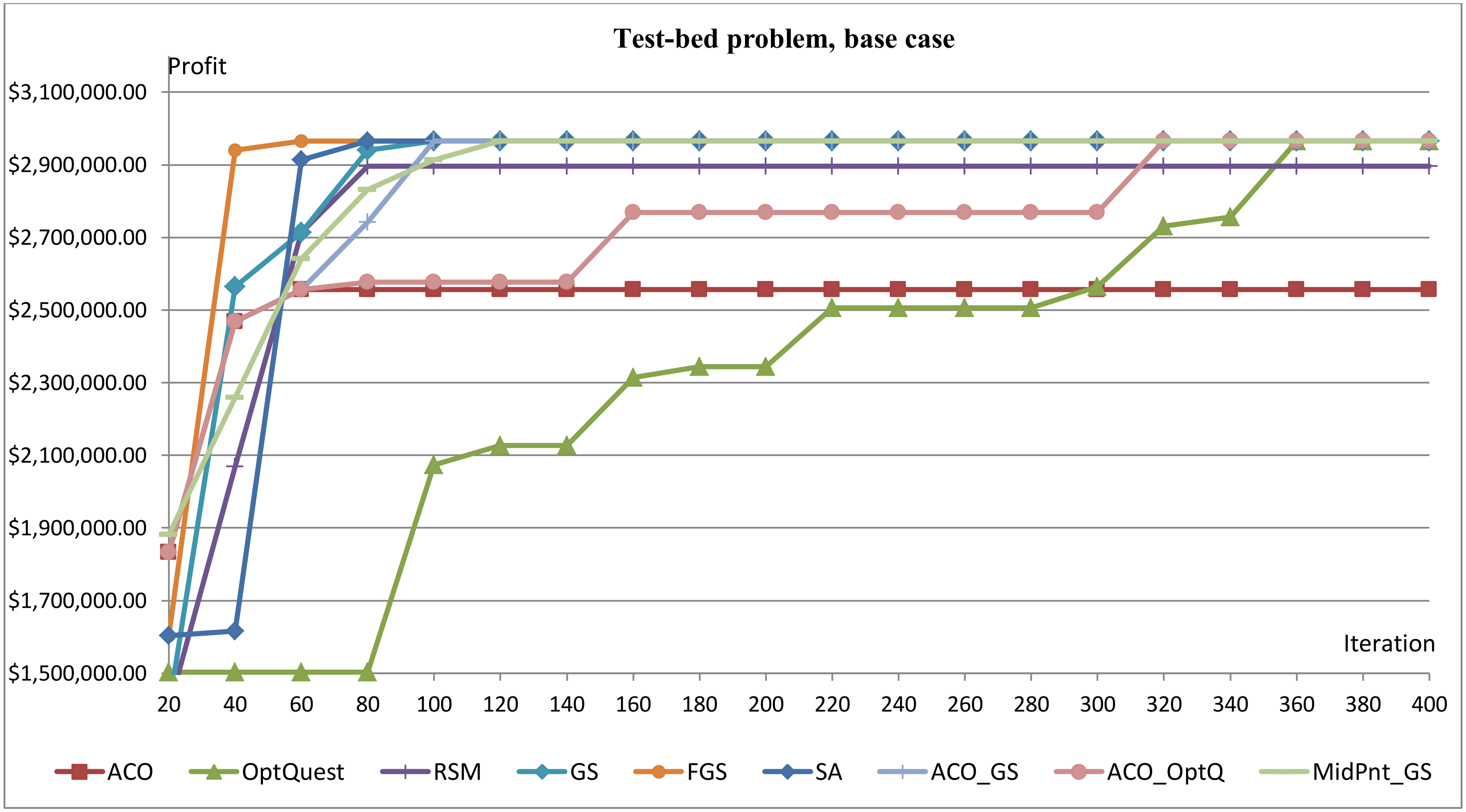


Figure 6.2 Convergence graphs for test-bed problem, base case & the cases from the sensitivity analysis

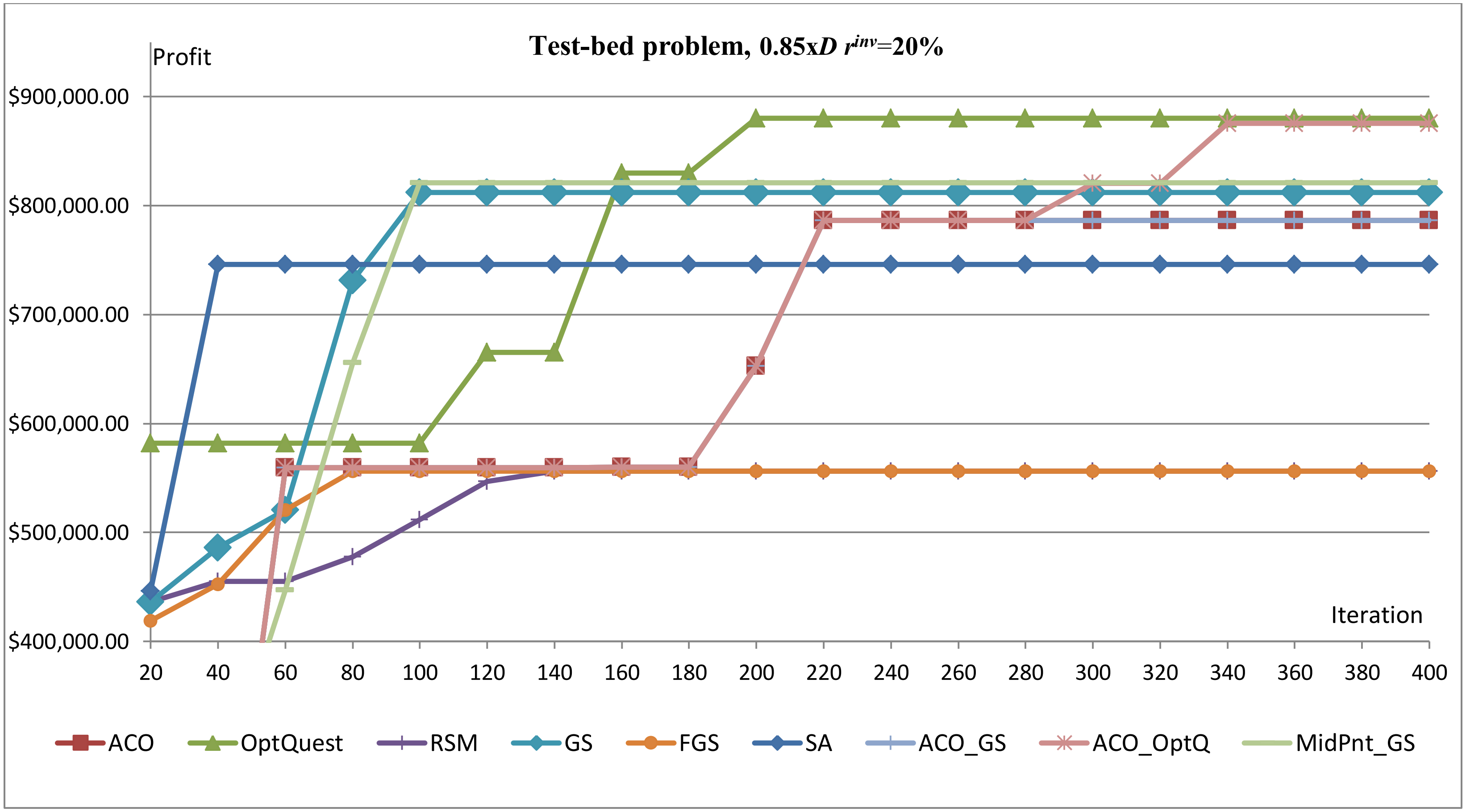


Figure 6.2 Convergence graphs for test-bed problem, base case & the cases from the sensitivity analysis – continued

