Tandem Queues Attended by a Moving Server

by:

Seyed Mohammad Reza Iravani

A thesis submitted in conformity with the requirements for the degree of Doctor of Philosophy
Graduate Department of Mechanical and Industrial Engineering
University of Toronto

©Copyright by Seyed Mohammad Reza Iravani. 1997
The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author’s permission.

L’auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L’auteur conserve la propriété du droit d’auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.
Tandem Queues Attended by a Moving Server

A thesis for the degree of Doctor of Philosophy

1997

Seyed Mohammad Reza Iravani
Department of Mechanical and Industrial Engineering
University of Toronto

Abstract

Optimization analysis of tandem queues attended by a moving server with holding and switching costs are the main concern in this research. First, as basic models, two-stage tandem queues attended by a moving server are analyzed to find the optimal policies in the first and the second stages in a two-stage tandem queue. It is shown that the optimal policy in the second stage which minimizes the total discounted and long-run average holding and switching costs is a greedy and exhaustive policy. Considering greedy and exhaustive policy for the second stage, three different service policies: (i) static, (ii) semi-dynamic (gated-limited) and (iii) dynamic (double-threshold) policies are defined for the first stage of a two-stage tandem queue. an $M/G_1 - G_2/1$ queue, to respectively deal with situations that (i) no information, (ii) partial information, or (iii) complete information about the number of customers in the system are available. To obtain the optimal static, gated-limited and double-threshold policies, three different models are developed and it is shown that the optimal gated-limited policy is almost independent of the arrival rate if switchover times are zero. For N-stage tandem queue $M/G_1 - G_2 - \cdots - G_N/1$, greedy and exhaustive policies downstream of stage 2 are defined and then three different models are developed to find the optimal static, gated-limited and double-threshold policies which minimize the total average holding and switching costs in these queues. Also, optimal policies in N-stage tandem queues with batch arrivals, $G^{(x)}/G_1 - G_2 - \cdots - G_N/1$ queues, are
studied through a new operational tool which reveals the properties of greedy and exhaustive policies and establishes two efficient and accurate heuristic algorithms to obtain the optimal policy. Finally, U-shaped lines which are actually tandem queues attended by moving servers are reviewed and a general definition and classification scheme are presented which organize the previous research and lead to the potential for further research on these lines. Then, by decomposing U-shaped lines into tandem queues attended by a moving server, the effects of switching costs and switchover times on some types of these lines are analyzed.
To my sisters, Parivash and Fatemeh,
and
to my brothers, Mahdi and Majid
Acknowledgement

My deepest gratitude is extended to Professor M.J.M. Posner and J.A. Buzacott who kindly agreed to be my supervisors. I am forever indebted to them for their insightful feedbacks, positive guidances and relentless support. It was and it will always be my honour to be their student.

I want to express my thanks to the members of my dissertation committee. Professor I. Duenyas, Professor A. Leon-Garcia, Professor D.L. Martell, Professor E.E. Pickett and Professor V. Makis for their insightful comments and feedback.

It is difficult to express in words my appreciation to my family for their constant love, support and encouragement. I am also forever indebted to all my teachers since my earliest school days until now, especially Dr. M.G. Aryanezhad who introduced me to queueing theory.
# Contents

1 Tandem Queues Attended by a Moving Server .................................................. 1
   1.1 Introduction ........................................................................................................ 1
   1.2 Applications ......................................................................................................... 3
       1.2.1 Robotic Systems ............................................................................................ 4
       1.2.2 Multi-Functional Machine Stochastic Scheduling .......................................... 6
       1.2.3 Transportation ............................................................................................... 6
       1.2.4 Computer and Communication Systems ....................................................... 7
   1.3 Service Policies ..................................................................................................... 8
   1.4 Background ........................................................................................................... 9
       1.4.1 Performance Analysis ..................................................................................... 10
       1.4.2 Optimization Analysis .................................................................................. 12
   1.5 The Basic Two-Stage Tandem Queue .................................................................. 15
       1.5.1 The Basic Two-Stage Tandem Queue or The BTQ Model ............................... 15
       1.5.2 The Basic Semi-Markov Decision Model or The BSD Model ...................... 16
   1.6 Motivation and Contribution ............................................................................... 17
   1.7 Scope of This Research ....................................................................................... 18

2 Optimal Policy in the Second Stage of a $G/G_1 - G_2/1$ Queue ......................... 22
   2.1 Introduction ........................................................................................................... 22
   2.2 Model and Problem Description .......................................................................... 22
   2.3 Optimality of Greedy Policies ............................................................................. 24
   2.4 Optimality of Exhaustive Policies ........................................................................ 28
3 An $M/G_1 - G_2/1$ Queue with Static Policy

3.1 Introduction .................................................. 34
3.2 Static Policy .................................................. 34
3.3 Model 1: An $M/G_1 - G_2/1$ queue .......................... 35
   3.3.1 Determination of $\Psi_{M_3}(x, y)$ ......................... 38
   3.3.2 Determination of $\Pi_2(x, 0)$ .......................... 40
   3.3.3 Determination of $\Pi_1(0, y)$, $\Theta_{M_{3-1}}(.)$ and $\pi^{(2)}_{00}$ .......................... 40
   3.3.4 Waiting times ............................................ 42
   3.3.5 Two-Stage Tandem Queue with Sequential Service Policy ... 44
   3.3.6 Average Number of Customers in the System ............. 45
   3.3.7 Optimal Static Policy ................................... 46
3.4 Model 2: An $M/G/1$ Queue with Cyclic Service Times ........ 48
   3.4.1 Background ............................................... 49
   3.4.2 Formulation .............................................. 50
   3.4.3 Probability Generating Function ........................ 52
   3.4.4 Average Number of Customers in the System ............. 57
   3.4.5 Applications ............................................. 60
   3.4.6 Approximation Model for $M/G^{(m)}/1$ .................... 67
   3.4.7 Numerical Examples ..................................... 68
   3.4.8 Optimal N-limited Strategy for a Single Vacation Queueing System .................................. 71

4 An $M/G_1 - G_2/1$ Queue with Semi-Dynamic Policy ......... 76

4.1 Introduction .................................................. 76
4.2 Semi-Dynamic Policies ....................................... 77
4.3 Optimal Semi-Dynamic Policy ................................ 78
4.4 An $M/G_1 - G_2/1$ with Gated-Limited Policy ............... 79
   4.4.1 Analysis of Steady State Probabilities ................... 80
   4.4.2 Probability Generating Functions ......................... 82
   4.4.3 Waiting Times ........................................... 89
4.4.4 Average Number of Customers in the System .......... 90
4.4.5 Average Number of Customers Served in a Cycle .......... 91
4.4.6 Optimal Gated-Limited Policy .......................... 93
4.5 System with Zero Switchover Times ........................ 94
4.5.1 Optimality Condition for Sequential Service Policy .......... 95
4.5.2 Sufficient Condition for Optimality of $M_2^* = 2$ .......... 99
4.5.3 Optimal Limits in High Traffic Intensity .................. 103
4.5.4 Numerical Study ........................................ 105

5 An $M/G_1 - G_2/1$ Queue with Dynamic Policy 111
5.1 Introduction .............................................. 111
5.2 Dynamic Policy ............................................ 111
5.3 Optimal Dynamic Policy ..................................... 112
5.4 An $M/G_1 - G_2/1$ with Double-Threshold Policy .......... 114
5.4.1 Analysis of Steady State Probabilities ..................... 114
5.4.2 Probability Generating Functions .......................... 115
5.4.3 Waiting times ............................................ 122
5.4.4 Average Number of Customers in the System ................. 124
5.4.5 Average Number of Customers Served in a Cycle .......... 125
5.5 Numerical Study ............................................ 126
5.5.1 Optimal Limited Policy and High Traffic Intensity .......... 126
5.5.2 Comparison of Optimal Policies in $M/G_1 - G_2/1$ .......... 129

6 Optimal Policies in $M/G_1 - G_2 - \cdots - G_N/1$ Queues 134
6.1 Introduction .............................................. 134
6.2 Problem Description ....................................... 134
6.3 Equivalent Two-Stage Tandem Queue (ETQ) .................... 137
6.4 Optimal Static Policy ..................................... 138
6.4.1 Average Holding Cost .................................... 138
6.4.2 Average Switching Cost .................................. 142
6.5 Optimal Semi-Dynamic Policy ................................ 142
6.5.1 Average Holding Cost ........................................ 143
6.5.2 Average Switching Costs .................................... 145
6.6 Optimal Dynamic Policy ......................................... 145
   6.6.1 Average Holding Costs ................................... 146
   6.6.2 Average Switching Cost .................................. 149
6.7 Numerical Study .................................................. 150

7 Optimal Policy in $G^{(x)}/G_1 - G_2 - \cdots - G_N/1$ Queues 153
   7.1 Introduction .................................................. 153
   7.2 Single Multi-Functional Machine Stochastic Scheduling or $G^{(x)}/G_1 - G_2 - \cdots - G_N/1$ 154
   7.3 Problem Formulation .......................................... 155
   7.4 Preliminaries .................................................. 159
   7.5 Basis of Heuristics ............................................ 167
   7.6 A.N.D Heuristic Algorithm .................................. 176
   7.7 SHORA Heuristic Algorithm .................................. 181
   7.8 Numerical Study ................................................ 186

8 U-Shaped Production Lines 190
   8.1 Introduction .................................................. 190
   8.2 U-Shaped Lines ................................................ 192
      8.2.1 Worker-Oriented U-Shaped Lines ......................... 193
      8.2.2 Machine-Oriented U-Shaped Lines ....................... 195
      8.2.3 Static Working Zones Vs Dynamic Working Zones ...... 196
      8.2.4 Sequenced Working Zones Vs Mixed Working Zones .... 197
      8.2.5 U-Shaped Lines with Multi-Functional Machines ....... 198
   8.3 Literature Review .............................................. 198
   8.4 U-Shaped Lines with Multi-Functional Machines and Switching Costs 203
   8.5 Worker-Oriented U-Shaped Lines with Sequenced Static Working Zones 206
   8.6 Bucket Brigades with Switching Costs ....................... 210
9 Summary

A Average Number of Customers in the $M/G_1 - G_2/1$ with Static Policy 225

A.1 Average Number of Customers in Stage 1 .................................. 225
   A.1.1 Determination of $\Psi_{M_2}^r(1.1)$ .................................. 226
   A.1.2 Determination of $\omega$ ............................................. 227
   A.1.3 Determination of $\Pi_{l_2}^r(1.0)$ .................................. 228

A.2 Average Number of Customers in Stage 2 .................................. 229
   A.2.1 Determination of $\Psi_{M_2}^r(x = 1, r = 1)$ ....................... 231
   A.2.2 Determination of $\delta$ ............................................. 231

B Average Number of Customers in the $M/G_1 - G_2/1$ with Gated-Limited Policy 232

B.1 Average Number of Customers in Stage 1 .................................. 232
   B.1.1 Determination of $\Pi_{l_2}^r(1.0.0)$ and $\Pi_{l_1}^r(1.0.1)$ ............ 233
   B.1.2 Determination of $\Delta$ ............................................. 235

B.2 Average Number of Customers in Stage 2 .................................. 236
   B.2.1 Determination of $\Pi_{l_1}^r(x = 1.0, x = 1)$ ....................... 237
   B.2.2 Determination of $\Upsilon$ ........................................... 238

C Average Number of Customers in the $M/G_1 - G_2/1$ with Double-Threshold Policy 240

C.1 Average Number of Customers in Stage 1 .................................. 240
   C.1.1 Determination of $\Psi_{M_n}^r(1.1)$ .................................. 241
   C.1.2 Determination of $\beta$ ............................................. 242
   C.1.3 Determination of $\Pi_{l_2}^r(1.0)$ .................................. 243

C.2 Average Number of Customers in Stage 2 .................................. 245
   C.2.1 Determination of $\Psi_{M_n}^r(x = 1, x = 1)$ ....................... 246
   C.2.2 Determination of $\xi$ ............................................. 247

Bibliography 248
List of Figures

1.1 A basic polling system................................. 2
1.2 N-stage tandem queue attended by a moving server........ 3
1.3 A robotic cell for drilling and trimming.................. 4
1.4 An automatic milling machine with a loading/unloading robot 5
1.5 A queue-cart model...................................... 7
1.6 Queueing model for call processing........................ 8
1.7 The BTQ model............................................ 16
1.8 The scope of this research............................... 19
2.1 Behaviour of π and γ in [τi−, τi+, S2 + I]..................... 25
2.2 Behaviour of π and γ in [τi−, τi+, I + D21]................. 27
2.3 Behaviour of γ and ξ in [τ21, τ3,12 + D12 + S2]............. 29
3.1 Interdeparture times of $M/G_1 - G_2/1$ and $M/G^{(M_S)}/1$ 66
4.1 Two-stage tandem queue with gated-limited policy........... 80
4.2 Behaviour of γ and π in $[τ_i, τ_{i+1}]$........................ 96
4.3 Sample path of cycles under policies π and β.............. 100
4.4 $E[TC_{M_G}]$ for different values of λ..................... 106
4.5 Optimal limit curves $L^G_ϕ$ and $L^G_1$ for gated-limited policy 108
5.1 An example of $L(x_1, x_2)$. ... and $L(x_1, x_2)$....... 113
5.2 Optimal limit curves $L^L_ϕ$ and $L^L_1$ for limited policy ($M_e = 1$) 128
6.1 An N-stage tandem queue attended by a moving server........ 135
6.2 Number of customers in stage 2 during a cycle when static policy with 139
  parameter $M_S = 4$ is applied......................... 139
7.1 N-stage tandem queue attended by a moving server with batch arrivals 155
7.2 Examples of decomposed tandem queues. ........................................ 160
7.3 The A.VD decomposition method for a 6-stage tandem queues. .......... 176
7.4 Shortest route problem for nodes 2 to \( N + 1 \). ............................ 182
7.5 Shortest route problem for example 7.12. ................................... 185
8.1 A U-shaped line with \( N = 12 \) work stations and \( M = 3 \) workers. ... 193
8.2 A U-shaped line with (a) Sequenced working zones. (b) Mixed working
zones.  ........................................................................................................ 197
8.3 The static working zone of worker \( m \) in a bucket brigade. ............... 212
8.3 Working zones of balanced bucket brigade in example 8.3. .............. 217
List of Tables


3.2 Optimal static policy for a system with exponential service times, deterministic switchover times and $h_1 = 1$. $h_2 = 2$. $K_{12} = 9$ and $K_{21} = 6$. 48

3.3 Comparison of Relative Error ($RE$) for $\theta_i^{(1)} = i$ and $\theta_i^{(2)} = m^{-1}$ in an $M/M^{(3)}/1$. ............................................ 70

3.4 Comparison of Relative Error ($RE$) for $\theta_i^{(1)} = i$ and $\theta_i^{(2)} = m^{-1}$ in an $M/E_k^{(3)}/1$. ............................................ 70

3.5 Comparison of Relative Error ($RE$) for $\theta_i^{(1)} = i$ and $\theta_i^{(2)} = m^{-1}$ in an $M/HE_2^{(3)}/1$. ............................................ 70

3.6 Optimal costs for different tool replacement policies. ................. 75

4.1 Parameters of problems 1 to 12. .................................. 94

4.2 Optimal gated-limited policy for problems 1 to 12. ................. 94

4.3 Characteristics of the $M/HE_2 - D/1$ when $\lambda = 2$ ($\rho = 0.3$). .......................... 105

4.4 Characteristics of the $M/HE_2 - D/1$ when $\lambda = 4$ ($\rho = 0.6$). .......................... 105

4.5 Characteristics of the $M/HE_2 - D/1$ when $\lambda = 6$ ($\rho = 0.9$). .......................... 105

4.6 $E[TC_{M0}]$ for $M/HE_2 - D/1$ queue with $h_1 = 16$. $h_2 = 20$ and $K_{12} = 10$. 106

4.7 Comparison of Relative Error $\delta$ for points $a$ and $b$ in Figure 4.5. .......................... 107

4.8 Comparison of Relative Error $\delta$ for points $c$ and $d$ in Figure 4.5. .......................... 107

4.9 Comparison of Relative Error $\delta$ for points $e$ and $f$ in Figure 4.5. .......................... 107

5.1 Characteristics of $M/HE_2 - D/1$ under limited policy. .................. 127

5.2 $E[TC_{(1,M0)}]$ for $M/HE_2 - D/1$ queue with $h_1 = 16$. $h_2 = 20$ and $K_{12} = 10$. .................. 127
Chapter 1

Tandem Queues Attended by a Moving Server

1.1 Introduction

Queueing theory has made significant progress during the last few decades. Two important reasons for its success are, (i) a solid mathematical foundation, and (ii) a positive response to the challenges posed by new applications. The use of efficient mathematical tools to analyze queues and also the productive interactions with the areas of production, communications and computer science have made queueing theory an important field of research in operations research and performance evaluation.

One of the good examples of this positive response to the encounter with new practical problems is polling systems. A polling system is a system of multiple queues, attended by a moving server which attends to the queues in some order. The term "polling" originates in polling data link control schemes. In such systems, each terminal is interrogated by a central computer on a multidrop communication line to determine if it has data to transmit. Then, the data is transmitted and the computer examines the next terminal. From a queueing system perspective, the server represents the computer and the queue corresponds to a terminal.

A basic polling system consists of $N$ unbounded queues and a single server which
serves them one at a time. The arrival process to queue $i$ ($i = 1, 2, \ldots, N$) is typically assumed to be an independent Poisson process with rate $\lambda_i$. The arriving customers at queue $i$ are called type $i$ customers and are assumed to have common random service time $S_i$. After being served at queue $i$, a type $i$ customer is assumed to leave the system (Figure 1.1).

Although the basic model is suitable for analyzing a large variety of systems, it may fail to model some important features of certain applications. For example, the assumption that customers must leave the system upon completion of service is a major limitation of the basic model. Consequently, in a basic polling system, simple features such as the generation of a job at one queue upon the completion of a job in another queue can not be modeled. Therefore, the basic polling system has been generalized by considering a feature called customer routing. In polling systems with customer routing, a type $i$ customer is either routed to queue $j$ ($1 \leq j \leq N$) with probability $p_{ij}$, or leaves the system with probability $p_{i0}$. A customer routed to queue $j$ becomes a type $j$ customer and then waits for service at queue $j$.

Considering $\lambda_i = 0$ for $i = 2, 3, \ldots, N$, $p_{N0} = 1$ and

$$p_{ij} = \begin{cases} 1 & ; \ i = 1, 2, \ldots, N - 1, \ j = i + 1 \\ 0 & ; \ \text{otherwise} \end{cases},$$

the polling system with this customer routing becomes a tandem queue attended by a moving server with arrivals to the first queue. Upon each service completion in queue $i$, the served customer is routed to queue $i + 1$, and finally leaves the system from

\begin{figure}
\centering
\includegraphics[width=\textwidth]{polling_system}
\caption{A basic polling system.}
\end{figure}
queue \( Q \) after receiving service there. We use notation \( G/G_1 - G_2 - \cdots - G_N/1 \) to address this type of \( N \)-stage tandem queue attended by a moving server with general interarrival and service times (Figure 1.2).

Sometimes servers need time to go from one queue to another. Suppose \( D_{ij} \) is the time required to move from queue \( i \) to queue \( j \) \((i \neq j)\) when the server finishes his work in queue \( i \) and decides to go to queue \( j \). The time \( D_{ij} \) is called the switchover time or walking time, and \( D_{ij} = 0 \) means that the server can switch from queue \( i \) to \( j \) immediately (after deciding to switch).

### 1.2 Applications

There are several practical applications of tandem queues served by moving servers: a labour and machine-limited production system [54], a repairable system with a repairman [67] and operating systems in computer and telephone switching systems [30, 28, 29, 49, 37, 22, 39]. In general, the applications of tandem queues served by moving servers can be classified as follows:

- Robotic Systems
- Stochastic Scheduling
- Transportation
- Computer and Communication systems

In this section each of the above applications is introduced and described using real world examples.
1.2.1 Robotic Systems

The application of tandem queues with moving servers in robotic systems can be described considering two cases (1) Robotic Cells and (2) Loading/Unloading Robots.

1. *Robotic Cells*: Consider a robotic cell which is designed to perform $N$ consecutive operations. Work pieces arrive according to a random process to the robotic cell and $N$ operations must be performed on each unit by the single robot. Each operation needs a particular type of tool and the single robot has the ability to change its tool automatically, taking a random amount of time. If we consider random times for each operation $i$ ($i = 1, 2, \ldots, N$), the system can be viewed as an $N$-stage tandem queue attended by a moving server (the robot) and each queue in stage $i$ ($i = 1, 2, \ldots, N$) comprises the units which wait for operation $i$. A specific example of manufacturing in this area can be considered in the aerospace industry in the process of drilling and trimming of complex contoured parts such as aircraft skins, fairing and doors (Nazemetz et al [53]). The advent of large precision robots and low cost micro-processors provided an opportunity to automate this aspect of aircraft manufacturing, (Figure 1.3). The robot is capable of moving from the drilling work station to the trimming work station, automatically changing tools and performing the programmed function neces-
sary to produce the part configuration. The cell was designed so the operator's function would be performed outside the range of the robot, on the cell perimeter, so as to eliminate the human/robot interface problem. The function of the operator is to set up the work, load/unload parts from work stations and also move the parts from one work station to another.

2. **Loading/Unloading Robots**: Consider a work station with a milling machine and a loading/unloading robot (Figure 1.4). Work pieces arrive to this work station according to some random process, and each requires \( N \) consecutive machining processes which take different random times. Considering random times for loading, unloading and tool changing, the system can be considered as a \( N \)-stage tandem queue with one moving server. The server is the milling machine and the service time of each unit in queue \( i \) (machining process \( i \)) is the summation of loading, unloading and machining times in that process. A typical example of loading/unloading robots can be found in Bollinger and Duffie [8], in flexible manufacturing cells which are used for machining large marine seals (Anon [2]) and in rail-cart-mounted smart robot systems (Talvage and Hannam [65]).
### 1.2.2 Multi-Functional Machine Stochastic Scheduling

Consider a single machine scheduling problem in which jobs arrive according to a random process. Each job requires $V$ different consecutive operations $1, 2, \ldots, V$. The machine is able to perform the $V$ different operations. Switching from one operation to another takes a random time for some adjustments, but once the machine is ready to perform operation $i$ ($i = 1, 2, \ldots, V$), it can do any number of that operation without additional delays. The times for performing operations $1, 2, \ldots, V$ are independent random variables. Considering holding costs for jobs waiting for operation $i$, and also a switching cost from operation $i$ to $j$, this problem can be considered as a single multi-functional machine stochastic scheduling problem with the goal being to minimize the total average cost. This model is actually an $V$-stage tandem queue attended by a moving server in which the server is the single machine and the customers are jobs waiting for operation $i$.

### 1.2.3 Transportation

Suppose units arrive at a loading facility in accordance with some renewal process. These units must be transported to some other place after loading onto a carrier having finite (or infinite) capacity. There is an unloading facility at the destination to handle the units from the carrier. Loading and unloading times for each unit are random variables. Also, traveling time from the loading facility to the unloading facility can be considered as a random variable. This transportation model can be formulated as a two-stage tandem queue attended by a moving server in which customers are the units, the loading and unloading facility are two stages and the carrier acts like the switching process from the first stage (loading) to the second stage (unloading), and vice versa.

Another transportation application can be found in industry, especially in material handling systems in which heavy units must be transferred between workshops using a carrier and lift-truck. The queue-cart problem is another well-known application of
tandem queues attended by a moving server in transportation (Figure 1.5). A queue-cart problem is a work station in which the server serves items that arrive at random and then await processing in a buffer of infinite capacity (buffer of stage 1). Served items enter a second buffer that in a queue-cart model is actually a cart (buffer of stage 2). From time to time, the server leaves the queue to deliver the cart to another location and empty it. The server might be a factory worker who both fills orders for the parts and delivers them to the next station. During the delivery time new items may arrive and join the queue. In the literature on production scheduling (e.g. see Coffman et al [13], Dobson et al [15], Karmarkar [26, 27]), the cart may take another form such as a pallet moved by a forklift.

1.2.4 **Computer and Communication Systems**

In recent years, many queueing models have been analyzed to evaluate system performance and to obtain dynamic characteristics of call processing programs in certain types of stored program controlled switching systems (Kuhn [40]). Call processing in an electronic switching system consists of the following three basic processing functions: (i) input processing, (ii) internal processing and (iii) output processing (Figure 1.6). Input processing performs scanning for recognition of input signals, and internal processing performs analysis of received digits as well as seizure of idle trunks and idle paths in a switching network. The output processing executes speed path orders to establish network path connections and performs read/write processing for
Figure 1.6. Queueing model for call processing (adapted from Katayama [30]).

the secondary memories.

To ensure continuity of service in a single processor system, such as the NO.1 ESS and D10 (Hawashima et al [22] and Katayama [30]) all three processing steps are performed by a single central processor. This system can be modeled as a queueing system as follows: the processor control point is actually a moving server which makes the round of each processing program in accordance with a moving rule (switching rule) and performs call servicing in some queues in tandem. This queueing model is represented as a tandem queue attended by a moving server. Different kinds of switching policies were analyzed for call processing systems in Katayama [28, 29, 31].

1.3 Service Policies

Since there is only one server in charge of serving all queues, he may attend to the queues according to some specific service policy. A service policy is a rule which describes when the server must switch from one queue to another queue. Four basic service policies are defined for polling systems as well as for tandem queues attended by a moving server. These policies are

- **Exhaustive Policy**: When the server visits a queue, he serves its customers until it becomes empty, whereupon he then switches to the other queue.
- **Gated Policy**: When the server switches to a queue, he only serves customers who were in that queue at his arrival epoch.

- **Semi-exhaustive**: When the server visits a queue, he continues serving customers until the number present is one less than the number found upon his arrival at that queue. This is also called a *decrementing policy*.

- **Limited Policy**: When the server visits a queue, he continues serving until this queue becomes either empty or at most a predetermined number of customers are served, whichever occurs first.

Recently, new policies have been introduced for polling systems. These policies are constructed by combining the four basic policies or by adding further features to some of those. For example, the limit of the limited policy can be probabilistically chosen after each service completion. In *Bernoulli* policies, the distribution of the limit is geometric with parameter $0 \leq p \leq 1$ (see Keilson and Servi [36] and Servi [60]). Therefore, in a Bernoulli-exhaustive policy, after each service completion, the server decides either to serve the next customer in that queue with probability $p$ or to switch to the other queue with probability $(1 - p)$, continuing until that queue becomes empty. The Bernoulli-gated policy is similar except that the server does not serve more than the number of customers he found in the queue upon his arrival at that queue. Another probabilistic policy is the *binomial-gated* policy (see Levi [41, 42]). In this case, the number of served customers is distributed according to a binomial distribution with parameter $0 \leq p \leq 1$ and $X$, where $X$ is the number of customers present at a queue when the server arrives there.

### 1.4 Background

In this section we review the literature on tandem queues attended by a moving server. These studies can be divided into two groups: *(i)* research that deals with the *performance analysis* of different characteristics of tandem queues attended by a moving server such as waiting times, cycle times, number of customers in the system.
etc. and (ii) research which focuses on the optimization analysis of these types of systems.

1.4.1 Performance Analysis

The first paper in tandem queue attended by a moving server was presented in 1970 by Nair [50]. He considered an $M/G_1 - G_2/1$ model with zero-switchover times and exhaustive policies in both stages. By using an imbedded semi-Markov sequence, he performed a transient analysis and studied the distribution of the queue length and virtual waiting times. He also analyzed the convergence and the limiting properties of the transition probabilities of the semi-Markov sequence. The same model, but with a limited policy, was also studied by Nair [51]. He analyzed the distribution of the busy period, virtual waiting times and queue length and their limiting behaviour. In a similar work, the steady state analysis of an $M/G_1 - G_2/1$ queue with exhaustive service policy was performed by Taube-Netto [67]. He obtained the steady state probabilities and the Laplace-Stieltjes transforms of the waiting times, the busy period of the server and the busy period of each stage. Katayama [29] considered the same model but with nonzero switchover times and derived the generating function of the steady state probabilities and Laplace-Stieltjes transforms of the waiting times and cycle times.

An $M/G_1 - G_2/1$ model with zero switchover times and gated policy was analyzed by Katayama [30]. He found the generating function of the steady state probabilities, and the distribution of the sojourn time and the cycle time. Using some numerical results, he concluded that it is possible to shorten the mean total sojourn time by applying a gated policy instead of an exhaustive policy. He also performed the same analysis for the $M/G_1 - G_2/1$ model with nonzero switchover times and a gated policy [28], nonzero switchover times and a limited policy [31], zero switchover times and a semi-exhaustive policy [34] and zero switchover times and a break-in policy [32]. In the break-in policy, the server applies an exhaustive policy at the first stage, and then switches to the second stage: if a customer arrives at stage 1 while the server is busy at stage 2, the server completes the service at hand and then switches back to
stage 1. This service policy is also called a nonpreemptive priority discipline in stage 1.

The particular case of a two-stage tandem queue within the context of a queue-cart problem with deterministic interarrival and service times was analyzed in Coffman et al [14], Dobson et al [15] and Karmarkar [26, 27]. The more complete case was considered by Coffman and Gilbert [13] with the following cart delivery strategy: delivery begins when $K$ units are in the cart or when the queue is empty and at least $M$ units are in the cart ($M \leq K$). Strategies with $K = M$ are called full cart strategies. because, with the cart holding $K$ units, the server delivers the cart only when it is full. $K = \infty$ and $M = 0$ are called empty queue and never idle strategies, because in first case the cart leaves after it has acquired $M$ items and the queue empties. and in second case the server always takes the cart away instead of waiting idle for the next arrival to an empty queue. Coffman and Gilbert [13] derived the generating function for the numbers in the queue, cart and system for the three strategies $K = M$, $K = \infty$ and $M = 0$.

There are only a few studies treating the multi-stage tandem queue attended by a moving server. Three-stage tandem queues were studied by Murakami et al [49]. Katayama [33] analyzed an $N$-stage tandem queue with zero switchover times. He found an explicit expression for the mean sojourn times in each stage of an $M/G_1 - G_2 - \cdots - G_N / 1$ model when exhaustive, limited or gated service policies are applied. He also obtained the upper and lower bounds for the mean sojourn time when the server implements a semi-exhaustive policy or a preemptive priority policy. In a preemptive priority policy, customers in queue $i$ have priority over customers in queue $i + 1$, $(i = 1, 2, \ldots, N - 1)$. If a customer arrives at queue $i$ when the server is at queue $j$ $(j > i)$, he interrupts the current service and immediately the server starts processing customers in queue $i$. The service of the customer in queue $j$ is resumed when there are no customers in queues 1, 2, \ldots, $j - 1$. König and Schmidt [38] studied the $N$-stage tandem queue with zero switchover times in which customers can either depart the system after their service completion in stage $i$ or go to stage $i + 1$. Considering a general switching rules, they focused on the relationship between
time-stationary and customer-stationary queue length characteristics.

1.4.2 Optimization Analysis

From an optimization point of view, there are also a few studies in tandem queueing systems attended by a moving server. However, similar types of problems were analyzed in terms of polling systems in which the serial routing behaviour of customers was not considered. Although these two types of problems are not exactly the same, the approaches for finding an optimal policy could be the same.

Optimization of polling system can be considered through two different approaches: (i) Controlling the order by which the server visits the queues, and (ii) Controlling how many customers must be served at a given queue (controlling the service policy).

Related to controlling the visit order, Boxma et al [9] introduced the square root rule for minimizing the mean waiting time in the system. According to this rule, the order in which the visit frequency of a station is selected is in proportion to the square root of its utilization ($\rho_i$) minimizes the mean waiting time of a customer in the system. For systems with zero switchover times it has been shown that the weighted sum of mean waiting times in each stage can be minimized using the $c\mu$ rule (cf. Warland [70]). In the $c\mu$ rule, the queues are ranked according to their ratio $c_i\mu_i = c_i/E[S_i]$ ($c_i$'s are weights) from the highest ratio to the lowest ratio and the server always serves a customer from the highest ranked queue which is nonempty.

A simple and optimal rule for selecting the visit order when the goal is to minimize the mean duration of the next visit cycle was provided by Browne and Yechiali [10]. Using Markov decision processes, they determined a semi-dynamic policy in which the server, at the beginning of a cycle, chooses a visiting order of the queues for this cycle that minimizes the mean duration of the cycle. The solution to this problem was surprisingly simple: the mean cycle time following the instant at which there are $(k_1, k_2, \ldots, k_N)$ customers in the system when the server leaves a queue is minimized if the server visits queues according to increasing values of $k_i/\lambda_i$. A similar approach is presented in Browne and Yechiali [11] for queues with unit buffers with losses of customers. They showed that an index policy can minimize the sum of holding
costs and customer loss costs over a cycle. Liu and Nain [45] identified the optimal visit order for a particular polling system (videotex system) with switchover times. Based on the amount of information available to the server, they obtain the optimal scheduling policy under fairly general assumptions. Towsley [66] proved that in a polling system with finite equal size buffers and negligible switchover times, the policy that serves the queues with the largest queue length first minimizes the number of customers that are lost.

Controlling the service policy in polling systems has a lot in common with optimization problems in tandem queues with a moving server. One of the first attempts was made by Hofri and Ross [23]. They proved that for a symmetric two-queue polling system but with different arrival rates, the policy that minimizes the sum of discounted holding and switching costs is exhaustive service in a nonempty queue. They also conjectured that the optimal policy is the threshold-type policy for switching from an empty queue. In Levy et al [43], it was proved that when nonidling policies are applied, the exhaustive policy dominates all other nonidling service policies in a system with zero switchover times based on minimizing the total amount of work found in the system at any time. Rajan and Agrawal [58] analyzed a system of $N$ symmetric queues with switching costs. They concentrated on nonidling policies and defined optimality in terms of a stochastic dominance of the holding and switching costs. They showed that the optimal policy is an exhaustive policy, and the server must switch to the longest available queue at each exhaustion epoch. A similar problem with switchover times under the objective of stochastically minimizing either the total unfinished work or the total number of jobs in the system was studied by Liu, Nain and Towsley [44]. They characterized the optimal policies under different information patterns such as complete, partial, periodic, nearest neighbor or delayed. They determined the conditions under which optimal policies can be exhaustive, nonidling, greedy, patient (the server stays at the last visited queue when the system is empty) or impatient (the server leaves the queue as soon as it is empty). It should be noted that most of their analyses are based on the assumption that the polling system is symmetric. Duenyas and Van Oyen [16] studied a similar problem but with
switching costs and zero switchover times. They partially characterized the optimal policy and developed a simple heuristic scheduling policy which minimizes the total expected cost.

In contrast to polling systems, there is no complete work in optimization analysis of tandem queues attended by a moving server. Katayama [35] considered a two-stage tandem queue attended by a moving server with zero switchover times in which in the second stage there are $n$ types of service facilities demanded by $n$ types of arriving customers. Considering an exhaustive policy for the first stage, he showed that in the second stage the well-known $c\mu$ rule minimizes the total waiting time of customers. The problem of scheduling a single server in a forest network of $N$ queues with $m$ jobs (without arrivals) subject to switching and holding costs was analyzed by Van Oyen and Teneketsis [69]. In their model, a job completed in queue $n$ either leaves the system or reenters the system at queue $I(n)$, where $I(n) \in \{1, 2, \ldots, n-1, n+1, \ldots, N\}$. Using reward rate notions they derived conditions on the holding costs and service times for which an exhaustive policy is optimal. Johri and Katehakis [25] considered an $N$-stage tandem queue attended by a moving server with zero switchover times. For Poisson arrivals to the first stage and independent exponential service times in each stage, they showed that the policy which always assigns the server to the non-empty queue closest to the exit stochastically minimizes the number of customers in the system. Duenyas et al [17] considered the same model but with general service and setup times and holding costs included. They developed an exact analysis of exhaustive and gated policies and assuming that holding costs in the stages are non-decreasing functions of the stage number, they introduced a heuristic algorithm to find a suboptimal policy which minimizes the average holding cost.

A new type of production line called the U-shaped line has recently been designed in which the number of workers is less than the number of work stations. These lines have become very popular because they are highly flexible in terms of their production rates. The production rate of a U-shaped line can be increased or decreased by changing the number of workers in the line. U-shaped production lines with $M$

\footnote{U-shaped lines are introduced in Chapter 8.}
workers and \( N (N > M) \) work stations are actually \( N \) multi-stage tandem queues each attended by a moving server. The behaviour of each tandem queue and the interactions among them determine the behaviour of the line. Therefore, the first step to analyze the behaviour of U-shaped lines is to study the behaviour of tandem queues attended by a moving server. This fact, along with the other applications of tandem queues attended by a moving server in manufacturing and communications as well as the lack of literature on optimization analysis of these type of queues, imply the need for more studies in the theory and optimization analysis of these queues and their new applications in manufacturing. Therefore, in this research, tandem queues attended by a moving server are studied as the first step towards the analysis of more complex systems which are combinations of these basic models.

1.5 The Basic Two-Stage Tandem Queue

In this section we introduce the basic two-stage tandem queue attended by a moving server, which will be used in Chapters 2 to 6 as the basis of the performance analysis. The semi-Markov decision model for the two-stage tandem queue is also presented for better demonstration of the optimization problem and will be used to establish new policies.

1.5.1 The Basic Two-Stage Tandem Queue or The BTQ Model

The basic two-stage tandem queue, or the BTQ, is actually a system of two queues in tandem in which only one server is assigned responsibility to serve customers in both queues (Figure 1.7). In this \( M/G_1 - G_2/1 \) model:

- Customers arrive according to a homogeneous Poisson process with rate \( \lambda \) to stage 1.

- The buffers or waiting spaces of stages 1 and 2 have infinite capacities.

- Service times \( S_i \) in stage \( i \) (\( i = 1, 2 \)) are independent random variables with distribution functions \( F_i(.) \) and Laplace-Stieltjes transforms \( F_i^*(.) \).
- The switchover times from stage 1 to 2 (or 2 to 1) are also independent random variables $D_{12}$ (or $D_{21}$) with distribution function $B_{12}(\cdot)$ (or $B_{21}(\cdot)$) and Laplace-Stieltjes transform $B_{12}^*(\cdot)$ (or $B_{21}^*(\cdot)$).

- There is a fixed switching cost $K_{12}$ ($K_{21}$) which is charged at the start of switchover time $D_{12}$ ($D_{21}$).

- the holding cost rate for each customer present in stage $i$ ($i = 1, 2$) per unit time is $h_i$.

Remark 1.1

Considering the special case of $S_2 = 0$, the model becomes a two-stage tandem queue with bulk service in the second stage (Queue-Cart problem).

1.5.2 The Basic Semi-Markov Decision Model or The BSD Model

Since the server is responsible for both stages 1 and 2, therefore, upon any changes in the system such as a new arrival at the first stage or a service completion, he must decide about his next action in order to minimize the total holding and switching cost. Considering nonpreemptive policies which imply that once service of a customer has started it cannot be interrupted by switching or serving other customers, this problem can be formulated as a semi-Markov decision process in which:

- *Decision epochs* are service completion times, switch completion times and arrival instants when the server is idle at one of the queues.
The state of the system at any decision epoch consists of the location of the server \( y \), the number of customers in the first stage \( x_1 \) and the number of customers in the second stage \( x_2 \): therefore, the state space of the system is
\[
SS = \{ (y, x_1, x_2) | y = 1, 2, \quad x_1, x_2 \in \mathbb{Z}^+ \}
\]
where \( \mathbb{Z}^+ \) is the set of nonnegative integers.

At each decision epoch, the server decides one of the followings: serve the next customer \( (S_r.) \), switch to the other stage \( (S_w.) \) or be idle \( (I_d.) \). Thus, the action space is \( A_c. = \{ S_r., S_w., I_d. \} \).

Based on the properties of the optimal policy in the second stage, the capabilities of the server and availability of information about the number of customers in the first stage, the basic semi-Markov decision process, or BSD, will be revised and analyzed in Chapters 3 to 5.

### 1.6 Motivation and Contribution

This research develops analytical models which can be used in performance and optimization analysis of tandem queues attended by moving servers. The motivation for developing these models stems from the lack of optimization models in dealing with these types of systems on the one hand, and the capability of these models in analysis of new ideas\(^2\) in production and communication systems, on the other hand. The contribution of this research in optimization of queueing systems can be summarized as follows:

- Analysis and optimization of \( M/G^{(m)}/1 \) queues with cyclic service times, which has a wide range of applications in queues with vacations, tool replacement problems and tandem queues attended by a moving server.

- Optimization analysis of tandem queues attended by a moving server when switching costs and switchover times are considered. This procedure consists of:

\(^2\)Such as U-shaped production lines, see Chapter 8.
1. Obtaining some properties of optimal policies.

2. Introducing new policies such as static, semi-dynamic (gated-limited) and dynamic (double-threshold) policies, based on the properties of the optimal policy, capabilities of the server and uncertainty.

3. Developing exact models for characteristics and optimization analysis of these policies.

- A new operational tool is developed which can be used in optimization analysis of connected queues with moving servers (polling systems, tandem queues attended by moving servers).

- Some properties of greedy and exhaustive policies are obtained and used to construct two efficient and accurate heuristic algorithms to find the optimal policy in a tandem queue with batch arrivals.

- U-shaped lines are actually made up of a number of tandem queues attended by moving servers. They are classified, and by decomposing these lines into tandem queues each attended by a moving server, the effects of switching costs and switchover times on different policies are studied.

### 1.7 Scope of This Research

Tandem queues attended by a moving server is the main concern of this research and it consists of 8 chapters. Figure 1.8 shows the interrelationships among these chapters. Tandem queues attended by a moving server are introduced in Chapter 1 as a type of polling system with deterministic customer routing. The literature on performance and optimization analysis of tandem queues attended by a moving server, along with the motivation and contribution of this research, are also described in this chapter. Furthermore, the basic two-stage tandem queue and the basic semi-Markov decision model are introduced in Chapter 1, which form the framework of the analysis in Chapters 2 to 5.
Figure 1.8. The scope of this research.
Chapter 2 is devoted to finding the optimal policy in the second stage which minimizes the total long-run average and discounted cost in a two-stage tandem queue attended by a moving server. It is shown that the optimal policy in stage 2 of a two-stage tandem queue is a greedy and exhaustive policy.

Considering a greedy and exhaustive service policy in the second stage, we then focus on the optimization analysis of service policies in the first stage. Based on the capabilities of the server and uncertainty, three different decision environments and, therefore, three different policies are introduced in Chapters 3, 4 and 5. The first decision environment deals with the situation that no information about the number of customers in the first stage is available or the server is not able to adapt to different states of the system. This usually occurs when the server is a machine and either the machine cannot be adjusted for different capacities or the adjustment costs are too high. Therefore, the machine is usually adjusted to complete a fixed number of units in each cycle. This policy is called a static policy and the optimal static policy is obtained in Chapter 3.

For the case that the server has partial information about the number of customers in the first stage, the semi-dynamic policy is introduced in Chapter 4. When partial information is available, the server knows the number of customers waiting in stage 1, but only when he actually arrives there. Thus, to avoid idling in stage 1, he must choose some customers from among those present to serve upon his arrival. In Chapter 4, optimal gated-limited policies (a class of semi-dynamic policies) are found in order to minimize the total long-run average holding and switching costs per unit time.

In Chapter 5 a dynamic policy is introduced for the BTQ model as an applicable policy for the case that the server has complete information about the number of customers in the system after each service completion. Double-threshold policies are analyzed in this chapter to approximate the optimal dynamic policies that minimize the total long-run average holding and switching cost per unit time. Finally, using some numerical results, static, limited, gated-limited and double-threshold policies are compared at the end of the chapter.
Chapter 6 focuses on the $N$-stage tandem queue attended by a moving server. Greedy and exhaustive service policies downstream of stage 2 are introduced for stages 2 to $N$, and then three different models are developed to find the optimal static, gated-limited and double-threshold policies which minimize the total long-run average holding and switching costs per unit time.

An $N$-stage tandem queue attended by a moving server with batch arrivals to the first stage is analyzed in Chapter 7. By focusing on the class of greedy and exhaustive policies, a new operational tool is developed which leads to the properties of the optimal greedy and exhaustive policy. Two heuristic algorithms are presented in this chapter to approximate the optimal policy which minimizes the total long-run average holding and switching costs.