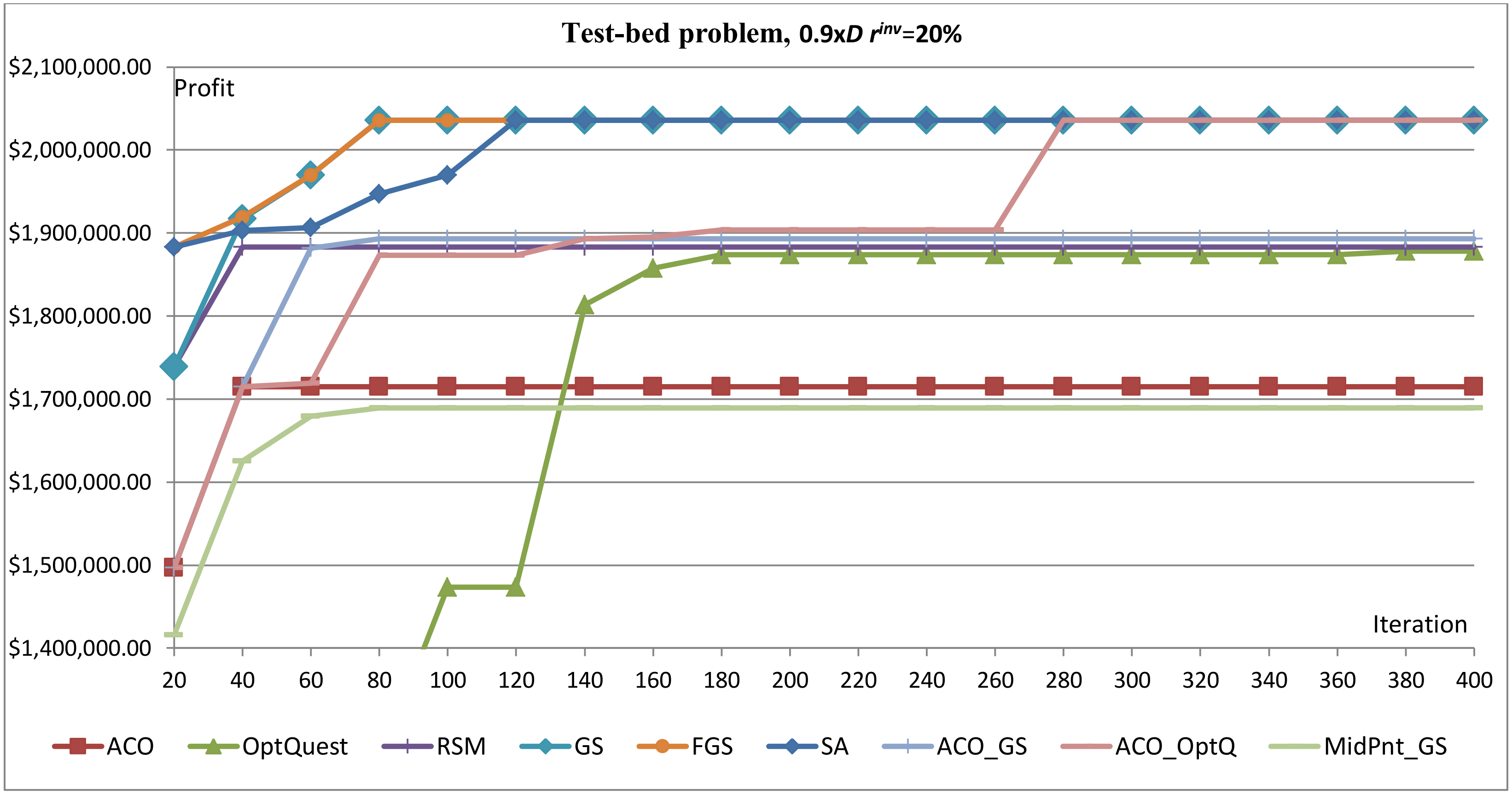


Figure 6.2 Convergence graphs for test-bed problem, base case & the cases from the sensitivity analysis – continued

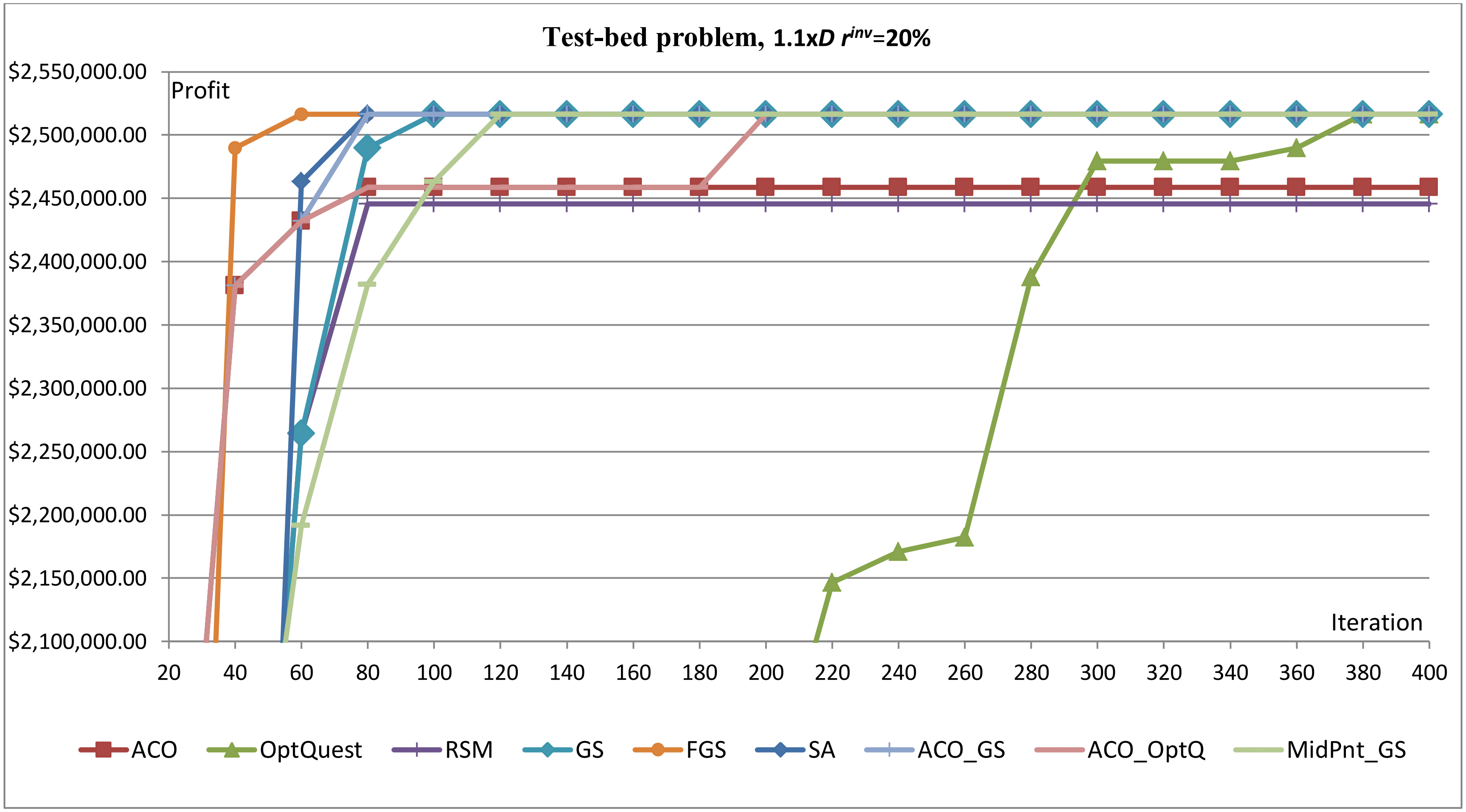


Figure 6.2 Convergence graphs for test-bed problem, base case & the cases from the sensitivity analysis – continued

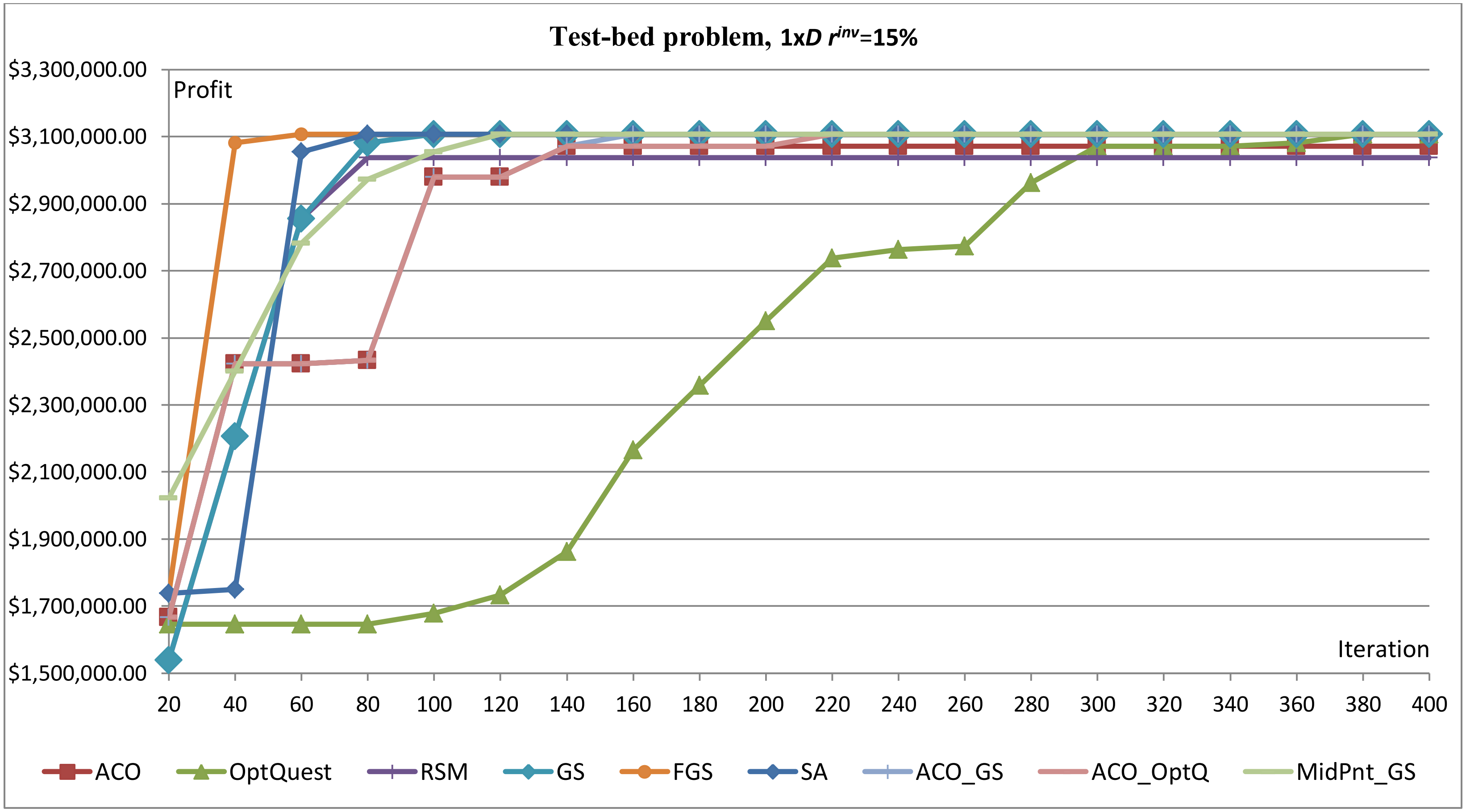


Figure 6.2 Convergence graphs for test-bed problem, base case & the cases from the sensitivity analysis – continued

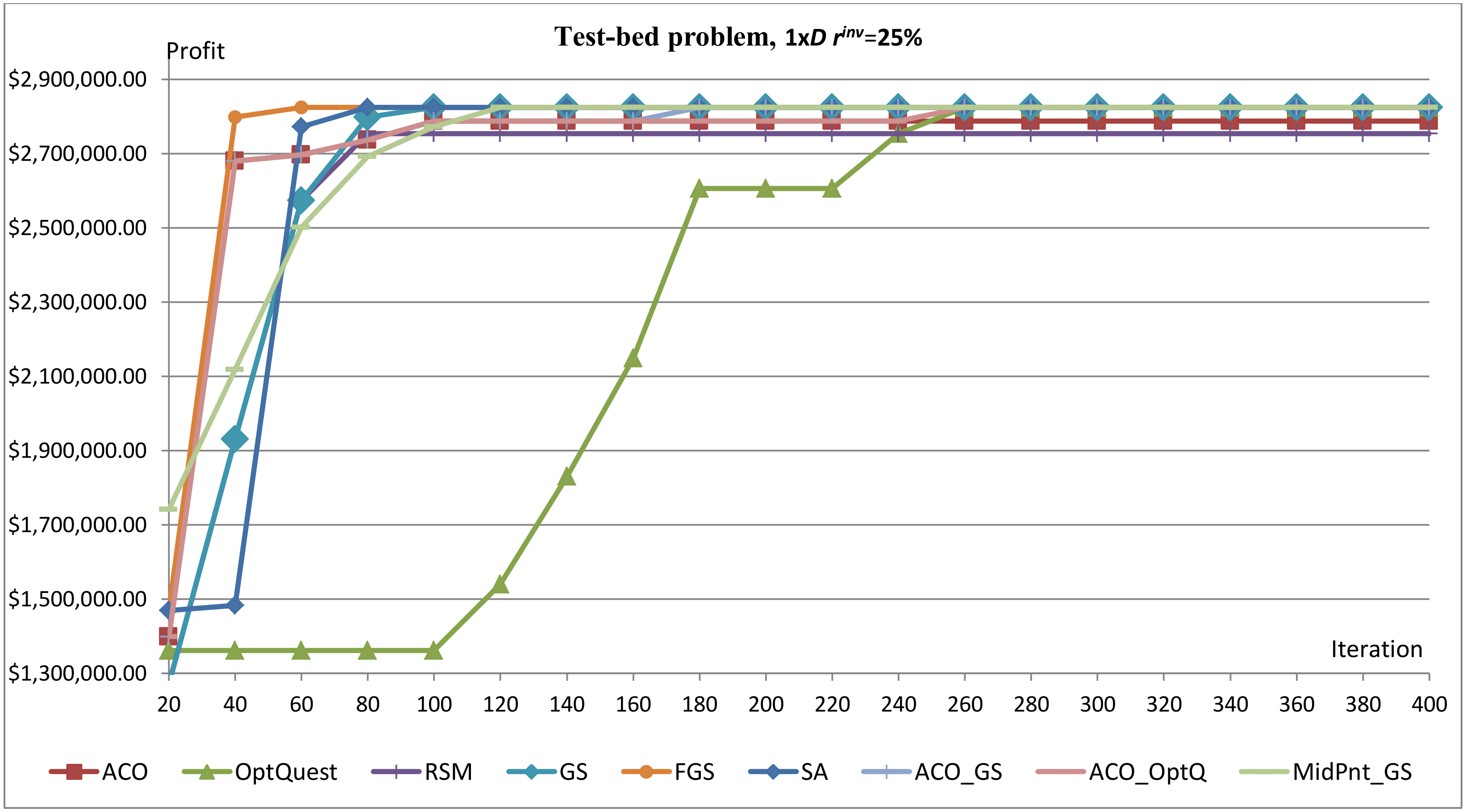


Figure 6.2 Convergence graphs for test-bed problem, base case & the cases from the sensitivity analysis – continued

6.3 Results for the Real-life Problem

The comparisons on the real-life case in Table 6.2 show how the problem size may affect the performances of the proposed simulation optimization approaches. Note that, the profit values for the real-life case are divided by an undeclared real number for confidentiality purposes. In the real-life problem, both FGS and OptQuest deviate from the best solution in most cases; however, the maximum deviation from the best solution is lower for OptQuest (100% - 95.55%=4.4%) compared to that for FGS (100% - 91.34% =8.66%). ACO_GS finds the best solutions for all problem instances. The solution quality of SA and GS are similar to each other and relatively lower than that of ACO_GS while the maximum deviation from the best solution is much lower for SA compared to GS.

According to these results, ACO_GS is likely to provide the best result for large size problems. However, if there are large amount of station structure options to be evaluated, practitioners may prefer to use SA or GS and trade some of the solution quality with a significant relief from the computation time. Table 6.2 also illustrates that the solution quality for ACO_GS and OptQuest does not change significantly with the varying levels of demand and inventory holding cost. Similarly, GS and FGS are robust to the changes in inventory holding costs. However, their solution quality is better when the demand is high (Demand≥1.1xD). In case of low demand, there may be limited number of effective system configurations with high solution quality; thus, a myopic search algorithm may get stuck into local optima more easily.

Convergence graphs for methods applied to the real-life case are given in Figure 6.3. Graphs show the best value found at every 20 iterations; i.e. the best value for the first 20 iterations followed by the best value found for the first 40 iterations and so on. Convergence patterns for the real-life problem are similar to those of the test-bed problem.

All calculations assume that machines work with 85% efficiency. To understand how the system can be affected if machines performed at higher or lower efficiency, extra analysis was done. Problem was solved using ACO_GS both for 80% efficiency and 90% efficiency. Results show that 5% increase creates a 1% rise in the profit, while 5% reduction causes 14% drop in the overall annual profit. This significant fall is a result of the failure of the system in achieving the desired throughput level with 80% efficiency using the current possible values of decision variables. At 90% efficiency, even though throughput increases, it has a minor impact on profit; because the demand is fixed.

Table 6.2 Results for the real-life case

PROBLEM TYPE		METHODS									**Max. Expected Profit**
		OptQuest	**SA**	**ACO**	**RSM**	**GS**	**FGS**	**ACO_GS**	**ACO_OptQ**	**MidPnt_GS**	(Approximately optimal)
Base Case	**BEP***	99.85%	99.95%	95.48%	99.51%	**100%**	**100%**	**100%**	99.85%	99.84%	$3,839,504
	Max @	253	74	**48**	49	84	55	130	293	84	
	Iterations	500	89	**60**	74	106	65	149	500	103	
SENSITIVITY ANALYSIS											
0.85xD r^{inv}=20%	**BEP**	**100%**	91.34%	97.98%	96.47%	96.15%	91.34%	**100%**	99.97%	91.34%	$1,913,895
	Max @	238	42	66	72	50	**39**	117	471	74	
	Iterations	500	61	74	106	66	**54**	135	500	88	
0.9xD r^{inv}=20%	**BEP**	99.92%	99.84%	97.09%	99.24%	99.94%	99.84%	**100%**	99.94%	99.88%	$2,603,223
	Max @	490	117	144	182	108	126	216	378	**87**	
	Iterations	500	131	149	190	142	140	234	500	**101**	
1.1xD r^{inv}=20%	**BEP**	**100%**	**100%**	99.14%	**100%**	**100%**	**100%**	**100%**	99.84%	**100%**	$3,541,028
	Max @	338	97	80	256	119	**60**	118	278	63	
	Iterations	500	114	82	260	138	**75**	142	500	87	
1xD r^{inv}=15%	**BEP**	99.83%	**100%**	95.55%	99.51%	**100%**	**100%**	**100%**	99.85%	99.29%	$3,904,821
	Max @	473	106	**48**	49	84	61	134	326	50	
	Iterations	500	121	**60**	74	106	76	153	500	73	
1xD r^{inv}=25%	**BEP**	99.97%	99.98%	95.41%	99.50%	**100%**	**100%**	**100%**	99.97%	99.86%	$3,779,095
	Max @	343	74	**48**	49	84	57	149	327	83	
	Iterations	500	89	**60**	74	106	76	168	500	99	