Chapter 8 is devoted to U-shaped production lines which are tandem queues attended by moving servers. In this chapter a general definition for U-shaped lines is presented and different types of U-shaped lines are introduced. Then the literature on U-shaped lines is studied, and finally, by decomposing these lines into tandem queues attended by a moving server, the influences of switching costs and switchover times on some types of these lines are presented.

Finally, in Chapter 9, the research is summarized and the areas for further research are discussed.
Chapter 2

Optimal Policy in the Second Stage of a $G/G_1 - G_2/1$ Queue

2.1 Introduction

The optimal policy in a two-stage tandem queue attended by a moving server consists of the optimal policy in the first stage and the second stage. In this chapter we concentrate on the optimal policy in the second stage of a two-stage tandem queue attended by a moving server with general interarrival, service and switchover times. Two different holding costs in stages 1 and 2, and two different switching costs from one stage to the other are assumed and the total discounted and average holding and switching costs are considered as the criteria in the optimization analysis of this system. It is shown that the optimal policy in the second stage is greedy; and if the holding cost rate in the second stage is greater or equal to the rate in the first stage, then the optimal policy in the second stage is also exhaustive.

2.2 Model and Problem Description

Suppose that in the BTQ model, customers arrive according to a stochastic process $AR$ with rate $\lambda$ which is independent of the state of the system (a $G/G_1 - G_2/1$ model), and let $X_i^{r,n}(t)$ be the number of customers in queue $i$ ($i = 1, 2$) at time
when admissible policy $\pi \in \Pi$ is applied and the initial state is $z_0 \in S$. The set of admissible policies $\Pi$ consists of nonpreemptive policies which imply that once service on a customer has started, it cannot be interrupted by switching or serving other customers. Also, let $Q^{\pi, z_0}_{12}(t)$ and $Q^{\pi, z_0}_{21}(t)$ be the number of switches from stage 1 to 2 and 2 to 1, respectively, up to time $t$ under policy $\pi$ and initial state $z_0$. Then the expected total discounted holding and switching cost $V^*_\pi(z_0)$ is given by

$$V^*_\pi(z_0) = E[\int_0^\infty e^{-zt}(h_1X^{\pi, z_0}_1(t)+h_2X^{\pi, z_0}_2(t))dt + \int_0^\infty e^{-zt}(K_{12}dQ^{\pi, z_0}_{12}(t)+K_{21}dQ^{\pi, z_0}_{21}(t))]$$

where $\beta$ is the discount factor. The corresponding total long-run average holding and switching cost $\bar{V}^*_\pi(z_0)$ is given by

$$\bar{V}^*_\pi(z_0) = \lim_{t \to \infty} \frac{1}{t} E[\int_0^t (h_1X^{\pi, z_0}_1(u)+h_2X^{\pi, z_0}_2(u))du + K_{12}Q^{\pi, z_0}_{12}(t)+K_{21}Q^{\pi, z_0}_{21}(t)] \tag{2.1}$$

The optimal discounted cost policy $\pi^*_\pi$ and the optimal long-run average cost policy $\pi^*$ are policies that minimize $V^*_\pi(z_0)$ and $\bar{V}^*_\pi(z_0)$, respectively.

Instead of optimality criteria (2.1) and (2.2), other equivalent criteria which are more flexible to analyze can be considered. Suppose that $AR^{\pi, z_0}_i(t)$ and $DP^{\pi, z_0}_i(t)$ are the number of arrivals to, and departures from stage $i$ $(i = 1, 2)$ up to time $t$, respectively, under policy $\pi$ and initial state $z_0 = (y^0, x^0_1, x^0_2)$. Then,

$$X^{\pi, z_0}_1(t) = x^0_1 + AR^{\pi, z_0}_1(t) - DP^{\pi, z_0}_1(t) \tag{2.3}$$

$$X^{\pi, z_0}_2(t) = x^0_2 + DP^{\pi, z_0}_2(t) - DP^{\pi, z_0}_2(t) \tag{2.4}$$

Substituting (2.3) and (2.4) into (2.1), we have

$$V^*_\pi(z_0) = E[\frac{(h_1x^0_1+h_2x^0_2)}{\beta} + h_1\int_0^\infty e^{-zt}AR^{\pi, z_0}_1(t)dt - h_2\int_0^\infty e^{-zt}DP^{\pi, z_0}_1(t)dt$$

$$+ (h_2-h_1)\int_0^\infty e^{-zt}DP^{\pi, z_0}_2(t)dt + \int_0^\infty e^{-zt}(K_{12}dQ^{\pi, z_0}_{12}(t)+K_{21}dQ^{\pi, z_0}_{21}(t))]$$

Therefore,

$$V^*_\pi(z_0) = \frac{h_1x^0_1+h_2x^0_2}{\beta} - E[ h_2\int_0^\infty e^{-zt}DP^{\pi, z_0}_2(t)dt - h_1\int_0^\infty e^{-zt}AR^{\pi, z_0}_1(t)dt$$

$$-(h_2-h_1)\int_0^\infty e^{-zt}DP^{\pi, z_0}_2(t)dt - \int_0^\infty e^{-zt}(K_{12}dQ^{\pi, z_0}_{12}(t)+K_{21}dQ^{\pi, z_0}_{21}(t))]$$
and the minimization of $V_\pi(z_0)$ is equivalent to maximization of $U_\pi(z_0)$ where.

\[
U_\pi(z_0) = E\left[ h_2 \int_0^\infty e^{-\lambda t} DP_2(x_2) dt - h_1 \int_0^\infty e^{-\lambda t} AR_1(x_0) dt \\
- (h_2 - h_1) \int_0^\infty e^{-\lambda t} DP_1(x_0) dt \\
- \int_0^\infty e^{-\lambda t} (K_{12} Q_{12} + K_{21} Q_{21}) dt \right]. \tag{2.5}
\]

Also using (2.3) and (2.4) in (2.2), we have.

\[
V_\pi(z_0) = (h_1 x_1^0 + h_2 x_2^0) - \lim_{t \to \infty} \frac{1}{t} E\left[ h_2 \int_0^t DP_2(u) du - h_1 \int_0^t AR_1(u) du \\
- (h_2 - h_1) \int_0^t DP_1(u) du - K_{12} Q_{12} - K_{21} Q_{21} \right]
\]

and minimization of $V_\pi$ is equivalent to maximization of $U_\pi$, where.

\[
U_\pi(z_0) = \lim_{t \to \infty} \frac{1}{t} E\left[ h_2 \int_0^t DP_2(u) du - h_1 \int_0^t AR_1(u) du \\
- (h_2 - h_1) \int_0^t DP_1(u) du - K_{12} Q_{12} - K_{21} Q_{21} \right]. \tag{2.6}
\]

As mentioned, the optimal policy for the system consisting of a two-stage tandem queue attended by a moving server consists of the optimal policies in stage 1 and stage 2. In the next sections we focus on the optimal policy in the second stage. In other words, considering the BSD model we are looking for the optimal decisions in stages $(2, x_1, x_2)$ where the server has just switched to the second stage or completed a service in that stage.

### 2.3 Optimality of Greedy Policies

In this section, we compare the decisions between "serving" and "idling" when the server is at nonempty stage 2. It will be shown that any admissible policy in the second stage can be improved by a greedy policy. Consider admissible policy $\pi \in \Pi$ then.

**Definition 2.1.** Policy $\pi$ is said to be a greedy policy if the server never idles at a nonempty queue. Let $\Gamma_2 \subset \Pi$ be the set of all policies that are greedy in stage 2.
Definition 2.2. Policy \( \pi \) is said to be a *non-breaking* policy if the server never idles between two consecutive services at a nonempty queue. Let \( \Gamma_2^B \subset \Pi \) be the set of all policies that are non-breaking in stage 2.

Since non-breaking policies may have an idle time after serving the last customer in the second stage, therefore \( \Gamma_2 \subset \Gamma_2^B \).

Proposition 2.1

For any policy \( \pi \in \Pi \), there exists a policy \( \gamma^* \in \Gamma_2 \) and \( \gamma^*_B \in \Gamma_2^B \) such that:

\[
U_{\gamma^*_B}(z_0) \geq U_\pi(z_0) \tag{2.7}
\]

\[
\bar{U}_{\gamma^*}(z_0) \geq \bar{U}_\pi(z_0) \tag{2.8}
\]

Proof:

Consider \( \tau^*_I \) as the first time under policy \( \pi \in \Pi \) that the server starts idling in nonempty stage 2. Consider state \( z_{\tau_I} = (2, x_{1}', x_{2}') \) at time \( \tau^*_I \) (where \( x_{2}' > 0 \)); then after some idle time \( I \), the server may decide to (i) serve the next customer in stage 2, or (ii) switch to the first stage.

(i) In the case that after idle time \( I \), the server decides to serve another customer in stage 2 with service time \( S_2^1 \), a new service policy \( \gamma \) can be constructed as follows (Figure 2.1):

![Diagram of service policies](https://example.com/diagram.png)

**Figure 2.1.** Behaviour of \( \pi \) and \( \gamma \) in \( [\tau^*_I, \tau^*_I + S_2^1 + I] \)
• $\gamma$ follows $\pi$ in $[0, \tau^*_f)$

• $\gamma$ serves a customer in stage 2 then idles the same as $\pi$

• $\gamma$ follows $\pi$ in $[\tau^*_f + S^1_1 + I, \infty)$

Since $\gamma$ behaves like $\pi$ in $[0, \tau^*_f)$, the problem can be analyzed by setting $\tau^*_f = 0$, so that $z_0 = (2. x_1^{T'}, x_2^{T'})$. Thus, for all $t \geq 0$.

$$AR^T_{\gamma}(t) = AR^T_{\pi}(t) \tag{2.9}$$

$$DP^T_{\gamma}(t) = DP^T_{\pi}(t) \tag{2.10}$$

$$Q^T_{12}(t) = Q^T_{12}(t) \tag{2.11}$$

$$Q^T_{21}(t) = Q^T_{21}(t) \tag{2.12}$$

and

$$DP^T_{2}(t) = \begin{cases} 
DP^T_{1}(t) & : 0 \leq t < S^1_2 \\
DP^T_{2}(t) + 1 & : S^1_2 \leq t < S^1_2 + I \\
DP^T_{2}(t) & : S^1_2 + I \leq t.
\end{cases} \tag{2.13}$$

Therefore, considering total discounted holding and switching costs, and using (2.9) to (2.13), we get

$$U_\gamma(z_0) - U_\pi(z_0) = E[h_2 \int_0^{\infty} e^{-3t} (DP^T_{2}(t) - DP^T_{2}(t)) dt]$$

$$= E[h_2 \int_{S^1_2}^{S^1_2 + I} e^{-3t} dt]$$

$$= \frac{h_2}{3} E[e^{-3S^1_2} (1 - e^{-3I})]$$

$$= \frac{h_2}{3} F_2(\beta) E[1 - e^{-3I}] . \tag{2.14}$$

Since idle time $I \geq 0$, it follows that (2.14) is nonnegative. In other words,

$$U_\gamma(z_0) \geq U_\pi(z_0) \quad \forall \ z_0 .$$

It can be simply shown that in terms of long-run average switching and holding costs policies $\pi$ and $\gamma$ behave the same. Therefore, policy $\gamma$, which is non-breaking up to time $\tau^*_f$, improved the total discounted costs and behaves like $\pi$ in terms of
long-run average cost. Now, under policy \( \gamma \), if the server decides to serve another customer in stage 2 after idle time \( I \). before switching to stage 1. then policy \( \gamma' \in \Gamma_2^B \) can be constructed from \( \gamma \) and it is non-breaking up to time \( \tau_{\gamma'} = \tau_{\gamma} + S_2' \), so that \( U_{\gamma'}(z_0) \geq U_{\gamma}(z_0) \) . \( U_{\gamma'}(z_0) \geq U_{\gamma} \). Iterating this procedure, we finally obtain policy \( \gamma^*_B \in \Gamma_2^B \) which is a non-breaking policy in stage 2.

To show that the optimal policy in the second stage is also a greedy policy when the long-run average costs are considered, we analyze the second case in which the server decides to switch to stage 1 after idle time \( I \).

(ii) If the server switches to stage 1 at time \( \tau_{\gamma} + I \), the new policy \( \gamma \) can be constructed as follows (Figure 2.2):

![Figure 2.2. Behaviour of \( \pi \) and \( \gamma \) in \( [\tau_{\gamma}, \tau_{\gamma} + I + D_{21}] \)](image)

- \( \gamma \) follows \( \pi \) in \([0, \tau_{\gamma}]\)
- \( \gamma \) switches to the first stage and then idles the same as \( \pi \)
- \( \gamma \) follows \( \pi \) in \([\tau_{\gamma} + I + D_{21}, \infty)\)

Assuming that a switch from stage 1 to 2 (or 2 to 1) occurs at the first instant \( D_{12} \) (or \( D_{21} \)) and setting \( \tau_{\gamma} = 0 \) and \( z_0 = (2, x_1', x_2') \), for all \( t \geq 0 \), we will have

\[
AR_1^{\gamma, z_0}(t) = AR_1^{\pi, z_0}(t) \quad (2.15)
\]

\[
DP_1^{\gamma, z_0}(t) = DP_1^{\pi, z_0}(t) \quad (2.16)
\]
\[ DP_2^{\gamma,z_0}(t) = DP_2^{\pi,z_0}(t) \]  
\[ Q_{12}^{\gamma,z_0}(t) = Q_{12}^{\pi,z_0}(t) \]

and

\[ Q_{21}^{\gamma,z_0}(t) = \begin{cases} Q_{21}^{\pi,z_0}(t) + 1 & : 0 \leq t < I \\ Q_{21}^{\pi,z_0}(t) & : I \leq t \end{cases} \]  

Therefore,

\[ \bar{C}_\gamma(z_0) = \bar{C}_\pi(z_0) \quad \forall \ z_0 \]

In other words, for each policy \( \pi \in \Pi \) in which there is an idle period before switching to stage 1, there exists a policy \( \gamma \in \Gamma_2 \) which behaves the same in the sense of long-run average cost.

Combining the analyses of cases (i) and (ii), it can be concluded that the optimal policy in the second stage is a greedy policy when the long-run average cost is considered, and is a non-breaking policy when the total discounted cost is considered.

### 2.4 Optimality of Exhaustive Policies

Considering the optimality of greedy and non-breaking policies, in this section we compare the decision between "serving" and "switching" in the second stage in order to find the optimal decision when the server is at nonempty stage 2. We will show that when \( h_2 \geq h_1 \), any policy \( \pi \in \Pi \) is improved by a greedy and exhaustive policy in stage 2. In other words, the optimal policy in a two-stage tandem queue attended by a moving server with holding and switching costs is a policy in which the server is never idle at nonempty stage 2 and never leaves stage 2 before it is empty.

**Definition 2.3**

Policy \( \xi \) is said to be an exhaustive policy if the server never leaves a queue until it becomes empty. Let \( \Xi_2 \subset \Pi \) be the set of all policies that are exhaustive in stage 2.
Proposition 2.2

If \( h_2 \geq h_1 \), then for any policy \( \pi \in \Pi \), there exist policies \( \xi^*_h \in \Xi_2 \cap \Gamma_2^h \) and \( \xi^* \in \Xi_2 \cap \Gamma_2 \) such that

\[
U_{\xi^*_h}(z_0) \geq U_\pi(z_0), \quad U_{\xi^*}(z_0) \geq U_\pi(z_0).
\]

Proof

Let \( \tau_{S,21}^\gamma \) be the first time under policy \( \gamma \in \Gamma_2 \) that the server decides to switch from nonempty stage 2 to stage 1. Also, suppose that \( \tau_{S,12}^\gamma \) is the first time after \( \tau_{S,21}^\gamma \) that under policy \( \gamma \in \Gamma_2 \) the server switches back to stage 2 to serve a customer. We construct a new policy \( \xi \) as follows (Figure 2.3):

\[\begin{align*}
\text{Policy } \gamma & \quad \tau_{S,21}^\gamma \quad D_{21} \quad \tau_{S,12}^\gamma \quad D_{12} \quad S_2^1 \\
\text{Policy } \xi & \quad \tau_{S,21}^\gamma \quad S_2^1 \quad D_{21} \quad \square \quad D_{12}
\end{align*}\]

**Figure 2.3.** Behaviour of \( \gamma \) and \( \xi \) in \([\tau_{S,21}^\gamma, \tau_{S,12}^\gamma + D_{12} + S_2^1]\).

- \( \xi \) follows \( \gamma \) in \([0, \tau_{S,21}^\gamma]\)
- \( \xi \) serves next customer in stage 2 and then switches to stage 1 and follows policy \( \gamma \) until \( \gamma \) switches back to stage 2 to serve a customer
- \( \xi \) follows \( \gamma \) in \([\tau_{S,12}^\gamma + D_{12} + S_2^1, \infty)\)

Setting \( \tau_{S,21}^\gamma = 0 \) and considering state \( z_0 = (x_1^{\tau_{S,21}^\gamma}, x_2^{\tau_{S,21}^\gamma}) \) at time \( \tau_{S,21}^\gamma \), we will have

\begin{align*}
AR_{1}^{\xi,z_0}(t) & = AR_{1}^{\gamma,z_0}(t) \\
DP_{1}^{\xi,z_0}(t) & \leq DP_{1}^{\gamma,z_0}(t)
\end{align*}

(2.20) (2.21)
Thus, from (2.5) and using (2.20) to (2.24) we get

\[ U'_{\xi}(z_0) - L'_{\chi}(z_0) = E \left[ h_2 \int_{S_2^1 + r_{5,12}^2 + D_{12}} S_2^1 e^{-\beta t} dt \right. \]
\[ - (h_2 - h_1) \int_0^{\infty} e^{-\beta t} (DP_{2}^{\xi, z_0}(t) - DP_{1}^{\gamma, z_0}(t)) dt \]
\[ - K_{12} \int_0^{\infty} e^{-\beta t} (dQ_{12}^{\xi, z_0}(t) - dQ_{12}^{\gamma, z_0}(t)) \]
\[ - K_{21} \int_0^{\infty} e^{-\beta t} (dQ_{21}^{\xi, z_0}(t) - dQ_{21}^{\gamma, z_0}(t)) \]
\[ = E \left[ \frac{h_2}{\beta} (e^{-\beta S_2^1} - e^{-\beta (S_2^1 + r_{5,12}^2 + D_{12})}) \right] \]
\[ + (h_2 - h_1) E \left[ \int_0^{\infty} e^{-\beta t} (DP_{1}^{\gamma, z_0}(t) - DP_{1}^{\xi, z_0}(t)) dt \right] \]
\[ + K_{12} E \left[ \int_0^{\infty} e^{-\beta t} (dQ_{12}^{\gamma, z_0}(t) - dQ_{12}^{\xi, z_0}(t)) \right] \]
\[ + K_{21} E \left[ \int_0^{\infty} e^{-\beta t} (dQ_{21}^{\gamma, z_0}(t) - dQ_{21}^{\xi, z_0}(t)) \right]. \]  \hspace{1cm} (2.25)

Consider the first term of the right hand side of (2.25):

\[ E \left[ \frac{h_2}{\beta} (e^{-\beta S_2^1} - e^{-\beta (S_2^1 + r_{5,12}^2 + D_{12})}) \right] = \frac{h_2}{\beta} E[e^{-\beta S_2^1} (1 - e^{-\beta (r_{5,12}^2 + D_{12})})] \]
\[ = \frac{h_2}{\beta} F_{2}^\prime (\beta) E[1 - e^{-\beta (r_{5,12}^2 + D_{12})}]. \]  \hspace{1cm} (2.26)

Since \( r_{5,12}^2 + D_{12} > 0 \), therefore \( 1 - e^{-\beta (r_{5,12}^2 + D_{12})} \geq 0 \) and (2.26) is nonnegative; and in the case of \( h_2, \beta > 0 \), it is strictly positive.

On the other hand,

\[ K_{12} E \left[ \int_0^{\infty} e^{-\beta t} (dQ_{12}^{\gamma, z_0}(t) - dQ_{12}^{\xi, z_0}(t)) \right] = K_{12} E[e^{-\beta r_{5,12}^2} - e^{-\beta (r_{5,12}^2 + S_2^1)}] \]
and for $K_{12} > 0$, equation (2.27) is strictly positive. Also,

$$K_{21} E\left[ \int_0^\infty e^{-\beta t} (dQ_{21}^{\xi,\gamma}(t) - dQ_{21}^{\xi,\gamma}(t)) \right] = K_{21} E[1 - e^{-\beta S}] = K_{21}[1 - F^*_2(\beta)]. \quad (2.28)$$

and (2.28) is also strictly positive if $K_{21} > 0$.

Now consider the second term of the right hand side of equation (2.25). Using (2.21), since $DP_1^{\gamma,\tau}(t) - DP_1^{\xi,\tau}(t) \geq 0$ and also $h_2 - h_1 \geq 0$, then

$$(h_2 - h_1) E\left[ \int_0^\infty e^{-\beta t} \right. \left. (DP_1^{\gamma,\tau}(t) - DP_1^{\xi,\tau}(t)) dt \right] \geq 0. \quad (2.29)$$

Aggregating the results (2.26) to (2.29), we obtain

$$U^\xi(z_0) - U^\gamma(z_0) \geq 0 : \forall z_0$$

Therefore, policy $\xi$, which is exhaustive up to time $\tau_{S,21}^\xi$, improves the total discounted holding and switching costs. If at time $\tau_{S,21}^\xi$ stage 2 is nonempty, then policy $\xi'$, which is exhaustive up to time $\tau_{S,21}^{\xi'} = \tau_{S,21}^\xi + S^\xi_2$, can be constructed from $\xi$, where $U^\xi(z_0) \geq U^\xi(z_0)$. Applying this approach iteratively, we will finally end up with policy $\xi_B^* \in \Xi_2 \cap \Gamma_2^B$, which is exhaustive and nonbreaking.

When the long-run average costs are considered, then using equation (2.20) to (2.24), yields

$$U^\xi(z_0) - U^\gamma(z_0) = \lim_{t \to \infty} \frac{1}{t} E\left[ h_2 \int_{S^\xi_2}^{S^\xi_1 + \tau_{S,12} + D_{12}} dt \right.$$

$$+ (h_2 - h_1) \int_0^t (DP_1^{\gamma,\tau}(t) - DP_1^{\xi,\tau}(t)) dt$$

$$+ K_{12} (Q_{12}^{\gamma,\tau}(t) - Q_{12}^{\xi,\tau}(t)) + K_{21} (Q_{21}^{\gamma,\tau}(t) - Q_{21}^{\xi,\tau}(t)) \right].$$

When $t \to \infty$, then $Q_{12}^{\xi,\tau}(t) = Q_{12}^{\gamma,\tau}(t)$ and $Q_{21}^{\xi,\tau}(t) = Q_{21}^{\gamma,\tau}(t)$; therefore,

$$U^\xi(z_0) - U^\gamma(z_0) = \lim_{t \to \infty} \frac{1}{t} E\left[ h_2 (\tau_{S,12} + D_{12}) + (h_2 - h_1) \int_0^t (DP_1^{\gamma,\tau}(t) - DP_1^{\xi,\tau}(t)) dt \right]$$

$$= K_{12} E\left[ e^{-\beta \tau^2_{S,12}} (1 - e^{-\beta S^2_2}) \right]$$

$$= K_{12} [1 - F^*_2(\beta)] E[e^{-\beta \tau^2_{S,12}}]. \quad (2.27)$$
and since \( DP_1^{\gamma,z_0}(t) - DP_1^{\xi,z_0}(t) \geq 0 \), and also \( h_2 \geq h_1 \).

\[
(h_2 - h_1) \int_0^t (DP_1^{\gamma,z_0}(t) - DP_1^{\xi,z_0}(t)) dt \geq 0 .
\]

Then, since \( h_2(\tau_{S,12} + D_{12}) \geq 0 \) we will have

\[
\overline{U}_\xi(z_0) - \overline{U}_\gamma(z_0) \geq 0 : \forall z_0 .
\]

Using an iterative approach as before, we conclude that when \( h_2 \geq h_1 \) the policy \( \xi^* \in \Xi_2 \cap \Gamma_2 \) is always the optimal policy in the second stage of a two-stage tandem queue attended by a moving server with holding and switching costs. \( \square \)

**Remark 2.1**

For an \( N \)-stage tandem queue attended by a moving server we will have

\[
U_\pi(z_0) = E[ h_N \int_0^\infty e^{-\beta t} DP_N^{\pi,z_0}(t) dt - h_1 \int_0^\infty e^{-\beta t} AR_1^{\pi,z_0}(t) dt - \sum_{i=1}^{N-1} (h_{i+1} - h_i) \int_0^\infty e^{-\beta t} DP_i^{\pi,z_0}(t) dt - \int_0^\infty e^{-\beta t} \sum_{i,j} K_{ij} dQ_{ij}^{\pi,z_0}(t) ] .
\]

and

\[
\overline{U}_\pi(z_0) = \lim_{t \to \infty} \frac{1}{t} E[ h_N \int_0^t DP_N^{\pi,z_0}(u) du - h_1 \int_0^t AR_1^{\pi,z_0}(u) du - \sum_{i=1}^{N-2} (h_{i+1} - h_i) \int_0^t DP_i^{\pi,z_0}(u) du - \sum_{i,j} K_{ij} Q_{ij}^{\pi,z_0}(t) ] .
\]

where \( Q_{ij}^{\pi,z_0}(t) \) is the number of switches from stage \( i \) to \( j \) up to time \( t \) under policy \( \pi \) and initial state \( z_0 \). Using the same approach as Propositions 1 and 2, it can be shown that if \( h_{i+1} - h_i \geq 0 \) (\( i = 1, 2, \ldots, N-1 \)), then the optimal policies in stage \( N \) which minimize \( V_\pi(z_0) \) and \( \overline{V}_\pi(z_0) \) are nonbreaking and exhaustive, and greedy and exhaustive, respectively. \( \square \)

**Remark 2.2**

When \( h_2 \geq h_1 \), stage 1 may have higher priority than stage 2. In other words, sometimes the server may switch to stage 1 when there are some customers waiting
in stage 2. However, since assumption $h_{i+1} - h_i \geq 0$ $(i = 1, 2, \ldots, N - 1)$ covers the more realistic situations, especially in manufacturing systems where the item's value increases after each operation, this research deals only with this case.

Since the optimal policy for the $G/G_1 - G_2/1$ queue consists of optimal policies in stages 1 and 2; therefore, considering greedy and exhaustive service policy in the second stage, three different policies for the first stage are introduced and analyzed in Chapters 3, 4 and 5. These policies are called static, semi-dynamic and dynamic policies and are defined based on the capabilities of the server and the uncertainty about the number of customers in the system.
Chapter 3

An $M/G_1 - G_2/1$ Queue with Static Policy

3.1 Introduction

Suppose that in the basic two-stage tandem queue, the BTQ model, a greedy and exhaustive service policy is applied in stage 2 and the server always serves a fixed number of customers, $M_S$, in stage 1 before switching to stage 2. This means that sometimes the server must wait in stage 1 for a customer to come because he hasn’t served $M_S$ customers yet. This policy is called a static policy. In this chapter we introduce two different models which can be used to analyze the two-stage tandem queue attended by a moving server with a static policy. Although the first model alone is sufficient to find the optimal static policy, the second model is developed because of the broad range of its capabilities in analyzing other queueing systems.

3.2 Static Policy

Consider our basic two-stage tandem queue, the BTQ model, and suppose that the server implements the optimal greedy and exhaustive policy in the second stage whenever he switches to that stage. Also, suppose that when the server switches to stage 1, he serves a predetermined number, $M_S$, of customers there before switching back
to stage 2. As was previously mentioned, this policy is called a static policy. Therefore, defining a cycle as the time elapsed between two consecutive switches to stage 1. $M_S$ customers are always served in each cycle when a static policy with parameter $M_S$ is implemented. This usually occurs when the server is a machine and either the machine cannot be adjusted for different capacities in each cycle, or the adjustment costs are too high. Therefore, the machine is adjusted to complete exactly $M_S$ units in each cycle.

A static policy is actually the full cart strategy in the queue-cart problem. The difference between this problem and the queue-cart problem analyzed by Coffman and Gilbert [13] is that the delivery time in our problem depends on the number of items in the cart, which is clearly a more realistic situation.

### 3.3 Model 1: An $M/G_1 - G_2/1$ queue

To analyze our basic model when static policy is applied, let $I^{(n)}$ be the location of the server at the end of the $n$th service completion. ($I^{(n)} \in \{1, 2\}$), and let $X_1^{(n)}$ and $X_2^{(n)}$ be the number of customers in stages 1 and 2 at that epoch, respectively. Then, a model for this system can be represented by the imbedded Markov chain $< X_1^{(n)}, X_2^{(n)}, I^{(n)} >$, and assuming that steady state conditions prevail, we have $\lim_{n \to \infty} (X_1^{(n)}, X_2^{(n)}, I^{(n)}) = (X_1, X_2, I)$ in distribution. so that the stationary probabilities can be defined as

$$
\pi_{ij}^{(k)} = P\{X_1 = i, X_2 = j, I = k\} \quad : \quad k = 1, 2.
$$

Let $Y_k^s$ be the number of customers that arrive during a service time in stage $k$, and $Y_k^D$ the number of customers that arrive during switchover times $D_k$. (switchover time from stage $k$, $k = 1, 2$). Then, for $k = 1, 2$,

$$
p_l^{(k)} = P\{Y_k^s = l\} = \int_0^\infty \frac{e^{-\lambda t}(\lambda t)^l}{l!} dB_k(t)
$$

$$
d_l^{(k)} = P\{Y_k^D = l\} = \int_0^\infty \frac{e^{-\lambda t}(\lambda t)^l}{l!} dB_k(t)
$$

Therefore, $p_l^{(k)}$ and $d_l^{(k)}$ represent the probabilities that $l$ customers arrive at stage 1 during service time $S_k$ and switchover times $D_k$, ($k = 1, 2$), respectively.
For a system with a static policy, the balance equations for \( i \geq 0 \) are:

\[
\pi_{ij}^{(1)} = \begin{cases} 
0 & : j = 0 \\
\pi_{00}^{(2)} d_0^{(2)} p_i^{(1)} + \sum_{m=0}^{i} \pi_{i-m}^{(2)} d_{m}^{(2)} p_m^{(1)} & : j = 1 \\
\pi_{0,j-1}^{(1)} p_i^{(1)} + \sum_{m=0}^{j} \pi_{i-m+1,j-1}^{(1)} p_m^{(1)} & : 2 \leq j \leq M_S \\
0 & : j \geq M_S + 1 
\end{cases}
\]

\[
\pi_{ij}^{(2)} = \begin{cases} 
\sum_{m=0}^{\infty} \pi_{i-m,j+1}^{(2)} p_m^{(1)} & : 0 \leq j \leq M_S - 2 \\
\sum_{m=0}^{\infty} \sum_{n=0}^{\infty} \pi_{i-m-n,M_s}^{(1)} d_{n}^{(1)} p_m^{(2)} & : j = M_S - 1 \\
0 & : j \geq M_S 
\end{cases}
\]

and

\[
\sum_{i=0}^{\infty} \sum_{j=0}^{\infty} \pi_{ij}^{(1)} + \pi_{ij}^{(2)} = 1 .
\]

If the probability generating function for probabilities \( \pi_{ij}^{(k)} \), \( k = 1,2 \), is defined by:

\[
\Pi_k(x,y) = \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} \pi_{ij}^{(k)} x^i y^j \quad : k = 1,2 \quad |x|, |y| \leq 1
\]

then, multiplying both sides of (3.1) by appropriate \( x^i \) and \( y^j \), and then summing over \( i \) and \( j \) yields:

\[
\Pi_k(x,y) = \sum_{i=0}^{\infty} \pi_{00}^{(2)} d_0^{(2)} p_i^{(1)} x^i y + \sum_{i=0}^{\infty} \sum_{m=0}^{i} \pi_{i-m}^{(2)} d_{m}^{(2)} p_m^{(1)} x^i y \\
+ \sum_{i=0}^{\infty} \sum_{m=0}^{\infty} \pi_{i-m+1,j}^{(2)} p_m^{(1)} x^i y + \sum_{i=0}^{\infty} \sum_{j=2}^{M_S} \pi_{0,j-1}^{(1)} p_i^{(1)} x^i y \\
+ \sum_{i=0}^{\infty} \sum_{j=2}^{M_S} \sum_{m=0}^{\infty} \pi_{i-m+1,j-1}^{(1)} p_m^{(1)} x^i y^j .
\]

Defining:

\[
P_k(x) = \sum_{i=0}^{\infty} \pi_{i}^{(k)} x^i \quad : k = 1,2 \quad |x| \leq 1
\]

\[
D_k(x) = \sum_{i=0}^{\infty} d_i^{(k)} x^i \quad : k = 1,2 \quad |x| \leq 1
\]
we will have
\[
\sum_{i=0}^{\infty} \pi_{i}^{(2)} d_{i}^{(2)} p_{i}^{(1)} x^{i} y = \pi_{00}^{(2)} d_{0}^{(2)} y P_{1}(x) .
\]
(3.6)

and
\[
\sum_{i=0}^{\infty} \sum_{m=0}^{i} \sum_{n=0}^{i-m} \pi_{i-n-m+1,0}^{(2)} d_{n}^{(2)} p_{m}^{(1)} x^{i} y = \frac{y}{x} \sum_{i=0}^{\infty} \sum_{m=0}^{i} \sum_{n=0}^{i-m} \pi_{i-n-m+1,0}^{(2)} d_{n}^{(2)} x^{i-n-m+1} y^{n} p_{m}^{(1)} x^{m}
\]
\[
= \frac{y}{x} [\Pi_{2}(x, 0) - \pi_{00}^{(2)}] D_{2}(x) P_{1}(x).
\]
(3.7)

Also,
\[
\sum_{i=0}^{\infty} \sum_{m=0}^{i} \pi_{00}^{(2)} d_{i-m+1,0}^{(2)} p_{m}^{(1)} x^{i} y = \pi_{00}^{(2)} \frac{y}{x} [D_{2}(x) - d_{0}^{(2)}] P_{1}(x)
\]
(3.8)

and
\[
\sum_{i=0}^{\infty} \sum_{m=0}^{i} \pi_{00}^{(1)} p_{i-m}^{(1)} x^{i} y^{j} = y P_{1}(x) \left[ \sum_{j=1}^{M_{S}-1} \pi_{00}^{(1)} y^{j} \right].
\]
(3.9)

Finally,
\[
\sum_{i=0}^{\infty} \sum_{m=0}^{i} \sum_{j=0}^{i-m+1} \pi_{i-m-j+1,0}^{(1)} p_{m}^{(1)} x^{i} y^{j} = \frac{y}{x} [\Pi_{1}(x, y) - \Pi_{1}(0, y) - \Psi_{M_{S}}(x, y)] P_{1}(x).
\]
(3.10)

where
\[
\Psi_{M_{S}}(x, y) = \sum_{i=1}^{\infty} \pi_{i,M_{S}}^{(1)} x^{i} y^{M_{S}}.
\]
(3.11)

Substituting (3.6) to (3.10) into (3.5), after some algebra we get
\[
\Pi_{1}(x, y) = \frac{y P_{1}(x)}{x - y P_{1}(x)} \left[ D_{2}(x) \Pi_{2}(x, 0) - \Pi_{1}(0, y) - \Psi_{M_{S}}(x, y) \right.
\]
\[
- \pi_{00}^{(2)} d_{0}^{(2)} (1 - x) + x \Theta_{M_{S}-1}(y) \right] .
\]
(3.12)

in which
\[
\Theta_{M_{S}-1}(y) = \sum_{j=1}^{M_{S}-1} \pi_{00}^{(1)} y^{j} .
\]
(3.13)

Now, multiply both sides of (3.2) by appropriate $x^{i}$ and $y^{j}$ and then sum over $i$ and $j$ to yield
\[
\Pi_{2}(x, y) = \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} \sum_{m=0}^{i} \pi_{i-m,j}^{(2)} p_{m}^{(1)} x^{i} y^{j} + \sum_{i=0}^{\infty} \sum_{m=0}^{i} \sum_{n=0}^{i-m} \pi_{i-n-M_{S}}^{(1)} d_{m}^{(1)} p_{m}^{(2)} x^{i} y^{M_{S}-1}
\]
\[
= \frac{1}{y} [\Pi_{2}(x, y) - \Pi_{2}(x, 0)] P_{2}(x) + \frac{1}{y} [\Psi_{M_{S}}(x, y) + \pi_{00}^{(1)} y^{M_{S}}] D_{1}(x) P_{2}(x)
\]
(3.14)
which upon rearrangement yields

$$\Pi_2(x, y) = \frac{P_2(x)}{y - P_2(x)} \{ D_1(x) \left[ \Psi_{M_S}(x, y) + \pi_{0M_S}^{(1)} y^{M_S} \right] - \Pi_2(x, 0) \}.$$  \hspace{1cm} (3.15)

To establish the generating functions $\Pi_1(x, y)$ and $\Pi_2(x, y)$ it is now necessary to determine unknown functions $\Pi_1(0, y)$, $\Pi_2(x, 0)$, $\Psi_{M_S}(x, y)$, $\Theta_{M_S-1}(y)$, $\pi_{0M_S}^{(1)}$ and $\pi_{00}^{(2)}$.

### 3.3.1 Determination of $\Psi_{M_S}(x, y)$

To determine $\Psi_{M_S}(x, y)$ we first observe that we can write

$$\Psi_{M_S}(x, y) = \frac{y^{M_S}}{M_S!} \left[ \frac{d^{M_S}}{dy^{M_S}} \Pi_1(x, y) \right]_{y=0} - \frac{d^{M_S}}{dy^{M_S}} \Pi_1(x, y) \bigg|_{y=0}.$$ \hspace{1cm} (3.16)

Using (3.12) we have.

$$\frac{d^{M_S}}{dy^{M_S}} \Pi_1(x, y) = \left[ D_2(x) \Pi_2(x, 0) - \pi_{00}^{(2)} d_0^{(2)} (1 - x) \right] \frac{d^{M_S}}{dy^{M_S}} \left\{ \frac{y P_1(x)}{x - y P_1(x)} \right\} + \frac{d^{M_S}}{dy^{M_S}} \left\{ - \Pi_1(0, y) - \Psi_{M_S}(x, y) + x \Theta_{M_S-1}(y) \right\} \frac{y P_1(x)}{x - y P_1(x)}.$$ \hspace{1cm} (3.17)

But it can be shown that

$$\frac{d^k}{dy^k} \left[ \frac{y P_1(x)}{x - y P_1(x)} \right] = \frac{k! [P_1(x)]^k}{(x - y P_1(x))^{k+1}},$$  \hspace{1cm} (3.18)

whence.

$$\frac{d^k}{dy^k} \left[ \frac{y P_1(x)}{x - y P_1(x)} \right] \bigg|_{y=0} = k! \left[ \frac{P_1(x)}{x} \right]^k.$$ \hspace{1cm} (3.19)

On the other hand

$$\frac{d^{M_S}}{dy^{M_S}} \left[ \Pi_1(0, y) \frac{y P_1(x)}{x - y P_1(x)} \right] \bigg|_{y=0} = \sum_{k=0}^{M_S} C_k^{M_S} \frac{d^k}{dy^k} \left[ \Pi_1(0, y) \right] \bigg|_{y=0} \frac{d^{M_S-k}}{dy^{M_S-k}} \left[ \frac{y P_1(x)}{x - y P_1(x)} \right] \bigg|_{y=0}$$ \hspace{1cm} (3.20)

where $C_k^n = \frac{n!}{k!(n-k)!}$. Therefore, since

$$\frac{d^k}{dy^k} \Pi_1(0, y) \bigg|_{y=0} = k! \pi_{0k}^{(1)}.$$  \hspace{1cm} (3.21)
using (3.19) and (3.21) in (3.20), we will have

\[
\frac{d^{M_s}}{dy^{M_s}} \left[ \Pi_1(0, y) \frac{y P_1(x)}{x - y P_1(x)} \right]_{y=0} = \sum_{k=1}^{M_s-1} C_k^{M_s} k! \pi_{0k}^{(1)} (M_s - k)! \left[ \frac{P_1(x)}{x} \right]^{M_s-k} = M_s! \left[ \frac{P_1(x)}{x} \right]^{M_s} \Theta_{M_s-1} \left( \frac{x}{P_1(x)} \right). \tag{3.22}
\]

Furthermore, since

\[
\frac{d^k}{dy^k} x \Theta_{M_s-1}(y) \bigg|_{y=0} = \begin{cases} 
  x! \pi_{0k}^{(1)} & : k \leq M_s - 1 \\
  0 & : k \geq M_s
\end{cases}
\]

then

\[
\frac{d^{M_s}}{dy^{M_s}} x \Theta_{M_s-1}(y) \left[ \frac{y P_1(x)}{x - y P_1(x)} \right]_{y=0} = \sum_{k=1}^{M_s-1} C_k^{M_s} k! \pi_{0k}^{(1)} (M_s - k)! \left[ \frac{P_1(x)}{x} \right]^{M_s-k} = M_s! x \left[ \frac{P_1(x)}{x} \right]^{M_s} \Theta_{M_s-1} \left( \frac{x}{P_1(x)} \right). \tag{3.23}
\]

Finally, considering the definition of $\Psi_{M_s}(x, y)$ from (3.11), it is clear that

\[
\frac{d^{M_s}}{dy^{M_s}} \left[ \Psi_{M_s}(x, y) \frac{y P_1(x)}{x - y P_1(x)} \right]_{y=0} = 0. \tag{3.24}
\]

Substituting (3.22) to (3.24) into (3.17), we now have

\[
\frac{d^{M_s}}{dy^{M_s}} \Pi_1(x, y) \bigg|_{y=0} = \left[ D_2(x) \Pi_2(x, 0) - \pi_{00}^{(2)} d_0^{(2)} (1 - x) \right] M_s! \left[ \frac{P_1(x)}{x} \right]^{M_s} - M_s! \left[ \frac{P_1(x)}{x} \right]^{M_s} \Theta_{M_s-1} \left( \frac{x}{P_1(x)} \right) + M_s! \left[ \frac{P_1(x)}{x} \right]^{M_s} x \Theta_{M_s-1} \left( \frac{x}{P_1(x)} \right). \tag{3.25}
\]

On the other hand,

\[
\frac{d^{M_s}}{dy^{M_s}} \Pi_1(x, y) \bigg|_{x=y=0} = M_s! \pi_{0M_s}^{(1)}, \tag{3.26}
\]

so that using (3.25) and (3.26) in (3.16) gives

\[
\Psi_{M_s}(x, y) = \left( \frac{y P_1(x)}{x} \right)^{M_s} \left[ D_2(x) \Pi_2(x, 0) - \pi_{00}^{(2)} d_0^{(2)} (1 - x) - \Theta_{M_s} \left( \frac{x}{P_1(x)} \right) \right] + x \Theta_{M_s-1} \left( \frac{x}{P_1(x)} \right). \tag{3.27}
\]
3.3.2 Determination of $\Pi_2(x, 0)$

From the denominator (3.15), it is clear that $y - P_2(x)$ has one root $y = P_2(x)$ inside the unit circle $|y| \leq 1$. Therefore, substituting this root into numerator of (3.15), we get

$$D_1(x)[\Psi_{MS}(x, P_2(x)) + \pi^{(1)}_{0MS}[P_2(x)]^MS] - \Pi_2(x, 0) = 0. \quad (3.28)$$

so that

$$\Pi_2(x, 0) = D_1(x)[\Psi_{MS}(x, P_2(x)) + \pi^{(1)}_{0MS}[P_2(x)]^MS]. \quad (3.29)$$

Now, substituting (3.27) into (3.29),

$$\Pi_2(x, 0) = D_1(x)[\left(\frac{P_1(x)P_2(x)}{x}\right)^MS (D_2(x)\Pi_2(x, 0) - \pi^{(2)}_{00}d^{(2)}_0(1 - x) - \Theta_{MS}\left(\frac{x}{P_1(x)}\right))$$

$$+ x\Theta_{MS-1}\left(\frac{x}{P_1(x)}\right) + \pi^{(1)}_{0MS}[P_2(x)]^MS]. \quad (3.30)$$

and rearranging, yields

$$\Pi_2(x, 0) = \frac{D_1(x)[P_1(x)P_2(x)]^MS(x - 1)}{x^MS - D_1(x)D_2(x)[P_1(x)P_2(x)]^MS[\pi^{(2)}_{00}d^{(2)}_0 + \Theta_{MS-1}\left(\frac{x}{P_1(x)}\right)]}. \quad (3.30)$$

3.3.3 Determination of $\Pi_1(0, y)$, $\Theta_{MS-1}(\cdot)$ and $\pi^{(2)}_{00}$

From (3.13) we observe that $\Theta_{MS-1}(\cdot)$ is a function of the unknown probabilities $\pi^{(1)}_{0j}$, $j = 1, 2, \ldots, MS - 1$. Using Takacs' lemma [62] we can show that if $[\rho_1 + \rho_2 + (\eta_1 + \eta_2)/MS] \leq 1$ (where $\rho_i = \lambda E[S_i]$ and $\eta_i = \lambda E[D_i]$), then the denominator in the right hand side of (3.30), $x^MS - D_1(x)D_2(x)[P_1(x)P_2(x)]^MS = 0$, has $MS$ roots $\xi_j$, $j = 1, 2, \ldots, MS - 1$ in the unit circle, while $\xi_{MS} = 1$. Since $|\Pi_2(x, 0)| \leq 1$ in domain $|x| \leq 1$, it follows that numerator in the right hand side of (3.30) must be zero for $\xi_j$ ($j = 1, 2, \ldots, MS - 1$); therefore for $j = 1, 2, \ldots, MS - 1$,

$$\pi^{(2)}_{00}d^{(2)}_0 + \Theta_{MS-1}\left(\frac{\xi_j}{P_1(\xi_j)}\right) = 0. \quad (3.31)$$

Since the system of $MS - 1$ equations (3.31) has $MS$ unknowns $\pi^{(1)}_{01}$, $\pi^{(1)}_{02}$, $\ldots$, $\pi^{(1)}_{0, MS - 1}$ and $\pi^{(2)}_{00}$, it is necessary to use one more relation among those unknowns, namely $\Pi_1(1, 1) + \Pi_2(1, 1) = 1$. 
Using (3.27), we get

\[ \Psi_{M_S}(1, 1) = \Pi_2(1, 0) - \pi_{0M_S}^{(1)} . \]  

(3.32)

If for any function \( g(x_1, x_2, \ldots, x_n) \) we define

\[ g''(1, 1, \ldots, 1) = \frac{d}{dx_i} g(1, \ldots, 1, x_i, 1, \ldots, 1) \bigg|_{x_i=1} . \]  

(3.33)

then, considering (3.12) and using L'Hopital's rule, we get,

\[ \Pi_1(1, 1) = \Pi_1(0, 1) + \Pi'_{M_S}(0, 1) + \Psi_{M_S}'(1, 1) - \Theta_{M_S}(1) - \Theta_{M_S-1}'(1) \]  

(3.34)

where,

\[ \Pi'_{M_S}(0, 1) = \sum_{r=1}^{M_S} r\pi_{Or}^{(1)} \]  

(3.35)

\[ \Psi_{M_S}'(1, 1) = \frac{d}{dy} \left\{ y^{M_S} \left[ \Pi_2(1, 0) - \Theta_{M_S}(1) + \Theta_{M_S-1}(1) \right] \right\} \bigg|_{y=1} \]

\[ = M_S \left[ \Pi_2(1, 0) - \pi_{0M_S}^{(1)} \right] \]  

(3.36)

and

\[ \Theta_{M_S-1}'(1) = \frac{d}{dy} \Theta_{M_S-1}(y) \bigg|_{y=1} = \sum_{r=1}^{M_S-1} r\pi_{Or}^{(1)} . \]  

(3.37)

Substituting (3.35) to (3.37) into (3.34), and noting that \( \Pi_1(0, 1) = \Theta_{M_S}(1) \), yields

\[ \Pi_1(1, 1) = M_S \Pi_2(1, 0) \]  

(3.38)

To find \( \Pi_2(1, 0) \), using L' Hospital's rule in (3.30) we get

\[ \Pi_2(1, 0) = \frac{\pi_{00}^{(2)} d_0^{(2)} + \Theta_{M_S-1}(1)}{M_S(1 - \bar{p})} . \]  

(3.39)

where \( \bar{p} = \rho + \eta/M_S \) and \( \rho = \rho_1 + \rho_2, \quad \eta = \eta_1 + \eta_2 \).