*BEP: Best Expected Profit

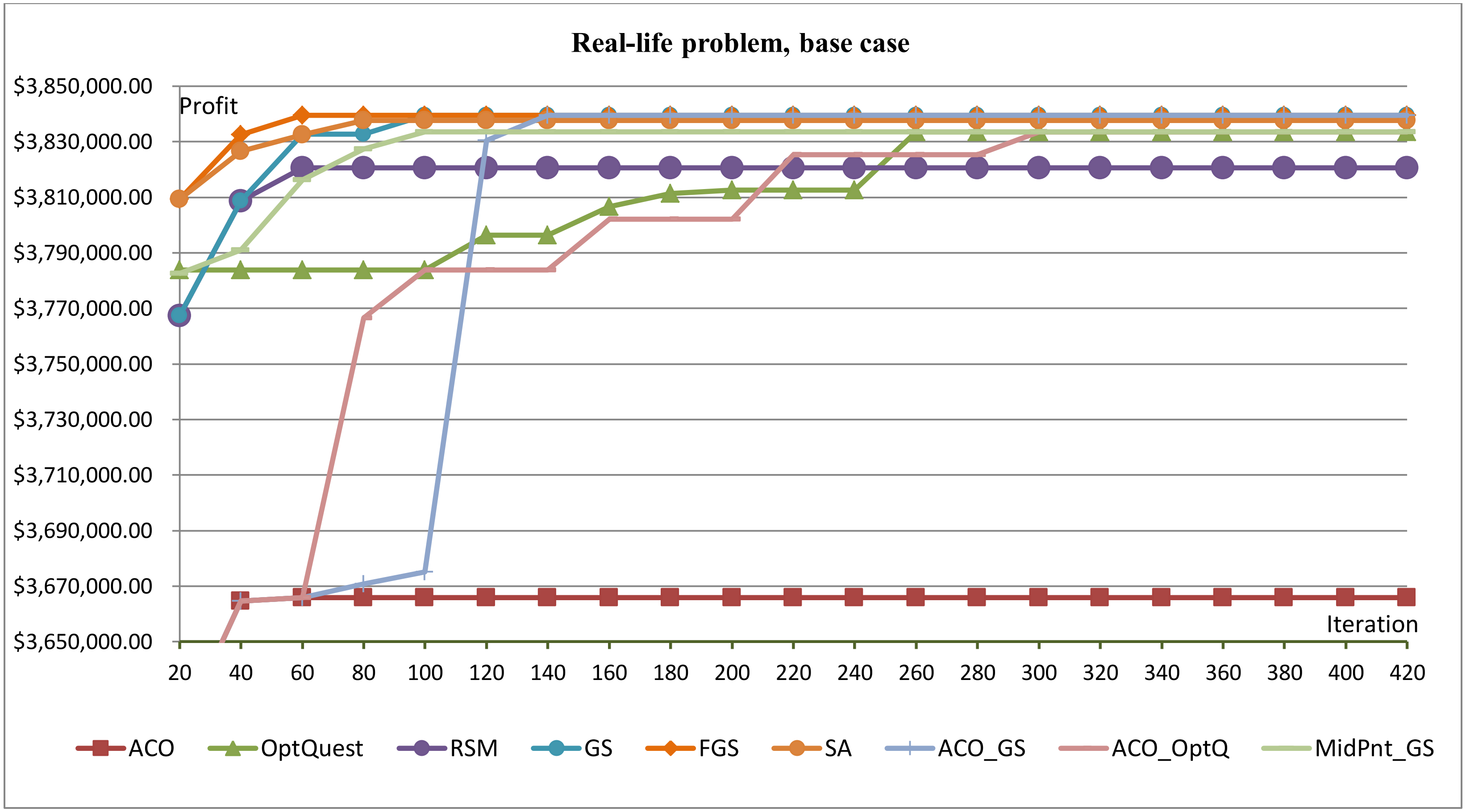


Figure 6.3 Convergence graphs for real-life problem, base case & the cases from the sensitivity analysis

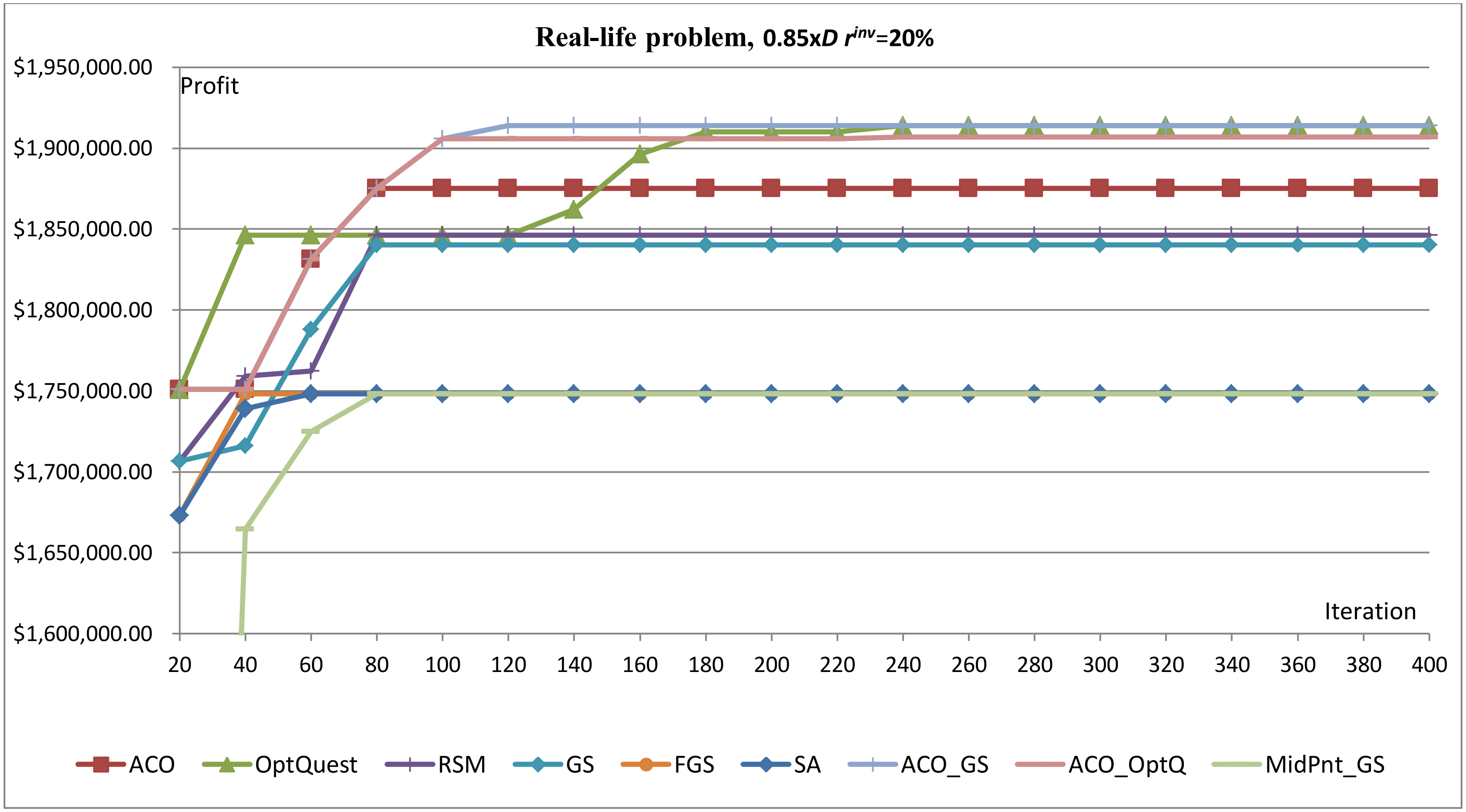


Figure 6.3 Convergence graphs for real-life problem, base case & the cases from the sensitivity analysis - continued

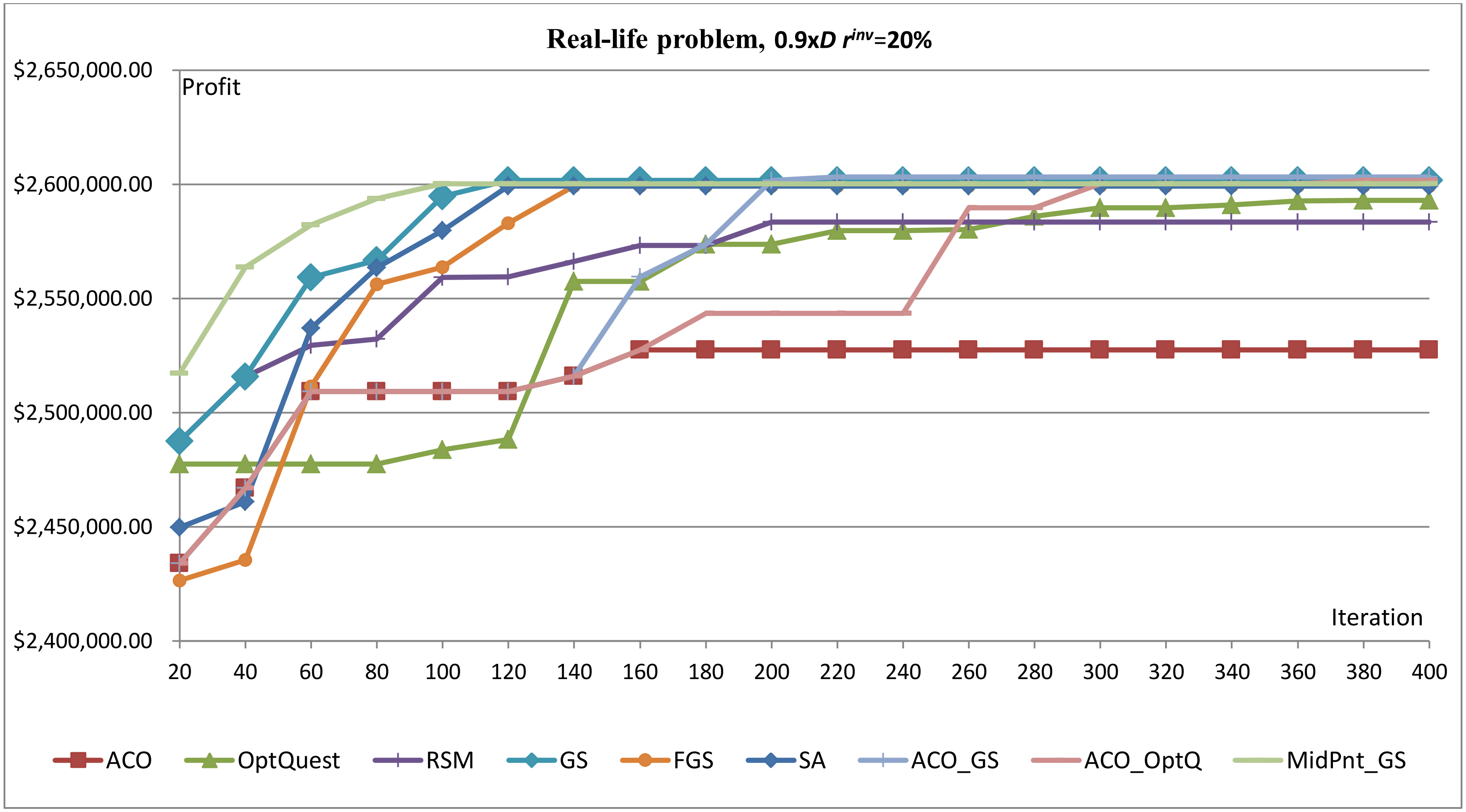


Figure 6.3 Convergence graphs for real-life problem, base case & the cases from the sensitivity analysis - continued

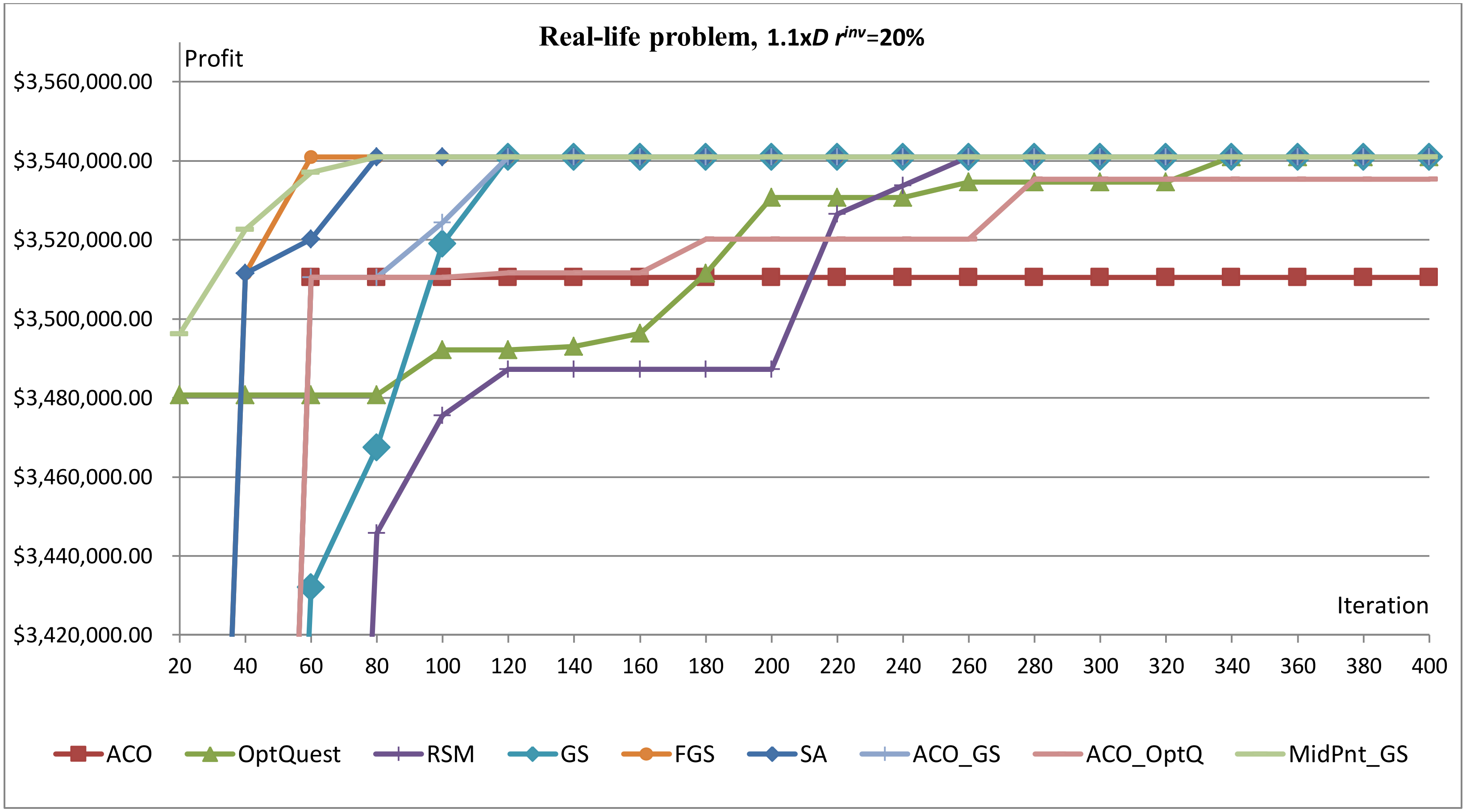


Figure 6.3 Convergence graphs for real-life problem, base case & the cases from the sensitivity analysis - continued

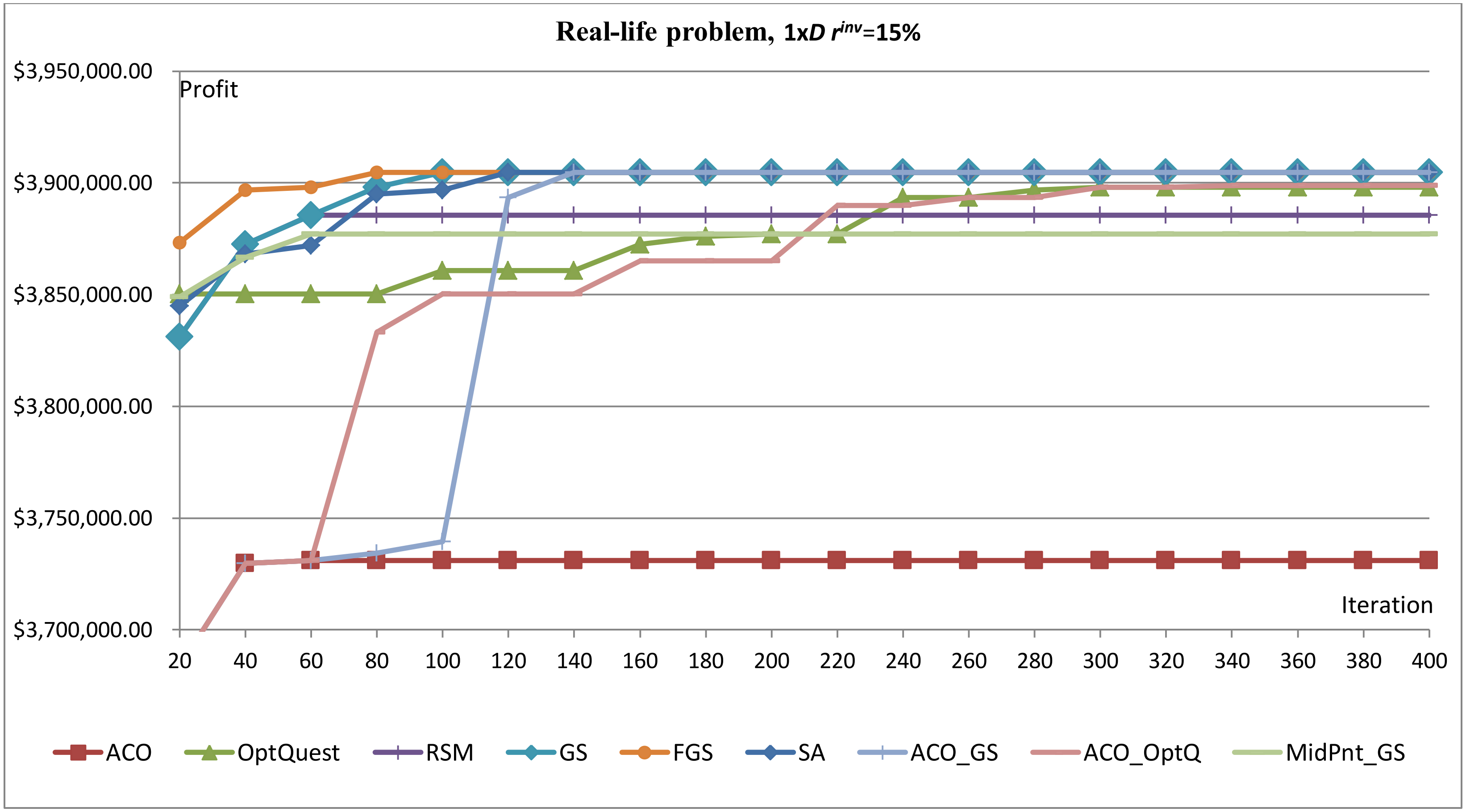


Figure 6.3 Convergence graphs for real-life problem, base case & the cases from the sensitivity analysis - continued