On the other hand, using L' Hospital's rule in (3.15) we will have

\[ \Pi_2(1, 1) = \Psi_{M_S}'(1, 1) + M_S\pi_{0M_S}^{(1)} \]

\[ = M_S \Pi_2(1, 0) . \]  

(3.40)

Therefore considering (3.38) and (3.40) and the normalizing condition \( \Pi_1(1, 1) + \Pi_2(1, 1) = 1 \), we obtain

\[ \Pi_1(1, 1) = \Pi_2(1, 1) = \frac{1}{2} , \] 

(3.41)
and using (3.39).

\[ \pi_{00}^{(2)} d_0^{(2)} + \Theta_{M_{S-1}}(1) = \frac{(1 - \bar{p})}{\bar{p}} . \]  

(3.42)

By using equation (3.42) along with system of linear equations (3.31), the unknown probabilities \( \pi_{0j}^{(1)} \), \( j = 1, 2, \ldots, M_S - 1 \) (or \( \Theta_{M_{S-1}}(\cdot) \)) and \( \pi_{00}^{(2)} \) will be obtained.

From (3.4) we observe that

\[ \Pi_1(0, y) = \sum_{j=1}^{M_S} \pi_{0j}^{(1)} y^j . \]  

(3.43)

Therefore, to obtain \( \Pi_1(0, y) \) it only remains to find \( \pi_{0M_S}^{(1)} \).

**Determination of \( \pi_{0M_S}^{(1)} \)**

Considering the equilibrium equations (3.2), we will have

\[ \pi_{0j}^{(2)} = \pi_{0,j+1}^{(2)} p_0^{(2)} : 0 \leq j \leq M_S - 2 . \]  

(3.44)

Thus, by using (3.44) iteratively, we get

\[ \pi_{0,M_S-1}^{(2)} = \pi_{00}^{(2)} \frac{1}{p_0^{(2)}}^{M_S-1} . \]  

(3.45)

Also, using (3.2) for \( j = M_S - 1 \), with (3.45) yields

\[ \pi_{0M_S}^{(1)} = \pi_{0,M_S-1}^{(2)} \frac{1}{d_0^{(1)} p_0^{(2)}} = \pi_{00}^{(2)} \frac{1}{d_0^{(1)} p_0^{(2)}}^{M_S} . \]  

(3.46)

**3.3.4 Waiting times**

Define the waiting time at stage 1 of an arbitrary customer, \( W_1 \), as measured from his arrival instant to the first stage until his service completion at that stage and let \( W_1(t) \) be the corresponding waiting time distribution. When the service of a customer is completed at stage 1, the number of customers in that stage at his departure epoch is equal to the number of customers that arrived during his waiting time at stage 1.
If $\tilde{A}_1(x)$ is defined as the probability generating function of the number of arrivals during the waiting time of an arbitrary customer in stage 1, then

$$
\tilde{A}_1(x) = \sum_{n=0}^{\infty} x^n \int_{0}^{\infty} \frac{e^{-\lambda t} (\lambda t)^n}{n!} dW_1(t)
= \int_{0}^{\infty} e^{-\lambda (1-s)t} dW_1(t)
= W_1^*(\lambda - \lambda x).
$$

(3.47)

where $W_1^*(\cdot)$ is the Laplace-Stieltjes Transform of $W_1(\cdot)$.

On the other hand,

$$
\tilde{A}_1(x) = \sum_{n=0}^{\infty} \left[ \frac{\sum_{j=1}^{\infty} \pi_{nj}^{(1)}}{\sum_{i=0}^{\infty} \sum_{j=1}^{\infty} \pi_{ij}^{(1)}} \right] x^n
= \frac{\Pi_1(x,1)}{\Pi_1(1,1)}
= 2\Pi_1(x,1).
$$

(3.48)

using (3.41).

If $s = \lambda - \lambda x$, then $x = 1 - s/\lambda$ and using (3.47) and (3.48), we will have

$$
W_1^*(s) = 2\Pi_1(1 - s/\lambda,1).
$$

(3.49)

Now, define the total waiting time at stages 1 and 2 of an arbitrary customer, $W_{1,2}$, as measured from his arrival instant to the first stage until his service completion at the second stage and let $W_{1,2}(t)$ be the corresponding waiting time distribution. Also, consider $\tilde{A}_{1,2}(x)$ as the probability generating function of number of arrivals during the waiting time of an arbitrary customer in the system (stages 1 and 2), then

$$
\tilde{A}_{1,2}(x) = W_{1,2}^*(\lambda - \lambda x).
$$

(3.50)

and

$$
\tilde{A}_{1,2}(x) = \sum_{n=0}^{\infty} \left[ \frac{\sum_{i=0}^{n} \pi_{i,n-i}^{(2)}}{\sum_{i=0}^{\infty} \sum_{j=0}^{\infty} \pi_{ij}^{(2)}} \right] x^n
= \frac{\Pi_2(x,x)}{\Pi_2(1,1)}
= 2\Pi_2(x,x).
$$

(3.51)
so that
\[ W_{1,2}^*(s) = 2\Pi_2(1 - s/\lambda, 1 - s/\lambda) \quad . \]  \hspace{1cm} (3.52)

and the average waiting time in stage \( k(k = 1, 2) \), \( W_k \) is
\[ E[W_k] = \left( \frac{2}{\lambda} \right) \frac{d}{dx} [\Pi_k(x, 1)] \bigg|_{x=1} \]
\[ E[W_2] = \left( \frac{2}{\lambda} \right) \frac{d}{dx} [\Pi_2(x, x)] \bigg|_{x=1} - E[W_1] \quad . \]  \hspace{1cm} (3.54)

3.3.5 Two-Stage Tandem Queue with Sequential Service Policy

By a sequential service policy we mean a static policy with \( M_S = 1 \). In other words, when a sequential service policy is applied, then after serving a customer in stage 1, the server switches to stage 2 to complete his service at that stage.

Thus, the equilibrium equations (3.1) and (3.2) reduce to
\[
\pi_{i1}^{(1)} = \pi_{00}^{(2)} d_0^{(2)} p_i^{(1)} + \sum_{m=0}^{i} \sum_{n=0}^{i-m} \pi_{i-n-m+1,0}^{(2)} d_n^{(2)} p_m^{(1)} + \sum_{m=0}^{i} \pi_{00}^{(2)} d_{i-m+1}^{(2)} p_{i-m}^{(1)} \quad : \; i \geq 0 \]  \hspace{1cm} (3.55)
\[
\pi_{i0}^{(2)} = \sum_{m=0}^{i} \sum_{n=0}^{i-m} \pi_{i-m-n,1}^{(1)} d_n^{(1)} p_m^{(2)} \quad : \; i \geq 0 \quad . \]  \hspace{1cm} (3.56)

Defining
\[
\Pi_1(x) = \sum_{n=0}^{\infty} \pi_{i1}^{(1)} x^n
\]
\[
\Pi_2(x) = \sum_{n=0}^{\infty} \pi_{i0}^{(2)} x^n ,
\]

and then multiplying both sides of (3.55) and (3.56) by \( x^i \) and summing over \( i \) separately, we obtain
\[
\Pi_1(x) = \frac{P_1(x)}{x} \left[ D_2(x)\Pi_2(x) - \pi_{00}^{(2)} d_0^{(2)} (1 - x) \right] \]  \hspace{1cm} (3.57)
\[
\Pi_2(x) = \Pi_1(x)D_1(x)P_2(x) .\]  \hspace{1cm} (3.58)

Substituting (3.58) into (3.57) yields,
\[
\Pi_1(x) = \frac{\pi_{00}^{(2)} d_0^{(2)} (x - 1) P_1(x)}{x - D_1(x)D_2(x)P_1(x)P_2(x)} , \]  \hspace{1cm} (3.59)
and since $\Pi_1(1) = 0.5$, the unknown probability $\pi_{00}^{(2)}$ is obtained as:

$$
\pi_{00}^{(2)} = \frac{1 - \bar{\rho}}{2d_0^{(2)}}. 
$$

(3.60)

For the average number of customers in stage 1, $L_1$, we have

$$
L_1 = \frac{d}{dx} \Pi_1(x) \bigg|_{x=1} = \frac{\lambda^2 (E[S_1^2] + E[D_1^2] + E[D_2^2]) + 2(\eta \rho + \rho_1 \rho_2 + \eta_1 \eta_2)}{2(1 - \bar{\rho})} + \rho_1 .
$$

(3.61)

analogous to the standard Pollaczek-Khintchine formula, and it is clear that.

$$
L_2 = \eta_1 + \rho_2 .
$$

(3.62)

### 3.3.6 Average Number of Customers in the System

The average number of customers in stages 1 and 2, $L_1$ and $L_2$, are obtained in Appendix A. In this section only the results are presented for completeness of this chapter.

$$
L_1 = \frac{1}{1 - \rho_1} \left\{ \Pi_2(1,0)[P_1''(1) + D_1''(1) + 2\rho_1 \eta_2] - P_1''(1)[\Pi_1(0,1) + \Psi_{Ms}(1,1)] \\
+ \Theta_{Ms-1}(1)[P_1''(1) + 2\rho_1] + 2\Pi_2''(1,0)[\rho_1 + \eta_2] - 2\rho_1 \Psi_{Ms}^{z'}(1,1) \\
+ 2\rho_1 \pi_{00}^{(2)} d_0^{(2)} - \omega + \frac{P_1''(1)}{2} \right\} .
$$

(3.63)

where $P_1''(1) = \lambda^2 E[S_1^2]$, $D_1''(1) = \lambda^2 E[D_1^2]$ and

$$
\Psi_{Ms}^{z'}(1,1) = (M_S \rho_1 + \eta_2) \Pi_2(1,0) - M_S \Psi_{Ms}(1,1) + \pi_{00}^{(2)} d_0^{(2)} \\
+ \Theta_{Ms-1}(1) - M_S \pi_{0Ms}^{(1)} + \Pi_2'(1,0)
$$

(3.64)

$$
\omega = [M_S P_1''(1) + D_2''(1) + 2M_S \rho_1 \eta_2 + M_S(M_S - 1) \rho_1^2] \Pi_2(1,0) \\
- M_S(M_S - 1) \Psi_{Ms}(1,1) + 2M_S \rho_1 [\pi_{00}^{(2)} d_0^{(2)} + \Theta_{Ms-1}(1)] - M_S(M_S - 1) \pi_{0Ms}^{(1)} \\
+ 2(1 - \rho_1) \Theta_{Ms-1}'(1) + 2(M_S \rho_1 + \eta_2) \Pi_2'(1,0) - 2M_S \Psi_{Ms}^{z'}(1,1)
$$

(3.65)
\[ \Pi_2'(1.0) = \frac{1}{2MS(1 - \rho)} \left\{ 2(\eta_1 + MS\rho)[\pi_{00}^{(2)}d_0^{(2)} + \Theta_{MS-1}(1)] + 2(1 - \rho_1)\Theta_{MS-1}(1) - \Pi_2(1.0)f''(1) \right\} \] (3.66)

\[ f''(1) = MS(MS - 1)(1 - \rho^2) - [D_1''(1) + D_2''(1) + MS(P_1''(1) + P_2''(1))] - 2MS(\eta\rho + \rho_1\rho_2) - 2\eta_1\eta_2. \] (3.67)

For the average number of customers in the second stage, \( L_2 \), we have,

\[ L_2 = L_{1,2} - L_1 \]

where \( L_{1,2} \) is the average total number of customers in stages 1 and 2, and

\[ L_{1,2} = \frac{1}{1 - \rho_2} \left\{ [P_2''(1) + D_1''(1) + 2\rho_2\eta_1]\Psi_{MS}(1,1) - P_2''(1)\Pi_2(1.0) - 2\rho_2\Pi_2'(1.0) + 2(\rho_2 + \eta_1)\Psi_{MS}'(x = 1, x = 1) + [P_2''(1) + D_1''(1) + 2\rho_2\eta_1 + 2MS(\rho_2 + \eta_1) + MS(MS - 1)]\pi_{0,MS}^{(1)} - \delta + \frac{P_2''(1)}{2} \right\} \] (3.68)

in which

\[ \Psi_{MS}'(x = 1, x = 1) = MS\Psi_{MS}(1,1) + \Psi_{MS}'(1,1) \] (3.69)

and

\[ \delta = -[\omega + MS(MS - 1)\Psi_{MS}(1,1) + 2MS\Psi_{MS}'(1,1)]. \] (3.70)

### 3.3.7 Optimal Static Policy

In this section we consider a two-stage tandem queue attended by a moving server with holding costs \( h_2 \geq h_1 \), switching costs \( K_{12} \) and \( K_{21} \), and we want to find the optimal static policy. In other words, in terms of a Queue-Cart problem, we are actually looking for the optimal capacity of the cart, when the full cart strategy is applied and the delivery time depends on the number of items in the cart.

The optimization model is

\[ \text{Min } E[TC_{MS}] = (K_{12} + K_{21})\frac{\lambda}{MS} + h_1L_1(MS) + h_2L_2(MS) \]
where $E[TC_{M_S}]$ is the total average cost per unit time and $L_k(M_S)$ is the average number of customers in stage $k$ when static policy with parameter $M_S$ is applied.

Consider an $M/M_1 - M_2/1$ queue with deterministic switchover times and $h_1 = 1, h_2 = 2$. $K_{12} = 9$ and $K_{21} = 6$. To obtain the average numbers of customers in stages 1 and 2, for different values of $M_S$, model 1 can be used as follows: first Chaudhry [12] is used to find the complex roots of $f(z)$; then the resulting system of linear equations is solved to find the probabilities $\pi_{00}^{(2)}, \pi_{01}^{(1)}, \pi_{02}^{(1)}, \ldots \pi_{0,M_S-1}^{(1)}$. Equation (3.46) yields $\pi_{0,M_S}^{(1)}$, and finally, sections 3.3.6 obtains the average number of customers in stages 1 and 2. These results are summarized in Table 3.1.

### Table 3.1. Characteristics of the $M/M - M/1$ when $\lambda = 1.3(/\text{hour})$, $E[S_1] = 10, E[S_2] = 12, E[D_{12}] = 4, E[D_{21}] = 2 (\text{min})$.

<table>
<thead>
<tr>
<th>$M_S$</th>
<th>Roots of $f(z)$</th>
<th>$\bar{p}$</th>
<th>$L_1$</th>
<th>$L_2$</th>
<th>$L_{1,2}$</th>
<th>$E[TC_{M_S}]$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>0.6067</td>
<td>0.6977</td>
<td>0.3467</td>
<td>1.0493</td>
<td>20.8910</td>
</tr>
<tr>
<td>2</td>
<td>-0.4938</td>
<td>0.5417</td>
<td>0.6957</td>
<td>0.8622</td>
<td>1.5789</td>
<td>12.1701</td>
</tr>
<tr>
<td>3</td>
<td>-0.3432±0.4151i</td>
<td>0.5200</td>
<td>0.6855</td>
<td>1.4108</td>
<td>2.0963</td>
<td>10.0070</td>
</tr>
<tr>
<td>4</td>
<td>-0.5143, -0.1362±0.5742i</td>
<td>0.5092</td>
<td>0.7011</td>
<td>1.9719</td>
<td>2.6731</td>
<td>9.5200*</td>
</tr>
<tr>
<td>5</td>
<td>0.0340±0.6340i</td>
<td>0.5027</td>
<td>0.7271</td>
<td>2.5402</td>
<td>3.2673</td>
<td>9.7075</td>
</tr>
<tr>
<td></td>
<td>-0.4575±0.2664i</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>-0.5214, 0.1768±0.6515i</td>
<td>0.4983</td>
<td>0.7583</td>
<td>3.1130</td>
<td>3.8712</td>
<td>10.2342</td>
</tr>
<tr>
<td></td>
<td>-0.3504±0.4276i</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Some other numerical examples are presented in Table 3.2. The results in this table indicate that the optimal value $M_S$ increases as the arrival rate or switchover times increase. Also, it shows that if switchover times $D_{12}$ and $D_{21}$ can be interchanged, it is always better to design a two-stage tandem queue in which the average switchover time to stage 1 is greater than the average switchover time to stage 2. This is because the customers in stage 2 always wait during switchover time $D_{12}$; however, customers in stage 1 wait during switchover times $D_{12} + D_{21}$.
Table 3.2. Optimal static policy for a system with exponential service times, deterministic switchover times and $h_1 = 1, h_2 = 2, K_{12} = 9$ and $K_{21} = 6$.

<table>
<thead>
<tr>
<th>$\lambda$ (/hour)</th>
<th>$E[S_1]$ (min.)</th>
<th>$E[S_2]$ (min.)</th>
<th>$E[D_{12}]$ (min.)</th>
<th>$E[D_{21}]$ (min.)</th>
<th>$\bar{p}$</th>
<th>$L_1$</th>
<th>$L_2$</th>
<th>$L_{1,2}$</th>
<th>$E[TC_{M_S}]$</th>
<th>$M_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
<td>12</td>
<td>4</td>
<td>2</td>
<td>0.3917</td>
<td>0.4132</td>
<td>1.9042</td>
<td>2.3175</td>
<td>7.9717</td>
<td>4</td>
</tr>
<tr>
<td>1.3</td>
<td>10</td>
<td>12</td>
<td>4</td>
<td>2</td>
<td>0.5092</td>
<td>0.7011</td>
<td>1.9719</td>
<td>2.6731</td>
<td>9.5200</td>
<td>4</td>
</tr>
<tr>
<td>1.6</td>
<td>10</td>
<td>12</td>
<td>4</td>
<td>2</td>
<td>0.6187</td>
<td>1.2007</td>
<td>2.5840</td>
<td>3.7843</td>
<td>11.1683</td>
<td>5</td>
</tr>
<tr>
<td>1.9</td>
<td>10</td>
<td>12</td>
<td>4</td>
<td>2</td>
<td>0.7347</td>
<td>2.0705</td>
<td>2.6057</td>
<td>4.6762</td>
<td>12.9819</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td>12</td>
<td>6</td>
<td>2</td>
<td>0.7867</td>
<td>2.7381</td>
<td>2.6434</td>
<td>5.3815</td>
<td>14.0249</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td>12</td>
<td>8</td>
<td>2</td>
<td>0.8000</td>
<td>2.9482</td>
<td>2.6778</td>
<td>5.6260</td>
<td>14.3038</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td>12</td>
<td>12</td>
<td>2</td>
<td>0.8111</td>
<td>3.2588</td>
<td>3.2899</td>
<td>6.5487</td>
<td>14.8386</td>
<td>6</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>15</td>
<td>2</td>
<td>12</td>
<td>0.7267</td>
<td>2.0917</td>
<td>2.6115</td>
<td>4.7033</td>
<td>13.3148</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>15</td>
<td>12</td>
<td>2</td>
<td>0.7267</td>
<td>2.0917</td>
<td>2.9449</td>
<td>5.0366</td>
<td>13.9815</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>15</td>
<td>4</td>
<td>12</td>
<td>2</td>
<td>0.7111</td>
<td>1.9512</td>
<td>2.9928</td>
<td>4.9440</td>
<td>12.9369</td>
<td>6</td>
</tr>
<tr>
<td>2</td>
<td>15</td>
<td>4</td>
<td>2</td>
<td>12</td>
<td>0.7111</td>
<td>1.9512</td>
<td>2.6595</td>
<td>4.6107</td>
<td>12.2702</td>
<td>6</td>
</tr>
</tbody>
</table>

3.4 Model 2: An $M/G/1$ Queue with Cyclic Service Times

When a static policy with parameter $M_S$ is applied, then during a cycle we have $M_S$ service times $S_1$ and $S_2$ and switchover times $D_{12}$ and $D_{21}$ occurring in the following sequence:

$$\underbrace{S_1, S_1, \ldots, S_1}_{M_S\text{times}}, D_{12}, \underbrace{S_2, S_2, \ldots, S_2}_{M_S\text{times}}, D_{21}$$

In this sequence, the first stage can be viewed as an $M/G/1$ queue in which the service times are changed cyclically according to the customer sequence number. Hence, when a static policy is applied, the queue in the first stage can be analyzed as an $M/G/1$ queue with cyclic service times as follows:

$$\underbrace{S_1, S_1, \ldots, S_1}_{M_S-1\text{times}}, S_1 + D_{12} + \underbrace{S_2 + S_2 + \ldots + S_2}_{M_S\text{times}} + D_{21}$$

In the next section we analyze an $M/G/1$ queue with cyclic service times and introduce a broad range of other capabilities of this model in the optimization analysis of other queueing systems.
3.4.1 Background

Consider a queueing system comprising \( m \) servers. At any time only one of these \( m \) servers is assigned responsibility to serve a customer while the other \( m - 1 \) servers await their turn in an idle server pool. After serving a customer, the server joins the idle server pool, and one of these idle servers then becomes active to process the next customer. Suppose servers are specified by index \( i, (i = 1, 2, \ldots, m) \); then, if the servers become active according to the cyclic order \( 1, 2, \ldots, m, 1, 2, \ldots \), this system can be considered as a single server queue with \( m \) cyclic service times.

The idea of cyclic behaviour of different elements of a queueing system was considered by various authors. Agrawala and Tripathi [1] provided an approach to calculate the steady state customer waiting time distribution for a model with general cyclic independent interarrival times and exponential service times. The idea of cyclic visits of one server to \( N \) independent queues is the basis of polling systems which was summarized in Takagi [63]. To the best of our knowledge, queueing systems with cyclic independent service times were only considered by Morrice et al [48]. They considered a single server queue with FIFO service discipline and exponential interarrival and service times in which the arrival and/or service rates are a deterministic cyclic function of the customer sequence number. They provided steady state results for the mean number in the system for the model with fixed arrival and cyclic service rates. They also provided upper and lower bounds for the steady state mean waiting time in the system.

In the next section we consider the single server queue with homogeneous Poisson arrivals and \( m \) cyclic service times, each having a different general distribution function. Using notation adapted from Agrawala and Tripathi [1] and Morrice et al [48], this model can be designated by \( M/G^m/1 \). Because of the similarity of this notation with bulk queues, we use the notation \( M/G^{(m)}/1 \) in this chapter. In the next section the stationary probability generating function for the number of customers in the system is obtained, whence a closed form expression is found for the expected number in the system. As special cases of \( M/G^{(m)}/1 \), some models in queueing systems with
vacations, tool replacement problems and queue-cart problems will be analyzed. In queueing systems with vacations, the new idea of an $N$-limited service policy in which the server goes on vacation after serving $N$ consecutive customers will be analyzed using $M/G^{(m)}/1$. Coffman and Gilbert [13] studied a queue-cart problem in which the cart delivery time is independent of the number of items in the cart. We will show that our model has the ability to analyze the queue-cart problem with full cart strategy even when the cart delivery times depend on the number of items in the cart.

We also introduce an approximation method to evaluate the average number of customers in an $M/G^{(m)}/1$ system. Using a broad range of examples, we show that this approximation method is very efficient and accurate. Finally, the optimal $N$-limited service policy for a single vacation queueing system will be found by minimizing the total average holding, operating and vacation costs.

3.4.2 Formulation

The $M/G^{(m)}/1$ model can be interpreted from the point of view of arriving customers. Instead of $m$ servers, we can consider $m$ types of customers arriving at a single server system according to a homogeneous Poisson process with rate $\lambda$. The customers are labeled upon arrival as type $i$ : ($i = 1, 2, \ldots, m$) in cyclic order $1, 2, \ldots, m, 1, 2, \ldots$ and are served according to FIFO. Each type $i$ requires service time $T_i$ with distribution $F_i(t)$ and Laplace-Stieltjes transform, $F_i^*(s)$. If the server is capable of providing all the different service types, then this system will be denoted by an $M/G^{(m)}/1$ model.

Let $I_r$ be the type of the $r$th departing customer, $I_r \in \mathcal{I} = \{1, 2, \ldots, m\}$ and let $X_r$ be the number of customers in the system at the $r$th departure epoch, $X_r \in \mathcal{X} = \{0, 1, 2, \ldots\}$. Let $Y_i$ be the number of customers that arrive during the service time of a type $i$ customer. Then,

$$a_{ki} = P\{Y_i = k\} = \int_0^\infty e^{-\lambda t}(\lambda t)^k \frac{dF_i(t)}{k!}.$$

A model for this system can now be represented by the imbedded Markov chain $<X_r, I_r>$, where its one-step transition probability matrix is depicted by the compact
matrix $\mathcal{P}$:

$$\mathcal{P} = \begin{pmatrix}
P_0 & P_1 & P_2 & P_3 & \cdots \\
P_0 & P_1 & P_2 & P_3 & \cdots \\
0 & P_0 & P_1 & P_2 & \cdots \\
0 & 0 & P_0 & P_1 & \cdots \\
\vdots & \vdots & \vdots & \vdots & \ddots \\
0 & 0 & 0 & 0 & \cdots & P_0 \\
\end{pmatrix}$$

in which each submatrix $P_n : n = (0,1,2,\ldots)$ is defined as:

$$P_n = \begin{pmatrix}
0 & a_{n2} & 0 & 0 & \cdots & 0 \\
0 & 0 & a_{n3} & 0 & \cdots & 0 \\
0 & 0 & 0 & a_{n4} & \cdots & 0 \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & 0 & \cdots & a_{nm} \\
0 & 0 & 0 & 0 & \cdots & 0 \\
\end{pmatrix}.$$ 

Assuming that steady state conditions prevail, we have $\lim_{r \to \infty} (X_r, I_r) = (X, I)$ in distribution, and we define the stationary probabilities $\pi_{ni} = P\{X = n, I = i\}$. The stationary balance equations for the imbedded Markov chain $<X, I>$ for $n = 0,1,2,\ldots$ and $i \in \mathcal{I}$ are then

$$\pi_{n1} = \pi_{0m}a_{n1} + \pi_{1m}a_{n1} + \pi_{2m}a_{n-1,1} + \cdots + \pi_{n+1,m}a_{01},$$

$$\pi_{n2} = \pi_{01}a_{n2} + \pi_{11}a_{n2} + \pi_{21}a_{n-1,2} + \cdots + \pi_{n+1,1}a_{02},$$

$$\vdots$$

$$\pi_{nm} = \pi_{0,m-1}a_{nm} + \pi_{1,m-1}a_{nm} + \pi_{2,m-1}a_{n-1,m} + \cdots + \pi_{n+1,m-1}a_{0m}.$$ 

Considering the cyclic nature of services, $\pi_{nm} = \pi_{n0}$, and the balance equations for $n = 0,1,2,\ldots$ and $i \in \mathcal{I}$ can be summarized in

$$\pi_{ni} = \pi_{0,i-1}a_{ni} + \pi_{1,i-1}a_{ni} + \pi_{2,i-1}a_{n-1,i} + \cdots + \pi_{n+1,i-1}a_{0i},$$

and we have

$$\sum_{n=0}^{\infty} \sum_{i=1}^{m} \pi_{ni} = 1.$$  

\footnote{It is shown in [55] page 89 that if the mean number of arrivals during an "average" service time, $\bar{\lambda}$, does not exceed one, then the steady state probabilities for systems with the same matrix structure as $\mathcal{P}$ (the $M/SM/1$ queue) exist.}
3.4.3 Probability Generating Function

Multiplying (3.72) by $z^n$ and then summing over $n$, we obtain

\[
\sum_{n=0}^{\infty} \pi_{n1} z^n = \pi_{0m} \sum_{n=0}^{\infty} a_{n1} z^n + \pi_{1m} \sum_{n=0}^{\infty} a_{n1} z^n + \pi_{2m} z \sum_{n=1}^{\infty} a_{n-1,1} z^{n-1} + \cdots + \\
+ \pi_{jm} z^{j-1} \sum_{n=j-1}^{\infty} a_{n-j+1,1} z^{n-j+1} + \cdots \\
= \pi_{0m} \sum_{n=0}^{\infty} a_{n1} z^n + \pi_{1m} \sum_{n=0}^{\infty} a_{n1} z^n + \sum_{u=2}^{\infty} \pi_{um} z^{u-1} \sum_{n=u-1}^{\infty} a_{n-u+1,1} z^{n-u+1} .
\]

(3.73)

Now introduce generating functions $\Pi_i(z)$ and $A_i(z)$ as follows:

\[
\Pi_i(z) = \sum_{n=0}^{\infty} \pi_{ni} z^n : |z| < 1 \quad i \in \mathcal{I} \\
A_i(z) = \sum_{n=0}^{\infty} a_{ni} z^n : |z| < 1 \quad i \in \mathcal{I} .
\]

Then, (3.73) becomes

\[
\Pi_1(z) = \pi_{0m} A_1(z) + \pi_{1m} A_1(z) + A_1(z) \sum_{u=2}^{\infty} \pi_{um} z^{u-1} .
\]

Finally yielding

\[
z \Pi_1(z) - A_1(z) \Pi_m(z) = A_1(z)(z - 1) \pi_{0m} .
\]

An analogous approach for $\pi_{n2}, \pi_{n3}, \ldots, \pi_{nm}$ will generate the general result for $i = 2, 3, \ldots, m$ as

\[
z \Pi_i(z) - A_i(z) \Pi_{i-1}(z) = A_i(z)(z - 1) \pi_{0,i-1} .
\]

(3.74)

This system of generating function relations can be aggregated into the matrix form

\[
A(z) \Omega(z) = \Psi(z) .
\]

(3.75)

where

\[
\Omega(z) = (\Pi_1(z), \Pi_2(z), \ldots, \Pi_m(z))^T \\
\Psi(z) = (z - 1)(A_1(z) \pi_{0m}, A_2(z) \pi_{01}, \ldots, A_m(z) \pi_{0,m-1})^T
\]
and
\[ A(z) = \begin{pmatrix} 1.2 \ldots & z & 0 & 0 & \ldots & 0 & -A_1(z) \\ -A_2(z) & z & 0 & 0 & \ldots & 0 & 0 \\ 0 & -A_3(z) & z & 0 & \ldots & 0 & 0 \\ 0 & 0 & -A_4(z) & z & \ldots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \ldots & -A_m(z) & z \end{pmatrix}. \]

Using Cramer's rule for \( i = 1, 2, \ldots, m \), we have
\[ \Pi_i(z) = \frac{D_i(z)}{D(z)}. \quad (3.76) \]
in which \( D(z) \) is the determinant of matrix \( A(z) \) and \( D_i(z) \) is the determinant of matrix \( A(z) \) with column \( i \) replaced by vector \( \Psi(z) \).

Considering the Laplace expansion of the first row of matrix \( A(z) \), we obtain
\[ D(z) = z^m - \prod_{i=1}^{m} A_i(z). \quad (3.77) \]

**Lemma 3.1**

\( D_i(z) \) is obtained from
\[ (D_1(z), D_2(z), \ldots, D_m(z))^T = (z - 1)C(z)(\pi_0, \pi_2, \ldots, \pi_{0m})^T \quad (3.78) \]

where
\[ C(z) = \begin{pmatrix} A_1A_2 \ldots A_m & zA_3 \ldots A_mA_1 & z^2A_4 \ldots A_mA_1 & \ldots & z^{m-1}A_1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ z^{m-1}A_2 & A_1A_2 \ldots A_m & zA_4 \ldots A_mA_1 & \ldots & z^mA_1A_2 \\ z^{m-2}A_2A_3 & z^{m-1}A_3 & A_1A_2 \ldots A_m & \ldots & z^{m-2}A_1A_2A_3 \\ z^{m-3}A_2A_3A_4 & z^{m-2}A_3A_4 & z^{m-1}A_4 & \ldots & z^{m-3}A_1A_2A_3 \end{pmatrix}. \]

(and where we have used a simplified representation of matrix \( C(z) \) in which \( z \) is suppressed in \( A_i(z) \).)

**Proof**

Before constructing matrix \( C(z) \), we first denote an equivalent form for \( D_i(z) \) for an \( M/G^{(m)}/1 \) problem with \( m \) types of customers by \( D_i^{(m)}(z) \), and then we show that there is a recursive relation between \( D_i^{(m)}(z) \) and \( D_i^{(m-1)}(z) \).
To show this recursive structure, we first see that

$$D_1^{(m)}(z) = \text{det} \begin{vmatrix} (z-1)A_1(z)\pi_{0m} & 0 & 0 & \ldots & 0 & -A_1(z) \\ (z-1)A_2(z)\pi_{01} & z & 0 & \ldots & 0 & 0 \\ (z-1)A_3(z)\pi_{02} & -A_3(z) & z & \ldots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ (z-1)A_m(z)\pi_{0,m-1} & 0 & 0 & \ldots & -A_m(z) & z \end{vmatrix}$$

or

$$D_1^{(m)}(z) = (z-1)[A_1(z)\pi_{0m}z^{m-1} + (-1)^{1+m}(-A_1(z))\text{det}$$

$$\begin{vmatrix} A_2(z)\pi_{01} & z & 0 & 0 & \ldots & 0 \\ A_3(z)\pi_{02} & -A_3(z) & z & 0 & \ldots & 0 \\ A_4(z)\pi_{03} & 0 & -A_4(z) & z & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ A_m(z)\pi_{0,m-1} & 0 & 0 & 0 & \ldots & -A_m(z) \end{vmatrix}].$$

Rearranging,

$$D_1^{(m)}(z) = z^{m-1}A_1(z)\pi_{0m}(z-1)$$

$$+(-1)^{m+2}A_m(z)(-1)^{m-2}\text{det} \begin{vmatrix} (z-1)A_1(z)\pi_{0,m-1} & 0 & 0 & \ldots & -A_1(z) \\ (z-1)A_2(z)\pi_{01} & z & 0 & \ldots & 0 \\ (z-1)A_3(z)\pi_{02} & -A_3(z) & z & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ (z-1)A_{m-1}(z)\pi_{0,m-2} & 0 & 0 & \ldots & z \end{vmatrix}$$

$$= z^{m-1}A_1(z)\pi_{0m}(z-1) + A_m(z)D_1^{(m-1)}(z).$$

For $m = 1$, we have $D_1^{(1)} = (z-1)A_1(z)\pi_{01}$. Then using (3.76) and (3.77), we can write

$$\Pi_1(z) = \frac{D_1^{(1)}(z)}{D(z)} = \frac{(z-1)A_1(z)\pi_{01}}{z - A_1(z)},$$

and, by using the recursive equation for $m = 2$, we now have

$$D_1^{(2)}(z) = zA_1(z)\pi_{02}(z-1) + A_2(z)[(z-1)A_1(z)\pi_{01}]$$

$$= (z-1)(A_1(z)A_2(z), zA_1(z)) (\pi_{01}, \pi_{02})^T.$$

By continuing to use the recursive equation, the first row of matrix $C(z)$ is generated as: (we use abbreviation $A_i$ in place of $A_i(z)$)

$$(A_1 \ldots A_m, zA_3 \ldots A_mA_1, z^2A_4 \ldots A_mA_1, \ldots, z^{m-1}A_1).$$
The 2nd, 3rd... mth rows of matrix $C(z)$ can also be constructed in the same manner using the cyclic behaviour of the service rates. Because systems with cyclic order 2, 3, ..., $m$, 1, 2, ..., behave in the same way as systems with cyclic order 1, 2, ..., $m$, 1, 2, ..., we consider customer type 2 in the first system as a type 1 customer (and therefore all customers $i$ should be considered as type $i - 1$); and then using the first row of matrix $C(z)$ for customer 2 (which acts like customer type 1), we have

$$D_2^{(m)}(z) = (z - 1)(A_2 \ldots A_m A_1, z A_4 \ldots A_1 A_2, \ldots, z^{m-2} A_1 A_2, z^{m-1} A_2) \begin{pmatrix} \pi_{02} \\ \pi_{03} \\ \vdots \\ \pi_{0m} \\ \pi_{01} \end{pmatrix}$$

or,

$$D_2^{(m)}(z) = (z - 1)(z^{m-1} A_2, A_1 \ldots A_m, z A_4 \ldots A_1 A_2, \ldots, z^{m-2} A_1 A_2) \begin{pmatrix} \pi_{01} \\ \pi_{02} \\ \pi_{03} \\ \vdots \\ \pi_{0m} \end{pmatrix}.$$ 

Comparing this with (3.78) gives the second row of matrix $C(z)$, namely

$$(z^{m-1} A_2, A_1 \ldots A_m, z A_4 \ldots A_1 A_2, \ldots, z^{m-2} A_1 A_2).$$

The other rows of matrix $C(z)$ are generated in the same manner and the proof is now complete. □

Let $\pi_n$ be the steady state probability that $n$ customers are left behind by a departing customer. Then,

$$\pi_n = \sum_{i=1}^{m} \pi_{ni}, \quad (3.79)$$

and considering generating function $\Pi(z)$ as

$$\Pi(z) = \sum_{n=0}^{\infty} \pi_n z^n \quad |z| < 1,$$

we will have

$$\Pi(z) = \sum_{i=1}^{m} \Pi_i(z) = \frac{\sum_{i=1}^{m} D_i(z)}{D(z)}. \quad (3.80)$$
Finally, because of the PASTA property (Wolff [71]), these steady state probabilities at departure epochs are also the steady state probabilities at arbitrary time points.

**Lemma 3.2**

The probability that the server is idle in an $M/G^{(m)}/1$ queue with $m$ types of customers is

$$\pi_0 = 1 - \bar{\rho},$$  \hspace{1cm} (3.81)

in which $\bar{\rho} = (\sum_{i=1}^{m} \rho_i)/m$ and $\rho_i = \lambda E[T_i]$, for $i \in \mathcal{I}$.

**Proof**

Because of factor $(z - 1)$ in $D_i(z)$ and also because $z = 1$ is a root of $D(z)$ in (3.77), both numerator and denominator of (3.80) will be zero when $z = 1$. Furthermore, from (3.77)

$$D'(1) = m - \sum_{i=1}^{m} \frac{d}{dz} A_i(z) \bigg|_{z=1}$$

$$= m - \sum_{i=1}^{m} \lambda E[T_i].$$  \hspace{1cm} (3.82)

Denoting the $j$th row of matrix $C(z)$ by $C_j(z)$, we have from (3.78) that

$$D_j(z) = (z-1)C_j(z)(\pi_0, \pi_0, \ldots, \pi_0)^T,$$

and, therefore,

$$D'_j(1) = C_j(1)(\pi_0, \pi_0, \ldots, \pi_0)^T$$

$$= \pi_0 + \pi_0 + \ldots + \pi_0$$

$$= \pi_0,$$

since $C_j(1) = (1, 1, \ldots, 1)$. Hence using L'Hospital's rule in (3.80) we have

$$\Pi(1) = \sum_{i=1}^{m} \frac{D'_i(1)}{D'(1)}$$

$$= \sum_{i=1}^{m} \frac{\pi_0}{m - \sum_{j=1}^{m} \lambda E[T_j]}$$

$$= \frac{\pi_0}{1 - [(\rho_1 + \rho_2 + \ldots + \rho_m)/m]} = 1.$$
whence
\[ \pi_0 = 1 - \frac{\rho_1 + \rho_2 + \cdots + \rho_m}{m} = 1 - \bar{\rho}. \]

Because \( \Pi(z) \) is an analytic function in the unit circle \( |z| < 1 \), any zero of \( D(z) \) in the unit circle must also be a root of \( \sum_{i=1}^{m} D_i(z) \). Using Takacs [62], Lemma 2, it can be shown that under the condition \( \bar{\rho} \leq 1 \), \( D(z) \) has exactly \( m - 1 \) roots inside the unit circle and one root equal to 1 on the unit circle. Thus, the unknown probabilities \( \pi_{01}, \pi_{02}, \ldots, \pi_{0m} \) can be found for an \( M/G^{(m)}/1 \) model by performing the following steps:

1. Find \( D(z) \) and \( D_i(z) \), \( i \in I \) using equations \( (3.77) \) and \( (3.78) \), respectively.

2. Find the \( m - 1 \) roots of \( D(z) \) that are inside the unit circle \( (|z| < 1) \).

3. These roots are also the roots of \( \sum_{i=1}^{m} D_i(z) \); hence, by putting each of these roots into \( \sum_{i=1}^{m} D_i(z) = 0 \) a relation among the \( \pi_{0i}, i \in I \), will be obtained.