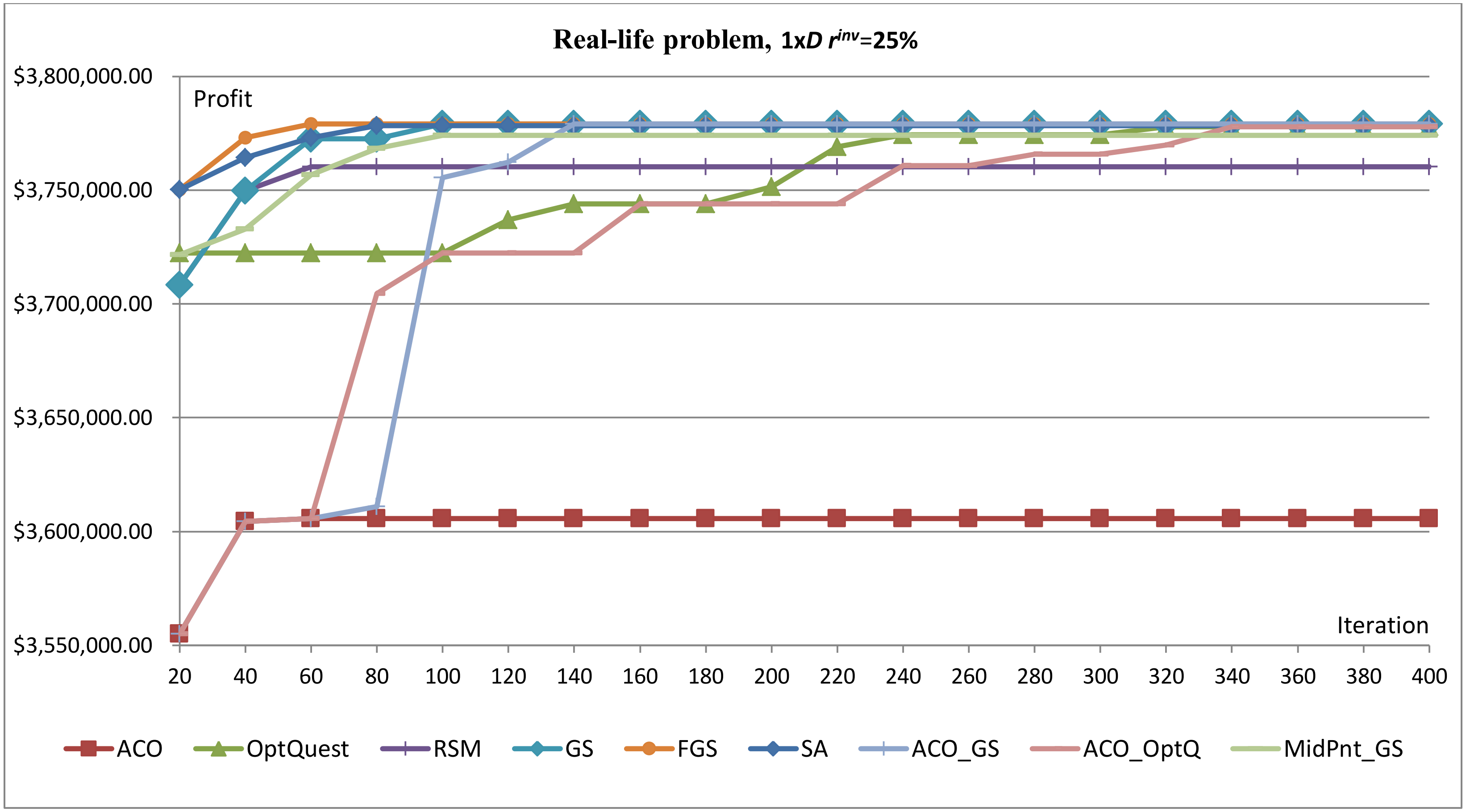


Figure 6.3 Convergence graphs for real-life problem, base case & the cases from the sensitivity analysis - continued

Chapter 7
Conclusion and Recommendations

Reverse transfer of manufacturing systems as defined above has the potential to become a common phenomenon for capital-intense Western and Japanese manufacturers. The general approach and proposed simulation optimization methods presented in this thesis may help practitioners make more informed strategic investment decisions towards efficient production line reconfiguration.

Transferring production lines from a labor-intensive environment to a capital-intensive environment usually requires reconfiguration in order to increase productivity and capacity. Newly designed automatized processes can be incorporated to the stations with the intention of increasing throughput. Some cases may involve several feasible options for any given station, where decision makers need to know which option would yield better results. This makes the problem in hand two tiered. First is determining which station options to use, and second is finding the best configuration for that particular option.

The experience I had during my employment with ABC showed that decision makers should consider the variance and randomness in the complex nature of the reconfigured manufacturing system in order to prevent significant efficiency losses due to a sub-optimal implementation. In this context, practitioners facing similar challenges may significantly benefit from developing simulation-based laboratory environments where they can test what-if questions regarding the redesigned\transferred manufacturing system. For example, I observed significant performance differences between the approximately optimal production line configurations found by the proposed approaches and the initial production line design recommended to ABC by the labor-intense Asian company. The simulation results show that the approximately optimal configurations are able to produce about 35% more end-products than the initial design which even failed to satisfy the targeted production capacity[1].

It is also observed that the proposed simulation optimization methods that can be applied to similar problems lie in a particular solution quality and speed spectrum. The practitioners facing similar problems may prefer to employ easy-to-implement software packages such as OptQuest. This might be a reasonable approach for small or large problems given that there is limited number of possible

[1] As per the confidentiality agreement with ABC, details of this calculation are not provided.

station structure options for which the simulation optimization procedure needs to be repeated. However, the decision makers may need to invest in custom-developed faster approaches when the number of possible station structure combinations is high. In such cases, the proposed ACO_GS and GS approaches has the potential to reduce the computational burden and improve the solution quality.

One limitation of this study is the lack of historical data associated with machines and labor, for which I had to rely on the specifications provided by the machine producers and the assumptions of the engineering team. In addition, I was not able to develop an analytical framework for this problem because the considered system requires many generalizations and simplifications in order to be modeled analytically. Such generalizations and simplifications would greatly reduce the accuracy of the results. Another limitation is the absence of randomly generated test problems that could enable further analysis of the performance of the proposed simulation optimization methods. Since the system is complex, random instances may not be able to generate feasible production lines.

As future research, simulation optimization methods proposed in this thesis can be tested on other similar real-life problems. Researchers can also work on creating feasible random test-bed instances to further analyze the performance of the approach. It is also possible to develop other heuristic algorithms to benchmark with the ones presented in this study.

Bibliography

Alabas, C., Altiparmak, F., & Dengiz, B. (2002). A comparison of the performance of artificial intelligence techniques for optimizing the number of kanbans. *Journal of the Operational Research Society*, 53(8), 907-914.

Alrefaei, M. H., & Diabat, A. H. (2009). A simulated annealing technique for multi-objective simulation optimization. *Applied Mathematics and Computation,* 215(8), 3029-3035.

Altiparmak, F., Dengiz, B., & Bulgak, A. A. (2002). Optimization of buffer sizes in assembly systems using intelligent techniques. *Proceedings of the 2002 Winter Simulation Conference*, San Diego.

Altiparmak, F., Dengiz, B., & Bulgak, A. A. (2007). Buffer allocation and performance modeling in asynchronous assembly system operations: An artificial neural network metamodeling approach. *Applied Soft Computing Journal,* 7(3), 946-956.

Anderson, D. R., & Moodie, C. L. (1969). Optimal buffer storage capacity in production line systems. *International Journal of Production Research,* 7(3), 233-240.

Andradottir, S. (1998). A review of simulation optimization techniques. *Proceedings of the 1994 Winter Simulation Conference,* Orlando.

Angun, E., Kleijnen, J., Den Hertog, D., & Gurkan, G. (2009). Response surface methodology with stochastic constraints for expensive simulation. *Journal of the Operational Research Society,* 60(6), 735-746.

Azadeh, A., Shirkouhi, S., & Rezaie, K. (2010). A robust decision-making methodology for evaluation and selection of simulation software package. *The International Journal of Advanced Manufacturing Technology,* 47(1), 381-393.

Azadivar, F. (1992). A tutorial on simulation optimization. *Proceedings of the 1992 Winter Simulation Conference,* Arlington.