4. The probabilities \( \pi_{01}, \pi_{02}, \ldots, \pi_{0m} \) are then found by solving a system of \( m \) linear equations (\( m - 1 \) linear equations from step 3 along with \( \pi_0 = \pi_{01} + \pi_{02} + \cdots + \pi_{0m} = 1 - \bar{\rho} \)).

**Remark 3.1**

Considering \( \bar{\rho} < 1 \) and \( P\{X = n|I = i\} = \frac{P\{X = n, I = i\}}{P\{I = i\}} = m\pi_{ni} \), we have,

\[ m\pi_{ni} = \int_{0}^{\infty} e^{-\lambda t} (\lambda t)^n \frac{1}{n!} dW_i(t) \]

in which \( W_i(t) \) is the waiting time distribution of customer type \( i \) in the system. Therefore, \( m\Pi_i(z) = W_i^*(\lambda - \lambda z) \), or, substituting \( s = \lambda - \lambda z \), for \( i \in I \), we have \( W_i^*(s) = m\Pi_i(1 - \frac{s}{\lambda}) \), where \( W_i^*(s) \) is the Laplace-Stieltjes transform of \( W_i(t) \).

### 3.4.4 Average Number of Customers in the System

Let \( L \) denote the average number of customers in the system, and let \( C_{kj}(z) \) denote the element of the \( k \)-th row and \( j \)-th column of matrix \( C(z) \). Then, from \( (3.80) \),
\[ L = \frac{d}{dz} \left[ \sum_{i=1}^{m} \frac{D_i(z)}{D(z)} \right] \bigg|_{z=1} . \]

Noting that \( \sum_{k=1}^{m} C_{kj}(z) \) is the summation of the elements of the \( j \)th column of matrix \( C(z) \),

\[ L = \frac{d}{dz} \left[ \sum_{k=1}^{m} C_{k1}(z) \pi_{01} + \sum_{k=1}^{m} C_{k2}(z) \pi_{02} + \cdots + \sum_{k=1}^{m} C_{km}(z) \pi_{0m} (z-1) \right] \bigg|_{z=1} . \]

Set
\[ \varphi_j(z) = \sum_{k=1}^{m} C_{kj}(z) . \] (3.83)

and let.
\[ \Phi(z) = \sum_{i=1}^{m} \varphi_i(z) \pi_{0i} . \] (3.84)

Then.
\[ L = \frac{d}{dz} \left[ \frac{(z-1) \sum_{i=1}^{m} \varphi_i(z) \pi_{0i}}{D(z)} \right] \bigg|_{z=1} = \frac{d}{dz} \left[ \frac{\Phi(z)}{D(z)} \right] \bigg|_{z=1} = \frac{(z-1)[\Phi'(z)D(z) - \Phi(z)D'(z)] + \Phi(z)D(z)}{[D(z)]^2} \bigg|_{z=1} . \] (3.85)

Since both numerator and denominator of (3.85) are zero at \( z = 1 \), applying L'Hospital's rule twice yields:
\[ L = \frac{\Phi''(1)}{D'(1)} - \frac{\Phi'(1)D''(1)}{2[D'(1)]^2} . \] (3.86)

In order to find \( L \), first \( D'(1), D''(1), \Phi(1) \) and \( \Phi'(1) \) must be obtained. Using relation (3.82)
\[ D'(1) = m(1 - \bar{\rho}) = m\pi_0 , \] (3.87)

and from (3.77)
\[ D''(1) = m(m-1) - \left( \sum_{i=1}^{m} A''_i(z) + 2 \sum_{i=1}^{m} A'_i(z) \sum_{j=i+1}^{m} A'_j(z) \right) \bigg|_{z=1} . \]

Since \( A''_i(1) = \lambda^2 E[T_i^2] \) and \( A'_i(1) = \lambda E[T_i] = \rho_i \), we have
\[ \mathcal{D}'(1) = m(m-1) - \lambda^2 \sum_{i=1}^{m} E[T_i^2] - 2 \sum_{i=1}^{m-1} \rho_i \sum_{j=i+1}^{m} \rho_j. \] (3.88)

Now from (3.83), \( \varphi_i(1) \) is the summation of column \( i \) of matrix \( C(z) \) when \( z = 1 \); therefore it is clear that for \( i = 1, 2, \ldots, m \)
\[ \varphi_i(1) = m. \] (3.89)
so that from (3.84) and (3.79)
\[ \Phi(1) = \sum_{i=1}^{m} \varphi_i(1) \pi_0 \]
\[ = m \pi_0. \] (3.90)

To find \( \Phi'(z) \) we first observe that
\[ \varphi_1(z) = \prod_{j=1}^{m} A_j(z) + z^{-1} A_2(z) + z^{-2} A_3(z) + \cdots + z A_2(z) A_3(z) \cdots A_m(z). \]
\[ \varphi_2(z) = \prod_{j=1}^{m} A_j(z) + z^{-1} A_3(z) + z^{-2} A_4(z) + \cdots + z A_3(z) A_4(z) \cdots A_1(z) \]
\[ \cdots \]
\[ \varphi_m(z) = \prod_{j=1}^{m} A_j(z) + z^{-1} A_1(z) + z^{-2} A_2(z) + \cdots + z A_1(z) A_2(z) \cdots A_{m-1}(z). \]
Since \( A'_i(1) = \rho_i \), we will then have
\[ \varphi'_1(1) = \rho_1 + \rho_2 + \cdots + \rho_m + [(m-1) + \rho_2] + [(m-2) + \rho_2 + \rho_3] + \cdots + [1 + \rho_2 + \rho_3 + \cdots + \rho_m] \]
\[ \varphi'_2(1) = \rho_1 + \rho_2 + \cdots + \rho_m + [(m-1) + \rho_3] + [(m-2) + \rho_3 + \rho_4] + \cdots + [1 + \rho_3 + \rho_4 + \cdots + \rho_1] \]
\[ \cdots \]
\[ \varphi'_m(1) = \rho_1 + \rho_2 + \cdots + \rho_m + [(m-1) + \rho_1] + [(m-2) + \rho_1 + \rho_2] + \cdots + [1 + \rho_1 + \rho_2 + \cdots + \rho_{m-1}]. \]
Consequently,
\[ \Phi'(1) = \sum_{i=1}^{m} \varphi'_i(1) \pi_0 \]
\[ = [\rho_1 + \rho_2 + \cdots + \rho_m + (m-1) + (m-2) + \cdots + 1][\pi_{01} + \pi_0 + \cdots + \pi_{0m}] \]
\[ + \sum_{i=1}^{m} \pi_{0i} \sum_{j=1}^{m-1} (m-j) \rho_{i+j} \]
\[ = [m \varphi + \frac{(m-1)m}{2}][1 - \varphi] + \sum_{i=1}^{m} \pi_{0i} \sum_{j=1}^{m-1} (m-j) \theta_{i+j}, \] (3.91)
in which \( \rho_0 = \rho_m \) and \( \varphi_{i+j} = \rho_k \) : \( k \equiv i + j \mod m \) for \( 1 \leq k \leq m \).

Finally, using (3.87), (3.88), (3.90) and (3.91) in (3.86), we have

\[
L = \frac{m(\bar{\rho} + \frac{m-1}{2})(1 - \bar{\rho}) + \sum_{i=1}^{m} \pi_{0i} \sum_{j=1}^{m-1} (m-j) \varphi_{i+j}}{m(1 - \bar{\rho})} \\
- \frac{m \pi_0 m(m-1) - \lambda^2 \sum_{i=1}^{m} E[T_i^2] - 2 \sum_{i=1}^{m-1} \rho_i \sum_{j=i+1}^{m} \rho_j}{2(m \pi_0)^2} \\
= \bar{\rho} + \frac{m-1}{2} + \frac{1}{m(1 - \bar{\rho})} \sum_{i=1}^{m} \pi_{0i} \sum_{j=1}^{m-1} (m-j) \varphi_{i+j} \\
+ \frac{\lambda^2 \sum_{i=1}^{m} E[T_i^2]}{2m(1 - \bar{\rho})} + \frac{\sum_{i=1}^{m-1} \rho_i \sum_{j=i+1}^{m} \rho_j}{m(1 - \bar{\rho})} - \frac{m-1}{2(1 - \bar{\rho})} ,
\]

or.

\[
L = \frac{\lambda^2 \sum_{i=1}^{m} E[T_i^2/m]}{2(1 - \bar{\rho})} + \bar{\rho} + \frac{1}{m(1 - \bar{\rho})} \sum_{i=1}^{m} \pi_{0i} \sum_{j=1}^{m-1} (m-j) \varphi_{i+j} \\
+ \frac{\sum_{i=1}^{m-1} \rho_i \sum_{j=i+1}^{m} \rho_j}{m(1 - \bar{\rho})} - \frac{\bar{\rho}(m-1)}{2(1 - \bar{\rho})} .
\] (3.92)

**Remark 3.2**

Considering \( F_i(t) = F(t) \) , \( i = 1, 2, \ldots, m \), and by using (3.92), the standard Pollaczek-Khinchin formula for the average number of customers in the system is obtained. \( \square \)

### 3.4.5 Applications

In this section we focus on three major applications of \( M/G^{(m)}/1 \):

1. Queues with vacations
2. Replacement models
3. Queue-Cart problems
4. The \( M/G_1 - G_2/1 \) queue with static policy

**Queues with Vacations**

In the following sections we show that the \( M/G^{(m)}/1 \) model has the ability to analyze two particular types of queueing systems with vacations:
1. Single vacation queueing systems with \textit{N-limited} service strategies.

2. Multiple vacation queueing systems with cyclic vacations.

\textbf{Single Vacation Queueing Systems with N-limited Service Strategies}  Consider an $M/G/1$ model in which the server goes on vacation after serving $N$ consecutive customers. A real application of this model may be to a preventive maintenance problem in which the machine requires preventive maintenance after producing $N$ units. The maintenance operation takes time $V$ to complete and the time for producing a unit is $T$ where $V$ and $T$ are independent random variables.

If the service time of a customer, $T$, has distribution function $F(.)$, and the vacation time $V$ has distribution function $F(.)$, then this model can be considered as an $M/G(N)/1$ queue with cyclic service times $T_1, T_2, \ldots, T_N, T, \ldots$ in which

$$T_i = \begin{cases} T & ; i = 1, 2, \ldots, N - 1 \\ T + V & ; i = N \end{cases}$$  \hspace{1cm} (3.93)

Therefore, for this queue we will have

$$\rho_i = \begin{cases} \rho = \lambda E[T] & ; i = 1, 2, \ldots, N - 1 \\ \rho_N = \lambda E[T + V] = \rho + \rho_v & ; i = N \end{cases}$$  \hspace{1cm} (3.94)

and

$$\bar{p} = \frac{1}{N}[(N - 1)\rho + \rho + \rho_v]$$

$$= \rho + \frac{\rho_v}{N}.$$  \hspace{1cm} (3.95)

Since the service time of a type $N$ customer, which we artificially define as $T_N = T + V$ is actually $T$, these customers leave the system after time $T$, and during the following time $V$ there is nobody in service facility. Thus, $\hat{p}_N$ is the fraction of times that there is nobody in the service facility and the server is on vacation. Considering $L$ as the average number of customers in an $M/G(N)/1$ queue with cyclic service times (3.92), the average number of customers in a single vacation queueing system with N-limited strategy, $L_v$, is given by

$$L_v = L - \frac{\rho_v}{N},$$  \hspace{1cm} (3.95)
and, using (3.92),

\[ L_v = \frac{\lambda^2 \left( \frac{(N-1)E[T^2] + E[(T+V)^2]}{N} \right)}{2(1 - \bar{\rho})} + \frac{\sum_{i=1}^{N} \pi_{0i} \sum_{j=i+1}^{N-1} (N - j) \theta_{i+j}}{N(1 - \bar{\rho})} + \frac{\sum_{i=1}^{N} \rho_i \sum_{j=i+1}^{N} \rho_j}{N(1 - \bar{\rho})} + \frac{\bar{\rho}(N - 1)}{2(1 - \bar{\rho})} - \frac{\rho_v}{N}. \]  

(3.96)

Considering relation (3.94) we can deduce that

\[ \sum_{i=1}^{N} \pi_{0i} \sum_{j=i+1}^{N-1} (N - j) \theta_{i+j} = \frac{\rho N(N - 1)(1 - \bar{\rho})}{2} + \rho_v \sum_{i=1}^{N-1} i \pi_{0i} \]  

(3.97)

after considerable algebraic manipulation, and in addition.

\[ \sum_{i=1}^{N} \rho_i \sum_{j=i+1}^{N} \rho_j = \rho_1(\rho_2 + \rho_3 + \cdots + \rho_N) + \rho_2(\rho_3 + \rho_4 + \cdots + \rho_N) + \cdots + \rho_{N-1}(\rho_N) \]

\[ = \rho((N - 1)\rho + \rho_v) + \rho[(N - 2)\rho + \rho_v] + \cdots + \rho[\rho + \rho_v] \]

\[ = \rho \frac{(N - 1)}{2}(N\rho + 2\rho_v). \]  

(3.98)

upon using (3.94). Inserting (3.97) and (3.98) into (3.96) yields

\[ L_v = \frac{\lambda^2 \left( \frac{(N-1)E[T^2] + E[(T+V)^2]}{N} \right)}{2(1 - \bar{\rho})} + \rho \]

\[ + \frac{\rho N(N - 1)(1 - \bar{\rho})}{2} + \rho_v \sum_{i=1}^{N-1} i \pi_{0i} \frac{\rho \frac{N-1}{2}(N\rho + 2\rho_v)}{N(1 - \bar{\rho})} - \frac{\bar{\rho}(N - 1)}{2(1 - \bar{\rho})}, \]

which finally reduces to

\[ L_v = \frac{\lambda^2 \left( E[T^2] + \frac{E[V^2] + 2E[V]E[T]}{N} \right)}{2(1 - \bar{\rho})} + \rho \]

\[ + \frac{\rho_v \sum_{i=1}^{N-1} i \pi_{0i}}{N(1 - \bar{\rho})} - \frac{(N - 1)\rho_v(1 - \rho)}{2N(1 - \bar{\rho})}. \]  

(3.99)

Multiple Vacation Systems with Cyclic Vacations  Consider an M/G/1 queue in which the server has a predetermined schedule for his vacations. The schedule consists of N different vacation times V_1, V_2, ..., V_N which are cyclically repeated after the server returns from vacation time V_N. Vacation time V_i starts after serving N_i consecutive customers (i = 1, 2, ..., N).

Considering service times and vacation times as random variables T and V_i, respectively, this system with cyclic multiple vacations can be analyzed using an
$M/G^{(N+N)}/1$ model in which

$$N = \sum_{i=1}^{N} N_i.$$ 

and the cyclic service times are

$$\overbrace{T, T, \ldots, T, V_1}^{N_1 \text{ times}}, \overbrace{T, T, \ldots, T, V_2}^{N_2 \text{ times}}, \ldots, \overbrace{T, T, \ldots, T, V_N}^{N_N \text{ times}}.$$ 

Therefore,

$$\rho_i = \begin{cases} \rho_{v1} & : i = N_1 + 1 \\ \rho_{v2} & : i = N_1 + N_2 + 2 \\ \vdots & \\ \rho_{vN} & : i = N + N \\ \rho & : \text{otherwise} \end{cases}$$

in which $\rho_{vi} = \lambda E[V_i]$ and $\rho = \lambda E[T]$. It should finally be noted that the $M/G^{(N+N)}/1$ model can be used to analyze queueing systems with cyclic multiple vacations, even when the vacation times have different distributions.

**Tool Replacement Models**

Consider a production tool whose performance characteristics deteriorate with each successive job in such a way that successive jobs require more time on average. Suppose that the tool is changed (or sharpened) after every $N$ jobs. Now, if the time needed to complete the $ith$ job after changing the tool is a random variable $T_i$ with distribution $F_i(.)$ and average $E[T_i]$, where

$$E[T_{i+1}] = d_i E[T_i].$$

($d_i$ is an increasing function of $i$), then the model can be formulated as an $M/G^{(N)}/1$ queue with cyclic service times $T_1, T_2, \ldots, T_N + V, T_1, T_2, \ldots,$ or as an $M/G^{(N+1)}/1$ queue with cyclic service times $T_1, T_2, \ldots, T_N, V, T_1, T_2, \ldots,$ in which $V$ is now a random variable denoting the time needed for changing or sharpening the tool.

**Queue-Cart Problem**

Consider the Queue-Cart problem as an ordinary $M/G/1$ queue serving items that arrive and await service in a buffer of infinite capacity. The server serves items and
put them into a cart. From time to time, the server leaves the queue to deliver the
cart and empty it. During the delivery period new items may arrive and join the
queue.

In different papers on production scheduling such as Coffman et al [14], Dobson
et al [15], Karmarkar [6, 7], different forms of carts such as pallets or carriers were
considered. They all assumed a predetermined service policy and delivery times con-
fined to a given collection of items all available at time zero. Coffman and Gilbert
[3] studied a more realistic case in which items arrive according to a Poisson process.
and random service times are considered. In an initial stochastic model of this type.
Karmarkar [7] studied the waiting time using an approximation based on the $M/M/1$
quickest. Coffman and Gilbert [13] considered cart delivery strategies as follows: deliv-
ery begins when $N$ items are in the cart, or when the queue is empty and at least
$M$ items are in the cart ($M \leq N$). When $M = N$ the strategies are called full cart
strategies, because with the cart holding $N$ items, the server delivers the cart when it
is full. Strategies with $N = \infty$ are empty queue strategies, and strategies with $M = 0$
are called never-idle strategies. Coffman and Gilbert [3] found the stationary proba-
bility generating function for a model of an $M/G/1$ Queue-Cart system in which the
cart delivery times are independent of the number of items in the cart. They applied
their results for full cart, empty queue and never-idle strategies.

In real cases the cart delivery time depends on the number of units in the cart,
because as the number of items in the cart increases, the cart will be heavier and
may require more time to be transferred and unloaded. To the best of our knowledge
there are no studies which consider the dependency between the cart delivery time
and the number of items in the cart. Fortunately, the $M/G^{(m)}/1$ model studied here
can be used to analyze this problem when the full cart strategy is applied.

Suppose the capacity of the cart is $N$. Then, considering the full cart strategy
and service time $T$ for items in the queue and also delivery time $V_N$, the problem can
be formulated as an $M/G/1$ queue with vacation in which the server goes on vacation
after $N$ service completions and the vacation time is a random variable $V_N$. This
problem can then be analyzed using an $M/G^{(m)}/1$ model as described in Section 3.1.
The $M/G_1 - G_2/1$ Queue with Static Policy

As it was described, when a static policy in the first stage and a greedy and exhaustive policy in the second stage are implemented in an $M/G_1 - G_2/1$ queue, then during a cycle which serves $M_S$ customers, the sequence of the service and switchover times are cyclically changed as follow:

\[
\begin{array}{c}
\overset{M_{\text{times}}}{S_1, S_1, \ldots, S_1} , D_{12} , \\
\overset{M_{\text{times}}}{S_2, S_2, \ldots, S_2} , D_{21}.
\end{array}
\]

By focusing on the first queue, the performance analysis of the first stage can be studied using an $M/G(M_S)/1$ queue with cyclic service times

\[
\dot{S}_i = \begin{cases} 
S_i & : i = 1, 2, \ldots, M_S - 1 \\
S_{i} + D_{12} + S_2 + S_2 + \cdots + S_2 + D_{21} & : i = M_S .
\end{cases}
\]  

(3.100)

This is actually a single vacation queueing system with $M_S$-limited service strategy where vacation time $\nu = S_2 + S_2 + \cdots + S_2 + D_{12} + D_{21}$, or

\[ \rho_\nu = M_S \rho_2 + \eta . \]

Therefore, the average number of customers in the first stage of the $M/G_1 - G_2/1$ model can be obtained using (3.99).

To obtain the average number of customers in the second stage, the interdeparture times of stage 1 must first be analyzed. Let $T_{M_S}^{cy}$ be the cycle time elapsed from the server’s arrival at stage 1 to his next arrival at stage 1 when static policy with parameter $M_S$ is applied. Also, suppose $R_{i, i+1}^{(1)}$ denotes the time between $i$th and $(i + 1)$th customer departures from stage 1. Thus, the average number of customers in the second stage when static policy with parameter $M_S$ is applied, $L_2(M_S)$, is

\[
L_2(M_S) = \frac{1}{E[T_{M_S}^{cy}]} \left( \sum_{i=1}^{M_S-1} i E[R_{i, i+1}^{(1)}] + M_S E[D_{12}] + \frac{M_S(M_S + 1)}{2} E[S_2] \right) , \tag{3.101}
\]

in which

\[ E[T_{M_S}^{cy}] = \frac{M_S}{\lambda} . \]

Hence, it only remains to obtain the average interdeparture times from stage 1. Suppose $\tau_k^{(M_S)}$ is the time measured from the start of a cycle to the time that the $k$th
(customer in that cycle departs from stage 1, when static policy with parameter $M_S$ is implemented in the $M/G_1 - G_2/1$ queue. Also, consider $t_k^{(M_S)}$ as the departure time of the $k$th customer ($k = 1, 2, \ldots, M_S$) in our $M/G^{(M_S)}/1$ queue measured from the start of each cycle (the time that the server finishes the service of the type $M_S$ customer). Then considering Figure 3.1, we will have

$$
\tau_{k+1}^{(M_S)} - \tau_k^{(M_S)} = \begin{cases} 
 t_{k+1}^{(M_S)} - t_k^{(M_S)} & : k = 1, 2, \ldots, M_S - 2 \\
 t_{k+1}^{(M_S)} - (S_2 + S_2 + \cdots + S_2 + D_{12} + D_{21}) & : i = M_S - 1.
\end{cases}
$$

Hence, for $i = 1, 2, \ldots, M_S - 2$,

$$
E[R_{i,i+1}^{(1)}] = E[t_{i+1}^{(M_S)} - t_i^{(M_S)} | I = i] \\
= \sum_{n=0}^{\infty} E[t_{i+1}^{(M_S)} - t_i^{(M_S)} | I = i, X = n]P\{X = n | I = i\} \\
= (\frac{1}{\lambda} + E[S_1])P\{X = 0 | I = i\} + E[S_1](1 - P\{X = 0 | I = i\}) \\
= (\frac{1}{\lambda} + E[S_1])m\pi_0i + E[S_1](1 - m\pi_0i) \\
= \frac{m}{\lambda} \pi_0i + E[S_1], \tag{3.102}
$$

and

$$
E[R_{M_S-1,M_S}^{(1)}] = E[t_{M_S}^{(M_S)} - t_{M_S-1}^{(M_S)} - (S_2 + S_2 + \cdots + S_2 + D_{12} + D_{21}) | I = i] \\
= \frac{m}{\lambda} \pi_{0,M_S-1} + E[S_1]. \tag{3.103}
$$
Using (3.102) and (3.103) in (3.101), the average number of customers in stage 2 is obtained and the optimization problem to find the optimal limit \( M_S \) is the same as (3.71).

### 3.4.6 Approximation Model for \( M/G^{(m)}/1 \)

In section 3.4.3, we noted that the first step in solving the \( M/G^{(m)}/1 \) model requires determination of the \( \pi_{0i}, \ (i = 1, 2, \ldots, m) \). In order to find these probabilities, \( m - 1 \) roots of \( D(z) \) inside the unit circle \( |z| < 1 \) must be evaluated. However, this approach is a time consuming process that reduces the flexibility of using the \( M/G^{(m)}/1 \) model in optimization problems. Therefore, in this section an approximation method is introduced to find \( \pi_{0i} \) which then will be used to compute the average number of customers in the system. Using numerical examples, we will show that this approximation method is very efficient in providing accurate results.

Consider an \( M/G^{(m)}/1 \) model with \( m \) types of customers. If the service rate of customer type \( j \) \( (\mu_j = \frac{1}{E[T_j]}, \ j = 1, 2, \ldots, m) \) were to increase to \( \mu_j + \delta_j \), the customer would receive his service faster and the probability that he leaves an empty system \( (\pi_{0j}) \) would increase to \( \pi_{0j} + \Delta_{\pi_{0j}}^i \). This would also increase the probability that customer type \( j + 1 \), who always waits behind customer type \( j \), leaves an empty system \( (\pi_{0j+1} + \Delta_{\pi_{0j+1}}^i) \). In other words, if \( \mu_i \) were increased to \( \mu_i + \delta_i \), then all \( \pi_{0j} \) would increase to \( \pi_{0j} + \Delta_{\pi_{0j}}^i \) in such a way that usually,

\[
\Delta_{\pi_{0j}}^i \geq \Delta_{\pi_{0,j+1}}^i \geq \Delta_{\pi_{0,j+2}}^i \geq \ldots \geq \Delta_{\pi_{0m}}^i \geq \Delta_{\pi_{01}}^i \geq \ldots \geq \Delta_{\pi_{0,j-1}}^i. \tag{3.104}
\]

We now introduce \( \hat{\pi}_{0j} \) as an approximation for \( \pi_{0j} \) \((j = 1, 2, \ldots, m)\) as follows:

\[
\hat{\pi}_{0j} = \Theta_j \pi_0, \tag{3.105}
\]

in which

\[
\Theta_j = \frac{\theta_m \mu_j + \theta_{m-1} \mu_{j-1} + \cdots + \theta_{m-j+1} \mu_1 + \theta_{m-j} \mu_m + \cdots + \theta_2 \mu_j + 2 + \theta_1 \mu_{j+1}}{\sum_{i=1}^m \theta_i (\mu_1 + \mu_2 + \cdots + \mu_m)},
\]

and \( \theta_i \) is a nondecreasing function of \( i \).
The main idea for this approximation is that the values of $\mu_1, \mu_2, \ldots, \mu_m$ play the primary role in distributing amount $\pi_0$ among the $\pi_{0j}$, $j = 1, 2, \ldots, m$: and considering each $\pi_{0j}$, these values, in order of their importance, are

$$\mu_j, \mu_{j-1}, \ldots, \mu_1, \mu_m, \mu_{m-1}, \ldots, \mu_{j+2}, \mu_{j+1}$$

Therefore, $\mu_j$ in $\Theta_j$ has the highest coefficient, $\theta_m$, and $\mu_{j+1}$ has the lowest coefficient, $\theta_1$, in the numerator of $\Theta_j$. Thus, $\pi_{0j}$ ($j = 1, 2, \ldots, m$) satisfies equation (3.104) as well as

$$\sum_{j=1}^{m} \hat{\pi}_{0j} = \pi_0,$$

since

$$\sum_{j=1}^{m} \Theta_j = 1.$$

Here we consider two different functions $\theta_i^{(1)}$ and $\theta_i^{(2)}$, as simple linear and exponential functions.

$$\theta_i^{(1)} = i \quad ; \quad i = 1, 2, \ldots, m \quad (3.106)$$

$$\theta_i^{(2)} = m^{i-1} \quad ; \quad i = 1, 2, \ldots, m$$

Then,

$$\hat{\Theta}_j^{(1)} = \frac{m \mu_j + (m - 1) \mu_{j-1} + \cdots + (m - j + 1) \mu_1 + (m - j) \mu_m + \cdots + 2 \mu_{j+2} + \mu_{j+1}}{m(m+1) \sum_{j=1}^{m} (\mu_1 + \mu_2 + \cdots + \mu_m)}$$

$$\hat{\Theta}_j^{(2)} = \frac{m^{m-1} \mu_j + m^{m-2} \mu_{j-1} + \cdots + m^{m-j} \mu_1 + m^{m-j-1} \mu_m + \cdots + m \mu_{j+2} + \mu_{j+1}}{m^{m-1} \sum_{j=1}^{m} (\mu_1 + \mu_2 + \cdots + \mu_m)}$$

By using different examples in the next section, we will show that the approach which chooses $\theta_i = i$ is very efficient and provides good results in approximating the average number of customers in the system.

### 3.4.7 Numerical Examples

In this section we use $\hat{\pi}_{0j}$ in (3.92) to approximate the average number of customers in the system ($\hat{L}$). For each example, Chaudhry [12] was used to find the roots of $\mathcal{D}(z)$ and then the resulting system of linear equations was solved to find the real values of the $\pi_{0j}$ which were then used to find the true average number of customers.
in the system \( (L) \). We considered three types of problems, \( M/M^{(3)}/1, M/E_k^{(3)}/1 \) and \( M/HE_2^{(3)}/1 \), in which the coefficients of variation of service times, \( C_v \), are \( C_v = 1, C_v < 1 \) and \( C_v > 1 \) respectively. The approximation approach using \( \theta_i^{(1)} \) and \( \theta_i^{(2)} \) was then applied to a broad range of problems and in each case, the resulting \( \hat{L} \) was compared with the corresponding exact result \( (L) \) using the Relative Error \( (RE) \)

\[
RE = \frac{L - \hat{L}}{L}.
\]

Some of these results are presented in Tables 3.3, 3.4 and 3.5.

We find that the average absolute Relative Errors is around 1.5\% and 2.5\%, respectively, for \( \theta_i^{(1)} \) and \( \theta_i^{(2)} \). In general, the performance of both approximations is similar when \( C_v < 1 \); however, when \( C_v \geq 1 \), \( \theta_i^{(1)} \) provides better approximations than \( \theta_i^{(2)} \). Overall, we may therefore conclude that \( \theta_i^{(1)} \) is superior to \( \theta_i^{(2)} \). This same conclusion was also found when comparing \( \theta_i^{(1)} \) with other approximations as well.
Table 3.3. Comparison of Relative Error $(RE)$ for $\theta_i^{(1)} = i$ and $\theta_i^{(2)} = m^{-1}$ in an $M/M^{(3)}/1$.

<table>
<thead>
<tr>
<th>$\lambda$</th>
<th>$(\mu_1, \mu_2, \mu_3)$</th>
<th>$\pi_0$</th>
<th>$L$</th>
<th>$\theta_i^{(1)} = i$</th>
<th>$\theta_i^{(2)} = m^{-1}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>(4.6, 14)</td>
<td>0.0238</td>
<td>45.0654</td>
<td>+0.01</td>
<td>-0.08</td>
</tr>
<tr>
<td>5</td>
<td>(4.6, 20)</td>
<td>0.2222</td>
<td>3.9215</td>
<td>-0.08</td>
<td>-1.29</td>
</tr>
<tr>
<td>3</td>
<td>(19, 17, 10)</td>
<td>0.7885</td>
<td>0.2721</td>
<td>-0.44</td>
<td>-1.21</td>
</tr>
<tr>
<td>2</td>
<td>(1.8, 4)</td>
<td>0.0833</td>
<td>14.6837</td>
<td>+0.17</td>
<td>-0.13</td>
</tr>
<tr>
<td>7</td>
<td>(11, 13, 12)</td>
<td>0.4139</td>
<td>1.4181</td>
<td>+0.01</td>
<td>-0.03</td>
</tr>
<tr>
<td>10</td>
<td>(20, 7, 16)</td>
<td>0.1488</td>
<td>6.3231</td>
<td>+0.14</td>
<td>-0.24</td>
</tr>
<tr>
<td>8</td>
<td>(12, 7, 13)</td>
<td>0.1917</td>
<td>4.3751</td>
<td>+0.04</td>
<td>-0.14</td>
</tr>
<tr>
<td>1</td>
<td>(1.8, 4)</td>
<td>0.5417</td>
<td>1.0211</td>
<td>-0.08</td>
<td>-2.21</td>
</tr>
<tr>
<td>19</td>
<td>(97, 21, 12)</td>
<td>0.1053</td>
<td>10.0777</td>
<td>-0.14</td>
<td>-0.60</td>
</tr>
<tr>
<td>4</td>
<td>(83, 39, 2)</td>
<td>0.2831</td>
<td>4.1492</td>
<td>+1.41</td>
<td>-0.11</td>
</tr>
<tr>
<td>2</td>
<td>(1.4, 8)</td>
<td>0.2333</td>
<td>5.0343</td>
<td>-0.06</td>
<td>-0.73</td>
</tr>
<tr>
<td>3</td>
<td>(47, 54, 2)</td>
<td>0.4602</td>
<td>1.7819</td>
<td>+1.84</td>
<td>-2.02</td>
</tr>
</tbody>
</table>

Table 3.4. Comparison of Relative Error $(RE)$ for $\theta_i^{(1)} = i$ and $\theta_i^{(2)} = m^{-1}$ in an $M/E_k^{(3)}/1$.

<table>
<thead>
<tr>
<th>$\lambda$</th>
<th>$(\mu_1, \mu_2, \mu_3)$</th>
<th>$(k_1, k_2, k_3)$</th>
<th>$\pi_0$</th>
<th>$L$</th>
<th>$\theta_i^{(1)} = i$</th>
<th>$\theta_i^{(2)} = m^{-1}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>(19, 17, 10)</td>
<td>(2.5, 3)</td>
<td>0.7885</td>
<td>0.2518</td>
<td>-0.37</td>
<td>-1.20</td>
</tr>
<tr>
<td>2</td>
<td>(1, 8, 4)</td>
<td>(9, 7, 2)</td>
<td>0.0833</td>
<td>7.2991</td>
<td>+1.48</td>
<td>+0.89</td>
</tr>
<tr>
<td>6</td>
<td>(3.15, 15)</td>
<td>(8, 12, 10)</td>
<td>0.0667</td>
<td>8.9691</td>
<td>+1.28</td>
<td>+0.67</td>
</tr>
<tr>
<td>6</td>
<td>(4.6, 14)</td>
<td>(2.8, 20)</td>
<td>0.0238</td>
<td>29.8577</td>
<td>+0.06</td>
<td>-0.07</td>
</tr>
<tr>
<td>10</td>
<td>(20, 7, 16)</td>
<td>(10, 5, 16)</td>
<td>0.1488</td>
<td>3.8546</td>
<td>+0.80</td>
<td>+0.18</td>
</tr>
<tr>
<td>8</td>
<td>(12, 7, 13)</td>
<td>(8, 20, 5)</td>
<td>0.1917</td>
<td>2.7055</td>
<td>+0.46</td>
<td>+0.17</td>
</tr>
<tr>
<td>0.5</td>
<td>(5, 10, 7)</td>
<td>(18, 6, 4)</td>
<td>0.9262</td>
<td>0.0773</td>
<td>-0.96</td>
<td>-1.86</td>
</tr>
<tr>
<td>1</td>
<td>(1, 1, 16)</td>
<td>(1, 1, 16)</td>
<td>0.3125</td>
<td>2.5656</td>
<td>-1.24</td>
<td>-4.10</td>
</tr>
<tr>
<td>7</td>
<td>(11, 13, 12)</td>
<td>(4, 4, 1)</td>
<td>0.4139</td>
<td>1.2090</td>
<td>+0.06</td>
<td>+0.02</td>
</tr>
<tr>
<td>5</td>
<td>(18, 11, 10)</td>
<td>(5, 7, 3)</td>
<td>0.5894</td>
<td>0.5937</td>
<td>-0.15</td>
<td>-0.74</td>
</tr>
<tr>
<td>20</td>
<td>(24, 15, 60)</td>
<td>(17, 8, 26)</td>
<td>0.1667</td>
<td>3.2378</td>
<td>+0.49</td>
<td>-0.28</td>
</tr>
<tr>
<td>9</td>
<td>(96, 60, 90)</td>
<td>(4, 2, 15)</td>
<td>0.5854</td>
<td>0.6519</td>
<td>+3.96</td>
<td>-0.14</td>
</tr>
</tbody>
</table>

Table 3.5. Comparison of Relative Error $(RE)$ for $\theta_i^{(1)} = i$ and $\theta_i^{(2)} = m^{-1}$ in an $M/H E_k^{(3)}/1$.

<table>
<thead>
<tr>
<th>$\lambda$</th>
<th>$(p_1, \mu_{11}, \mu_{12})$</th>
<th>$(p_2, \mu_{21}, \mu_{22})$</th>
<th>$(p_3, \mu_{31}, \mu_{32})$</th>
<th>$\pi_0$</th>
<th>$L$</th>
<th>$\theta_i^{(1)} = i$</th>
<th>$\theta_i^{(2)} = m^{-1}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(0.2, 25, 5)</td>
<td>(0.4, 25, 5)</td>
<td>(0.5, 10, 10)</td>
<td>0.8173</td>
<td>0.2344</td>
<td>-1.22</td>
<td>-2.64</td>
</tr>
<tr>
<td>27</td>
<td>(0.64, 87, 22)</td>
<td>(0.18, 78, 72)</td>
<td>(0.3, 6, 44)</td>
<td>0.0701</td>
<td>31.9852</td>
<td>-0.04</td>
<td>-0.21</td>
</tr>
<tr>
<td>13</td>
<td>(0.9, 7, 37)</td>
<td>(0.37, 84, 93)</td>
<td>(0.02, 9, 39)</td>
<td>0.2642</td>
<td>4.0641</td>
<td>+0.27</td>
<td>-0.65</td>
</tr>
<tr>
<td>35</td>
<td>(0.73, 79, 70)</td>
<td>(0.71, 53, 71)</td>
<td>(0.82, 77, 25)</td>
<td>0.4350</td>
<td>1.4125</td>
<td>-0.05</td>
<td>-0.16</td>
</tr>
<tr>
<td>21</td>
<td>(0.21, 35)</td>
<td>(0.48, 27, 47)</td>
<td>(0.2, 35, 58)</td>
<td>0.4627</td>
<td>1.1981</td>
<td>-0.08</td>
<td>-0.29</td>
</tr>
<tr>
<td>2</td>
<td>(0.2, 5, 2.5)</td>
<td>(0.2, 70, 125)</td>
<td>(0.5, 15, 15)</td>
<td>0.2870</td>
<td>3.3946</td>
<td>-0.92</td>
<td>-2.11</td>
</tr>
<tr>
<td>7</td>
<td>(0.2, 15, 10)</td>
<td>(0.4, 10, 15)</td>
<td>(0.15, 16, 8)</td>
<td>0.3768</td>
<td>1.7185</td>
<td>-0.01</td>
<td>-0.04</td>
</tr>
<tr>
<td>43</td>
<td>(0.05, 9, 61)</td>
<td>(0.12, 33, 97)</td>
<td>(0.13, 89, 34)</td>
<td>0.1273</td>
<td>9.5694</td>
<td>+0.01</td>
<td>-0.14</td>
</tr>
<tr>
<td>4</td>
<td>(0.32, 35, 94)</td>
<td>(0.83, 54, 89)</td>
<td>(0.43, 72, 35)</td>
<td>0.9254</td>
<td>0.0814</td>
<td>-2.06</td>
<td>-4.30</td>
</tr>
<tr>
<td>2</td>
<td>(1, 1, 4)</td>
<td>(0.2, 20, 5)</td>
<td>(0.3, 10, 10)</td>
<td>0.0330</td>
<td>37.7521</td>
<td>+0.08</td>
<td>-0.03</td>
</tr>
<tr>
<td>5</td>
<td>(0.5, 8, 28)</td>
<td>(0.9, 10, 20)</td>
<td>(0.2, 30, 5)</td>
<td>0.4300</td>
<td>1.5037</td>
<td>-0.01</td>
<td>-0.53</td>
</tr>
<tr>
<td>18</td>
<td>(0.63, 7, 23)</td>
<td>(0.89, 64, 34)</td>
<td>(0.44, 42, 53)</td>
<td>0.1344</td>
<td>10.4264</td>
<td>+0.17</td>
<td>-0.31</td>
</tr>
</tbody>
</table>
3.4.8 Optimal N-limited Strategy for a Single Vacation Queueing System

Consider an $M/G/1$ model in which the server goes on vacation after serving $N$ consecutive customers. Suppose that decisions about $N$, as well as the average vacation time, $E[V]$, must be made in order to minimize the total average holding and vacation costs.

Let:

- $h$ = The holding cost of each waiting customer per unit time
- $R(V)$ = The cost per unit time while the server is on vacation
- $K$ = The normal operating cost per unit time
- $O(N)$ = The average of the additional operating costs per unit time while the system is working with N-limited strategy
- $O(\infty)$ = The average of the additional operating costs per unit time while the system is working without vacation.

This problem can be analyzed as an $M/G^{(N)}/1$ model with the total average cost per unit time ($E[TC]$),

$$
E[TC] = hL_v + R(V)\frac{\rho_v}{N} + [K + O(N)]\rho .
$$

in which $L_v$ is obtained from (3.99), $\frac{\rho_v}{N}$ is the fraction of time that the server spends on vacation and $\rho$ is the fraction of time that system is working.

For $E[V] = 0$, the system will behave like an $M/G/1$ system with total average cost per unit time ($E[TC]$),

$$
E[TC] = hL + [K + O(\infty)]\rho .
$$

For $E[V] > 0$, the exact $M/G^{(N)}/1$ model cannot be effectively used to find the optimal values of $N$ and $E[V]$ analytically. This is so because for each value of $N$ and $E[V]$, the roots of $D(z)$ must be determined, and this can only be accomplished practically by implementing a numerical procedure. Then a search process must be carried
out over feasible values of \( N \) and \( E[V] \). Instead of this time consuming approach which also requires the distribution function of the vacation time, the approximation method only needs the first and second moments of the vacation time to obtain an analytic form.

Using equation (3.106) and considering

\[
\mu_i = \begin{cases} 
\frac{\mu}{E[T]} & : i = 1, 2, \ldots, N - 1 \\
\frac{1}{E[T] + E[V]} & : i = N 
\end{cases}
\]

we have.

\[
\hat{\tau}^{(1)}_i = \begin{cases} 
\frac{N(N+1)\mu + (N-1)(\mu - \mu)}{2(NN+1)[(N-1)\mu + \mu]} & : i = 1, 2, \ldots, N - 1 \\
\frac{N(N+1)\mu + N(\mu - \mu)}{2(NN+1)[(N-1)\mu + \mu]} & : i = N 
\end{cases}
\]

and therefore.

\[
\sum_{i=1}^{N-1} i \hat{\tau}_{0i} = \sum_{i=1}^{N-1} (1 - \bar{p}) \frac{N(N+1)\mu + (N-1)(\mu - \mu)}{2(NN+1)[(N-1)\mu + \mu]} \sum_{i=1}^{N-1} i + (\mu_v - \mu) \sum_{i=1}^{N-1} i(N - i)
\]

\[
= (1 - \bar{p})(N - 1) \frac{N\mu + (\mu_v - \mu)}{2} + \frac{(\mu_v - \mu)}{3}.
\]  

(3.109)

Considering (3.109) after some algebra,

\[
\sum_{i=1}^{N-1} i \hat{\tau}_{0i} = (1 - \bar{p}) \frac{3NE[T] + (3N - 2)E[V]}{6(\frac{N}{N-1}E[T] + E[V])},
\]

or

\[
\sum_{i=1}^{N-1} i \hat{\tau}_{0i} = (1 - \bar{p}) \frac{3N\rho + (3N - 2)\rho_v}{6(\frac{N}{N-1}\rho + \rho_v)}.
\]

Finally, using (3.99), the average number of customers in the system can be approximated by

\[
\bar{L}_v = \frac{\lambda^2(NE[T^2] + E[V^2]) + (1 - \rho)[2N\rho - (N - 1)\rho_v]}{2N(1 - \rho) - 2\rho_v} + \frac{\rho_v(3N\rho + (3N - 2)\rho_v)}{6N(\frac{N}{N-1}\rho + \rho_v)}
\]  

(3.110)
Therefore, using the approximation method, the optimization problem is:

\[
\begin{align*}
\text{Min } E[TC_v] &= h\bar{L}_v + R(V)\frac{\rho_v}{N} + [K + O(N)]\rho \\
\text{Subject to:} & \\
\rho + \frac{\rho_v}{N} < 1 \\
\mathcal{N} &\in \mathcal{N}, \ E[V] \geq 0
\end{align*}
\]

where \( \mathcal{N} = \{2, 3, \ldots\} \).

This model can be widely used in maintenance and tool replacement problems. Suppose that the server is a production machine which requires a preventive maintenance operation (readjustment or tool replacement) after producing \( N \) consecutive items. Considering the maintenance, operating and holding costs, the problem is to find the optimal number of items that should be produced before each maintenance operation and also the optimal average time for maintenance operations. In this case, the maintenance cost (vacation cost) is assumed to be a nonincreasing function of average maintenance operation time, \( E[V] \), because usually the faster maintenance operation results in higher costs. On the other hand, operating cost \( O(N) \) is considered as a nondecreasing function of the number of items processed \( (N) \) between maintenance operations.

**Example 3.1**

Suppose a tool replacement operation is to be planned for a production machine in a work station. Items arrive at the work station according to a homogeneous Poisson process with rate \( \lambda = 6 \) and the production times are exponential random variable with mean \( E[T] = 0.1 \). The holding cost for each item per unit time is \( h = 10 \), and the normal operating cost per unit time is \( K = 3000 \). The average additional operating cost per unit time is

\[
O(N) = 350 - \frac{350}{N}.
\]

The tool replacement time is assumed to be an exponential random variable with average \( E[V] \), and the average cost per unit time of using the tool replacement operation
with average time $E[V]$ is

$$R(V) = 600 - 3000E[V].$$

where the fastest and slowest tool replacement operation have average times $E[V_F] = .08$ and $E[V_S] = .18$, respectively. The optimal number of items that should be produced by each tool and the optimal average tool replacement time should be found in order to minimize the total average holding, operating and tool replacement costs.

Using (3.111) the optimization problem will be

$$\text{Min } E[TC_v] = \frac{720(E[V])^2 + (96 - 24N)E[V] + 7.2N}{0.8N - 12E[V]} + \frac{3NE[V] + 10(3N - 2)(E[V])^2}{0.1(\frac{N^2}{N-1}) + .NE[V]} + \frac{1}{N}(3600E[V] - 18000(E[V])^2 - 288) + 2094$$

Subject to:

$$\frac{E[V]}{N} \leq 0.6666$$

$$0.08 \leq E[V] \leq 0.18$$

$$N \in \mathcal{N}$$

(3.112)

Table 3.6 depicts the optimal average replacement time, $E[V]^*$, and related costs for feasible values of $N$. As shown, producing less than 5 items with each tool requires the faster tool replacement operation, and producing 5 items or more needs the slower replacement operation in order to minimize the total average costs. But, the optimal replacement policy is to replace tools after producing $N^* = 7$ items, using the slowest replacement operation $E[V]^* = 0.18$ with minimum total average cost per unit time $E[TC_v]^* = 2017.71$. 
Table 3.6. Optimal costs for different tool replacement policies.

<table>
<thead>
<tr>
<th>N</th>
<th>$E[V]^*$</th>
<th>Holding costs</th>
<th>Tool replacement costs</th>
<th>Operating costs</th>
<th>Total costs</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>0.08</td>
<td>43.01</td>
<td>86.40</td>
<td>1905.00</td>
<td>2034.41</td>
</tr>
<tr>
<td>3</td>
<td>0.08</td>
<td>27.23</td>
<td>57.60</td>
<td>1940.00</td>
<td>2024.83</td>
</tr>
<tr>
<td>4</td>
<td>0.08</td>
<td>22.79</td>
<td>43.20</td>
<td>1957.50</td>
<td>2023.49</td>
</tr>
<tr>
<td>5</td>
<td>0.18</td>
<td>40.43</td>
<td>12.96</td>
<td>1968.00</td>
<td>2021.39</td>
</tr>
<tr>
<td>6</td>
<td>0.18</td>
<td>32.61</td>
<td>10.08</td>
<td>1975.00</td>
<td>2018.41</td>
</tr>
<tr>
<td>7</td>
<td>0.18</td>
<td>28.45</td>
<td>9.26</td>
<td>1980.00</td>
<td>2017.71</td>
</tr>
<tr>
<td>8</td>
<td>0.18</td>
<td>25.87</td>
<td>8.01</td>
<td>1983.75</td>
<td>2017.74</td>
</tr>
<tr>
<td>9</td>
<td>0.18</td>
<td>24.12</td>
<td>7.20</td>
<td>1986.67</td>
<td>2017.99</td>
</tr>
<tr>
<td>≥ 10</td>
<td>0.18</td>
<td>≤ 22.86</td>
<td>≤ 6.48</td>
<td>≥ 1989</td>
<td>≥ 2018.34</td>
</tr>
<tr>
<td>No replacement</td>
<td>15.00</td>
<td>0</td>
<td></td>
<td>2150.00</td>
<td>2165.00</td>
</tr>
</tbody>
</table>
Chapter 4

An $M/G_1 - G_2/1$ Queue with Semi-Dynamic Policy

4.1 Introduction

Consider the BTQ model in which information about the number of customers in stage 1 is only available at the arrival epoch of the server at that stage. Therefore, to avoid idleness, the server must choose some customers to serve from among those present in stage 1 upon his arrival at that stage. This policy is called a semi-dynamic policy. In this chapter we analyze the BTQ model when semi-dynamic policies are applied. First optimal semi-dynamic policies are studied, and then a model is developed to find the parameter of a class of semi-dynamic policy, gated-limited policy, in $M/G_1 - G_2/1$ queues with nonzero switchover times. For systems with zero switchover times, it is proved that the optimality conditions for sequential service policy and gated-limited policy with limit 2, are independent of the arrival process. Finally, the optimal limit of a gated-limited policy under high traffic intensity is obtained, and using numerical studies, it is shown that this optimal limit is robust with respect to the arrival process in systems with zero switchover times.
4.2 Semi-Dynamic Policies

Suppose that in our basic two-stage tandem queue model, the server applies a greedy and exhaustive policy in the second stage, and a semi-dynamic policy in the first stage. By semi-dynamic policies we mean policies in the first stage in which the server must choose some customers from among those present upon his arrival to that stage. Thus, the basic semi-Markov decision problem, the BSD model, can be revised as follows:

- **Decision Epochs**: Epochs at which the server returns to stage 1 (the end of switchover time $D_{21}$), or epochs at which a customer enters the system when the server is idle.

- **State ($n$)**: The number of waiting customers in stage 1 at each decision epoch. $n \in \mathbb{Z}^+ = \{0, 1, 2, \ldots\}$.

- **Action ($a$)**: Choosing the first $a$ units which are waiting in the first stage and completing their service in the first stage and then in the second stage, $a \in Ac = \{0, 1, 2, \ldots, n\}$.

Let $\zeta_n(a)$ be the expected time until the next decision epoch when action $a$ is used in state $n$; then

$$\zeta_n(a) = \begin{cases} \frac{1}{a} E[S_1] + aE[S_2] + E[D_{12}] + E[D_{21}] & : a = n = 0 \\ \frac{1}{a} E[S_1] + aE[S_2] + E[D_{12}] + E[D_{21}] & : 1 \leq a \leq n, \ n = 1, 2, \ldots \end{cases}$$

and the transition probabilities are

$$P_{ij}(a) = \begin{cases} 1 & : i = 0, j = 1, a = 0 \\ p_{j-i+2}(a) & : i = 1, 2, \ldots, j \geq i - a, a \leq i \\ 0 & : \text{otherwise} \end{cases}$$

where

$$p_j(a) = \int_0^\infty \frac{e^{-\lambda t}(\lambda t)^j}{j!} \text{d}(F_1^{*a} * F_2^{*a} * B_{12} * B_{21})$$

in which * indicates convolution. and $F_i^{*a}$ is the $a$-fold convolution of service time distribution $F_i$ with itself ($i = 1, 2$).
Semi-dynamic policies actually eliminate the possibility of idleness in stage 1 by not allowing the server to serve any more customers at stage 1 than were present there upon his arrival. As was described in Section 1.7 of Chapter 1, semi-dynamic policies are applicable in a partial information environment, where the information about the number of customers in stage 1 is only available at certain times. In the machine example, a semi-dynamic policy can be applied when the machine can be adjusted to serve different numbers of units at the beginning of each cycle, but cannot be readjusted during that cycle.

4.3 Optimal Semi-Dynamic Policy

A semi-dynamic policy in which the server chooses to serve \( a < n \) customers behaves like a limited policy, and when the server chooses to serve all \( n \) customers who are present in stage 1 upon his arrival, the semi-dynamic policy acts like a gated policy. Therefore, a semi-dynamic policy is actually a combination of a gated and a limited service policy in which the limit \( M_{LE} = U(n) \) is a function of state \( n \). These policies behave like a gated policy for some states and like a limited service policy for the remaining states. In other words, under each semi-dynamic policy \( \gamma \), and considering \( \Omega_G^* \) (\( \Omega_L^* \)) as the set of states in which policy \( \gamma \) acts like a gated (limited) service policy, we will have.

\[
\Omega_G^* = \{ n \mid n \in \mathcal{N}, U(n) = n \} \quad \Omega_L^* = \{ n \mid n \in \mathcal{N}, U(n) < n \}
\]

(4.1)

and \( \Omega_G^* \cup \Omega_L^* = \mathcal{N} \). \( \Omega_G^* \cap \Omega_L^* = \emptyset \). Suppose \( \gamma^* \) is the optimal semi-dynamic policy and consider state \( n \) in which policy \( \gamma^* \) behaves like a gated service policy; then, because of the switching costs and switchover times, we can conclude that \( \gamma^* \) also behaves like a gated policy for states \( n' \leq n \). In other words,

\[
n \in \Omega_G^* \implies \{ n - 1, n - 2, \ldots, 2, 1 \} \in \Omega_G^*.
\]

(4.2)

Now consider state \( n_G \), where \( n_G \in \Omega_G^* \) and \( n_G + 1 \in \Omega_L^* \). This means that, although in state \( n_G + 1 \), there is an opportunity to serve all customers, since an increase in holding cost at stage 2 results in optimal policy \( \gamma^* \) acting like a limited
policy. Therefore, if it is not optimal to serve all \( n_G + 1 \) customers in state \( n_G + 1 \), it won’t be optimal to serve all \( n_G + 2 \) customers in state \( n_G + 2 \). Hence,

\[
n_G + 1 \in \Omega_L^* \implies \{n_G + 2, n_G + 3, \ldots\} \in \Omega_L^*.
\]

(4.3)

and using (4.2), for the optimal semi-dynamic policy \( \gamma^* \), we have.

\[
\gamma^*(n) = \begin{cases} 
n & \text{if } \mathcal{U}(n) < n \\
\mathcal{U}(n) & \text{if } n_G + 1 \leq n \leq n_G 
\end{cases}
\]

(4.4)

where \( \gamma^*(n) \) is the optimal action in state \( n \).

In this chapter we concentrate on a semi-dynamic policy \( \pi_G \) with parameter \( M_G \).

where

\[
\pi_G(n) = \begin{cases} 
n & \text{if } \mathcal{U}(n) < n \\
M_G = n_G & \text{if } n \geq n_G + 1
\end{cases}
\]

(4.5)

Policy \( \pi_G \) is a gated-limited service policy which has the monotonicity property related to the number of customers in the system at a decision epoch. According to this policy, at each decision epoch, the server chooses \( a = \min\{n, M_G\} \) customers to serve in each cycle, where \( n \) is the number of customers in the system at each decision epoch and limit \( M_G \) is a predetermined number. In other words, policy \( \pi_G \) is actually a gated policy in a two-stage tandem queue attended by a moving server with intermediate waiting room of capacity \( M_G = n_G \). Policy \( \pi_G \) is easy to apply, and it can be considered as a suboptimal semi-dynamic policy.

### 4.4 An M/G₁ - G₂/1 with Gated-Limited Policy

Consider a two-stage tandem queue with the following assumptions (Figure 4.1):

- Customers arrive at queue \( Q_0 \) before gate \( G \) according to a homogeneous Poisson process with rate \( \lambda \).

- When the moving server arrives at stage 1, the gate is opened and, if there are fewer than \( M_G \) customers in \( Q_0 \), all of them move to the next queue, \( Q_1 \); otherwise, only \( M_G \) customers are allowed to move to \( Q_1 \), whereupon the gate is immediately closed.
Service times $S_i$ for a customer at stage $i$ are identically distributed, mutually independent random variables with general distribution functions $F_i(.)$, $i = 1, 2$.

The switchover time $D_{12}$ ($D_{21}$) from stage 1 to 2 (from 2 to 1) is a random variable with distribution function $B_{12}(.)$ ($B_{21}(.)$).

The server continues serving at stage 1 until it becomes empty, and then moves to stage 2 and serves all customers waiting there before returning to stage 1. The server begins to serve at stage 1 after the gate is closed. If there is no customer in queue $Q_0$, then the server stays at stage 1 to await the arrival of a customer, whereupon the gate is closed and the server begins to serve that customer in stages 1 and 2 sequentially.

4.4.1 Analysis of Steady State Probabilities

Let $I^{(n)}$ be the stage number where the $n$th service completion occurs, $I = \{1, 2\}$, and let $X_i^{(n)}$ be the number of customers in stage $i$ ($i = 0, 1, 2$), at the $n$th service completion. Then a model for this system can be considered as the imbedded Markov chain $\langle X_0^{(n)}, X_1^{(n)}, X_2^{(n)}, I^{(n)} \rangle$, and assuming that steady state conditions prevail, we will have $\lim_{n \to \infty}(X_0^{(n)}, X_1^{(n)}, X_2^{(n)}, I^{(n)}) = (X_0, X_1, X_2, I)$ in distribution, and the
stationary probabilities $\pi_{ijk}^{(u)}$ can be defined as

$$\pi_{ijk}^{(u)} = P\{X_0 = i, X_1 = j, X_2 = k, I = u\}.$$

Let $Y_r^*$ be the number of customers that arrive at $Q_0$ during a service time in stage $r$, and let $Y_r^D$ be the number of customers that arrive at $Q_0$ during switchover times $D_r$ (from stage $r$, $r = 1, 2$). Then, for $i = 1, 2$,

$$p_i^{(r)} = P\{Y_i^s = l\} = \int_0^{\infty} \frac{e^{-\lambda t}(\lambda t)^l}{l!}\,dF_r(t)$$

$$d_i^{(r)} = P\{Y_i^D = l\} = \int_0^{\infty} \frac{e^{-\lambda t}(\lambda t)^l}{l!}\,dB_r(t).$$

where $p_i^{(r)}$ and $d_i^{(r)}$ represent the probabilities that $l$ customers arrive at queue $Q_0$ during service times $S_r$ and switchover times $D_r$, respectively.

For systems with a gated-limited service policy, the balance equations for $i \geq 0$ are

$$\pi_{ijk} = \begin{cases} 
\pi_{000} p_i^{(1)} + \pi_{000} d_1^{(2)} p_i^{(1)} + \pi_{100} d_0^{(2)} p_i^{(1)} & ; j = 0, \quad k = 1 \\
\sum_{m=0}^{j+1} \pi_{m00} d_{j-m}^{(2)} p_i^{(1)} & ; 1 \leq j \leq M_G - 2, \quad k = 1 \\
\sum_{m=0}^{M_G-1} \sum_{n=M_G}^{M_G+i} \pi_{m00} d_{n-m}^{(2)} p_i^{(1)} + \sum_{m=M_G}^{M_G+i} \sum_{n=0}^{M_G+i-m} \pi_{m00} d_{n}^{(2)} p_i^{(1)} & ; j = M_G - 1, \quad k = 1 \\
\sum_{m=0}^{j} \pi_{i-m,j+m} p_m^{(1)} & ; 0 \leq j \leq M_G - k, \quad 2 \leq k \leq M_G \\
0 & ; j + k \geq M_G + 1. 
\end{cases} \quad (4.6)$$

and

$$\pi_{i0k} = \begin{cases} 
\sum_{m=0}^{i} \sum_{n=0}^{i-m} \pi_{i-m,n} p_m^{(1)} d_n^{(2)} p_m^{(1)} \\
+ \sum_{m=0}^{i} \pi_{m,0} p_{i-m}^{(2)} & ; 0 \leq k \leq M_G - 2 \\
\sum_{m=0}^{i} \sum_{n=0}^{i-m} \pi_{i-m,n} p_m^{(1)} d_n^{(2)} p_m^{(1)} & ; k = M_G - 1 \\
0 & ; k \geq M_G. 
\end{cases} \quad (4.7)$$
4.4.2 Probability Generating Functions

Define generating function $\Pi_u(x, y, z)$ for $u = 1, 2$, by

$$\Pi_u(x, y, z) = \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} \sum_{k=0}^{\infty} \pi_{i,j,k}^{(u)} x^i y^j z^k : |x|, |y|, |z| \leq 1. \quad (4.8)$$

Now, multiply both sides of (4.7) by appropriate $x^i$, $y^j$ and $z^k$, by summing over $i$ and $k$. after some algebra we obtain

$$\Pi_2(x, 0, z) = \sum_{i=0}^{M_2-1} \sum_{k=0}^{i} \sum_{m=0}^{i-m} \pi_{i-n-m,0,k+1}^{(2)} d_n^{(1)} p_m^{(2)} x^i y^j z^k + \sum_{i=0}^{M_2-2} \sum_{k=0}^{i} \sum_{m=0}^{i-m} \pi_{m,0,k+1}^{(2)} p_{i-m}^{(2)} x^i y^j z^k. \quad (4.9)$$

For the first term of the right hand side of (4.9) we have

$$\sum_{i=0}^{M_2-1} \sum_{k=0}^{i} \sum_{m=0}^{i-m} \pi_{i-n-m,0,k+1}^{(1)} d_n^{(1)} p_m^{(2)} x^i y^j z^k = \frac{1}{z} \Pi_1(x, 0, z) D_1(x) P_2(x). \quad (4.10)$$

where

$$D_u(x) = \sum_{i=0}^{\infty} d_i^{(u)} x^i : u = 1, 2, \ |x| \leq 1$$

$$P_u(x) = \sum_{i=0}^{\infty} p_i^{(u)} x^i : u = 1, 2, \ |x| \leq 1.$$ 

and for the second term of the right hand side of (4.9), we have

$$\sum_{i=0}^{M_2-2} \sum_{k=0}^{i} \sum_{m=0}^{i-m} \pi_{m,0,k+1}^{(2)} p_{i-m}^{(2)} x^i y^j z^k = \frac{1}{z} \sum_{i=0}^{M_2-2} \sum_{k=0}^{i} \sum_{m=0}^{i-m} \pi_{m,0,k+1}^{(2)} p_{i-m}^{(2)} x^i y^j z^k = \frac{1}{z} \left[ \Pi_2(x, 0, z) P_2(x) - \Pi_2(x, 0, 0) P_2(z) \right]$$

$$= \frac{P_2(x)}{z} \left[ \Pi_2(x, 0, z) - \Pi_2(x, 0, 0) \right]. \quad (4.11)$$

Using (4.10) and (4.11) in (4.9), after some algebra we get

$$\Pi_2(x, 0, z) = \frac{P_2(x)}{z - P_2(x)} \left[ D_1(x) \Pi_1(x, 0, z) - \Pi_2(x, 0, 0) \right]. \quad (4.12)$$

Now considering (4.6) for $j = M_2 - 1$ and $k = 1$, we obtain

$$\pi_{i,M_2-1,1}^{(1)} = \sum_{m=0}^{M_2-1} \pi_{m,0}^{(2)} \sum_{n=M_2}^{M_2+i} d_n^{(2)} p_{i-M_2-n}^{(1)} + \sum_{m=M_2}^{M_2+i} \pi_{m,0}^{(2)} \sum_{n=0}^{M_2} d_n^{(2)} p_{i-M_2-n}^{(1)}. \quad (4.13)$$
Setting \( u = n - m \), for the first term of the right hand side of (4.13) we will have

\[
\sum_{m=0}^{M_G-1} \sum_{n=M_G}^{M_G+i} \pi_m^{(2)} d_{n-m}^{(2)} p_i^{(1)} = \sum_{m=0}^{M_G-1} \sum_{n=M_G}^{M_G+i-m} \pi_m^{(2)} d_u^{(2)} p_i^{(1)}
\]

\[
= \sum_{m=0}^{M_G-1} \sum_{n=0}^{M_G+i-m} d_u^{(2)} p_i^{(1)}
\]

\[
- \sum_{m=0}^{M_G-1} \sum_{n=0}^{M_G+i-m} d_u^{(2)} p_i^{(1)}
\]

\[
- \sum_{m=0}^{M_G-1} \sum_{n=0}^{M_G+i-m} d_u^{(2)} p_i^{(1)}
\]

Therefore, for \( i \geq 0 \).

Substituting (4.14) into (4.13) yields

\[
\pi_i^{(1)} = \sum_{m=0}^{M_G-1} \sum_{n=0}^{M_G+i-m} \pi_m^{(2)} d_u^{(2)} p_i^{(1)}
\]

\[
+ \sum_{m=0}^{M_G+i} \sum_{n=0}^{M_G+i-m} d_n^{(2)} p_i^{(1)}
\]

\[
- \sum_{m=0}^{M_G-1} \sum_{n=0}^{M_G+i-m} d_u^{(2)} p_i^{(1)}
\]

\[
- \sum_{m=0}^{M_G-1} \sum_{n=0}^{M_G+i-m} d_u^{(2)} p_i^{(1)}
\]

Therefore, for \( i \geq 0 \).

\[
\pi_{i,j,k}^{(1)} = \begin{cases} 
\pi_{000}^{(2)} d_0^{(2)} p_i^{(1)} + \pi_{000}^{(2)} d_1^{(2)} p_i^{(1)} + \pi_{100}^{(2)} d_0^{(2)} p_i^{(1)} & : j = 0, \ k = 1 \\
\sum_{m=0}^{M_G+i} \sum_{n=0}^{M_G+i-m} \pi_m^{(2)} d_u^{(2)} p_i^{(1)} & : 1 \leq j \leq M_G - 2, \ k = 1 \\
\sum_{m=0}^{M_G+i} \sum_{n=0}^{M_G+i-m} \pi_m^{(2)} d_u^{(2)} p_i^{(1)} & : 1 \leq j \leq M_G - 2, \ k = 1 \\
\sum_{m=0}^{M_G-1} \sum_{n=0}^{M_G-1} \pi_m^{(2)} d_n^{(2)} p_i^{(1)} & : j = M_G - 1, \ k = 1 \\
\sum_{m=0}^{i} \pi_m^{(1)} & : 0 \leq j \leq M_G - k, \ 2 \leq k \leq M_G \\
0 & : j + k \geq M_G + 1 \ . 
\end{cases}
\]

Now, multiplying both sides of (4.16) by appropriate \( x^i, y^j, z^k \) and summing over \( i, j \) and \( k \), gives

\[
\Pi_1(x, y, z) = \sum_{i=0}^{\infty} (\pi_{000}^{(2)} d_0^{(2)} p_i^{(1)} + \pi_{000}^{(2)} d_1^{(2)} p_i^{(1)} + \pi_{100}^{(2)} d_0^{(2)} p_i^{(1)}) x^i z^j
\]
On the other hand.

\[ \sum_{i=0}^{\infty} \pi_{m00}^{(2)} p_i^{(1)} + \pi_{100}^{(2)} d_0^{(2)} p_i^{(1)} + \pi_{000}^{(2)} d_0^{(2)} p_i^{(1)} x^i y^j z = z P_1(x) \sum_{i=1}^{M_G-2} \sum_{j=0}^{M_G-2} \pi_{m00} y^{r_1} d_{j+1-m}^{(2)} y^{j+1-m} \]  \quad (4.18)

and also

\[ \sum_{i=0}^{\infty} \sum_{j=1}^{M_G-2} \sum_{m=0}^{M_G-2} \pi_{m00}^{(2)} d_{j+1-m}^{(2)} p_i^{(1)} x^i y^j z = z P_1(x) \sum_{i=1}^{M_G-2} \sum_{j=0}^{M_G-2} \pi_{m00} y^{r_1} d_{j+1-m}^{(2)} y^{j+1-m} \]  \quad (4.19)

Furthermore, by defining \( v = M_G + i \), we have

\[ \sum_{i=0}^{\infty} \sum_{m=0}^{M_G+i} \sum_{n=0}^{M_G+i-m} \pi_{m00}^{(2)} d_n^{(2)} p_{i+M_G-m-n}^{(1)} x^i y^{M_G-1} z = \sum_{v=M_G}^{\infty} \sum_{m=0}^{v} \pi_{m00}^{(2)} d_n^{(2)} p_{v-m-n}^{(1)} x^{v-M_G} y^{M_G-1} z \]

\[ = \frac{z y^{M_G-1}}{x^{M_G}} \sum_{v=M_G}^{\infty} \sum_{m=0}^{v} \sum_{n=0}^{v-m} \pi_{m00}^{(2)} d_n^{(2)} x^n p_{v-m-n}^{(1)} x^{v-m-n} \]

\[ = \frac{z y^{M_G-1}}{x^{M_G}} \left[ \Pi_2(x,0,0) D_2(x) P_1(x) - \sum_{v=0}^{M_G-1} \sum_{m=0}^{v} \sum_{n=0}^{v-m} \pi_{m00}^{(2)} d_n^{(2)} p_{v-m-n}^{(1)} x^n \right] \]  \quad (4.20)

and also

\[ \sum_{i=0}^{\infty} \sum_{m=0}^{M_G-1} \sum_{n=0}^{M_G-m-1} \pi_{m00}^{(2)} d_n^{(2)} p_{i+M_G-m-n}^{(1)} x^i y^{M_G-1} z = \frac{z y^{M_G-1}}{x^{M_G}} \left\{ \sum_{m=0}^{M_G-1} \sum_{n=0}^{M_G-m-1} \pi_{m00}^{(2)} d_n^{(2)} x^n \right\} \sum_{i=0}^{\infty} \sum_{m=0}^{M_G-m-n} x^{i+M_G-m-n} \]
For the last term of the right hand side (4.17), we have

\[
\sum_{i=0}^{\infty} \sum_{k=2}^{M_G} \sum_{j=0}^{M_G-1} \sum_{m=0}^{M_G-1} \pi_{i-m,j+1,k-1}^{(1)} P_{m}^{(1)} x^{i} y^{j} z^{k} = \frac{z}{y} \left[ \Pi_{1}(x,y,z) P_{1}(x) - \Pi_{1}(x,0,z) P_{1}(x) \right]
\]

\[
= \frac{z P_{1}(x)}{y} \left[ \Pi_{1}(x,y,z) - \Pi_{1}(x,0,z) \right]. \tag{4.22}
\]

Using (4.18) to (4.21) in (4.17), we get

\[
\Pi_{1}(x,y,z) = \frac{z}{y} P_{1}(x) \left( \pi_{000}^{(2)} d_{0}^{(2)} + \pi_{001}^{(2)} d_{1}^{(2)} + \pi_{100}^{(2)} d_{0}^{(2)} \right)
\]

\[
+ \frac{z}{y} P_{1}(x) \sum_{j=1}^{M_G-1} \sum_{m=0}^{M_G-1} \pi_{m00}^{(2)} d_{j+1-m}^{(2)} y^{j+1-m}
\]

\[
+ \frac{z}{y} \frac{M_G-1}{M_G} \left[ \Pi_{2}(x,0,0) D_{2}(x) P_{1}(x) - \sum_{u=0}^{M_G-1} \sum_{v=0}^{M_G-1} \sum_{n=0}^{M_G-1} \pi_{u00}^{(2)} d_{v}^{(2)} p_{u-v-n}^{(1)} x^{v} \right]
\]

\[
- \frac{z}{y} \frac{M_G-1}{M_G} \left[ P_{1}(x) \sum_{m=0}^{M_G-1} \sum_{n=0}^{M_G-1} \pi_{m00}^{(2)} d_{n}^{(2)} x^{n}
\]

\[
- \sum_{m=0}^{M_G-1} \sum_{n=0}^{M_G-1} \sum_{l=0}^{M_G-1} \pi_{m00}^{(2)} d_{n}^{(2)} x^{n} p_{l}^{(1)} x^{l} \right]
\]

\[
+ \frac{z P_{1}(x)}{y} \left[ \Pi_{1}(x,y,z) - \Pi_{1}(x,0,z) \right]. \tag{4.23}
\]

Set

\[
\Phi_{s}(x) = \sum_{m=0}^{s} \sum_{n=0}^{s-m} \pi_{m00}^{(2)} d_{n}^{(2)} x^{m+n}, \tag{4.24}
\]

then, for the second term of the right hand side of (4.23), we find that

\[
\frac{z}{y} P_{1}(x) \sum_{j=1}^{M_G-2} \sum_{m=0}^{j+1} \pi_{m00}^{(2)} d_{j+1-m}^{(2)} y^{j+1-m} = \frac{z P_{1}(x)}{y} \left( \frac{\Phi_{M_G-1}(y)}{y} - \frac{\pi_{000}^{(2)} d_{0}^{(2)}}{y}
\]

\[
- \frac{\pi_{000}^{(2)} d_{1}^{(2)}}{\pi_{000}^{(2)} d_{0}^{(2)}} \right). \tag{4.25}
\]
On the other hand, it can be shown by expansion that
\[ \sum_{v=0}^{M_G-1} \sum_{m=0}^{v-m} \sum_{n=0}^{\pi_m^{(2)} \pi_n^{(2)} \pi_{-m-n}^{(1)}} = \sum_{m=0}^{M_G-1} \sum_{n=0}^{M_G-m-1} \sum_{L=0}^{M_G-m-n-1} \pi_{-m-n}^{(2)} \pi_n^{(2)} x^m y^n z^L. \] (4.26)

Thus, using (4.24) to (4.26) in (4.23), after some algebra, yields
\[ \Pi_1(x, y, z) = \frac{zP_1(x)}{y - zP_1(x)} \left\{ \left[ \frac{y}{x} \right]^{M_G} [D_2(x) \Pi_2(x, 0, 0) - \Phi_{M_G-1}(x)] + \Phi_{M_G-1}(y) - \Pi_1(x, 0, z) - (1 - y)^{\pi_0^{(2)} d_0^{(2)}} \right\}. \] (4.27)

**Remark 4.1.**

If \( M_G \to \infty \), then \( \Phi_{M_G-1}(x) = D_2(x) \Pi_2(x, 0, 0) \), and
\[ \Pi_1(x, y, z) = \frac{zP_1(x)}{y - zP_1(x)} (D_2(y) \Pi_2(y, 0, 0) - \Pi_1(x, 0, z) - (1 - y)^{\pi_0^{(2)} d_0^{(2)}}). \]

This is the same result which was obtained by Katayama [28] for a system with a gated service policy. □

To complete generating functions \( \Pi_1(x, y, z) \) and \( \Pi_2(x, 0, z) \), it is necessary to determine unknown functions \( \Pi_1(x, 0, z) \), \( \Pi_2(x, 0, 0) \), \( \Phi_{M_G-1}(\cdot) \), and probability \( \pi_0^{(2)}. \)

**Determination of \( \Pi_1(x, 0, z) \)**

In (4.27), the root of denominator \( y - zP_1(x) = 0 \) is \( y = zP_1(x) \), which must also be the zero point of the numerator (4.27). Therefore,
\[ \left[ \frac{zP_1(x)}{x} \right]^{M_G} [D_2(x) \Pi_2(x, 0, 0) - \Phi_{M_G-1}(x)] + \Phi_{M_G-1}(zP_1(x)) - \Pi_1(x, 0, z) - (1 - zP_1(x))^\pi_0^{(2)} d_0^{(2)} = 0. \]

Hence,
\[ \Pi_1(x, 0, z) = \left[ \frac{zP_1(x)}{x} \right]^{M_G} [D_2(x) \Pi_2(x, 0, 0) - \Phi_{M_G-1}(x)] + \Phi_{M_G-1}(zP_1(x)) - [1 - zP_1(x)]^\pi_0^{(2)} d_0^{(2)}. \] (4.28)
Determination of $\Pi_2(x, 0, 0)$

The denominator of equation (4.12), $z - P_2(x) = 0$, has one root $z = P_2(x)$ inside the unit circle $|z| \leq 1$. Therefore, substituting this root in numerator (4.12), we get.

$$D_1(x)\Pi_1(x, 0, P_2(x)) - \Pi_2(x, 0, 0) = 0.$$ or

$$\Pi_2(x, 0, 0) = D_1(x)\Pi_1(x, 0, P_2(x)).$$ (4.29)

Using (4.28) in (4.29) yields,

$$\Pi_2(x, 0, 0) = \frac{x^{M_G}D_1(x)}{x^{M_G} - D_1(x)D_2(x)[P_1(x)P_2(x)]^{M_G}}\left\{-\frac{P_1(x)P_2(x)}{x}\Phi_{M_G-1}(x)\right\} + \Phi_{M_G-1}(P_1(x)P_2(x)) - [1 - P_1(x)P_2(x)]\pi^{(2)}_0d^{(2)}_0. \quad (4.30)$$

Determination of $\Phi_{M_G-1}(\cdot)$ and $\pi^{(2)}_0$

Since $\Phi_{M_G-1}(\cdot)$ is a function of $\pi^{(2)}_{100}$, $(i = 1, 2, \ldots, M_G - 1)$, therefore to obtain $\Phi_{M_G-1}(\cdot)$ and $\pi^{(2)}_0$, the unknown probabilities $\pi^{(2)}_{100}$ for $i = 1, 2, \ldots, M_G - 1$ must be found. Using Takacs lemma [62], it can be shown that if $\rho_1 + \rho_2 + (\eta_1 + \eta_2)/M_G < 1$ (where $\rho_i = \lambda E[S_i]$ and $\eta_i = \lambda E[D_i]$), then the denominator in the right hand side of (4.30), $x^{M_G} - D_1(x)D_2(x)[P_1(x)P_2(x)]^{M_G} = 0$, has just $M_G$ roots. $\xi_j$, $j = 1, 2, \ldots, M_G - 1$ in the unit circle, while $\xi_{M_G} = 1$. Since $|\Pi_2(x, 0, 0)| \leq 1$ in the domain $|x| \leq 1$, the numerator in the right hand side of (4.30) must be zero for $\xi_j$. $(j = 1, 2, \ldots, M_G)$. Therefore, for $j = 1, 2, \ldots, M_G - 1$

$$-\frac{P_1(\xi_j)P_2(\xi_j)}{\xi_j}\Phi_{M_G-1}(\xi_j) + \Phi_{M_G-1}(P_1(\xi_j)P_2(\xi_j)) - [1 - P_1(\xi_j)P_2(\xi_j)]\pi^{(2)}_0d^{(2)}_0 = 0. \quad (4.31)$$

Since the system of $(M_G - 1)$ equations (4.31) has $M_G$ unknowns $\pi^{(2)}_{100}, \pi^{(2)}_{101}, \ldots, \pi^{(2)}_{M_G-1,0,0}$, it is necessary to find one more relation among those unknowns. Using L'Hospital's rule in (4.12) and (4.27) yields,

$$-\rho_2\Pi_2(1, 0, 1) - (\rho_2 + \eta_1)\Pi_1(1, 0, 1) + \rho_2\Pi_2(1, 0, 0) - \Pi_1'(1, 0, 1) + \Pi_2'(1, 0, 0) = 0 \quad (4.32)$$

$$\Pi_2(1, 0, 1) - \Pi_1'(1, 0, 1) = 0 \quad (4.33)$$
\[ -\rho_1 \pi_1(1, 1, 1) - (\rho_1 - M_G + \eta_2)\pi_2(1, 0, 0) + \rho_1 \pi_1(1, 0, 1) - \pi_2'(1, 0, 0) + \pi_1'(1, 0, 1) - M_G \Phi_{M_G-1}(1) + \Phi'_{M_G-1}(1) = 0 \]  
\[ (4.34) \]

\[ \pi_1(1, 1, 1) - M_G \pi_2(1, 0, 0) + M_G \Phi_{M_G-1}(1) - \Phi'_{M_G-1}(1) - \pi_{000}'d_0^{(2)} = 0 \]  
\[ (4.35) \]

\[ \pi_1(1, 1, 1) + \pi_2(1, 0, 0) - \pi_1(1, 0, 1) - \pi_2'(1, 0, 1) = 0 \]  
\[ (4.36) \]

Since \( \pi_2(1, 0, 0) \) is the probability that after a service completion in stage 2 the server switches to stage 1, and \( \pi_1(1, 0, 1) \) is the probability that after a service completion in stage 1 the server switches to stage 2; thus,

\[ \pi_2(1, 0, 0) = \pi_1(1, 0, 1). \]
\[ (4.37) \]

By using (4.37) and relation \( \pi_1(1, 1, 1) + \pi_2(1, 0, 1) = 1 \), the system of equations (4.32) to (4.36) reduces to

\[ \eta_1 \pi_2(1, 0, 0) - \pi_1'(1, 0, 1) - \pi_2'(1, 0, 0) = -\frac{\rho_2}{2} \]  
\[ (4.38) \]

\[ (\eta_2 - M_G) \pi_2(1, 0, 0) - \pi_1'(1, 0, 1) + \pi_2'(1, 0, 0) = -\frac{\rho_1}{2} - M_G \Phi_{M_G-1}(1) + \Phi'_{M_G-1}(1) \]  
\[ (4.39) \]

\[ M_G \pi_2(1, 0, 0) + \pi_{000}'d_0^{(2)} = \frac{1}{2} + M_G \Phi_{M_G-1}(1) - \Phi'_{M_G-1}(1) \]  
\[ (4.40) \]

and therefore, we obtain

\[ \pi_1(1, 1, 1) = \pi_2(1, 0, 1) = \frac{1}{2} \]  
\[ (4.41) \]

\[ \pi_2(1, 0, 0) = \frac{\rho + 2[M_G \Phi_{M_G-1}(1) - \Phi'_{M_G-1}(1)]}{2(M_G - \eta)} \]  
\[ (4.42) \]

\[ \pi_{000}'^{(2)} = \frac{M_G(1 - \bar{\rho}) - 2\eta[M_G \Phi_{M_G-1}(1) - \Phi'_{M_G-1}(1)]}{2d_0^{(2)}(M_G - \eta)} \]  
\[ (4.43) \]

where \( \rho = \rho_1 + \rho_2, \quad \eta = \eta_1 + \eta_2 \) and \( \bar{\rho} = \rho + \eta/M_G \).

Now, by using equation (4.43) along with the system of linear equations (4.31), unknown probabilities \( \pi_{i00}'^{(2)} \) \( (i = 1, 2, \ldots, M_G - 1) \) can be determined.
4.4.3 Waiting Times

Define the waiting time at stage 1 of an arbitrary customer, $W_1$, as measured from his arrival instant to queue $Q_0$ until his service completion at stage 1 and let $W_1(t)$ be the corresponding waiting time distribution. When service of a customer is completed at stage 1, the total number of customers in that stage and in queue $Q_0$ is the same as the number of customers who arrived during his waiting time at $Q_0$ and $Q_1$. Therefore, if $\hat{A}_1(x)$ is defined as the probability generating function of the number of arrivals during the waiting time of an arbitrary customer in $Q_0$ and $Q_1$, we have:

$$\hat{A}_1(x) = \sum_{n=0}^{\infty} x^n \int_0^\infty \frac{e^{-\lambda t}(\lambda t)^n}{n!} dW_1(t)$$

$$= W_1^*(\lambda - \lambda x). \quad (4.44)$$

where $W_1^*(\cdot)$ is the Laplace Stieltjes transform of $W_1(\cdot)$.

On the other hand,

$$\hat{A}_1(x) = \sum_{n=0}^{\infty} \left( \frac{\sum_{k=1}^{\infty} \sum_{r=0}^{n} \pi_{r,n-r,k}^{(1)}}{\sum_{i=0}^{\infty} \sum_{j=0}^{\infty} \sum_{k=0}^{\infty} \pi_{i,j,k}^{(1)}} \right) x^n$$

$$= \sum_{n=0}^{\infty} \sum_{k=1}^{\infty} \sum_{r=0}^{n} \pi_{r,n-r,k}^{(1)} x^r x^{n-r} (1)^k$$

$$\frac{\Pi_1(1,1,1)}{}$$

$$= 2\Pi_1(x,x,1). \quad (4.45)$$

If $s = \lambda - \lambda x$, then $x = 1 - s/\lambda$, and using (4.44) and (4.45), we get

$$W_1^*(s) = 2\Pi_1(1 - s/\lambda, 1 - s/\lambda, 1) \quad (4.46)$$

Now suppose $W_{1,2}(t)$ is the distribution for the total waiting time, $W_{1,2}$, of an arbitrary customer in stages 1 and 2 (in $Q_0$, stage 1 and stage 2). Also assume that $\hat{A}_{1,2}(x)$ is the probability generating function of the number of arrivals during the waiting time of an arbitrary customer in $Q_0$, stage 1 and stage 2. Then,

$$\hat{A}_{1,2}(x) = W_{1,2}^*(\lambda - \lambda x). \quad (4.47)$$

On the other hand,

$$\hat{A}_{1,2}(x) = \sum_{n=0}^{\infty} \left( \frac{\sum_{r=0}^{n} \pi_{r,0,n-r}^{(2)}}{\sum_{i=0}^{\infty} \sum_{k=0}^{\infty} \pi_{i,0,k}^{(2)}} \right) x^n$$

$$= 2\Pi_2(x,0,x). \quad (4.48)$$
Thus,

$$W_{1,2}^*(s) = 2\Pi_2(1 - s/\lambda, 0.1 - s/\lambda). \tag{4.49}$$

and the average waiting times are

$$E[W_1] = \left( \frac{2}{\lambda} \right) \frac{d}{dx} \Pi_1(x, x, 1) \bigg|_{x=1} \tag{4.50}$$

and

$$E[W_2] = \left( \frac{2}{\lambda} \right) \frac{d}{dx} \Pi_2(x, 0, x) \bigg|_{x=0} - E[W_1]. \tag{4.51}$$

### 4.4.4 Average Number of Customers in the System

The average number of customers in stages 1 and 2, $L_1$ and $L_2$, are obtained in Appendix B. In this section only the final results are presented for completeness of this chapter.

$$L_1 = \frac{1}{1 - \rho_1} \left\{ \Pi_2(1, 0, 0)[D_2''(1) + 2\rho_1\eta_2] + 2(\rho_1 + \eta_2)\Pi_2'(1, 0, 0) - \Delta 
- 2\rho_1\Pi_1''(1, 0, 1) + 2\rho_1\pi_{000}^{(2)}d_{0}^{(2)} + \frac{P_1''(1)}{2} \right\} \tag{4.52}$$

where

$$\Pi_2'(1, 0, 0) = \frac{-1}{2M_G(1 - \bar{\rho})} \left\{ \Phi_{M_G-1}(1) [M_GP_1''(1) + M_GP_2''(1) 
- M_G(M_G - 1)(1 - \rho^2) 
- 2M_G\eta_1(1 - \rho) + 2M_G\rho_1\rho_2 ] 
+ \Phi'_{M_G-1}(1) [2\eta_1(1 - \rho) - P_1''(1) - P_2''(1) - 2\rho_1\rho_2] 
+ \Phi''_{M_G-1}(1) [1 - \rho^2] + f''(1)\Pi_2(1, 0, 0) 
- [P_1''(1) + P_2''(1) + 2(M_G + \eta_2)\rho + 2\rho_1\rho_2 ]\sigma_{000}^{(2)}d_{0}^{(2)} \right\}. \tag{4.53}$$

$$\Pi_1'(1, 0, 1) = -\frac{\rho_2}{2} - \eta_1\Pi_2(1, 0, 0) + \Pi_2'(1, 0, 0) \tag{4.54}$$
The average number of customers in the second stage, \( L_2 \), is

\[
L_2 = L_{1.2} - L_1
\]

where \( L_{1.2} \), the average number of customers in queue \( Q_0 \), stage 1 and stage 2, is

\[
L_{1.2} = \frac{1}{1 - \rho_2} \left\{ \Pi_1(1.0, 1)[D''(1) + 2\rho_2 \eta_1] + 2(\rho_2 + \eta_1)\Pi_2'(1.0, x = 1) \right\}
\]

\[
-2\rho_2\Pi_2'(1.0, 0) + \Upsilon + \frac{P_2''(1)}{2} \right\},
\]

where

\[
\Pi_2'(x = 1.0, x = 1) = \Pi_2'(1.0, 0) + (M_G\rho_1 + \eta_2)\Pi_2(1.0, 0)
\]

\[
-\rho_1[M_G\Phi_{M_G-1}(1) - \Phi'_{M_G-1}(1)] + (1 + \rho_1)\pi^{(2)}_{000d_0}^{(2)}.
\]

\[
\Upsilon = \Pi_2(1.0, 0) \left[ M_G P''(1) + D''(1) + 2M_G\rho_1 \eta_2 + M_G(M_G - 1)\rho_1^2 \right]
\]

\[
+2\Pi_2'(1.0, 0)[M_G\rho_1 + \eta_2] - \Phi_{M_G-1}(1) \left[ M_G P''(1) + M_G(M_G - 1)\rho_1^2 \right]
\]

\[
+\Phi'_{M_G-1}(1)[P_1''(1) - 2(M_G - 1)\rho_1] + \Phi''_{M_G-1}(1) [\rho_1(2 + \rho_1)]
\]

\[
+[P_1''(1) + 2\rho_1]\pi^{(2)}_{000d_0}^{(2)}.
\]

4.4.5 Average Number of Customers Served in a Cycle

Define a cycle as the time elapsed from the server's arrival at stage 1 to its next arrival at stage 1, and let \( N_{M_G}^{\text{cyc}} \) be the number of customers served during a cycle when gated-limited policy with parameter \( M_G \) is applied. Also suppose \( V_{ijk} \) is the
event that after the completion of switchover time \(D_{21}\) but before gate \(G\) is opened, \(i, j\) and \(k\) customers are in queues \(Q_0,\) stage \(1\) and stage \(2,\) respectively. Thus, if \(V_{ijk}^c\) denotes the event that a cycle starts with event \(V_{ijk},\) we have

\[
E[N_{MG}^{cu}] = \sum_{n=0}^{\infty} E[N_{MG}^{cu} | V_{n00}^c] P\{V_{n00}^c\}.
\]  

(4.59)

On the other hand,

\[
P\{V_{n00}^c\} = \frac{P\{V_{n00}\}}{\Pi_2(1, 0, 0)},
\]

(4.60)

where \(\Pi_2(1, 0, 0)\) is the probability that the server switches to stage \(1.\) and

\[
E[N_{MG}^{cu} | V_{n00}^c] = \begin{cases} 
1 & : \quad n = 0 \\
\frac{\pi_{m00}^{(2)} d_n^{(2)}}{\Pi_2(1, 0, 0)} & : \quad 1 \leq n \leq MG - 1 \\
MG & : \quad MG \leq n.
\end{cases}
\]

(4.61)

Since,

\[
P\{V_{n00}^c\} = \frac{\sum_{m=0}^{n} \pi_{m00}^{(2)} d_{n-m}}{\Pi_2(1, 0, 0)},
\]

(4.63)

then by using (4.61), (4.63) in (4.59) we obtain

\[
E[N_{MG}^{cu}] = \frac{1}{\Pi_2(1, 0, 0)}(\frac{\pi_{000}^{(2)} d_0^{(2)}}{\Pi_2(1, 0, 0)} + \sum_{n=1}^{MG-1} \frac{\pi_{m00}^{(2)} d_n^{(2)}}{\Pi_2(1, 0, 0)} + MG \sum_{n=MG}^{\infty} \frac{\pi_{m00}^{(2)} d_{n-m}}{\Pi_2(1, 0, 0)})
\]

(4.64)

It can be shown by expansion that

\[
\sum_{n=1}^{MG-1} \sum_{m=0}^{n} \pi_{m00}^{(2)} d_{n-m} = \sum_{m=0}^{MG-1} \pi_{m00}^{(2)} \sum_{n=0}^{MG-1-m} (n + m) d_n^{(2)}
\]

(4.65)

\[
\sum_{n=MG}^{\infty} \sum_{m=0}^{n} \pi_{m00}^{(2)} d_{n-m} = \sum_{m=0}^{MG-1} \pi_{m00}^{(2)} [1 - \sum_{n=0}^{MG-1-m} d_n^{(2)}] + \sum_{m=MG}^{\infty} \pi_{m00}^{(2)}
\]

(4.66)

Since \(\sum_{m=MG}^{\infty} \pi_{m00}^{(2)} = \Pi_2(1, 0, 0) - \sum_{m=0}^{MG-1} \pi_{m00}^{(2)},\) then by substituting (4.65) and (4.66) into (4.64), after some algebra we get

\[
E[N_{MG}^{cu}] = MG - \frac{MG \Phi_{MG-1}^{(1)} - \Phi_{MG-1}^{(1)} - \pi_{000}^{(2)} d_0^{(2)}}{\Pi_2(1, 0, 0)}.
\]

(4.67)
or considering (4.42),

\[ E[N_{MG}^{cu}] = \eta + \frac{2\pi_0^{(2)}d_0^{(2)}}{2\eta\lambda(1, 0, 0)}. \]  

(4.68)

and therefore, the average number of cycles per unit time, \( E[N_{MG}^{cu}] \), is

\[ E[N_{MG}^{cu}] = \frac{\lambda}{E[N_{MG}^{cu}]}. \]  

(4.69)

### 4.4.6 Optimal Gated-Limited Policy

To find the optimal gated-limited policy \( \pi_G \) which minimizes the long run average holding and switching costs per unit time, \( E[TC_{MG}] \), for a two-stage tandem queue attended by a moving server with switchover times, switching costs \( K_{12} \) and \( K_{21} \), and holding costs \( h_1 \) and \( h_2 \), the following problem must be solved:

\[
\text{Min } E[TC_{MG}] = E[N_{MG}^{cu}](K_{12} + K_{21}) + h_1 L_1(M_G) + h_2 L_2(M_G)
\]

Subject To:

\[
M_G \geq \frac{\eta}{1 - \rho} \\
M_G = 1, 2, 3, \ldots
\]

(4.70)

where \( L_i(M_G) \) is the average number of customers in stage \( i \) when a gated-limited policy with parameter \( M_G \) is implemented.