Aziz, A., Jarrahi, F., & Abdul-Kader, W. (2010). Modeling and performance evaluation of a series-parallel flow line system with finite buffers. *Information Systems and Operational Research,* 48(2), 103-120.

Battini, D., Persona, A., & Regattieri, A. (2009). Buffer size design linked to reliability performance: A simulative study. *Computers & Industrial Engineering,* 56(4), 1633-1641.

Baykasoglu, A. (2008). Gene expression programming based meta-modelling approach to production line design. *International Journal of Computer Integrated Manufacturing,* 21(6), 657-665.

Bergeron, D., Jamali, M. A., & Yamamoto, H. (2010). Modelling and analysis of manufacturing systems: A review of existing models. *International Journal of Product Development,* 10(1-3), 46-61.

Blumenfeld, D. E., & Li, J. (2005). An analytical formula for throughput of a production line with identical stations and random failures. *Mathematical Problems in Engineering,* 2005(3), 293-308.

Bonvik, A. M., Dallery, Y., & Gershwin, S. B. (2000). Approximate analysis of production systems operated by a CONWIP/finite buffer hybrid control policy. *International Journal of Production Research,* 38(13), 2845-2869.

Bulgak, A. A. (2006). Analysis and design of split and merge unpaced assembly systems by metamodelling and stochastic search. *International Journal of Production Research,* 44(18-19), 4067-4080.

Bulgak, A. A., Diwan, P. D., & Inozu, B. (1995). Buffer size optimization in asynchronous assembly systems using genetic algorithms. *Computers & Industrial Engineering,* 28(2), 309-322.

Burman, M. H. (1995). New results in flow line analysis. Thesis (PhD), OR Center, Massachusetts Institute of Technology.

Carlos, D. P., Jairo, R. M., & Aldo, F. (2008). Simulation-optimization using a reinforcement learning approach. *Proceedings of the 2008 Winter Simulation Conference,* Miami.

Carmichael, K. (2012). Canada's 'reshoring' opportunity. *Globe and Mail,* June 4, 2012

Chan, F. T. S. (2001). Simulation analysis of maintenance policies in a flow line production system. *International Journal of Computer Applications in Technology,* 14(1-3), 78-86.

Chan, F. T. S., & Ng, E. Y. H. (2002). Comparative evaluations of buffer allocation strategies in a serial production line. *International Journal of Advanced Manufacturing Technology,* 19(11), 789-800.

Chiang, S., Hu, A., & Meerkov, S. M. (2008). Lean buffering in serial production lines with nonidentical exponential machines. *IEEE Transactions on Automation Science and Engineering,* 5(2), 298-306.

Colledani, M., Gandola, F., Matta, A., & Tolio, T. (2008). Performance evaluation of linear and non-linear multi-product multi-stage lines with unreliable machines and finite homogeneous buffers. *IIE Transactions,* 40(6), 612-626.

Dallery, Y., & Gershwin, S. (1992). Manufacturing flow line systems: A review of models and analytical results. *Queueing Systems,* 12(1-2), 3-94.

De Vericourt, F., & Gershwin, S. B. (2004). Performance evaluation of a make-to-stock production line with a two-parameter-per-machine policy: The control point policy. *IIE Transactions,* 36(3), 221-236.

Demir, L., Tunali, S., & Lkketangen, A. (2011). A tabu search approach for buffer allocation in production lines with unreliable machines. *Engineering Optimization,* 43(2), 213-231.

Dengiz, B., Alabas-Uslu, C., & Dengiz, O. (2009). Optimization of manufacturing systems using a neural network metamodel with a new training approach. *Journal of the Operational Research Society,* 60(9), 1191-1197.

Dengiz, B., Bektas, T., & Ultanir, A. E. (2006). Simulation optimization based DSS application: A diamond tool production line in industry. *Simulation Modelling Practice and Theory,* 14(3), 296-312.

Dengiz, B., & Belgin, O. (2007). Paintshop production line optimization using response surface methodology. *Proceedings of the 2007 Winter Simulation Conference,* Washington D.C.

Durieux, S., & Pierreval, H. (2004). Regression metamodeling for the design of automated manufacturing system composed of parallel machines sharing a material handling resource. *International Journal of Production Economics,* 89(1), 21-30.

Economist (2013). Coming home; reshoring manufacturing. *The Economist,* 406 (Jan 19, 2013).

Ellram, L. M. (2013). Offshoring, reshoring and the manufacturing location decision. *Journal of Supply Chain Management,* 49(2), 3-5.

Elshout, J. (1995). Significance of reverse transfer of technology and experiences with its promotion in industrialised countries. *Technovation,* 15(7), 423-435.

Enginarlar, E., Li, J., & Meerkov, S. M. (2005). Lean buffering in serial production lines with non-exponential machines. *OR Spectrum,* 27(2-3), 195-219.

Feyzioglu, O., Pierreval, H., & Deflandre, D. (2005). A simulation-based optimization approach to size manufacturing systems. *International Journal of Production Research,* 43(2), 247-266.

Fishman, C. (2012). The insourcing boom. *The Atlantic Monthly,* 310 (Dec 2012).

Foroohar, R., & Saporito, B. (2013). Made in the USA. *Time,* 181 (Apr 22, 2013).

Fu, M. C. (1994). A tutorial review of techniques for simulation optimization. *Proceedings of the 1994 Winter Simulation Conference,* Orlando.

Fu, M. C., Glover, F. W., & April, J. (2005). Simulation optimization: A review, new developments, and applications. *Proceedings of the 2005 Winter Simulation Conference,* Orlando.

Gershwin, S. B., & Schor, J. E. (2000). Efficient algorithms for buffer space allocation. *Annals of Operations Research,* 93(1), 117-144.

Gershwin, S. B., & Burman, M. H. (2000). A decomposition method for analyzing inhomogeneous assembly/disassembly systems. *Annals of Operations Research,* 93(1-4), 91-115.

Ghiani, G., Legato, P., Musmanno, R., & Vocaturo, F. (2007). A combined procedure for discrete simulation-optimization problems based on the simulated annealing framework. *Computational Optimization and Applications,* 38(1), 133-45.

Gray, J. V., Skowronski, K., Esenduran, G., & Rungtusanatham, M. J. (2013). The reshoring phenomenon: What supply chain academics ought to know and should do. *Journal of Supply Chain Management,* 49(2), 27-33.

Gupta, U. G. (1997). Using citation analysis to explore the intellectual base, knowledge dissemination, and research impact of interfaces (1970-1992). *Interfaces,* 27(2), 85-102.

Haddock, J., & Mittenthal, J. (1992). Simulation optimization using simulated annealing. *Computers & Industrial Engineering,* 22(4), 387-395.

Harris, J. H., & Powell, S. G. (1999). An algorithm for optimal buffer placement in reliable serial lines. *IIE Transactions,* 31(4), 287-302.

Helber, S. (2000). Approximate analysis of unreliable transfer lines with splits in the flow of material. *Annals of Operations Research,* 93(1), 217-243.

Helber, S., Schimmelpfeng, K., Stolletz, R., & Lagershausen, S. (2011). Using linear programming to analyze and optimize stochastic flow lines. *Annals of Operations Research,* 182(1), 193-211.

Hemachandra, N., & Eedupuganti, S. K. (2003). Performance analysis and buffer allocations in some open assembly systems. *Computers & Operations Research,* 30(5), 695-704.

Ho, Y. C., Eyler, M. A., & Chien, T. T. (1979). A gradient technique for general buffer storage design in a production line. *International Journal of Production Research,* 17(6), 557-580.

Hong, L. J., & Nelson, B. L. (2006). Discrete optimization via simulation using COMPASS. *Operations Research,* 54(1), 115-129.

Horng, S., & Lin, S. (2009). An ordinal optimization theory-based algorithm for a class of simulation optimization problems and application. *Expert Systems with Applications,* 36(5), 9340-9349.

Hu, A. B., & Meerkov, S. M. (2006). Lean buffering in serial production lines with bernoulli machines. *Mathematical Problems in Engineering,* 2006, 1-24.

Jacobson, S. H., & Schruben, L. W. (1989). Techniques for simulation response optimization. *Operations Research Letters,* 8(1), 1-9.

Jeong, K. C., & Kim, Y. D. (2000). Heuristics for selecting machines and determining buffer capacities in assembly systems. *Computers & Industrial Engineering,* 38(3), 341-360.

Kim, S., Park, Y., & Jun, C. (2006). Performance evaluation of re-entrant manufacturing system with production loss using mean value analysis. *Computers and Operations Research,* 33(5), 1308-1325.

Kim, S. (2006). Gradient-based simulation optimization. *Proceedings of the 2006 Winter Simulation Conference,* Monterey.

Kleijnen, J. P. C. (2008a). Design and analysis of simulation experiments, Springer Science + Business Media.

Kleijnen, J. P. C. (2008b). Response surface methodology for constrained simulation optimization: An overview. *Simulation Modelling Practice and Theory,* 16(1), 50-64.

Kleijnen, J. P. C., & Wan, J. (2007). Optimization of simulated systems: OptQuest and alternatives. *Simulation Modelling Practice and Theory,* 15(3), 354-362.

Koren, Y., Hu, S. J., & Weber, T. W. (1998). Impact of manufacturing system configuration on performance. *CIRP Annals - Manufacturing Technology,* 47(1), 369-372.

Kouikoglou, V. S. (2000). Optimal rate allocation in unreliable, assembly/disassembly production networks with blocking. *IEEE Transactions on Robotics and Automation,* 16(4), 429-434.

Kouikoglou, V. S. (2002). An efficient discrete event model of assembly/disassembly production networks. *International Journal of Production Research,* 40(17), 4485-4503.

Lane, M. S., Mansour, A. H., & Harpell, J. L. (1993). Operations research techniques: A longitudinal update 1973-88. *Interfaces,* 23(2), 63-68.

Law, A. M. (2003). How to conduct a successful simulation study. *Proceedings of the 2003 Winter Simulation Conference,* New Orleans.

Law, A. M. (2007). Simulation modeling and analysis, fourth ed., McGraw-Hill, Boston.

Li, J. (2004). Performance analysis of production systems with rework loops. *IIE Transactions,* 36(8), 755-765.

Li, J. (2005). Overlapping decomposition: A system-theoretic method for modeling and analysis of complex manufacturing systems. *IEEE Transactions on Automation Science and Engineering,* 2(1), 40-53.

Lin, C., Madu, C. N., & Kuei, C. (1994). Experimental design and regression analysis in simulation: An automated flowline case study. *Microelectronics and Reliability,* 34(5), 845-861.

Liu, Y., & Li, J. (2010). Split and merge production systems: Performance analysis and structural properties. *IIE Transactions,* 42(6), 422-434.

Liu, Z. (2002). Foreign direct investment and technology spillover: Evidence from china. *Journal of Comparative Economics,* 30(3), 579-602.

Lutz, C. M., Davis, K. R., & Sun, M. (1998). Determining buffer location and size in production lines using tabu search. *European Journal of Operational Research,* 106(2-3), 301-316.

Manitz, M. (2008). Queueing-model based analysis of assembly lines with finite buffers and general service times. *Computers & Operations Research,* 35(8), 2520-2536.

Masmoudi, F. (2006). Sizing manufacturing cell machines based on the simulation and an expert system. *International Journal of Simulation Modelling,* 5(2), 45-55.

Masmoudi, W., Chtourou, H., & Maalej, A. Y. (2007). Labor and machine sizing through a simulation-expert-system-based approach. *Simulation Modelling Practice and Theory,* 15(1), 98-110.

Mebrahtu, H., Walker, R., Dionysopoulos, T., & Mileham, T. (2009). Manufacturing performance optimization: The simulation-expert mechanism approach. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture,* 223(12), 1625-1634.

Meketon, M. S. (1987). Optimization in simulation: A survey of recent results. *Proceedings of the 1987 Winter Simulation Conference,* Buckhead.

Mohan, B. C., & Baskaran, R. (2012). A survey: Ant colony optimization based recent research and implementation on several engineering domain. *Expert Systems with Applications,* 39(4), 4618-4627.

Nahas, N., Nourelfath, M., & Ait-Kadi, D. (2009). Selecting machines and buffers in unreliable series-parallel production lines. *International Journal of Production Research,* 47(14), 3741-3774.

Nahas, N., Nourelfath, M., & Ait-Kadi, D. (2008). Ant colonies for structure optimisation in a failure prone series-parallel production system. *Journal of Quality in Maintenance Engineering,* 14(1), 7-33.

Nikkei 2013. "Panasonic may bring white goods production back to japan." *Nikkei* (June 4, 2013).

Nourelfath, M., Nahas, N., & Ati-Kadi, D. (2005). Optimal design of series production lines with unreliable machines and finite buffers. *Journal of Quality in Maintenance Engineering,* 11(2), 121-138.

Papadopoulos, C. T., O'Kelly, M. E. J., & Tsadiras, A. K. (2013). A DSS for the buffer allocation of production lines based on a comparative evaluation of a set of search algorithms. *International Journal of Production Research,* 51(14), 4175-4199.

Papadopoulos, H. T., & Heavey, C. (1996). Queueing theory in manufacturing systems analysis and design: A classification of models for production and transfer lines. *European Journal of Operational Research,* 92(1), 1-27.

Pichitlamken, J., Nelson, B. L., & Hong, L. J. (2006). A sequential procedure for neighborhood selection-of-the-best in optimization via simulation. *European Journal of Operational Research,* 173(1), 283-298.

Powell, S. G. (1994). Buffer allocation in unbalanced three-station serial lines. *International Journal of Production Research,* 32(9), 2201-2217.

Prudius, A. A., & Andradottir, S. (2012). Averaging frameworks for simulation optimization with applications to simulated annealing. *Naval Research Logistics,* 59(6), 411-429.

Qudeiri, J. A., Yamamoto, H., & Ramli, R. (2009). Buffer size decision for flexible transfer line with rework paths using genetic algorithm. *International Journal of Intelligent Systems Technologies and Applications,* 7(2), 227-240.

Qudeiri, J. A., Yamamoto, H., Ramli, R., & Jamali, A. (2008). Genetic algorithm for buffer size and work station capacity in serial-parallel production lines. *Artificial Life and Robotics,* 12(1-2), 102-106.

Rosen, S. L., & Harmonosky, C. M. (2005). An improved simulated annealing simulation optimization method for discrete parameter stochastic systems. *Computers and Operations Research,* 32(2), 343-358.

Sabuncuoglu, I., Erel, E., & Kok, A. G. (2002). Analysis of assembly systems for interdeparture time variability and throughput. *IIE Transactions,* 34(1), 23-40.

Sadr, J., & Malhame, R. P. (2004). Unreliable transfer lines: Decomposition/aggregation and optimization. *Annals of Operations Research,* 125(1-4), 167-190.

Safizadeh, M. H. (1990). Optimization in simulation: Current issues and the future outlook. *Naval Research Logistics,* 37(6), 807-825.

Saggi, K. (2002). Trade, foreign direct investment, and international technology transfer: A survey. *The World Bank Research Observer,* 17(2), 191-235.

Sargent, R. G. (2010). Verification and validation of simulation models. *Proceedings of the 2010 Winter Simulation Conference,* Baltimore.

Shaaban, S., & McNamara, T. (2009). The effects of joint operations time means and variability unbalance on production line performance. *International Journal of Manufacturing Technology and Management,* 18(1), 59-78.

Shi, C., & Gershwin, S. B. (2009). An efficient buffer design algorithm for production line profit maximization. *International Journal of Production Economics,* 122(2), 725-740.

Spearman, M. L., Woodruff, D. L., & Hopp, W. J. (1990). CONWIP: A pull alternative to kanban. *International Journal of Production Research,* 28(5), 879-894.

Spieckermann, S., Gutenschwager, K., Heinzel, H., & Voss, S. (2000). Simulation-based optimization in the automotive industry - A case study on body shop design. *Simulation,* 75(5-6), 276-286.

Swain, J. J. (2009). 2009 simulation software survey. *OR/MS Today, 36*, 50-61.

Swisher, J. R., Hyden, P. D., Jacobson, S. H., & Schruben, L. W. (2000). A survey of simulation optimization techniques and procedures. *Proceedings of the 2000 Winter Simulation Conference,* Orlando.

Swisher, J. R., Jacobson, S. H., & Yucesan, E. (2003). Discrete-event simulation optimization using ranking, selection, and multiple comparison procedures: A survey. *ACM Transactions on Modeling and Computer Simulation,* 13(2), 134-154.

Swisher, J. R., Hyden, P. D., Jacobson, S. H., & Schruben, L. W. (2004). A survey of recent advances in discrete input parameter discrete-event simulation optimization. *IIE Transactions,* 36(6), 591-600.

Tekin, E., & Sabuncuoglu, I. (2004). Simulation optimization: A comprehensive review on theory and applications. *IIE Transactions,* 36(11), 1067-1081.

Tempelmeier, H. (2003). Practical considerations in the optimization of flow production systems. *International Journal of Production Research,* 41(1), 149-70.

Vergara, H. A., & Kim, D. S. (2009). A new method for the placement of buffers in serial production lines. *International Journal of Production Research,* 47(16), 4437-4456.

Vitanov, I. V., Vitanov, V. I., & Harrison, D. K. (2009). Buffer capacity allocation using ant colony optimisation algorithm. *Proceedings of the 2009 Winter Simulation Conference,* Austin.

Vouros, G. A., & Papadopoulos, H. T. (1998). Buffer allocation in unreliable production lines using a knowledge based system. *Computers & Operations Research,* 25(12), 1055-1067.

Wang, Q., & Chatwin, C. R. (2005). Key issues and developments in modelling and simulation-based methodologies for manufacturing systems analysis, design and performance evaluation. *The International Journal of Advanced Manufacturing Technology,* 25(11-12), 1254-1265.

Xia, B., Xi, L., & Zhou, B. (2010). A method for performance evaluation of transfer line with unreliable machines and finite transfer-delay buffers. *Advanced Materials Research,* 97-101, 3135-3138.

Xu, M., Abdul-Kader, W., & Ganjavi, O. (2009). Cycle time estimation of a multiple product production line: An approach with fuzzy and chance-constrained programming. *Information Systems and Operational Research,* 47(2), 93-103.

Yamamoto, H., Qudeiri, J. A., & Marui, E. (2008). Definition of FTL with bypass lines and its simulator for buffer size decision. *International Journal of Production Economics,* 112(1), 18-25.