Table 4.1 introduces 12 different problems and Table 4.2 shows the optimal limit \( M_G^* \) when gated-limited service policies \( \pi_G \) are implemented in these examples. Deterministic switchover times from stage 2 to stage 1 are considered to simplify the solution process. The arrival rate is \( \lambda \) per hour and notation \( Er(a, b) \) is used to show an Erlang random variable with average \( a \) minute and parameter \( b \). Also, \( Hp(l, m, n) \) is used to show a hyperexponential random variable with \( p = l, \mu_1 = m, \mu_2 = n \) (per hour). Finally, \( Ex(a) \) and \( Dt(a) \) indicate exponential and deterministic random variables each with average \( a \) minutes, respectively.
Table 4.1. Parameters of problems 1 to 12

<table>
<thead>
<tr>
<th>problem</th>
<th>$\lambda$</th>
<th>$S_1$</th>
<th>$S_2$</th>
<th>$D_{12}$</th>
<th>$D_{21}$</th>
<th>$h_1$</th>
<th>$h_2$</th>
<th>$K$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5</td>
<td>$E_r(1.5)$</td>
<td>$E_r(2.3)$</td>
<td>$E_x(6)$</td>
<td>$D_t(4)$</td>
<td>10</td>
<td>20</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>$E_r(1.5)$</td>
<td>$E_r(2.3)$</td>
<td>$E_x(6)$</td>
<td>$D_t(4)$</td>
<td>10</td>
<td>20</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>1.5</td>
<td>$H_p(0.2, 2, 12)$</td>
<td>$E_x(17)$</td>
<td>$E_x(3)$</td>
<td>$D_t(5)$</td>
<td>48</td>
<td>49</td>
<td>12.6</td>
</tr>
<tr>
<td>4</td>
<td>0.5</td>
<td>$H_p(0.2, 2, 12)$</td>
<td>$E_x(17)$</td>
<td>$E_x(3)$</td>
<td>$D_t(5)$</td>
<td>48</td>
<td>49</td>
<td>12.6</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>$E_x(10)$</td>
<td>$E_x(12)$</td>
<td>$D_t(14)$</td>
<td>$D_t(2)$</td>
<td>8</td>
<td>45</td>
<td>15</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>$E_x(10)$</td>
<td>$E_x(12)$</td>
<td>$D_t(14)$</td>
<td>$D_t(2)$</td>
<td>3</td>
<td>14</td>
<td>22</td>
</tr>
<tr>
<td>7</td>
<td>5</td>
<td>$D_t(3)$</td>
<td>$D_t(5)$</td>
<td>$D_t(2)$</td>
<td>$D_t(2)$</td>
<td>18</td>
<td>58</td>
<td>35</td>
</tr>
<tr>
<td>8</td>
<td>5</td>
<td>$D_t(3)$</td>
<td>$D_t(5)$</td>
<td>$D_t(2)$</td>
<td>$D_t(2)$</td>
<td>18</td>
<td>58</td>
<td>35</td>
</tr>
<tr>
<td>9</td>
<td>5</td>
<td>$H_p(0.6, 12, 6)$</td>
<td>$D_t(2)$</td>
<td>0</td>
<td>$D_t(2)$</td>
<td>16</td>
<td>16</td>
<td>12</td>
</tr>
<tr>
<td>10</td>
<td>6</td>
<td>$E_r(1.5)$</td>
<td>$H_p(0.8, 60, 10)$</td>
<td>$E_r(6.3)$</td>
<td>$D_t(4)$</td>
<td>34</td>
<td>81</td>
<td>1</td>
</tr>
<tr>
<td>11</td>
<td>7</td>
<td>$H_p(0.7, 60, 60)$</td>
<td>$H_p(0.5, 7.5, 30)$</td>
<td>$E_x(1)$</td>
<td>$D_t(1)$</td>
<td>2</td>
<td>120</td>
<td>2</td>
</tr>
<tr>
<td>12</td>
<td>8</td>
<td>$E_x(3)$</td>
<td>$E_r(2.7)$</td>
<td>$D_t(3)$</td>
<td>$D_t(2)$</td>
<td>57</td>
<td>179</td>
<td>31</td>
</tr>
</tbody>
</table>

Table 4.2. Optimal gated-limited policy for problems 1 to 12

<table>
<thead>
<tr>
<th>Problem</th>
<th>$L_1$</th>
<th>$L_2$</th>
<th>$L_{1,2}$</th>
<th>$E[TC_M]$</th>
<th>$M_G^*$</th>
<th>$M_0^*$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.72</td>
<td>0.81</td>
<td>1.53</td>
<td>29.84</td>
<td>8</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>0.49</td>
<td>0.61</td>
<td>1.10</td>
<td>22.99</td>
<td>6</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>1.63</td>
<td>1.09</td>
<td>2.72</td>
<td>142.43</td>
<td>6</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>0.16</td>
<td>0.18</td>
<td>0.34</td>
<td>22.31</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td>3.61</td>
<td>1.82</td>
<td>5.44</td>
<td>122.44</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>2.42</td>
<td>1.18</td>
<td>3.60</td>
<td>46.45</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>7</td>
<td>1.79</td>
<td>1.23</td>
<td>3.02</td>
<td>187.07</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>8</td>
<td>1.79</td>
<td>1.40</td>
<td>3.19</td>
<td>196.74</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>9</td>
<td>2.97</td>
<td>0.99</td>
<td>3.96</td>
<td>93.53</td>
<td>6</td>
<td>4</td>
</tr>
<tr>
<td>10</td>
<td>0.96</td>
<td>1.03</td>
<td>1.99</td>
<td>129.40</td>
<td>6</td>
<td>2</td>
</tr>
<tr>
<td>11</td>
<td>11.24</td>
<td>0.70</td>
<td>11.94</td>
<td>120.45</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>12</td>
<td>2.95</td>
<td>1.47</td>
<td>4.42</td>
<td>531.26</td>
<td>5</td>
<td>2</td>
</tr>
</tbody>
</table>

Table 4.2 also shows the optimal limit $M_0^*$ for each problem when switchover times are considered zero. As it is clear, $M_0^*$ can be used as a lower bound for the optimal limit $M_G^*$ for systems with non-zero switchover times and as switchover times increases, the gap between $M_G^*$ and $M_0^*$ increases. Therefore, in designing the capacity of a buffer between stages in a two-stage tandem queue attended by a moving server when using a gated service policy, it should be considered that the buffer capacity must be at least $M_0^*$ which is simply obtained in the next section.

4.5 System with Zero Switchover Times

In this section we consider a two-stage tandem queue attended by a moving server with holding costs $h_2 \geq h_1$, switching costs $K_{12}$ and $K_{21}$, and switchover times $D_{12} = D_{21} = 0$. First, in Proposition 4.1, we derive the optimality condition for sequential service
A sequential service policy is a semi-dynamic policy in which action \( a = 1 \) is always chosen for state \( n = 1, 2, \ldots \). In other words, each customer receives service in stage 2 immediately after completion service in stage 1. Then, in Proposition 4.2 a sufficient condition for the optimality of the limit \( M_G^* = 2 \) is obtained. Systems with zero switchover times in high traffic intensity are studied in Proposition 4.3 and the optimal limit \( M_G^* \) is derived. Finally, through several numerical studies, we will determine that the optimal limit \( M_G^* \) is almost independent of the arrival process in a two-stage tandem queue with zero switchover times.

### 4.5.1 Optimality Condition for Sequential Service Policy

Suppose that in the BTQ model with zero switchover times each customer receives his service in stage 2 immediately after his service completion in stage 1; consequently, only one customer is served in each cycle (sequential service policy). A sequential service policy minimizes the waiting time in stage 2 and it is the optimal policy when \( h_2 \geq h_1 \) and switchover times and switching costs are zero. Proposition 4.1 introduces the optimality condition for a sequential service policy in systems with nonzero switching costs and zero switchover times.

**Proposition 4.1**

In a two-stage tandem queue attended by a moving server with zero switchover times, holding costs \( h_2 \geq h_1 \), and switching costs \( K_{12}, K_{21} \), a sequential service policy minimizes the total average holding and switching cost if and only if

\[
K_{12} + K_{21} \leq (h_2 - h_1)E[S_2] + h_2E[S_1].
\] (4.71)

**Proof:**

Consider the \( i \)th decision epoch, \( \tau_i \), that the server returns to stage 1 where \( n \) customers await service. Suppose that the server uses policy \( \pi \) in which \( r \) customers \( (r \leq n) \) are chosen to serve during that cycle. Defining \( \tau_{S_{j,12}}^\xi \) (or \( \tau_{S_{j,21}}^\xi \)) as the time of the \( j \)th switch from stage 1 to 2 (or 2 to 1) after decision epoch \( \tau_i \) when policy \( \xi \)
is applied, a new policy $\gamma$ can be constructed as follows (Figure 4.2):

* $\gamma$ follows $\pi$ in $[0, \tau_1]$.

* $\gamma$ serves $r - k$ customers in stage 1 during interval $[\tau_1, \tau_{S1,12}]$ and then switches to stage 2.

* $\gamma$ serves $r - k$ customers in stage 2 during interval $[\tau_{S1,12}, \tau_{S1,21}]$ and then switches back to stage 1.

* $\gamma$ serves $k$ customers in stage 1 during interval $[\tau_{S1,21}, \tau_{S2,12}]$ and then switches to stage 2.

* $\gamma$ serves $k$ customers in stage 2 during interval $[\tau_{S2,12}, \tau_{S2,21}]$ and then switches back to stage 1.

* $\gamma$ follows $\pi$ in $[\tau_{i+1}, \infty)$.

Since $\gamma$ behaves like $\pi$ in $[0, \tau_1]$ and $[\tau_{i+1}, \infty)$, the problem can be analyzed only in interval $(\tau_i, \tau_{i+1})$. Let $E[C_{(a,b)}^\xi]$ be the total average holding and switching costs during interval $(a, b)$ when policy $\xi$ is applied, and let $E[H_{(a,b)}^u]^\xi$ be the average holding cost of $l$ units during interval $(a, b)$ in stage $u = 1, 2$ when policy $\xi$ is applied. We will
then have
\[
E[C_{(\tau_n, \tau_{n+1})}] = E[H_{(\tau_n, \tau_{n+1})}^A]_1^r + E[H_{(\tau_n, \tau_{n+1})}^{n-r}]_1^r + E[H_{(\tau_n, \tau_{S1,12})}]_1^r \\
+ E[H_{(\tau_n, \tau_{S2,12})}]_2^r + E[H_{(\tau_{S1,12}, \tau_{S2,21})}]_2^r + K_{12}
\]  
(4.72)

and
\[
E[C_{(\tau_n, \tau_{n+1})}] = E[H_{(\tau_n, \tau_{n+1})}^A]_1^r + E[H_{(\tau_n, \tau_{n+1})}^{n-r}]_1^r + E[H_{(\tau_n, \tau_{S1,12})}]_1^r + K_{12} \\
+ E[H_{(\tau_n, \tau_{S1,12})}]_2^r + E[H_{(\tau_{S1,12}, \tau_{S2,21})}]_2^r + K_{21} \\
+ E[H_{(\tau_n, \tau_{S2,21})}]_1^r + E[H_{(\tau_{S2,12}, \tau_{S2,21})}]_1^r \\
+ E[H_{(\tau_{S1,12}, \tau_{S2,21})}]_2^r + E[H_{(\tau_{S2,12}, \tau_{S2,21})}]_2^r + K_{12} .
\]  
(4.73)

where \(E[H_{(\tau_n, \tau_{n+1})}]_1^r\) is the average holding cost of arriving customers in stage 1 during interval \((\tau_n, \tau_{n+1})\) when policy \(\xi\) is applied. It can be shown that
\[
E[H_{(\tau_n, \tau_{S1,12})}]_1^r = h_1 \frac{r(r + 1)}{2} E[S_1] \\
E[H_{(\tau_n, \tau_{S1,12})}]_2^r = h_2 \frac{r(r - 1)}{2} E[S_1] \\
E[H_{(\tau_{S1,12}, \tau_{S2,21})}]_1^r = h_2 \frac{r(r + 1)}{2} E[S_2] .
\]

Therefore,
\[
E[C_{(\tau_n, \tau_{n+1})}] = E[H_{(\tau_n, \tau_{n+1})}^A]_1^r + E[H_{(\tau_n, \tau_{n+1})}^{n-r}]_1^r + h_1 \frac{r(r + 1)}{2} E[S_1] \\
+ h_2 \frac{r}{2} ((r - 1)E[S_1] + (r + 1)E[S_2]) + K_{21} .
\]  
(4.74)

On the other hand,
\[
E[H_{(\tau_n, \tau_{S1,12})}]_1^r = h_1 \frac{(r - k)(r - k + 1)}{2} E[S_1] \\
E[H_{(\tau_n, \tau_{S1,12})}]_2^r = h_2 \frac{(r - k)(r - k - 1)}{2} E[S_1] \\
E[H_{(\tau_{S1,12}, \tau_{S2,21})}]_1^r = h_2 \frac{(r - k)(r - k + 1)}{2} E[S_2] \\
E[H_{(\tau_n, \tau_{S2,21})}]_1^r = h_1 k(r - k)(E[S_1] + E[S_2]) \\
E[H_{(\tau_{S1,12}, \tau_{S2,21})}]_1^r = h_2 \frac{k(k + 1)}{2} E[S_1] \\
E[H_{(\tau_{S2,12}, \tau_{S2,21})}]_2^r = h_2 \frac{k(k - 1)}{2} E[S_1] \\
E[H_{(\tau_{S2,12}, \tau_{S2,21})}]_2^r = h_2 \frac{k(k + 1)}{2} E[S_2] .
\]
and thus,

\[ E[C_{(r, r+n+1)}^\gamma] = E[H_{(r, r+n+1)}^A]_{1} + E[H_{(r, r+n+1)}^{n-r}]_{1} \]
\[ + h_1 \left( \frac{(r-k)(r-k+1)}{2} + \frac{k(k+1)}{2} + k(r-k)E[S_1] \right) \]
\[ + h_2 \left( \frac{(r-k)(r-k-1)}{2} + \frac{k(k-1)}{2} \right) E[S_1] + h_1 k(r-k)E[S_2] \]
\[ + h_2 \left( \frac{(r-k)(r-k+1)}{2} + \frac{k(k+1)}{2} \right) E[S_2] + 2(K_{12} + K_{21}) \] (4.75)

Since \( E[H_{(r, r+n+1)}^A]_{1} = E[H_{(r, r+n+1)}^{n-r}]_{1} \) and \( E[H_{(r, r+n+1)}^{n-r}]_{1} = E[H_{(r, r+n+1)}^{n-r}]_{1} \), therefore after some algebra,

\[ E[C_{(r, r+n+1)}^\gamma] = E[C_{(r, r+n+1)}]_1 - E[C_{(r, r+n+1)}] = K_{12} + K_{21} - k(r-k)((h_2 - h_1)E[S_2] + h_2 E[S_1]) \] (4.76)

and if,

\[ E[C_{(0, r+n+1)}^\gamma] - E[C_{(0, r+n+1)}] \leq 0 \],

or, since \( h_2 \geq h_1 \), if

\[ \frac{K_{12} + K_{21}}{(h_2 - h_1)E[S_2] + h_2 E[S_1]} \leq k(r-k) \] (4.77)

then considering total average holding and switching costs, policy \( \gamma \) is always better than \( \pi \).

The least positive value of right-hand side (4.77) is obtained when \( r = 2 \) and \( k = 1 \). and therefore, if

\[ \frac{K_{12} + K_{21}}{(h_2 - h_1)E[S_2] + h_2 E[S_1]} \leq 1 \]

or

\[ K_{12} + K_{21} \leq (h_2 - h_1)E[S_2] + h_2 E[S_1] \] (4.78)

it is always better to apply policy \( \gamma \) which serves customers according to a sequential service policy.

To show that if the sequential service policy is optimal, then condition (4.71) is satisfied, we use contradiction. Suppose that the sequential service policy is optimal and we have

\[ K_{12} + K_{21} > (h_2 - h_1)E[S_2] + h_2 E[S_1] \] (4.79)
Then if a sequential service policy is compared with a gated limited policy with limit $M_G = 2$, it can be easily shown that the latter has less total average holding and switching costs, and therefore, the sequential service policy is not the optimal policy (contradiction). □

**Remark 4.2**

Condition (4.71) does not depend on the arrival distribution or service time distribution as long as stationary condition $\lambda(E[S_1] + E[S_2]) \leq 1$ holds. □

**Remark 4.3**

Condition (4.71) is a general result and does not depend on the definition of the action space in a semi-Markov decision process. And as long as $h_2 \geq h_1$ and (4.71) is satisfied, the optimal policy is a sequential service policy. □

### 4.5.2 Sufficient Condition for Optimality of $M_G^* = 2$

In this section we derive a sufficient condition for optimality of limit $M_G^* = 2$, and we show that this condition is also independent of the arrival process.

**Proposition 4.2**

In a two-stage tandem queue attended by a moving server with zero switchover times, holding costs $h_2 \geq h_1$, and switching costs $K_{12}, K_{21}$, the optimal limit $M_G^* = 2$ if

$$1 \leq \mathcal{R}_{12} \leq 2$$  \hspace{1cm} (4.80)

where

$$\mathcal{R}_{12} = \frac{K_{12} + K_{21}}{(h_2 - h_1)E[S_2] + h_2E[S_1]}.$$  \hspace{1cm} (4.81)

**Proof:**

Using Proposition 4.1 it is clear that if $\mathcal{R}_{12} > 1$, then a sequential service policy is not optimal and $M_G^* \geq 2$. We must show that if $\mathcal{R}_{12} > 1$, the gated-limited policy with limit 2 is better than a gated-limited policy with limits greater than 2. We
only show that if $R_{12} > 1$, the gated-limited policy with limit 2 is better than a gated-limited policy with limit 3. Comparison of limit 2 with limits $M_G > 3$ can be performed using the same line of argumentation and therefore will be omitted.

Consider the sample path of two gated-limited policies $\pi$ and $\beta$, with limits 3 and 2, respectively, from time $t = 0$. Suppose $t_{G}^{\pi}$ ($t_{G}^{\beta}$), measured from time $t = 0$, is the time that the server switches to stage 1 for the $k$th time, and gate $G$ is opened under policy $\pi$ ($\beta$). Also consider $\tau_{G}^{\pi}$, measured from time $t = 0$, as the time that gate $G$ is opened for the $k$th time at the same time under policies $\pi$ and $\beta$. Thus, $\tau_{G}^{\pi} = t_{G}^{\pi} = t_{G}^{\beta}$ for $k = 1, 2, \ldots, i$ implies that both policies $\pi$ and $\beta$ behave the same up to time $\tau_{G}^{i}$ (Figure 4.3). In other words, both policies serve fewer than three customers in each cycle up to time $\tau_{G}^{i}$, and $\tau_{G}^{i}$ is the first time that policy $\pi$ starts a cycle in which three customers are served and policy $\beta$ serves two customers in that cycle. Therefore, $\tau_{G}^{i+1} = t_{G}^{i} = t_{G}^{i+1}$, where $m \geq n$, and consequently, any sample path of policies $\pi$ and $\beta$ consists of intervals $[\tau_{G}^{i}, \tau_{G}^{i+1}]$ in which the same number of customers are served under policies $\pi$ and $\beta$.

The proof has two parts. In part (i) based on policy $\pi$, we construct policy $\gamma$ which is not a gated-limited policy, and we show that if $R_{12} \leq 2$, then policy $\gamma$ is always better than $\pi$. Then, in part (ii), based on policy $\gamma$, we construct policy $\beta$ which is a gated-limited policy with limit 2 and we show that if $1 \leq R_{12} \leq 2$, then policy $\beta$ is always better than policy $\gamma$.

(i) Suppose that policy $\gamma$ behaves exactly the same as $\pi$ except for cycles in which 3 customers are served, in which case, policy $\gamma$ serves 2 and 1 customers in
two consecutive cycles. Therefore, if \( N_{i,i+1}^\pi \) (\( N_{i,i+1}^\gamma \)) is the number of cycles in interval \([\tau_i^\pi, \tau_{i+1}^\pi]\) (\([\tau_i^\gamma, \tau_{i+1}^\gamma]\)) under policy \( \pi \) (\( \gamma \)), then policies \( \pi \) and \( \gamma \) behave the same in intervals \([\tau_j^\pi, \tau_{j+1}^\pi]\) (\([\tau_j^\gamma, \tau_{j+1}^\gamma]\)) in which \( N_{j,j+1}^\pi = N_{j,j+1}^\gamma \), and hence \( E[C_{[\tau_j^\pi, \tau_{j+1}^\pi]}^\pi] = E[C_{[\tau_j^\gamma, \tau_{j+1}^\gamma]}^\gamma] \) (\( E[C_{[\tau_j^\pi, \tau_{j+1}^\pi]}^\pi] \)) is the total average cost during interval \((a, b)\) under policy \( \xi \). Therefore, it is sufficient to compare these policies only in intervals \([\tau_i^\pi, \tau_{i+1}^\pi]\) where \( N_{i,i+1}^\pi < N_{i,i+1}^\gamma \). Using (4.71), it is clear that if \( R_{12} \leq 2 \), then \( E[C_{[\tau_i^\pi, \tau_{i+1}^\pi]}^\pi] \geq E[C_{[\tau_i^\gamma, \tau_{i+1}^\gamma]}^\gamma] \) for \( i = 1, 2, \ldots \), because if \( R_{12} \leq 2 \), it is always better to serve three customers in two consecutive cycles than in one cycle. Consequently, policy \( \gamma \) is always better than \( \pi \) in terms of total average holding and switching costs.

As previously noted policy \( \gamma \) is not a gated-limited policy because at time \( t_{i+1}^G \), according to policy \( \gamma \), the server switches to stage 1, but gate \( G \) is not opened at that time, and the server completes the service of the only customer waiting in stage 1. However, in part (ii) we compare \( \beta \) and \( \gamma \), and we show that \( \beta \) has less total average holding and switching cost than \( \gamma \) when \( R_{12} \geq 1 \).

(ii) Policies \( \beta \) and \( \gamma \) act the same as a gated-limited policy with limit 2 up to time \( \tau_i^G \) and also in intervals \([\tau_j^\beta, \tau_{j+1}^\beta]\) where \( N_{j,j+1}^\beta = N_{j,j+1}^\gamma = N_{j,j+1}^\gamma \). Thus, it only remains to show that policy \( \beta \) is better than \( \gamma \) in intervals \([\tau_i^\beta, \tau_{i+1}^\beta]\) in which \( N_{i,i+1}^\pi < N_{i,i+1}^\beta < N_{i,i+1}^\gamma \).

Let vector \( \Lambda_{[\tau_i^\xi, \tau_{i+1}^\xi]}^{\xi, \gamma, \beta} = (c_1^\xi, c_2^\xi, \ldots, c_k^\xi) \) be a sequence of \( k \) cycles in interval \([\tau_i^\xi, \tau_{i+1}^\xi]\) under policy \( \xi \), and let \( c_r^\xi \) be the number of customers served in the \( r \)th cycle in this interval. Also, let \( t_{cr}^\xi \) be the time that the \( r \)th cycle in interval \([\tau_i^\xi, \tau_{i+1}^\xi]\) starts under policy \( \xi \). Then, for policy \( \gamma \), we have \( \Lambda_{[\tau_i^\beta, \tau_{i+1}^\beta]}^{\gamma, \beta} = (2, \nu_1^\gamma, \nu_2^\gamma, \ldots, \nu_l^\gamma) \), where \( l = N_{i,i+1}^\gamma \).

If at time \( t_{c_2}^\gamma \), when the server switches to stage 1, there are no customers waiting in queue \( Q_0 \), then although according to policy \( \gamma \) gate \( G \) is not open, but since there is no one behind the gate, \( \gamma \) actually behaves like a gated-limited policy with limit 2. In other words, \( \gamma \) is a gated-limited policy up to time \( t_{c_2}^\gamma \). However, if at time \( t_{c_2}^\gamma \) when the server switches to stage 1, there is at least one customer behind the gate \( G \) in queue \( Q_0 \) (and obviously one in stage 1), then \( \gamma \) does not act like a gated-limited policy in interval \([\tau_i^G, \tau_{c_2}^\gamma] \). In this case, based on policy \( \gamma \), we construct policy \( \beta_1 \)
which is a gated-limited policy with limit 2 at least up to time \( t_{c_3}^* \) and it is shown that it has a lower total average cost than does \( \gamma \) when \( c_3^* = 1.2 \).

(a) If \( c_3^* = 1 \) and at time \( t_{c_2}^* \) there is at least one customer in queue \( Q_0 \), then policy \( \beta_1 \) can be constructed as follows: \( \beta_1 \) follows \( \gamma \) up to time \( t_{c_3}^* \), then \( \beta_1 \) serves 2 customers and follows \( \gamma \) in interval \([t_{c_4}^*, \infty)\). Thus \( \beta_1 \) is a gated-limited policy up to time \( t_{c_4}^* \), and we have

\[
E[C_{[t_{c_1}^*, t_{c_4}^*], t_{c_4}^*]}] - E[C_{[t_{c_1}^*, t_{c_4}^*], t_{c_4}^*]}] = E[C_{[t_{c_1}^*, t_{c_4}^*]}] - E[C_{[t_{c_1}^*, t_{c_4}^*]}].
\]

Policy \( \gamma \) actually serves two customers (who are available at time \( t_{c_1}^* \)) in two cycles; however, policy \( \beta_1 \) serves these customers in one cycle. Using (4.71) it is clear that if \( R_{12} \geq 1 \), then \( E[C_{[t_{c_1}^*, t_{c_4}^*]}] \leq E[C_{[t_{c_1}^*, t_{c_4}^*]}]. \) In other words, if \( R_{12} \geq 1 \), policy \( \beta_1 \), which is a gated-limited policy with limit 2 up to time \( t_{c_4}^* \), is always better than \( \gamma \).

(b) If \( c_3^* = 2 \) and at time \( t_{c_2}^* \) there is at least one customer in queue \( Q_0 \), then policy \( \beta_1 \) can be constructed as follows: \( \beta_1 \) follows \( \gamma \) up to time \( t_{c_3}^* \), \( \beta_1 \) serves 2 and 1 customers, respectively, in two consecutive cycles and then follows \( \gamma \) in interval \([t_{c_4}^*, \infty)\). Therefore, \( \beta_1 \) is a gated-limited policy up to time \( t_{c_3}^* \) and we also have

\[
E[C_{[t_{c_1}^*, t_{c_4}^*]}] - E[C_{[t_{c_1}^*, t_{c_4}^*]}] = E[C_{[t_{c_1}^*, t_{c_4}^*]}] - E[C_{[t_{c_1}^*, t_{c_4}^*]}].
\]

and \( \Lambda_{[t_{c_1}^*, t_{c_4}^*]} = (2, 1, 2) \), \( \Lambda_{[t_{c_1}^*, t_{c_4}^*]} = (2, 2, 1) \). However, it can be shown that in this case \( E[C_{[t_{c_1}^*, t_{c_4}^*]}] = E[C_{[t_{c_1}^*, t_{c_4}^*]}] \), and hence, policy \( \beta_1 \), which is a gated-limited policy up to time \( t_{c_4}^* \), is as good as policy \( \gamma \) in terms of total average cost.

Combining results in (a) and (b), we conclude that when \( R_{12} \geq 1 \), policy \( \gamma \) can always be improved by policy \( \beta_1 \) which is a gated-limited policy at least up to time \( t_{c_3}^* \). Using the same argument and constructing policy \( \beta_2 \) based on \( \beta_1 \), it can be shown that policy \( \beta_2 \), which is a gated-limited policy with limit 2 for a longer time than \( \beta_1 \), has a lower total average cost. Finally, iterating this approach we reach policy \( \beta \) which is a gated-limited policy with limit 2, and it is always better than \( \gamma \) when \( R_{12} \geq 1 \).

Based on results in (i) and (ii), we conclude that if \( 1 \leq R_{12} \leq 2 \) then the gated-limited policy \( \beta \) with limit 2 has lower total average holding and switching costs than gated-limited policy \( \pi \) with limit 3. □
Remark 4.4

Using a similar argument for limited policies, we can conclude that condition (4.80) is also a sufficient condition for optimality of the limited policy with limit 2. \(\square\)

4.5.3 Optimal Limits in High Traffic Intensity

As Propositions 4.1 and 4.2 show, it seems that the optimal limit \(M_G\) does not depend on the arrival process at least for limits 1 and 2 (if \(R_{12} \leq 2\)), in systems with zero switchover times. This independence property for \(M_G^* \geq 2\) can be investigated by analyzing the system in high traffic intensity, because in high traffic intensity, the optimal limits do not depend on the arrival process. Proposition 4.3 obtains these limits.

Proposition 4.3

In a two-stage tandem queue attended by a moving server with zero switchover times, holding costs \(h_2 \geq h_1\), and switching costs \(K_{12}\) and \(K_{21}\), the optimal limit \(M_G^*\) in high traffic intensity satisfies

\[
\frac{M_G^*(M_G^* - 1)}{2} \leq R_{12} \leq \frac{M_G^*(M_G^* + 1)}{2}.
\] (4.82)

Proof

In high traffic intensity where \(\rho = \lambda(E[S_1] + E[S_2]) \simeq 1\), there are a huge number of customers waiting in stage 1. In this system gated-limited policy \(\omega_0\) with parameter \(M_G\) actually chooses \(M_G\) customers to serve in each cycle. Considering \(E[C_{(0,M_{G}(M_{G}+1)E[S]),\omega_0}]\) as the total average holding and switching cost when policy \(\omega_0\) is used to serve \(M_G(M_G + 1)\) customers in time interval \((0,M_G(M_G + 1)E[S])\) (where \(S = S_1 + S_2\)), we will have

\[
E[C_{(0,M_{G}(M_{G}+1)E[S]),\omega_0}] = (M_G + 1)(h_1 \frac{M_G(M_G + 1)}{2} E[S_1] + h_2 \frac{M_G(M_G - 1)}{2} E[S_1] + h_2 \frac{M_G(M_G + 1)}{2} E[S_2]) + (M_G + 1)(K_{12} + K_{21})
\]
\[ + M_G h_1(M_G E[S]) \left( \frac{M_G(M_G + 1)}{2} \right). \quad (4.83) \]

Now suppose the same number of customers, \( M_G(M_G + 1) \), is served in the same time interval \( (0, M_G(M_G + 1) E[S]) \) using gated-limited policy \( \pi_1 \) with parameter \( M_G + 1 \). Therefore

\[
E[C^{\pi_1}_{(0, M_G(M_G + 1) E[S])}] = M_G \left( h_1 \left( \frac{M_G + 1}{2} \right) + h_2 \frac{M_G(M_G + 1)}{2} E[S_1] \right) + h_2 \left( \frac{M_G + 1}{2} \right) E[S_2] + M_G(K_{12} + K_{21}) + (M_G + 1) h_1(M_G + 1) E[S] \left( \frac{M_G(M_G - 1)}{2} \right). \quad (4.84)
\]

and after some algebra,

\[
E[C^{\pi_1}_{(0, M_G(M_G + 1) E[S])}] - E[C^{\pi_0}_{(0, M_G(M_G + 1) E[S])}] = \frac{M_G(M_G + 1)}{2} \left( (h_2 - h_1) E[S_2] + h_2 E[S_1] \right) - (K_{12} + K_{21}). \quad (4.85)
\]

and if \( E[C^{\pi_1}_{(0, M_G(M_G + 1) E[S])}] - E[C^{\pi_0}_{(0, M_G(M_G + 1) E[S])}] \geq 0 \), or in other words if

\[
\frac{K_{12} + K_{21}}{(h_2 - h_1) E[S_2] + h_2 E[S_1]} \leq \frac{M_G(M_G + 1)}{2}, \quad (4.86)
\]

then policy \( \pi_0 \) is always better than policy \( \pi_1 \). By using the same approach in comparing policies \( \pi_0 \) and \( \pi_{-1} \) with parameter \( M_G - 1 \) yields the result that if

\[
\frac{K_{12} + K_{21}}{(h_2 - h_1) E[S_2] + h_2 E[S_1]} \geq \frac{M_G(M_G - 1)}{2}, \quad (4.87)
\]

then policy \( \pi_0 \) is always better than policy \( \pi_{-1} \). Finally by considering (4.81) and combining (4.86) and (4.87), we can see that policy \( \pi_G^\ast \) with parameter \( M_G^\ast \) minimizes the total average holding and switching cost in high traffic intensity if

\[
\frac{M_G^\ast(M_G^\ast - 1)}{2} \leq \mathcal{R}_{12} \leq \frac{M_G^\ast(M_G^\ast + 1)}{2},
\]

and these intervals cover the results which were obtained in Propositions 4.1 and 4.2.

\[ \square \]

To further investigate this independence relation between the optimal limit \( M_G^\ast \) and the arrival process in a system with zero switchover times, in the next section a
numerical study is presented in which the optimal limits in systems with high traffic intensities are compared with optimal limits in systems with lower traffic intensities.

### 4.5.4 Numerical Study

Consider an $M/HE_2 - D/1$ queue in which service times in stage 1 are hyperexponential random variables with parameters $p = 0.6$, $\mu_{11} = 12$ and $\mu_{12} = 6$ per hour. Also, suppose that service times in stage 2 are considered to be deterministic with $S_2 = 2$ minutes. Tables 4.3, 4.4, and 4.5 show the characteristics of this model for $\lambda = 2$, $\lambda = 4$ and $\lambda = 6$ per hour, respectively, when gated-limited policies are applied.

#### Table 4.3. Characteristics of $M/HE_2 - D/1$ when $\lambda = 2$ ($\rho = 0.3$).

<table>
<thead>
<tr>
<th>$MG$</th>
<th>Roots of $f(z)$</th>
<th>$L_1$</th>
<th>$L_2$</th>
<th>$L_{1,2}$</th>
<th>$E[N_{Ma}^u]$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>0.3460</td>
<td>0.0667</td>
<td>0.4127</td>
<td>1.0000</td>
</tr>
<tr>
<td>2</td>
<td>-0.6524</td>
<td>0.3422</td>
<td>0.0837</td>
<td>0.4259</td>
<td>1.0602</td>
</tr>
<tr>
<td>3</td>
<td>-0.4156±0.5455i</td>
<td>0.3049</td>
<td>0.0897</td>
<td>0.4306</td>
<td>1.0691</td>
</tr>
<tr>
<td>4</td>
<td>-0.6524, -0.1273±0.7202i</td>
<td>0.3405</td>
<td>0.0917</td>
<td>0.4322</td>
<td>1.0706</td>
</tr>
<tr>
<td>5</td>
<td>0.0940±0.7652i</td>
<td>-0.5666±0.3464i</td>
<td>0.3403</td>
<td>0.0924</td>
<td>0.4327</td>
</tr>
<tr>
<td>6</td>
<td>-0.6524, 0.2586±0.7607i</td>
<td>-0.4156±0.5455i</td>
<td>0.3403</td>
<td>0.0925</td>
<td>0.4328</td>
</tr>
</tbody>
</table>

#### Table 4.4. Characteristics of $M/HE_2 - D/1$ when $\lambda = 4$ ($\rho = 0.6$).

<table>
<thead>
<tr>
<th>$MG$</th>
<th>Roots of $f(z)$</th>
<th>$L_1$</th>
<th>$L_2$</th>
<th>$L_{1,2}$</th>
<th>$E[N_{Ma}^u]$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>1.2556</td>
<td>0.1333</td>
<td>0.3889</td>
<td>1.0000</td>
</tr>
<tr>
<td>2</td>
<td>-0.4927</td>
<td>1.2293</td>
<td>0.2516</td>
<td>1.4808</td>
<td>1.2454</td>
</tr>
<tr>
<td>3</td>
<td>-0.3310±0.4037i</td>
<td>1.2131</td>
<td>0.3243</td>
<td>1.5374</td>
<td>1.3155</td>
</tr>
<tr>
<td>4</td>
<td>-0.4927, -0.1287±0.5488i</td>
<td>1.2035</td>
<td>0.3678</td>
<td>1.5712</td>
<td>1.3387</td>
</tr>
<tr>
<td>5</td>
<td>0.0321±0.6005i</td>
<td>-0.4345±0.2533i</td>
<td>1.1977</td>
<td>0.3937</td>
<td>1.5914</td>
</tr>
<tr>
<td>6</td>
<td>-0.4927, 0.1563±0.6143i</td>
<td>-0.3310±0.4037i</td>
<td>1.1946</td>
<td>0.4077</td>
<td>1.6023</td>
</tr>
</tbody>
</table>

#### Table 4.5. Characteristics of $M/HE_2 - D/1$ when $\lambda = 6$ ($\rho = 0.9$).

<table>
<thead>
<tr>
<th>$MG$</th>
<th>Roots of $f(z)$</th>
<th>$L_1$</th>
<th>$L_2$</th>
<th>$L_{1,2}$</th>
<th>$E[N_{Ma}^u]$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>7.8000</td>
<td>0.2000</td>
<td>8.0000</td>
<td>1.0000</td>
</tr>
<tr>
<td>2</td>
<td>-0.3940</td>
<td>7.7177</td>
<td>0.5704</td>
<td>8.2881</td>
<td>1.6994</td>
</tr>
<tr>
<td>3</td>
<td>-0.2694±0.3205i</td>
<td>7.6423</td>
<td>0.9114</td>
<td>8.5537</td>
<td>2.1537</td>
</tr>
<tr>
<td>4</td>
<td>-0.3940, -0.1121±0.4398i</td>
<td>7.5813</td>
<td>1.2031</td>
<td>8.7844</td>
<td>2.4571</td>
</tr>
<tr>
<td>5</td>
<td>0.0144±0.4859i</td>
<td>-0.3493±0.2002i</td>
<td>7.5243</td>
<td>1.4615</td>
<td>8.9858</td>
</tr>
<tr>
<td>6</td>
<td>-0.3940, 0.1132±0.5015i</td>
<td>-0.2694±0.3205i</td>
<td>7.4702</td>
<td>1.6928</td>
<td>9.1630</td>
</tr>
</tbody>
</table>
Remark 4.5

All computations in this research are performed in double precision using MATLAB software (The Mathworks, Inc) version 4.2.a.

For holding costs $h_1 = 16$, $h_2 = 20$ and switching cost $K_{12} + K_{21} = 10$, the optimal limit $M^*_G = 3$ is obtained for all cases ($\rho = 0.3$, 0.6, 0.9). Table 4.6 shows the total average holding and switching costs per unit time, $E[T_{CM_G}]$, and Figure 4.4 depicts $E[T_{CM_G}]$ for different values of $\lambda$.

<table>
<thead>
<tr>
<th>$M_G$</th>
<th>$\lambda = 2$</th>
<th>$\lambda = 4$</th>
<th>$\lambda = 6$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>26.8698</td>
<td>62.7556</td>
<td>188.8000</td>
</tr>
<tr>
<td>2</td>
<td>26.0146</td>
<td>56.8177</td>
<td>170.1979</td>
</tr>
<tr>
<td>3</td>
<td>25.9570</td>
<td>56.3017</td>
<td>168.2001</td>
</tr>
<tr>
<td>4</td>
<td>25.9638</td>
<td>56.4908</td>
<td>169.7398</td>
</tr>
<tr>
<td>5</td>
<td>25.9682</td>
<td>56.7290</td>
<td>172.3010</td>
</tr>
<tr>
<td>6</td>
<td>25.9685</td>
<td>56.8958</td>
<td>175.0656</td>
</tr>
</tbody>
</table>

It is clear from Figure 4.4 that the total average cost per unit time $E[T_{CM_G}]$ is flatter at the bottom for systems with lower $\rho$, and for our example, the variation of
\( \lambda \) does not influence the optimal limit \( M_G \). To investigate this result further, let us consider the influences of \( R_{12} \) on the optimal limit \( M_G \). Suppose curve \( L^G_{\rho} \) shows the variations of the optimal limit \( M_G \) in terms of changes in \( R_{12} \), for a system with traffic intensity \( \rho \) (\( \rho = 1 \) is considered as high traffic intensity). Then Figure 4.5 depicts \( L^G_{0.3}, L^G_{0.6}, L^G_{0.9} \) and \( L^G_{1} \) for fixed \( h_1 = 16, h_2 = 20 \) and different values of \( K_{12} + K_{21} \), and therefore, for different \( R_{12} \). It should be noted that for any combination of costs, Figure 4.5 remains the same. Let \( \delta \) be the cost relative error of the optimal limit obtained under high traffic intensity when it is used in a system with a lower traffic intensity. Therefore, considering Figure 4.5, for point a in the system with \( \rho = 0.3 \), we have

\[
\delta_a = \frac{E[TC_{M_G=2}] - E[TC_{M_G=3}]}{E[TC_{M_G=3}]} \bigg|_{R_{12}=2.56} = \frac{19.091268 - 19.091156}{19.091156} \approx 6 \times 10^{-6} .
\]

Tables 4.7, 4.8 and 4.9 show the total average cost per unit time and cost relative error \( \delta \) for points a to f in Figure 4.5.
Figure 4.5. Optimal limit curves $\mathcal{L}_\rho^G$ and $\mathcal{L}_\rho^G$ for gated-limited policy.
Several similar numerical analyses were also performed for problems with different \( \rho \) and different service distributions such as exponential, Erlang, deterministic and hyperexponential. From the results the following conclusions were obtained:

- The gap between optimal limit curves \( L^G_\rho \) and \( L^G_1 \) increases as \( \rho \) decreases or \( R_{12} \) increases.

- The bottom of curve \( E[TC_{M_G}] \) becomes flatter as \( \rho \) decreases.

- As \( \rho \) decreases, the gated-limited policy with limit \( M_G + 1 \) acts more like the gated-limited policy with limit \( M_G \), and in systems with low traffic intensity, gated-limited policies with limits \( M_G \geq 2 \) behave like a sequential service policy.

- If \( L^G_1 \) is used to find the optimal limit for a system with \( \rho < 1 \), the maximum cost relative error is less than 1\% and the average cost relative error, given that there is an error, is less than 0.01\%. This may occur for one of the following reasons:

  1. Similar to the cases \( M_G \leq 2 \), the optimal limit is independent of the arrival process, and the cost relative errors, which are very low, are created during computations on complex numbers.

  2. Since in systems with low traffic intensity \( E[TC_{M_G}] \) has a flat bottom, there is a slight difference between the total average cost of different limits. Therefore, although the optimal limits in high traffic intensity are not optimal in this case, the cost relative error is very low. On the other hand, as \( \rho \) increases toward 1, the gap between \( L^G_\rho \) and \( L^G_1 \) decreases and \( L^G_1 \) will yield the optimal limit. Consequently, in both high and low traffic intensities, \( L^G_1 \) yields a limit which differs insignificantly from the optimal total average cost.

These results, along with Proposition 4.1 and 4.2, corroborate the notion of independence between the optimal limit of a gated-limited policy and the arrival process, and makes the gated-limited policy an attractive policy in situations where there is
no information available regarding the arrival process (in a two-stage tandem queue attended by a moving server with zero switchover times).
Chapter 5

An $M/G_1 - G_2/1$ Queue with Dynamic Policy

5.1 Introduction

Consider the BTQ model with a greedy and exhaustive policy in the second stage and assume that the server has complete information about the number of customers in the system after each service completion. Thus, at each service completion epoch in stage 1, he must decide whether to serve the next customer in stage 1, switch to stage 2, or, if stage 1 is empty, wait for a customer to come and then serve him. This policy is called a dynamic policy. In this chapter, optimal dynamic policies are studied first, and then a model is developed to find the parameter of a class of dynamic policies with double-threshold, for the BTQ model. Finally, using numerical results, different properties of static, limited, gated-limited and double-threshold policies are compared.

5.2 Dynamic Policy

Suppose that the server implements an exhaustive and greedy policy in the second stage in our basic two-stage tandem queue. Also, suppose that after each service completion in stage 1, the server can decide whether to switch to stage 2, serve the
next customer in stage 1, or wait for the next customer to come and serve him if stage 1 is empty. We defined this policy as a dynamic policy. Assuming that the server never idles in stage 1 when there is at least one customer in that stage, and since idleness in stage 1 is not optimal before switching to stage 2 when total long-run average cost is considered, the BSD model can be revised as follows:

- *decision epochs* are now service completion times in stage 1,
- *state* of the system at any decision epoch consists of \((x_1, x_2)\), where \(x_i\) is the number of customers in stage \(i\) \((i = 1, 2)\).
- The server’s *actions* when \(x_1, x_2 > 0\), are: *switch* to stage 2 \((Sw.)\), or *serve* another customer in stage 1 \((Sr.)\). When \(x_1 = 0, x_2 > 0\), they are: *wait* for another customer to come and serve him \((Id.)\), or *switch* to stage 2 \((Sw.)\).

Therefore, using a dynamic policy the server has a chance to decide his next action based on the state of the system at each service completion in stage 1.

### 5.3 Optimal Dynamic Policy

Now we use an intuitive approach to characterize the optimal policy and we concentrate on a class of dynamic policy with a double-threshold. Suppose \(\Omega_a\) is the set of all states in which action \(a = \{Sr., Sw., Id.\}\) is optimal. Then we have

\[
(0, x_2) \in \Omega_{Id} \quad \Longrightarrow \quad \{(0, x_2 - 1), (0, x_2 - 2), \ldots, (0, 0)\} \in \Omega_{Id}
\]

\[
(0, x'_2) \in \Omega_{Sw} \quad \Longrightarrow \quad \{(0, x'_2 + 1), (0, x'_2 + 2), \ldots\} \in \Omega_{Sw}.
\]

Therefore, there should be an integer \(M_1\), where

\[
\{(0, 1), (0, 2), \ldots, (0, M_1 - 1)\} \in \Omega_{Id}, \quad \{(0, M_1), (0, M_1 + 1), \ldots\} \in \Omega_{Sw}.
\]

On the other hand, for \(x_1, x'_1 > 0,\)

\[
(x_1, x_2) \in \Omega_{Sr} \quad \Longrightarrow \quad \{(x_1 + 1, x_2), (x_1 + 2, x_2), \ldots\} \in \Omega_{Sr}
\]

\[
(x'_1, x'_2) \in \Omega_{Sw} \quad \Longrightarrow \quad \{(x'_1, x'_2 + 1), (x'_1, x'_2 + 2), \ldots\} \in \Omega_{Sw}.
\]
but the optimal action in states \((x_1, x_2 + 1)\) and \((x'_1 + 1, x'_2)\) may depend on \(x_1\) and \(x'_2\), respectively. Thus, we can conclude that the optimal dynamic policy can be fully described by switching curve \(\mathcal{L}(x_1, x_2)\), as shown in Figure 5.1.

We are looking for the best approximation for the multi-threshold switching curve \(\mathcal{L}(x_1, x_2)\) with thresholds \(M_1, M_2, \ldots\), using a double threshold switching curve \(\tilde{\mathcal{L}}(x_1, x_2)\) with thresholds \(M_e, M_n\). Applying double-threshold policy \(\tilde{\mathcal{L}}(x_1, x_2)\), the server continues serving at stage 1 either until \(M_n\) services have been completed without interruption, or until the first stage becomes empty and there are at least \(M_e\) customers in stage 2 \((M_e \leq M_n)\), whichever comes first. After servicing at stage 1, the server moves to stage 2 and serves there until it becomes empty, and then he returns to stage 1. When there are no customers in the system, the server waits for a new customer at stage 1.

**Remark 5.1.**

The double-threshold policy is a

- *sequential* service policy when \(M_e = 1\) and \(M_n = 1\),

- *limited* service policy when \(M_e = 1\) and \(M_n > 1\),

- *static* service policy when \(M_e = M_n\), and

- *exhaustive* service policy when \(M_e = 1\) and \(M_n \to \infty\).
5.4 An $M/G_1-G_2/1$ with Double-Threshold Policy

In this section we will analyze the BTQ model with a double-threshold policy. In this policy, as already described, the server continues serving at stage 1 until $M_n$ services have been completed without interruption, or until first stage becomes empty and there are at least $M_e$ customers in stage 2 ($M_e \leq M_n$), whichever comes first. After servicing at stage 1, the server moves to stage 2 and serves there until it becomes empty, and then he returns to stage 1. When there are no customers in the system, the server waits for a new customer at stage 1. In this section, first the steady state probabilities of the BTQ model under a double-threshold policy are analyzed and then their generating functions, the average number of customers in the system and the average number of customers served in each cycle are obtained.

5.4.1 Analysis of Steady State Probabilities

Let $I^{(n)}$ be the number of the stage at which the $n$th service completion occurs, $I^{(n)} \in \{1, 2\}$, and let $X_k^{(n)}$ be the number of customers in stage $k$, $(k = 1, 2)$ at the $n$th service completion. Then a model for this system can be represented by the imbedded Markov chain $\langle X_1^{(n)}, X_2^{(n)}, I^{(n)} \rangle$, and assuming that steady state conditions prevail, we have $\lim_{n \to \infty} (X_1^{(n)}, X_2^{(n)}, I^{(n)}) = (X_1, X_2, I)$ in distribution, and therefore, the stationary probabilities can be defined as

$$\pi_{ij}^{(k)} = P\{X_1 = i, X_2 = j, I = k\} \quad ; \quad k = 1, 2 .$$

Let $Y_k^s$ be the number of customers that arrive during a service time in stage $k$ and $Y_k^D$ the number of customers that arrive during switchover time $D_k$. (switchover time from stage $k, \quad k = 1, 2$). Then for $k = 1, 2$,

$$p_l^{(k)} = P\{Y_k^s = l\} = \int_0^\infty \frac{e^{-\lambda t} (\lambda t)^l}{l!} dB_k(t)$$

$$d_l^{(k)} = P\{Y_k^D = l\} = \int_0^\infty \frac{e^{-\lambda t} (\lambda t)^l}{l!} dF_k(t) .$$

Therefore, $p_l^{(k)}$ and $d_l^{(k)}$ represent the probabilities that $l$ customers arrive at stage 1 during service time $S_k$ and switchover time $D_k, \quad (k = 1, 2)$, respectively. For a system
with a double-threshold policy, the balance equations for \( i \geq 0 \) are:

\[
\pi_{ij}^{(1)} = \begin{cases} 
0 & : j = 0 \\
\pi_{00}^{(2)} p_i^{(1)} + \sum_{m=0}^{i} \sum_{n=0}^{i-m} \pi_{i-m-n+1,0}^{(2)} d_n^{(2)} p_m^{(1)} & : j = 1 \\
+ \sum_{m=0}^{i} \pi_{00}^{(2)} d_{-m+1}^{(2)} p_m^{(1)} & : 2 \leq j \leq M_e \\
\pi_{0,j-1}^{(1)} + \sum_{m=0}^{i} \pi_{i-m-j-1}^{(1)} p_m^{(1)} & : M_e + 1 \leq j \leq M_n \\
\sum_{m=0}^{i} \pi_{i-m+1,j-1}^{(1)} p_m^{(1)} & : j \geq M_n + 1 
\end{cases}
\tag{5.3}
\]

and

\[
\pi_{ij}^{(2)} = \begin{cases} 
\sum_{m=0}^{i} \pi_{i-m,j+1}^{(2)} p_m^{(2)} & : 0 \leq j \leq M_e - 2 \\
\sum_{m=0}^{i} \pi_{i-m,j+1}^{(2)} p_m^{(2)} + \sum_{m=0}^{i} \pi_{0,j+1}^{(1)} d_m^{(1)} p_{i-m}^{(2)} & : M_e - 1 \leq j \leq M_n - 2 \\
\sum_{m=0}^{i} \sum_{n=0}^{i-m} \pi_{i-m-n,M_n}^{(1)} d_n^{(1)} p_m^{(2)} & : j = M_n - 1 \\
0 & : j \geq M_n 
\end{cases}
\tag{5.4}
\]

along with

\[
\sum_{i=0}^{\infty} \sum_{j=0}^{\infty} \pi_{ij}^{(1)} + \pi_{ij}^{(2)} = 1 .
\tag{5.5}
\]

### 5.4.2 Probability Generating Functions

Define the probability generating function for probabilities \( \pi_{ij}^{(k)} \), \( k = 1, 2 \), by

\[
\Pi_k(x, y) = \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} \pi_{ij}^{(k)} x^i y^j ; k = 1, 2 \, , \, \, |x|, |y| \leq 1 .
\tag{5.6}
\]

Then multiplying both sides of (5.3) by appropriate \( x^i \) and \( y^j \) and summing over \( i \) and \( j \), yields

\[
\Pi_1(x, y) = \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} \pi_{00}^{(2)} p_i^{(1)} x^i y^j + \sum_{i=0}^{\infty} \sum_{m=0}^{i} \sum_{n=0}^{i-m} \pi_{i-m-n+1,0}^{(2)} d_n^{(2)} p_m^{(1)} x^i y^j \\
+ \sum_{i=0}^{\infty} \sum_{m=0}^{i} \pi_{00}^{(2)} d_{-m+1}^{(2)} p_m^{(1)} x^i y^j \\
+ \sum_{i=0}^{M_e} \sum_{j=2}^{i} \pi_{0,j-1}^{(1)} p_i^{(1)} x^i y^j + \sum_{i=0}^{M_e} \sum_{j=2}^{i} \sum_{m=0}^{i} \pi_{i-m-j-1}^{(1)} p_m^{(1)} x^i y^j \\
+ \sum_{i=0}^{M_n} \sum_{j=M_e+1}^{i} \sum_{m=0}^{i} \pi_{i-m+1,j-1}^{(1)} p_m^{(1)} x^i y^j .
\tag{5.7}
\]
Let 
\[ P_k(x) = \sum_{i=0}^{\infty} p_i^{(k)} x^i \quad : \quad k = 1, 2 \quad |x| \leq 1 \]
\[ D_k(x) = \sum_{i=0}^{\infty} d_i^{(k)} x^i \quad : \quad k = 1, 2 \quad |x| \leq 1 . \]

Then, the first term on the right-hand side of (5.7) can be summarized as
\[ \sum_{i=0}^{\infty} \pi_0^{(2)} d_0^{(2)} p_i^{(1)} x^i y = \pi_0^{(2)} d_0^{(2)} y P_1(x) . \] (5.8)

and, for the second term of the right-hand side of (5.7)
\[ \sum_{i=0}^{\infty} \sum_{m=0}^{i-m} \pi_{i-m+1,0}^{(2)} d_m^{(2)} p_m^{(1)} x^i y = \frac{y}{x} \pi_0^{(2)} [D_2(x) - d_0^{(2)}] P_1(x) . \] (5.9)

Also,
\[ \sum_{i=0}^{\infty} \sum_{m=0}^{i} \pi_{i-m+1,0}^{(2)} d_m^{(2)} p_m^{(1)} x^i y = \frac{y}{x} \pi_0^{(2)} [D_2(x) - d_0^{(2)}] P_1(x) . \] (5.10)

and finally, for the last two terms on the right-hand side of (5.7), we have
\[ \sum_{i=0}^{\infty} \sum_{j=2}^{M_n} \sum_{m=0}^{i} \pi_{i-m+1,j-1}^{(1)} p_m^{(1)} x^i y^j + \sum_{i=0}^{\infty} \sum_{j=M_n+1}^{M_n} \sum_{m=0}^{i} \pi_{i-m+1,j-1}^{(1)} p_m^{(1)} x^i y^j = \frac{y}{x} \pi_0^{(1)} [P_1(x,y) - \Pi_{1}(0,y) - \Psi_{M_n}(x,y)] P_1(x) \] (5.12)

where
\[ \Psi_{M_n}(x,y) = \sum_{i=1}^{\infty} \pi_{i,M_n}^{(1)} x^i y^{M_n} . \] (5.13)

Using (5.8) to (5.12) in (5.7), after some algebra we obtain
\[ \Pi_1(x,y) = \frac{y P_1(x)}{x - y P_1(x)} \left[ D_2(x) \Pi_2(x,0) - \Pi_1(0,y) - \Psi_{M_n}(x,y) - \pi_0^{(2)} d_0^{(2)} (1-x) + x \Theta_{1,M_n-1}(y) \right] , \] (5.14)
in which
\[ \Theta_{a,b}(y) = \sum_{r=a}^{b} \pi_{0r}^{(1)} y^r . \] (5.15)
Now, multiply both sides of (5.4) by appropriate $x^i$ and $y^j$ and then sum over $i$ and $j$ to yield

$$
\Pi_2(x, y) = \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} \sum_{m=0}^{M_2-1} \pi^{(2)}_{i-m,j+1} p_m^{(2)} x^i y^j + \sum_{i=0}^{\infty} \sum_{j=M_2-1}^{M_1-2} \sum_{m=0}^{M_1-2} \pi^{(1)}_{i-j+1} d_m^{(1)} p_{i-m}^{(2)} x^i y^j + \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} \sum_{m=0}^{M_2-1} \sum_{n=0}^{M_2-1} \pi^{(1)}_{i-j} d_n^{(1)} p_m^{(2)} x^i y^j y^{M_n-1} (5.16)
$$

The first term of the right hand side of (5.16) can be summarized as

$$
\sum_{i=0}^{\infty} \sum_{j=0}^{\infty} \sum_{m=0}^{M_2-1} \pi^{(2)}_{i-m,j+1} p_m^{(2)} x^i y^j = \frac{1}{y} \left[ \Pi_2(x, y) - \Pi_2(x, 0) \right] P_2(x) \quad (5.17)
$$

and for the second term we have

$$
\sum_{i=0}^{\infty} \sum_{j=M_2-1}^{M_1-2} \sum_{m=0}^{M_1-2} \pi^{(1)}_{i-j+1} d_m^{(1)} p_{i-m}^{(2)} x^i y^j = \frac{1}{y} \left[ \Theta_{M_1, M_2-1}(y) D_1(x) P_2(x) \right]. \quad (5.18)
$$

Finally,

$$
\sum_{i=0}^{\infty} \sum_{j=0}^{\infty} \sum_{m=0}^{M_2-1} \pi^{(1)}_{i-j} d_n^{(1)} p_m^{(2)} x^i y^j y^{M_n-1} = \frac{1}{y} \left[ \Psi_{M_2}(x, y) + \pi^{(1)}_{0,0,0} y^{M_n} \right] D_1(x) P_2(x). \quad (5.19)
$$

Therefore, using (5.17) to (5.19) in (5.16), after some algebra, we obtain

$$
\Pi_2(x, y) = \frac{P_2(x)}{y - P_2(x)} \left\{ D_1(x) \left[ \Psi_{M_2}(x, y) + \Theta_{M_1, M_2-1}(y) \right] - \Pi_2(x, 0) \right\}. \quad (5.20)
$$

To establish the generating functions $\Pi_1(x, y)$ and $\Pi_2(x, y)$ it is now necessary to determine unknown functions $\Pi_1(0,y)$, $\Pi_2(x,0)$, $\Psi_{M_n}(x,y)$, $\Theta_{1,M_2-1}(y)$, $\Theta_{M_1, M_2}(y)$ and $\pi^{(2)}_{00}$.

**Determination of $\Psi_{M_n}(x,y)$**

To determine $\Psi_{M_n}(x,y)$ we first observe that we can write

$$
\Psi_{M_n}(x,y) = \frac{y^{M_n}}{M_n!} \left[ \frac{d^{M_n}}{dy^{M_n}} \Pi_1(x,y) \right]_{y=0} - \frac{d^{M_n}}{dy^{M_n}} \Pi_1(x,y) \bigg|_{x=y=0}. \quad (5.21)
$$

Using (5.14) we have,

$$
\frac{d^{M_n}}{dy^{M_n}} \Pi_1(x,y) = \left[ D_2(x) \Pi_2(x,0) - \pi^{(2)}_{00} \xi_0(1-x) \right] \frac{d^{M_n}}{dy^{M_n}} \left\{ \frac{y P_1(x)}{x-y P_1(x)} \right\} + \frac{d^{M_n}}{dy^{M_n}} \left\{ \left[ -\Pi_1(0,y) - \Psi_{M_n}(x,y) + x \Theta_{1,M_2-1}(y) \right] \frac{y P_1(x)}{x-y P_1(x)} \right\}. \quad (5.22)
$$
But it can be shown that

\[
\frac{d^k}{dy^k} \left[ \frac{yP_1(x)}{x - yP_1(x)} \right] = \frac{k! x [P_1(x)]^k}{[x - yP_1(x)]^{k+1}}.
\]  

(5.23)

whence,

\[
\frac{d^k}{dy^k} \left( \frac{yP_1(x)}{x - yP_1(x)} \right) \bigg|_{y=0} = k! \left[ \frac{P_1(x)}{x} \right]^k.
\]  

(5.24)

On the other hand

\[
\frac{d^{M_n}}{dy^{M_n}} \left[ \Pi_1(0, y) \frac{yP_1(x)}{x - yP_1(x)} \right] \bigg|_{y=0} = \sum_{k=0}^{M_n} C_k^{M_n} \frac{d^k}{dy^k} \left[ \Pi_1(0, y) \right] \bigg|_{y=0}
\]

\[
= \sum_{k=0}^{M_n} C_k^{M_n} \frac{d^k}{dy^k} \left[ \Pi_1(0, y) \right] \bigg|_{y=0}
\]

\[
= \sum_{k=0}^{M_n} C_k^{M_n} \frac{d^{M_n-k}}{dy^{M_n-k}} \left[ \frac{yP_1(x)}{x - yP_1(x)} \right] \bigg|_{y=0}
\]  

(5.25)

where \( C_k^n = \frac{n!}{k!(n-k)!} \). Therefore, since

\[
\frac{d^k}{dy^k} \Pi_1(0, y) \bigg|_{y=0} = k! \pi^{(1)}_{0k}.
\]  

(5.26)

using (5.24) and (5.26) in (5.25), we will have.

\[
\frac{d^{M_n}}{dy^{M_n}} \left[ \Pi_1(0, y) \frac{yP_1(x)}{x - yP_1(x)} \right] \bigg|_{y=0} = \sum_{k=1}^{M_n-1} C_k^{M_n} k! \pi^{(1)}_{0k} (M_n - k)! \left[ \frac{P_1(x)}{x} \right]_{M_n-k}
\]

\[
= M_n! \left[ \frac{P_1(x)}{x} \right]_{M_n} \Theta_{1, M_n-1} \left( \frac{x}{P_1(x)} \right)
\]  

(5.27)

Furthermore, since,

\[
\frac{d^k}{dy^k} x \Theta_{1, M_n-1}(y) \bigg|_{y=0} = \begin{cases} 
  x k! \pi^{(1)}_{0k} & ; k \leq M_n - 1 \\
  0 & ; k \geq M_n
\end{cases}
\]

then

\[
\frac{d^{M_n}}{dy^{M_n}} x \Theta_{M_n-1}(y) = \sum_{k=1}^{M_n-1} C_k^{M_n} k! \pi^{(1)}_{0k} (M_n - k)! \left[ \frac{P_1(x)}{x} \right]_{M_n-k}
\]

\[
= M_n! x \left[ \frac{P_1(x)}{x} \right]_{M_n} \Theta_{1, M_n-1} \left( \frac{x}{P_1(x)} \right)
\]  

(5.28)

Finally, considering the definition of \( \Psi_{M_n}(x, y) \) from (5.13), it is clear that

\[
\frac{d^{M_n}}{dy^{M_n}} \left[ \Psi_{M_n}(x, y) \frac{yP_1(x)}{x - yP_1(x)} \right] \bigg|_{y=0} = 0
\]  

(5.29)
Substituting (5.27) to (5.29) into (5.22), we now have

\[
\frac{dM_n}{dyM_n} \Pi_1(x, y) \bigg|_{y=0} = \left[ D_2(x) \Pi_2(x, 0) - \pi_0^{(2)} d_0^{(2)} (1 - x) \right] M_n \left[ \frac{P_1(x)}{x} \right] M_n
\]

\[
- M_n \left[ \frac{P_1(x)}{x} \right] M_n \Theta_{1, M_{n-1}} \left( \frac{x}{P_1(x)} \right)
\]

\[
+ M_n \left[ \frac{P_1(x)}{x} \right] M_n x \Theta_{1, M_{n-1}} \left( \frac{x}{P_1(x)} \right).
\]

(5.30)

On the other hand,

\[
\frac{dM_n}{dyM_n} \Pi_1(x, y) \bigg|_{x=y=0} = M_n \pi_0^{(1)} M_n.
\]

(5.31)

so that using (5.30) and (5.31) in (5.21) gives

\[
\Psi_{\text{M}_n}(x, y) = \left( \frac{y P_1(x)}{x} \right) M_n \left[ D_2(x) \Pi_2(x, 0) - \pi_0^{(2)} d_0^{(2)} (1 - x) \right]
\]

\[
- \Theta_{1, M_n} \left( \frac{x}{P_1(x)} \right) + x \Theta_{1, M_{n-1}} \left( \frac{x}{P_1(x)} \right).
\]

(5.32)

**Determination of \Pi_2(x, 0)**

From the denominator (5.20), it is clear that \( y - P_2(x) \) has one root \( y = P_2(x) \) inside the unit circle \(|y| < 1\). Therefore, substituting this root into the numerator of (5.20), we get

\[
D_1(x) \left[ \Psi_{\text{M}_n}(x, P_2(x)) + \Theta_{M_e, M_n}(P_2(x)) \right] - \Pi_2(x, 0) = 0.
\]

(5.33)

so that,

\[
\Pi_2(x, 0) = D_1(x) \left[ \Psi_{\text{M}_n}(x, P_2(x)) + \Theta_{M_e, M_n}(P_2(x)) \right].
\]

(5.34)

Now substituting (5.32) into (5.34),

\[
\Pi_2(x, 0) = D_1(x) \left\{ \left( \frac{P_1(x) P_2(x)}{x} \right) M_n \left[ D_2(x) \Pi_2(x, 0) - \pi_0^{(2)} d_0^{(2)} (1 - x) - \Theta_{1, M_n} \left( \frac{x}{P_1(x)} \right) \right]
\]

\[
+ x \Theta_{1, M_{n-1}} \left( \frac{x}{P_1(x)} \right) + \Theta_{M_e, M_n}(P_2(x)) \right\},
\]

and rearranging, yields

\[
\Pi_2(x, 0) = \frac{x M_n D_1(x)}{x M_n - D_1(x) D_2(x) \left[ \frac{P_1(x) P_2(x)}{x} \right] M_n} \left\{ \Theta_{M_e, M_{n-1}}(P_2(x))
\]

\[
- \left( \frac{P_1(x) P_2(x)}{x} \right) M_n \pi_0^{(2)} d_0^{(2)} (1 - x)
\]

\[
+ \Theta_{1, M_{n-1}} \left( \frac{x}{P_1(x)} \right) - x \Theta_{1, M_{n-1}} \left( \frac{x}{P_1(x)} \right) \right\}.
\]

(5.35)
Determinination of \( \Pi_1(0, y) \), \( \Theta_{1, M_n-1}(.) \), \( \Theta_{M_n, M_n}(.) \) and \( \pi_{00}^{(2)} \)

Considering (5.6) and (5.15) it is clear that

\[
\Pi_1(0, y) = \Theta_{1, M_n-1}(y) + \Theta_{M_n, M_n}(y) .
\]

and \( \Theta_{a,b}(.) \) \((1 \leq a \leq b \leq M_n)\) is a function of the unknown probabilities \( \pi_{0j}^{(1)} \), \( j = 1, 2, \ldots, M_n \). Using Takacs' lemma [62] we can show that if \([\rho_1 + \rho_2 + (\eta_1 + \eta_2)/M_n] \leq 1\), (where \( \rho_i = \lambda E[S_i] \) and \( \eta_i = \lambda E[D_i] \)), then the denominator in the right hand side of (5.35), \( x^{M_n} - D_1(x)D_2(x)[P_1(x)P_2(x)]^{M_n} = 0 \) has \( M_n \) roots \( \xi_j \), \( j = 1, 2, \ldots, M_n - 1 \) in the unit circle. while \( \xi_{M_n} = 1 \). Since \(|\Pi_2(x, 0)| \leq 1 \) in domain \(|x| \leq 1 \), it follows that the numerator in the right-hand side of (5.35) must be zero for \( \xi_j \) \((j = 1, 2, \ldots, M_n - 1)\).

Therefore for \( j = 1, 2, \ldots, M_n - 1 \).

\[
\Theta_{M_n, M_n-1}(P_2(\xi_j)) - \left( \frac{P_1(\xi_j)P_2(\xi_j)}{\xi_j} \right)^{M_n} \pi_{00}^{(2)}(1 - \xi_j) \\
\quad + \Theta_{1, M_n-1}(\frac{\xi_j}{P_1(\xi_j)}) - \xi_j \Theta_{1, M_n-1}(\frac{\xi_j}{P_1(\xi_j)}) = 0 .
\]

(5.37)

Since the system of \( M_n - 1 \) equations (5.37) has \( M_n \) unknowns \( \pi_{01}^{(1)} \), \( \pi_{02}^{(1)} \), \ldots, \( \pi_{0,M_n-1}^{(1)} \) and \( \pi_{00}^{(2)} \), it is necessary to use one more relation among those unknowns, namely \( \Pi_1(1, 1) + \Pi_2(1, 1) = 1 \).

Using (5.32), we obtain.

\[
\Psi_{M_n}(1, 1) = \Pi_2(1, 0) - \Theta_{M_n, M_n}(1)
\]

(5.38)

Using L'Hopital's rule in (5.14) yields

\[
\Pi_1(1, 1) = \Pi_1(0, 1) + \Pi_1^*(0, 1) + \Psi_{M_n}(1, 1) - \Theta_{1, M_n}(1) - \Theta_{1, M_n-1}(1)
\]

(5.39)

where

\[
\Pi_1^*(0, 1) = \sum_{r=1}^{M_n} r \pi_{0r}^{(1)}
\]

(5.40)

\[
\Psi_{M_n}^*(1, 1) = M_n \left[ \Pi_2(1, 0) - \Theta_{M_n, M_n}(1) \right]
\]

(5.41)

and

\[
\Theta_{1, M_n-1}(1) = \sum_{r=1}^{M_n-1} r \pi_{0r}^{(1)}
\]

(5.42)
Substituting (5.40) - (5.42) into (5.39) gives

\[ \Pi_1(1, 1) = M_0 \Pi_2(1, 0) - \sum_{r=M_e}^{M_n-1} (M_n - r) \pi_{0r}^{(1)} . \]  

(5.43)

To find \( \Pi_2(1, 0) \) we use L'Hopital's rule in (5.35). For the denominator term we have

\[ \frac{d}{dx} \left[ x^{M_n} - D_1(x)D_2(x) \left[ P_1(x)P_2(x) \right]^{M_n} \right] \bigg|_{x=0} = M_n(1 - \bar{\rho}) . \]  

(5.44)

where \( \bar{\rho} = \rho + \eta/M_n \) and \( \rho = \rho_1 + \rho_2 \). \( \eta = \eta_1 + \eta_2 \). On the other hand, using L'Hopital's rule for the terms in the numerator of (5.35), we get

\[ \frac{d}{dx} \left[ x^{M_n} D_1(x) \Theta_{M_e,M_n-1}(P_2(x)) \right] \bigg|_{x=0} = (M_n + \eta_1) \Theta_{M_e,M_n-1}(1) + \rho_2 \Theta'_{M_e,M_n-1}(1) \]  

(5.45)

\[ \frac{d}{dx} \left[ D_1(x) \left[ P_1(x)P_2(x) \right]^{M_n(1-x)\pi_{00}^{(2)} r_0^{(2)}} \right] \bigg|_{x=0} = -\pi_{00}^{(2)} r_0^{(2)} \]  

(5.46)

\[ \frac{d}{dx} \left[ D_1(x) \left[ P_1(x)P_2(x) \right]^{M_n \Theta_{1,M_n-1}(x/P_1(x))} \right] \bigg|_{x=0} = (\eta_1 + M_n \rho) \Theta_{1,M_n-1}(1) + (1 - \rho_1) \Theta'_{1,M_n-1}(1) \]  

(5.47)

\[ \frac{d}{dx} \left[ x D_1(x) \left[ P_1(x)P_2(x) \right]^{M_n \Theta_{1,M_n-1}(x/P_1(x))} \right] \bigg|_{x=0} = (1 + \eta_1 + M_n \rho) \Theta_{1,M_n-1}(1) + (1 - \rho_1) \Theta'_{1,M_n-1}(1) \]  

(5.48)

Therefore, after some algebra,

\[ \Pi_2(1, 0) = \frac{M_n(1 - \rho) \Theta_{M_e,M_n-1}(1) + \Theta_{1,M_n-1}(1) - (1 - \rho) \Theta'_{M_e,M_n-1}(1) + \pi_{00}^{(2)} d_0^{(2)}}{M_n(1 - \bar{\rho})} . \]  

(5.49)

Now, using (5.49) in (5.43) we get

\[ \Pi_1(1, 1) = \frac{M_n \Theta_{1,M_n-1}(1) + M_n \pi_{00}^{(2)} d_0^{(2)} + \eta \sum_{r=M_e}^{M_n-1}(M_n - r) \pi_{0r}^{(1)}}{M_n(1 - \bar{\rho})} . \]  

(5.50)

On the other hand, using L'Hopital's rule in (5.20) we have,

\[ \Pi_2(1, 1) = \Psi_{M_n}^{(1,1)} + \Theta'_{M_e,M_n}(1) = M_n \Pi_2(1, 0) - \sum_{r=M_e}^{M_n-1} (M_n - r) \pi_{0r}^{(1)} \]  

(5.51)
upon using (5.41). Therefore, considering (5.43) and (5.51) and the normalizing condition $\Pi_1(1,1) + \Pi_2(1,1) = 1$, we have.

$$\Pi_1(1,1) = \Pi_2(1,1) = \frac{1}{2}. \quad (5.52)$$

By using equation (5.50), which is equal to $\frac{1}{2}$, along with the system of linear equations (5.37), the unknown probabilities $\pi_{O,j}^{(1)}$, $j = 1,2,\ldots,M_n - 1$ and $\pi_{\infty}^{(2)}$ will be uniquely determined. Therefore, to obtain $\Theta_{a,b}(.)$ ($1 \leq a \leq b \leq M_n$) it only remains to find $\pi_{0,M_n}^{(1)}$.

**Determination of $\pi_{0,M_n}^{(1)}$**

Considering the equilibrium equations (5.4), we will have.

$$\pi_{O,j}^{(2)} = \begin{cases} 
\pi_{O,j+1}^{(2)}P_0 & : 0 \leq j \leq M_e - 2 \\
\pi_{O,j+1}^{(2)}P_0 + \pi_{O,j+1}^{(1)}d_0^{(1)}P_0 & : M_e - 1 \leq j \leq M_n - 2 \\
\pi_{0,M_n}^{(1)}d_0^{(1)}P_0 & : j = M_n - 1.
\end{cases} \quad (5.53)$$

Thus, by using (5.53) iteratively, after some algebra we get,

$$\pi_{O,j}^{(2)} = \begin{cases} 
\pi_{00}^{(2)}(\frac{1}{P_0})^{2} & : 1 \leq j \leq M_e - 1 \\
(\frac{1}{P_0})^{2}\int\pi_{00}^{(2)} - d_0^{(1)}\Theta_{M_e,j}(P_0^{(2)}) & : M_e \leq j \leq M_n - 1
\end{cases} \quad (5.54)$$

and, using (5.53) for $j = M_n - 1$,

$$\pi_{0,M_n}^{(1)} = \frac{1}{[P_0^{(2)}]^{M_n}}[\pi_{00}^{(2)} - \Theta_{M_e,M_n-1}(P_0^{(2)})]. \quad (5.55)$$

**5.4.3 Waiting times**

To obtain the waiting time distribution, define the waiting time at stage 1 of an arbitrary customer, $W_1$, as measured from his arrival instant to the first stage until his service completion at that stage and let $W_1(t)$ be the corresponding waiting time distribution. When the service of a customer is completed at stage 1, the number of customers in that stage at his departure epoch is equal to the number of customers
that arrived during his waiting time at stage 1. If $\hat{A}_1(x)$ is defined as the probability generating function of number of arrivals during the waiting time of an arbitrary customer in stage 1, then

$$\hat{A}_1(x) = \sum_{n=0}^{\infty} x^n \int_{0}^{\infty} \frac{e^{-\lambda t} (\lambda t)^n}{n!} dW_1(t)$$

$$= \int_{0}^{\infty} e^{-\lambda(1-x)t} dW_1(t)$$

$$= W_1^*(\lambda - \lambda x) \quad (5.56)$$

where $W_1^*(.)$ is the Laplace-Stieltjes Transform of $W_1(.)$.

On the other hand,

$$\hat{A}_1(x) = \sum_{n=0}^{\infty} \left[ \frac{\sum_{j=1}^{\infty} \pi_{ij} (1)}{\sum_{i=0}^{\infty} \sum_{j=1}^{\infty} \pi_{ij}} \right] x^n$$

$$= \frac{\Pi_1(x, 1)}{\Pi_1(1, 1)}$$

$$= 2\Pi_1(x, 1) \quad (5.57)$$

using (5.52).

If $s = \lambda - \lambda x$, then $x = 1 - s/\lambda$; then using (5.56) and (5.57), we will have.

$$W_1^*(s) = 2\Pi_1(1 - s/\lambda, 1) \quad (5.58)$$

Now, define the total waiting time at stages 1 and 2 of an arbitrary customer, $W_{1,2}$, as measured from his arrival instant to the first stage until his service completion at the second stage and let $W_{1,2}(t)$ be the corresponding waiting time distribution. Also, let $\hat{A}_{1,2}(x)$ as the probability generating function of number of arrivals during the waiting time of an arbitrary customer in the system (stages 1 and 2), then

$$\hat{A}_{1,2}(x) = W_{1,2}^*(\lambda - \lambda x) \quad (5.59)$$

and

$$\hat{A}_{1,2}(x) = \sum_{n=0}^{\infty} \left[ \frac{\sum_{i=0}^{\infty} \pi_{i,n-i}^{(2)}}{\sum_{i=0}^{\infty} \sum_{j=0}^{\infty} \pi_{ij}^{(2)}} \right] x^n$$

$$= \frac{\Pi_2(x, x)}{\Pi_2(1, 1)}$$

$$= 2\Pi_2(x, x) \quad (5.60)$$
so that
\[ W_{1,2}(s) = 2\Pi_2(1-s/\lambda, 1-s/\lambda). \] (5.61)

and the average waiting time in stage \( k(k = 1, 2) \), \( W_k \) is
\[
E[W_1] = \left( \frac{2}{\lambda} \right) \frac{d}{dx} [\Pi_1(x, 1)] \bigg|_{x=1} 
\] (5.62)
\[
E[W_2] = \left( \frac{2}{\lambda} \right) \frac{d}{dx} [\Pi_2(x, x)] \bigg|_{x=1} - E[W_1]. \] (5.63)

### 5.4.4 Average Number of Customers in the System

The average number of customers in stages 1 and 2, \( L_1 \) and \( L_2 \), are obtained in Appendix C. In this section only the results are presented for completeness of this chapter.

\[
L_1 = \frac{1}{1-\rho_1} \left\{ \Pi_2(1,0)[P_1''(1) + D_2''(1) + 2\rho_1\eta_2] - P_1''(1)[\Pi_1(0,1) + \Psi_{M_n}(1,1)] \\
+ \Theta_{1, M_n-1}(1)[P_1''(1) + 2\rho_1] + 2\Pi_2'(1,0)[\rho_1 + \eta_2] - 2\rho_1\Psi_{M_n}'(1,1) \\
+ 2\rho_1\pi_{00}^2 d_0^2 - \beta + \frac{P_1''(1)}{2} \right\}. \] (5.64)

where
\[
\Psi_{M_n}'(1,1) = (M_n\rho_1 + \eta_2)\Pi_2(1,0) - M_n\Psi_{M_n}(1,1) + \pi_{00}^{(2)} d_0^{(2)} - M_n\rho_1\Theta_{M_n, M_n}(1) \\
+ \Theta_{1, M_n-1}(1) - (1-\rho_1)\Theta_{M_n, M_n}(1) + \Pi_2'(1,0) \] (5.65)

\[
\beta = [M_nP_1''(1) + D_2''(1) + 2M_n\rho_1\eta_2 + M_n(M_n - 1)\rho_1^2]\Pi_2(1,0) \\
- M_n(M_n - 1)\Psi_{M_n}(1,1) + 2M_n\rho_1[\pi_{00}^{(2)} d_0^{(2)} + \Theta_{1, M_n-1}(1)] \\
- [M_nP_1''(1) + M_n(M_n - 1)\rho_1^2]\Theta_{M_n, M_n}(1) \\
+ [P_1''(1) - 2(M_n - 1)\rho_1(1-\rho_1)]\Theta_{M_n, M_n}(1) \\
+ 2(1-\rho_1)\Theta_{1, M_n-1}(1) - (1-\rho_1)^2\Theta_{M_n, M_n}(1) \\
+ 2(M_n\rho_1 + \eta_2)\Pi_2'(1,0) - 2M_n\Psi_{M_n}'(1,1) \] (5.66)
\[ \Pi_2'(1, 0) = \frac{1}{2M_n(1 - \rho)} \left\{ M_n[-P_1''(1) - P_2''(1) + 2\eta_1(1 - \rho) - 2M_n\rho_1\rho_2 \\
+ (M_n - 1)(1 - \rho_1^2 - \rho_2^2)]\Theta_{Me, Mn-1}(1) \right. \\
- [2(1 - \rho)(\eta_1 + M_n\rho_1) - P_1''(1) - P_2''(1) - 2\rho_1(1 - \rho_1)]\Theta_{Me, Mn-1}'(1) \\
- [(1 - \rho_1)^2 - \rho_2^2]\Theta_{Me, Mn-1}''(1) + 2[\eta_1 + M_n\rho]\Theta_{1, Mn-1}'(1) \\
+ 2[1 - \rho_1]\Theta_{1, Mn-1}'(1) + 2(\eta_1 + M_n\rho)\pi_0^{(2)}d_0^{(2)} - f''(1)\Pi_2(1.0) \} \] (5.67)

For the average number of customers in the second stage, \( L_2 \), we have

\[ L_2 = L_{1.2} - L_1. \]

where

\[ L_{1.2} = \frac{1}{1 - \rho_2} \left\{ [P_2''(1) + D_1''(1) + 2\rho_2\eta_1]\psi_{Me, Mn}(1, 1) + \Theta_{Me, Mn}(1) \right. \\
- P_2''(1)\Pi_2(1.0) - 2\rho_2\Pi_2'(1.0) \\
+ [\rho_2 + \eta_1]\psi_{Me, Mn}'(x = 1, x = 1) + \Theta_{Me, Mn}'(1) \\
+ \Theta_{Me, Mn}''(1) - \xi + \frac{P_2''(1)}{2} \} \] (5.68)

and

\[ \psi_{Me, Mn}(x, x) \bigg|_{x=1} = M_n\psi_{Me, Mn}(1, 1) + \psi_{Me, Mn}'(1, 1) \] (5.69)

\[ \xi = -[\beta + M_n(M_n - 1)\psi_{Me, Mn}(1, 1) + 2M_n\psi_{Me, Mn}'(1, 1)] . \] (5.70)

### 5.4.5 Average Number of Customers Served in a Cycle

A cycle is defined as the time elapsed from the server’s arrival at stage 1 to the next arrival at stage 1. Let \( N_{Me, Mn}^{cu} \) be the number of customers served during a cycle when a double-threshold policy with parameter \( M_e \) and \( M_n \) is applied. Then \( M_e \leq N_C \leq M_n \), and

\[ E[N_{Me, Mn}^{cu}] = \sum_{n=M_e}^{M_n} nP\{N_{Me, Mn}^{cu} = n\} . \] (5.71)
where
\[ P\{N_{(M_e,M_n)}^{cu} = n\} = P\{X_2 = n \mid \text{Server switches}\} . \]

and therefore,
\[
P\{N_{(M_e,M_n)}^{cu} = n\} = \begin{cases} \frac{\pi_0^{(1)}}{\sum_{j=M_e}^{M_n} \pi_j^{(1)} + \psi_{M_n}^{(1,1)}} \quad ; \quad n = M_e, M_e + 1, \ldots, M_n - 1 \\ \frac{\pi_0^{(1)} + \psi_{M_n}^{(1,1)}}{\sum_{j=M_e}^{M_n} \pi_j^{(1)} + \psi_{M_n}^{(1,1)}} \quad ; \quad n = M_n . \end{cases} \tag{5.72} \]

Substituting (5.72) into (5.71) yields,
\[
E[N_{(M_e,M_n)}^{cu}] = \frac{\sum_{n=M_e}^{M_n} n \pi_0^{(1)} + M_n \psi_{M_n}^{(1,1)}}{\Theta_{M_e,M_n}^{(1,1)} + \psi_{M_n}^{(1,1)}} = \frac{\Theta'_{M_e,M_n}^{(1)} + M_n \psi_{M_n}^{(1,1)}}{\Theta_{M_e,M_n}^{(1)} + \psi_{M_n}^{(1,1)}} . \tag{5.73} \]

and the average number of cycles per unit time, \(E[N_{(M_e,M_n)}^{cy}]\), is
\[
E[N_{(M_e,M_n)}^{cy}] = \frac{\lambda}{E[N_{(M_e,M_n)}^{cu}]} .
\]

## 5.5 Numerical Study

In this section the optimal limit of the limited policy (dynamic policy with \(M_e = 1\)) for systems with zero switchover times is examined in terms of changes in \(R_{12}\). These limits are then compared with optimal limits in systems with high traffic intensity. It will be shown that, similar to the gated-limited policy, the optimal limit curve \(L_1^L\) can be used to find the limited policy at almost the optimal cost for systems with lower traffic intensities. Finally, static, gated-limited, limited and double-threshold policies are compared using some numerical examples.

### 5.5.1 Optimal Limited Policy and High Traffic Intensity

Consider the \(M/HE_2 - D/1\) queue in Section 4.5.4 with zero switchover times. Tables 5.1 and 5.2 show the characteristics of this queue under a limited policy with limit \(M_n\), and the total average cost per unit time when \(\lambda = 2, 4 \text{ and } 6\).
Table 5.1. Characteristics of $M/H_E_2 - D/1$ under limited policy.

<table>
<thead>
<tr>
<th>$M_n$</th>
<th>$\rho = 0.3$</th>
<th>$\rho = 0.5$</th>
<th>$\rho = 0.9$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$L_1$</td>
<td>$L_2$</td>
<td>$E[N_C]$</td>
</tr>
<tr>
<td>1</td>
<td>0.3460</td>
<td>0.0667</td>
<td>1.0000</td>
</tr>
<tr>
<td>2</td>
<td>0.3342</td>
<td>0.1198</td>
<td>1.2153</td>
</tr>
<tr>
<td>3</td>
<td>0.3278</td>
<td>0.1488</td>
<td>1.2751</td>
</tr>
<tr>
<td>4</td>
<td>0.3241</td>
<td>0.1655</td>
<td>1.2957</td>
</tr>
<tr>
<td>5</td>
<td>0.3219</td>
<td>0.1755</td>
<td>1.3040</td>
</tr>
<tr>
<td>6</td>
<td>0.3205</td>
<td>0.1814</td>
<td>1.3076</td>
</tr>
</tbody>
</table>

Table 5.2. $E[T_{C(1,M_n)}]$ for $M/H_E_2 - D/1$ queue with $h_1 = 16$, $h_2 = 20$
and $K_{12} = 10$.

<table>
<thead>
<tr>
<th>$M_n$</th>
<th>$\lambda = 2$</th>
<th>$\lambda = 4$</th>
<th>$\lambda = 6$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>26.8698</td>
<td>62.7556</td>
<td>188.8000</td>
</tr>
<tr>
<td>2</td>
<td>24.2007</td>
<td>53.1766</td>
<td>168.2815</td>
</tr>
<tr>
<td>3</td>
<td>23.9049</td>
<td>52.0420</td>
<td>165.8974</td>
</tr>
<tr>
<td>4</td>
<td>23.9306</td>
<td>52.5072</td>
<td>167.7805</td>
</tr>
<tr>
<td>5</td>
<td>23.9964</td>
<td>53.3736</td>
<td>171.2075</td>
</tr>
<tr>
<td>6</td>
<td>24.0526</td>
<td>54.3012</td>
<td>175.2892</td>
</tr>
</tbody>
</table>

As Table 5.2 shows, the total average cost per unit time $E[T_{C(1,M_n)}]$ is flatter at the bottom for systems with lower $\rho$, and similar to the gated-limited policy, for our example, variation in $\lambda$ does not influence the optimal limit $M_n^*$. Now, let us consider the influence of $R_{12}$ on the optimal limit $M_n^*$. Suppose curve $L_p^*$ shows the variations of the optimal limit of the limited policy, $M_n^*$, relative to changes in $R_{12}$. Figure 5.2 depicts optimal limit curves $L_{0.3}^*$, $L_{0.6}^*$, $L_{0.9}^*$ and $L_1^*$ for fixed $h_1 = 16$, $h_2 = 20$ and different value of $K_{12} + K_{21}$, and therefore, different $R_{12}$. It should be noted that for any combination of costs Figure 5.2 remains the same. Similar to the gated-limited case, let $\delta$ be the cost relative error of the optimal limit obtained under high traffic intensity when it is used in a system with a lower traffic intensity. Therefore, considering point $a$ in Figure 5.2 for system with $\rho = 0.3$ we have

$$
\delta_a = \frac{E[T_{C(1.2)}] - E[T_{C(1.3)}]}{E[T_{C(1.3)}]} \bigg|_{R_{12}=2.56} = \frac{17.897698 - 17.897575}{17.897575} = 7 \times 10^{-6}.
$$

Tables 5.3, 5.4 and 5.5 show the total average cost per unit time and cost relative error $\delta$ for points $a$ to $f$ in Figure 5.2.
Figure 5.2. Optimal limit curves $L_1^L$ and $L_p^L$ for limited policy ($M_e = 1$).
Table 5.3. Comparison of Relative Error $\delta$ for points a and b in Figure 5.2.

<table>
<thead>
<tr>
<th>$E[T_{C_{1,M_n}}]$</th>
<th>$\rho = 0.3$</th>
<th>$\rho = 0.6$</th>
<th>$\rho = 0.9$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>a</td>
<td>b</td>
<td>a</td>
</tr>
<tr>
<td>$M_n = 2$</td>
<td>17.897698</td>
<td>19.921891</td>
<td>44.554657</td>
</tr>
<tr>
<td>$M_n = 3$</td>
<td>17.897575</td>
<td>19.826828</td>
<td>44.554056</td>
</tr>
<tr>
<td>$\delta$</td>
<td>$5 \times 10^{-5}$</td>
<td>$3 \times 10^{-3}$</td>
<td>$1 \times 10^{-5}$</td>
</tr>
</tbody>
</table>

Table 5.4. Comparison of Relative Error $\delta$ for points c and d in Figure 5.2.

<table>
<thead>
<tr>
<th>$E[T_{C_{1,M_n}}]$</th>
<th>$\rho = 0.3$</th>
<th>$\rho = 0.6$</th>
<th>$\rho = 0.9$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>c</td>
<td>d</td>
<td>c</td>
</tr>
<tr>
<td>$M_n = 3$</td>
<td>25.536162</td>
<td>31.438716</td>
<td>59.292975</td>
</tr>
<tr>
<td>$M_n = 4$</td>
<td>25.536002</td>
<td>31.340023</td>
<td>59.292912</td>
</tr>
<tr>
<td>$\delta$</td>
<td>$6 \times 10^{-6}$</td>
<td>$3 \times 10^{-3}$</td>
<td>$1 \times 10^{-5}$</td>
</tr>
</tbody>
</table>

Table 5.5. Comparison of Relative Error $\delta$ for points e and f in Figure 5.2.

<table>
<thead>
<tr>
<th>$E[T_{C_{1,M_n}}]$</th>
<th>$\rho = 0.3$</th>
<th>$\rho = 0.6$</th>
<th>$\rho = 0.9$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>c</td>
<td>f</td>
<td>e</td>
</tr>
<tr>
<td>$M_n = 4$</td>
<td>34.257469</td>
<td>46.560140</td>
<td>77.498876</td>
</tr>
<tr>
<td>$M_n = 5$</td>
<td>34.257376</td>
<td>46.476138</td>
<td>77.498712</td>
</tr>
<tr>
<td>$\delta$</td>
<td>$3 \times 10^{-3}$</td>
<td>$2 \times 10^{-3}$</td>
<td>$2 \times 10^{-5}$</td>
</tr>
</tbody>
</table>

Similar results which were obtained in Section 4.5.4 for gated-limited policies are also applicable here. The only difference is the gap between $L_p^G$ and $L_1^G$, which is larger than the gap between $L_p^G$ and $L_1^G$. However, it does not influence the cost relative error significantly. If $L_1^G$ is used to find the optimal limit for the limited policies in systems with $\rho < 1$, then similar to the gated-limited policies the maximum cost relative error is less than 1%, and the average cost relative error, given that there is an error, is less than 0.01%. Consequently, referring to Remark 4.4, we find that similar to a gated-limited policy, the optimal limit of a limited policy can be considered to be independent of the arrival process.

5.5.2 Comparison of Optimal Policies in $M/G_1 - G_2/1$

To find the optimal static (S), gated-limited (G), double-threshold (D) and limited (L) policies which minimize the long-run average holding and switching costs, $E[T_{C_{\gamma}}]$, $\gamma \in \{S, G, D, L\}$, we must minimize

$$
\text{Min } E[T_{C_{\gamma}}] = E[N_{\gamma}^{T}] (K_{12} + K_{21}) + h_1 L_1^{(\gamma)} + h_2 L_2^{(\gamma)}
$$

subject to the stationary condition for policy $\gamma$. 
To investigate the relationships among the parameters of these policies, several problems were analyzed and the parameters of the optimal policy were determined. Some of these problems are introduced in Table 5.7 and the optimal policy parameters for these problems are shown in Table 5.8. The arrival rate is \( \lambda \) per hour and notation \( Er(t, b) \) is used to show an Erlang random variable with average \( t \) minutes and parameter \( b \). Also, \( Hyp(l, m, n) \) is used to show a hyperexponential random variable with \( p = l, \mu_1 = m, \mu_2 = n \) (per hour). Finally, \( Exp(t) \) and \( Det(t) \) indicate exponential and deterministic random variables with average \( t \) minutes, respectively. Also, \( M^*_S, M^*_G \) and \( M^*_L \) are the optimal limits of static, semi-dynamic (gated-limited) and limited policies, respectively.

In Tables 5.9 and 5.10, we compare the performance of our policies with the global optimal policy for systems with zero switching costs as given for a set of problems introduced in Duenyas et al.\(^1\) This set of problems covers a wide range of utilizations, service/switchover time ratios. Policy \( \gamma \), \( \gamma \in \{S, G, D, L\} \) is compared with the global optimal policy, \( \pi^* \), in terms of suboptimality cost ratio \( \delta_\gamma \), as follows:

\[
\delta_\gamma = \frac{E[TC_{(\gamma)}] - E[TC_{(\pi^*)}]}{E[TC_{(\pi^*)}]} \times 100
\]

Based on our observations we found that:

- For systems with zero switchover times or non-zero switchover times

\[
M^*_G \leq M^*_S \leq M^*_n = M^*_L
\]  \hfill(5.74)

- If \( E[TC_{(\gamma)}] \) is the optimal long-run average holding and switching cost of implementing policy \( \gamma \), then

\[
E[TC_{(D)}] \leq E[TC_{(L)}] \leq E[TC_{(G)}].
\]  \hfill(5.75)

- When \( \bar{p} \to 1 \), all policies behave the same; therefore,

\[
E[TC_{(D)}] \approx E[TC_{(L)}] \approx E[TC_{(S)}] \approx E[TC_{(G)}].
\]  \hfill(5.76)

\(^1\)To our knowledge, this is the only set of problems for which the cost of the optimal has been obtained.
If switchover times $D_{12}$ and $D_{21}$ are exchanged, then $L_1^{(\gamma)}$ and $E[\gamma(t)]$ remain the same while $L_2^{(\gamma)}$ increases (or decreases) by the same amount $\eta_2 - \eta_1$. Consequently, for all $\gamma$, the optimal limits $M_j^*$, $j \in \{S, L, G, E, n\}$ do not change, but the optimal average cost decreases when $D_{12} < D_{21}$.

When switching costs are zero, idling policies such as the static policy perform poorly, because idling in stage 1 increases the average number of customers in stage 2, $L_2$, and in the system, $L_{1,2}$. Therefore, as Tables 5.9 and 5.10 show, even the optimal double-threshold policy appears in its nonidling form, $M_e = 1$ (limited policy).

The average and maximum suboptimality cost ratio for static, gated-limited and double-threshold policies in Tables 5.9 and 5.10 are as follows:

<table>
<thead>
<tr>
<th>Policy</th>
<th>Average suboptimality cost ratio (%)</th>
<th>Maximum suboptimality cost ratio (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Static</td>
<td>33.5</td>
<td>44.5</td>
</tr>
<tr>
<td>Gated-Limited</td>
<td>6.2</td>
<td>12.6</td>
</tr>
<tr>
<td>Dynamic</td>
<td>0.8</td>
<td>2.3</td>
</tr>
</tbody>
</table>

As Table 5.6 depicts, our double-threshold policy with average suboptimality cost ratio less than 1% is almost as good as the optimal policy.

Idling policies such as static policy are not recommended for systems with zero switching costs; however, these policies perform better when switching costs are involved. This is so, because in systems with switching costs, it is sometimes optimal to wait for a customer in empty stage 1 in order to decrease the average switching cost.
Table 5.7- Parameters of problems 1 to 14

<table>
<thead>
<tr>
<th>Problem</th>
<th>$\lambda$</th>
<th>$S_1$</th>
<th>$S_2$</th>
<th>$D_1$</th>
<th>$D_2$</th>
<th>$h_1$</th>
<th>$h_2$</th>
<th>$K$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4</td>
<td>$Exp(0.6, 12.6)$</td>
<td>$Exp(2)$</td>
<td>$Exp(2)$</td>
<td>$Det(2)$</td>
<td>18</td>
<td>58</td>
<td>35</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>$Exp(0.6, 12.6)$</td>
<td>$Exp(2)$</td>
<td>$Exp(2)$</td>
<td>$Det(2)$</td>
<td>18</td>
<td>58</td>
<td>35</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>$Exp(0.6, 12.6)$</td>
<td>$Exp(2)$</td>
<td>0</td>
<td>0</td>
<td>18</td>
<td>58</td>
<td>35</td>
</tr>
<tr>
<td>4</td>
<td>6</td>
<td>$Exp(0.6, 12.6)$</td>
<td>$Exp(2)$</td>
<td>0</td>
<td>0</td>
<td>18</td>
<td>58</td>
<td>35</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>$Exp(0.6, 12.6)$</td>
<td>$Exp(2)$</td>
<td>$Exp(2)$</td>
<td>$Det(2)$</td>
<td>18</td>
<td>58</td>
<td>41</td>
</tr>
<tr>
<td>6</td>
<td>5</td>
<td>$Exp(0.6, 12.6)$</td>
<td>$Exp(2)$</td>
<td>$Exp(2)$</td>
<td>$Det(2)$</td>
<td>18</td>
<td>58</td>
<td>42</td>
</tr>
<tr>
<td>7</td>
<td>5</td>
<td>$Exp(0.6, 12.6)$</td>
<td>$Exp(2)$</td>
<td>$Exp(2)$</td>
<td>$Det(2)$</td>
<td>18</td>
<td>50</td>
<td>35</td>
</tr>
<tr>
<td>8</td>
<td>5</td>
<td>$Exp(0.6, 12.6)$</td>
<td>$Exp(2)$</td>
<td>$Exp(2)$</td>
<td>$Det(2)$</td>
<td>17</td>
<td>58</td>
<td>35</td>
</tr>
<tr>
<td>9</td>
<td>5</td>
<td>$Det(3)$</td>
<td>$Det(5)$</td>
<td>$Det(2)$</td>
<td>$Det(4)$</td>
<td>18</td>
<td>58</td>
<td>35</td>
</tr>
<tr>
<td>10</td>
<td>5</td>
<td>$Det(3)$</td>
<td>$Det(5)$</td>
<td>$Det(4)$</td>
<td>$Det(2)$</td>
<td>18</td>
<td>58</td>
<td>35</td>
</tr>
<tr>
<td>11</td>
<td>5</td>
<td>$Det(5)$</td>
<td>$Det(3)$</td>
<td>$Det(4)$</td>
<td>$Det(2)$</td>
<td>18</td>
<td>58</td>
<td>35</td>
</tr>
<tr>
<td>12</td>
<td>5</td>
<td>$Det(5)$</td>
<td>$Det(3)$</td>
<td>$Det(2)$</td>
<td>$Det(4)$</td>
<td>18</td>
<td>58</td>
<td>35</td>
</tr>
<tr>
<td>13</td>
<td>2</td>
<td>$Exp(10)$</td>
<td>$Exp(12)$</td>
<td>$Det(2)$</td>
<td>$Det(14)$</td>
<td>8</td>
<td>45</td>
<td>15</td>
</tr>
<tr>
<td>14</td>
<td>2</td>
<td>$Exp(10)$</td>
<td>$Exp(12)$</td>
<td>$Det(2)$</td>
<td>$Det(5)$</td>
<td>8</td>
<td>45</td>
<td>15</td>
</tr>
</tbody>
</table>

Table 5.8- Comparison of optimal policies for problems 1 to 14

<table>
<thead>
<tr>
<th>Problem</th>
<th>Static Policy</th>
<th>Semi-dynamic Policy</th>
<th>Dynamic Policy</th>
<th>Limited Policy</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$M_s^*$</td>
<td>Cost</td>
<td>$M_G^*$</td>
<td>Cost</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>148.8</td>
<td>3</td>
<td>154.7</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>219.8</td>
<td>4</td>
<td>217.1</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>75.2</td>
<td>3</td>
<td>76.8</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>285.0</td>
<td>3</td>
<td>290.6</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td>227.3</td>
<td>4</td>
<td>230.4</td>
</tr>
<tr>
<td>6</td>
<td>4</td>
<td>228.5</td>
<td>5</td>
<td>231.9</td>
</tr>
<tr>
<td>7</td>
<td>4</td>
<td>205.7</td>
<td>5</td>
<td>207.0</td>
</tr>
<tr>
<td>8</td>
<td>3</td>
<td>214.9</td>
<td>4</td>
<td>212.8</td>
</tr>
<tr>
<td>9</td>
<td>3</td>
<td>190.9</td>
<td>5</td>
<td>187.1</td>
</tr>
<tr>
<td>10</td>
<td>3</td>
<td>200.5</td>
<td>5</td>
<td>196.7</td>
</tr>
<tr>
<td>11</td>
<td>3</td>
<td>190.8</td>
<td>4</td>
<td>193.1</td>
</tr>
<tr>
<td>12</td>
<td>3</td>
<td>181.2</td>
<td>4</td>
<td>183.4</td>
</tr>
<tr>
<td>13</td>
<td>3</td>
<td>123.4</td>
<td>4</td>
<td>104.2</td>
</tr>
<tr>
<td>14</td>
<td>2</td>
<td>86.6</td>
<td>3</td>
<td>80.5</td>
</tr>
</tbody>
</table>
Table 5.9. Comparison of policies with optimal policies in $M/M_1 - M_2/1$ with exponential switchover times, zero switching costs, $\lambda = 0.4$, $h_1 = 10$ and $h_2 = 20$.

<table>
<thead>
<tr>
<th>$\bar{S}_1$</th>
<th>$\bar{S}_1$</th>
<th>$\bar{D}_{12}$</th>
<th>$\bar{D}_{21}$</th>
<th>Static</th>
<th>Gated-Limited</th>
<th>Dynamic</th>
<th>Opt. cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>7</td>
<td>152.04</td>
<td>70</td>
<td>(1.11)</td>
</tr>
<tr>
<td>1</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td>66.58</td>
<td>7</td>
<td>(1.6)</td>
</tr>
<tr>
<td>0.5</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td>71.03</td>
<td>8</td>
<td>(1.7)</td>
</tr>
<tr>
<td>0.5</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>41.65</td>
<td>6</td>
<td>(1.5)</td>
</tr>
</tbody>
</table>

| 1           | 1           | 1              | 2              | 10     | 191.61        | 9.5      | (1.16)   | 142.07   | 0.8      | 140.96   |
| 1           | 0.5         | 1              | 2              | 5      | 84.80         | 1.8      | (1.8)    | 63.46    | 0.7      | 62.97    |
| 0.5         | 1           | 1              | 2              | 5      | 90.15         | 6.5      | (1.0)    | 64.08    | 0.2      | 63.98    |
| 0.5         | 0.5         | 1              | 2              | 4      | 53.51         | 8.3      | (1.7)    | 36.69    | 0.1      | 36.65    |

| 1           | 1           | 2              | 1              | 10     | 199.61        | 9.0      | (1.16)   | 150.07   | 0.7      | 148.96   |
| 1           | 0.5         | 2              | 1              | 5      | 92.80         | 10.2     | (1.5)    | 71.46    | 0.6      | 70.97    |
| 0.5         | 1           | 2              | 1              | 5      | 98.15         | 5.8      | (1.10)   | 72.08    | 0.1      | 71.98    |
| 0.5         | 0.5         | 2              | 1              | 4      | 61.51         | 6.8      | (1.7)    | 44.96    | 0.1      | 44.65    |

| 1           | 1           | 2              | 2              | 13     | 236.49        | 11.3     | (1.20)   | 175.65   | 0.8      | 174.33   |
| 1           | 0.5         | 2              | 2              | 7      | 109.88        | 12.6     | (1.10)   | 83.43    | 1.0      | 82.48    |
| 0.5         | 1           | 2              | 2              | 7      | 117.24        | 6.7      | (1.13)   | 83.99    | 0.0      | 83.99    |
| 0.5         | 0.5         | 2              | 2              | 4      | 71.76         | 7.8      | (1.9)    | 51.52    | 0.1      | 51.77    |

Table 5.10. Comparison of policies with optimal policies in $M/M_1 - M_2/1$ with exponential switchover times, zero switching costs, $\lambda = 0.4$, $h_1 = 10$ and $h_2 = 40$.

<table>
<thead>
<tr>
<th>$\bar{S}_1$</th>
<th>$\bar{S}_1$</th>
<th>$\bar{D}_{12}$</th>
<th>$\bar{D}_{21}$</th>
<th>Static</th>
<th>Gated-Limited</th>
<th>Dynamic</th>
<th>Opt. cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>7</td>
<td>223.38</td>
<td>26.9</td>
<td>(1, 8)</td>
</tr>
<tr>
<td>1</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td>104.36</td>
<td>34.7</td>
<td>(1, 4)</td>
</tr>
<tr>
<td>0.5</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>107.79</td>
<td>30.2</td>
<td>(1, 5)</td>
</tr>
<tr>
<td>0.5</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>62.32</td>
<td>32.4</td>
<td>(1, 4)</td>
</tr>
</tbody>
</table>

| 1           | 1           | 1              | 2              | 9      | 285.65        | 27.2    | (1, 12)   | 228.69   | 1.8      | 224.55   |
| 1           | 0.5         | 1              | 2              | 5      | 129.35        | 31.1    | (1, 6)    | 104.20   | 1.5      | 98.69    |
| 0.5         | 1           | 1              | 2              | 5      | 139.37        | 34.4    | (1, 7)    | 104.56   | 0.8      | 103.74   |
| 0.5         | 0.5         | 1              | 2              | 3      | 79.69         | 35.8    | (1, 5)    | 61.28    | 4.4      | 58.68    |

| 1           | 1           | 2              | 1              | 9      | 301.65        | 25.4    | (1, 12)   | 244.69   | 1.7      | 240.55   |
| 1           | 0.5         | 2              | 1              | 5      | 145.35        | 26.7    | (1, 6)    | 120.93   | 5.4      | 116.20   |
| 0.5         | 1           | 2              | 1              | 5      | 155.37        | 29.8    | (1, 7)    | 123.28   | 3.0      | 119.74   |
| 0.5         | 0.5         | 2              | 1              | 3      | 95.69         | 28.1    | (1, 5)    | 77.28    | 3.5      | 74.96    |

| 1           | 1           | 2              | 2              | 12     | 362.21        | 26.8    | (1, 15)   | 289.97   | 1.5      | 285.57   |
| 1           | 0.5         | 2              | 2              | 6      | 170.18        | 6.3     | (1, 8)    | 142.80   | 0.3      | 134.33   |
| 0.5         | 1           | 2              | 2              | 6      | 181.14        | 3.4     | (1, 9)    | 140.06   | 0.5      | 139.24   |
| 0.5         | 0.5         | 2              | 2              | 4      | 112.07        | 4.2     | (1, 6)    | 88.84    | 0.4      | 85.41    |
Chapter 6
Optimal Policies in $M/G_1 - G_2 - \cdots - G_N/1$ Queues

6.1 Introduction

The BTQ model which is a two-stage tandem queue attended by a moving server is extended to an N-stage tandem queue, $M/G_1 - G_2 - \cdots - G_N/1$, in this chapter. First, a greedy and exhaustive policy downstream of stage 2 is defined for an N-stage tandem queue, and then, considering that the server applies a greedy and exhaustive policy downstream of stage 2, an equivalent two-stage tandem queue is introduced which can be used to analyze the $M/G_1 - G_2 - \cdots - G_N/1$ queue with nonzero switchover times. Based on equivalent two-stage tandem queues, three models are developed to find the optimal static, gated-limited and double-threshold policies in stage 1. Finally, some numerical results are provided and used to compare the parameters of the optimal policies.

6.2 Problem Description

Consider an N-stage tandem queue in which only one server is assigned responsibility to serve customers in all stages (Figure 6.1). Also, suppose
• Customers arrive according to a homogeneous Poisson process with rate $\lambda$ to stage 1.

• The buffers or waiting spaces of stages $i$ ($i = 1, 2, \ldots, N$) have infinite capacities.

• Service times $S_i$ in stage $i$ ($i = 1, 2, \ldots, N$) are independent random variables with distribution functions $F_i(.)$ and Laplace-Stieltjes transform $F^*_i(.)$.

• The switchover times from stage $i$ to $j$ ($i \neq j; i, j = 1, 2, \ldots, N$) are also independent random variables $D_{ij}$ with distribution functions $B_{ij}(.)$ and Laplace-Stieltjes transform $B^*_{ij}(.)$.

• There is a fixed switching cost $K_{ij}$ which is charged at the start of switchover time $D_{ij}$.

• The holding cost rate for each customer present in stage $i$ ($i = 1, 2, \ldots, N$) per unit time is $h_i$, which is non-decreasing in $i$.

If we only include the set of admissible policies which imply that once a customer's service has started, it cannot be interrupted by switching or by serving other customers, then this problem can be viewed as a semi-Markov decision process in which

• decision epochs are service completion times, switch completion times, and arrival instants whenever the server is idle at one of the stages,

• the state of the system at any decision epoch is defined in terms of the location of the server ($y$) and the number of customers, $x_i$, in stage $i$ ($i = 1, 2, \ldots, N$).
Therefore, the state space of the system is

\[ SS = \{(y, x_1, x_2, \ldots, x_n) \mid y \in I_n, x_i \in \mathbb{Z}^+ \} \]

where \( I_n = \{1, 2, \ldots, N\} \) and \( \mathbb{Z}^+ \) denotes the set of nonnegative integers.

- at each decision epoch, the server decides whether to serve the next customer in the present stage, to switch to another stage, or to be idle.

Let \( X^{\pi,z_0}_i(t) \) be the number of customers in queue \( i \) at time \( t \) when admissible policy \( \pi \) is applied and the initial state is \( z_0 \). and also let \( Q^{\pi,z_0}_{ij}(t) \) be the number of corresponding switches from stage \( i \) to \( j \) up to time \( t \). Then, the total long-run average holding and switching cost when policy \( \pi \) is applied with initial state \( z_0 \) is

\[
\bar{V}_\pi(z_0) = \lim_{t \to \infty} \frac{1}{t} \mathbb{E}\left[ \sum_{i=1}^{N} \int_0^t h_i X^{\pi,z_0}_i(u) du + \sum_{j \neq i} K_{ij} Q^{\pi,z_0}_{ij}(t) \right]. \tag{6.1}
\]

Since the action that the server chooses at each decision epoch depends on the number of customers at other stages as well as his present stage, the structure of the optimal policy is rather complex and progressively worsens as the number of stages and customers in the system increases. Thus, we only concentrate on a class of policies which we call greedy and exhaustive downstream of stage 2.

**Definition 6.1:** *Greedy and exhaustive policy downstream of stage* \( i \) *is a policy in which the server applies greedy and exhaustive policies in stages* \( i, i + 1, \ldots, N \) *and switches from stage* \( j \) *to* \( j + 1 \) \((j = i, i + 1, \ldots, N - 1)\); *and when stages* \( i \) *to* \( N \) *all become empty he switches to one of the earlier stages* \( 1, 2, \ldots, i - 1 \).

For the first stage, similar to the BTQ model, we consider three policies as follows:

- **Static policy:** The server serves a fixed number of customers, \( M_S \), upon his arrival at stage 1, and then he switches to stage 2.

- **Semi-dynamic policy** (or gated-limited policy): The server serves \( \min\{n, M_G\} \) customers in stage 1, where \( n \) is the number of customers in stage 1 when he arrives there, and \( M_G \) is a predetermined number. Then the server switches to stage 2.
- **Dynamic policy** (or double-threshold policy): The server continues serving in stage 1 until \( M_n \) services have been completed without interruption, or until the first stage becomes empty and there are at least \( M_e \) customers in stage 2 (\( M_e \leq M_n \)). whichever comes first. Then the server switches to stage 2.

### 6.3 Equivalent Two-Stage Tandem Queue (ETQ)

When a greedy and exhaustive policy downstream of stage 2 is applied in an \( N \)-stage tandem queue attended by moving server, an equivalent two-stage tandem queue, or ETQ, can be introduced in which

\[
\begin{align*}
S_1^Q &= S_1^N \\
S_2^Q &= S_2^N + S_3^N + \cdots + S_N^N \\
D_{12}^Q &= D_{12}^N + D_{23}^N + \cdots + D_{N-1,N}^N \\
D_{21}^Q &= D_{N1}^N
\end{align*}
\]

where \( S_i^Q \) and \( D_i^Q \) are the service time in stage \( i \) and the switchover time from stage \( i \) to the other stage, respectively, in the ETQ model, and \( S_i^N \) and \( D_{ij}^N \) are the service time in stage \( i \) and the switchover time from stage \( i \) to \( j \) in the \( N \)-stage tandem queue.

If policy \( \gamma \) with parameter \( M_\gamma \) is applied in the first stage of the \( N \)-stage tandem queue, and a greedy and exhaustive policy downstream of stage 2 is considered, then

\[
L_1^{(N)}(M_\gamma) = L_1^{(Q)}(M_\gamma)
\]  

(6.2)

where \( L_i^{(N)}(M_\gamma) \) and \( L_i^{(Q)}(M_\gamma) \) are the average numbers of customers in stage \( i \) in the \( N \)-stage and ETQ model, respectively. Therefore, if a static, semi-dynamic or dynamic policy is applied in stage 1 in an \( N \)-stage tandem queue attended by a moving server, where a greedy and exhaustive policy downstream of stage 2 is implemented, then

\[
\begin{align*}
L_1^{(N)}(M_S) &= L_1^{(Q)}(M_S) \\
L_1^{(N)}(M_G) &= L_1^{(Q)}(M_G) \\
L_1^{(N)}(M_e, M_n) &= L_1^{(Q)}(M_e, M_n)
\end{align*}
\]
6.4 Optimal Static Policy

In this section we seek the optimal value of $M_S$ which minimizes the total long-run average holding and switching cost in an $N$-stage tandem queue with non-zero switchover times. where the server applies a static policy in stage 1 and a greedy and exhaustive policy downstream of stage 2. Let $E[T_{M_S}^{cy}]$ be the average cycle time measured from the server's arrival at stage 1 to his next arrival at stage 1, when a static policy with parameter $M_S$ in stage 1 and a greedy and exhaustive policy downstream of stage 2 are applied. Then, considering $E[TC_{M_S}]$ as the total long-run average holding and switching cost per unit time, the optimization problem is

$$
\text{Min } E[TC_{M_S}] = \sum_{i=1}^{N} h_i L_i^{(N)}(M_S) + \frac{1}{E[T_{M_S}^{cy}]} \sum_{i=1}^{N} K_{i,i+1}
$$

subject to

$$\rho + \frac{\eta}{M_S} \leq 1$$

$$M_S \in \{1, 2, 3, \ldots\}$$

(6.3)

where

$$\rho = \sum_{i=1}^{N} \rho_i = \lambda \sum_{i=1}^{N} E[S_i]$$

$$\eta = \sum_{i=1}^{N} \eta_i = \lambda \sum_{i=1}^{N} E[D_{i,i+1}]$$

and $D_{N,N+1} = D_{N1}$. $K_{N,N+1} = K_{N1}$. Therefore, to find the optimal limit $M_S^*$, the average number of customers in each stage. $L_i^{(N)}(M_S)$, and the mean cycle time $E[T_{M_S}^{cy}]$ must be found.

6.4.1 Average Holding Cost

The average holding cost per unit time in stage 1 is $h_1 L_1^{(N)}(M_S)$, where $L_1^{(N)}(M_S)$ can be found using the ETQ model. However, to find the average number of customers, and therefore, the average holding costs per unit time in stages 2, 3, $\ldots$, $N$, the average interdeparture times from stage 1 must be analyzed. This can be done by analogy
Figure 6.2. Number of customers in stage 2 during a cycle when static policy with parameter $M_S = 4$ is applied.

with Figure 6.2. Figure 6.2 depicts the number of customers in stage 2: $N_2(t)$, during a cycle which serves $M_S = 4$ customers (static policy with parameter $M_S = 4$).

Let $R^{(n)}_{i, i+1}$ denote the time between the $i$th and $(i+1)$th customer departures from stage $n$ during a cycle. Therefore,

$$L_2^{(N)}(M_S) = \frac{1}{E[T_{M_S}^{cy}]} \left\{ \sum_{i=1}^{M_S-1} i E[R^{(1)}_{i, i+1}] + M_S E[D_{12}] + \frac{M_S(M_S + 1)}{2} E[S_2] \right\} ,$$

in which

$$E[T_{M_S}^{cy}] = \frac{M_S}{\lambda} ,$$

and, assuming stationary conditions, it only remains to find the average interdeparture times $E[R^{(1)}_{i, i+1}]$, for $i = 1, 2, \ldots, M_S - 1$.

Suppose $\tau_k^{(M_S)}$ is the time measured from the start of a cycle to the time that the $k$th customer departs from stage 1, in a cycle which serves $M_S$ customers. Using the ETQ model, and implementing an imbedded Markov chain for systems with a static policy $^1$, we will have

$$E[R^{(1)}_{k,k+1}] = E[\tau^{(M_S)}_{k+1} - \tau^{(M_S)}_k | X_2 = k, I = 1] ,$$

$^1$The state of the imbedded Markov chain for a two-stage tandem queue with static policy is defined as $< X_1^{(n)}, X_2^{(n)}, I^{(n)} >$ where $I^{(n)}$ is the number of the stage at which the nth service completion occurs and $X_i^{(n)}$ is the number of customers in stage $i$ at the nth service completion. Assuming that steady state conditions prevail, $lim_{n \rightarrow \infty} (X_1^{(n)}, X_2^{(n)}, I^{(n)}) = (X_1, X_2, I)$ in distribution, and the stationary probabilities are defined by $\pi_{ij}^{(k)} = P\{X_1 = i, X_2 = j, I = k\}$. 

---

Figure 6.2: Number of customers in stage 2 during a cycle when static policy with parameter $M_S = 4$ is applied.
or.
\[
E[R_{k,k+1}^{(1)}] = \sum_{n=0}^{\infty} E[\tau_{k+1}^{(M_S)} - \tau_k^{(M_S)} | X_1 = n, X_2 = k, I = 1] P\{X_1 = n | X_2 = k, I = 1\}.
\]

On the other hand,
\[
P\{X_1 = n | I = 1, X_2 = k\} = \frac{P\{X_1 = n, X_2 = k, I = 1\}}{P\{X_2 = k, I = 1\}} = \frac{\pi_{nk}^{(1)}}{\sum_{n=0}^{\infty} \pi_{nk}^{(1)}}.
\]

so that
\[
E[R_{k,k+1}^{(1)}] = \frac{1}{\sum_{r=0}^{\infty} \pi_{rk}^{(1)}} \left\{ E[\tau_{k+1}^{(M_S)} - \tau_k^{(M_S)} | X_1 = 0, X_2 = k, I = 1] \pi_{0k}^{(1)} + \sum_{n=1}^{\infty} E[\tau_{k+1}^{(M_S)} - \tau_k^{(M_S)} | X_1 = n, X_2 = k, I = 1] \pi_{nk}^{(1)} \right\}
\]
\[
= \frac{1}{\sum_{r=0}^{\infty} \pi_{rk}^{(1)}} \left\{ \left( \frac{1}{\lambda} + E[S_1] \right) \pi_{0k}^{(1)} + E[S_1] \left( \sum_{n=0}^{\infty} \pi_{nk}^{(1)} - \pi_{0k}^{(1)} \right) \right\}
\]
\[
= \frac{\pi_{0k}^{(1)}}{\sum_{n=0}^{\infty} \pi_{nk}^{(1)}} \left( \frac{1}{\lambda} \right) + E[S_1].
\]

Using the results which were obtained in chapter 3 for the ETQ model with a static policy and considering (3.4), we will have
\[
\sum_{n=0}^{\infty} \pi_{nk}^{(1)} = \frac{1}{k!} \frac{d^k}{dy^k} \Pi_1(1,y) \bigg|_{y=0} : k = 1, 2, \ldots, M_S - 1
\]

where
\[
\Pi_1(1,y) = \frac{y}{1-y} \left[ \Pi_2(1,0) - \Pi_1(0,y) - \Psi_{M_S}(1,y) + \Theta_{M_S-1}(1) \right]
\]
\[
= \frac{y}{1-y} \left[ \Pi_2(1,0) - \pi_{0M_S}^{(1)} y^{M_S} - \Psi_{M_S}(1,y) \right].
\]

and
\[
\Psi_{M_S}(1,y) = y^{M_S} \left[ \Pi_2(1,0) - \Theta_{M_S}(1) + \Theta_{M_S-1}(1) \right]
\]
\[
= y^{M_S} \left[ \Pi_2(1,0) - \pi_{0M_S}^{(1)} \right].
\]

Therefore, using (6.11) in (6.10), we get
\[
\Pi_1(1,y) = \Pi_2(1,0) \left[ \frac{y(1-y^{M_S})}{1-y} \right].
\]
Assume
\[ \Pi_1^{(k)}(1, 0) = \left. \frac{d^k}{y^k} \Pi_1(1, y) \right|_{y=0} \]  
(6.13)
then, using (6.12), it can be shown that
\[ \Pi_1^{(k)}(1, 0) = k! \Pi_2(1, 0) \quad : \quad k = 1, 2, \ldots, M_S - 1 \]  
(6.14)
and using (6.14) in (6.9), we get
\[ \sum_{n=0}^{\infty} \pi_{n1}^{(1)} = \Pi_2(1, 0) \quad : \quad k = 1, 2, \ldots, M_S - 1. \]  
(6.15)

**Remark 6.1**

Equation (6.15) can also be obtained using an intuitive argument as follows: 
\( \Pi_2(1, 0) \) is the probability that the server switches to stage 1 after completing a service in stage 2. On the other hand, \( \sum_{n=0}^{\infty} \pi_{n1}^{(1)} \) is the probability that after a service completion in stage 1, there are \( k \) customers in stage 2. Since \( k = 1, 2, \ldots, M_S - 1 \), therefore it is clear that when the server switches back to stage 1 after a service completion in that stage, there will be exactly one customer in stage 2. Thus, \( \sum_{n=0}^{\infty} \pi_{n1}^{(1)} = \Pi_2(1, 0) \):
and after the next service completion in stage 1, the number of customers in stage 2 will be 2, so that \( \sum_{n=0}^{\infty} \pi_{n2}^{(1)} = \sum_{n=0}^{\infty} \pi_{n1}^{(1)} \). Using the same argument iteratively yields equation (6.15).

Substituting (6.15) into (6.8) we get,
\[ E[R_{k,k+1}^{(1)}] = \frac{\pi_{0k}^{(1)}}{\Pi_2(1, 0)} \left( \frac{1}{\lambda} + E[S_1] \right) \quad : \quad k = 1, 2, \ldots, M_S - 1. \]  
(6.16)
where \( \Pi_2(1, 0) \) and \( \pi_{0k}^{(1)} \) can be obtained using the ETQ model with static policy. Finally, using (6.16) and (6.5) in (6.4), the average number of customers in stage 2. \( L_2^{(N)}(M_S) \), is obtained.

To find the average number of customers in stages 3, 4, \ldots, \( N \), the same approach can be used, and, since a greedy and exhaustive policy downstream of stage 2 is applied. \( E[R_{i,i+1}^{(j)}] = E[S_{j-1}] \). Therefore,
\[ L_j^{(N)}(M_S) = \frac{1}{E[T_{M_S}]} \left( \frac{M_S(M_S - 1)}{2} E[S_{j-1}] + M_SE[D_{j-1,j}] + \frac{M_S(M_S + 1)}{2} E[S_j] \right) \]
\[ = \frac{\lambda}{2} \left( (M_S - 1)E[S_{j-1}] + (M_S + 1)E[S_j] + E[D_{j-1,j}] \right). \]  
(6.17)
and, consequently, the average holding cost per unit time, \( \sum_{i=1}^{N} h_i L_i^{(N)}(M_S) \), is determined.

### 6.4.2 Average Switching Cost

Since exactly \( M_S \) customers are served in each cycle, and assuming that under stationary conditions we have \( \rho + \frac{\eta}{M_S} < 1 \), then the average number of cycles per unit time, \( E[N_{M_S}^{cy}] \), is

\[
E[N_{M_S}^{cy}] = \frac{1}{E[T_{M_S}]} .
\]

Using (6.5), the average switching cost per unit time is \( E[N_{M_S}^{cy}] \sum_{i=1}^{N} K_{i,i+1} \).

### 6.5 Optimal Semi-Dynamic Policy

Applying a semi-dynamic (gated-limited) policy in stage 1, the server serves the minimum of (i) the number of customers waiting in stage 1 when he arrives there, \( n \), and (ii) a limit, \( M_G \). In this section we find the optimal value of \( M_G \) which minimizes the total long-run average holding and switching cost per unit time in an \( N \)-stage tandem queue, where the server applies a greedy and exhaustive policy downstream of stage 2. The optimization problem is similar to (6.3), and is expressed as

\[
\text{Min } E[T_{M_G}] = \sum_{i=1}^{N} h_i L_i^{(N)}(M_G) + E[N_{M_G}^{cy}] \sum_{i=1}^{N} K_{i,i+1}
\]

subject to:

\[
\rho + \frac{\eta}{M_G} \leq 1
\]

\[
M_G \in \{1, 2, 3, \ldots \}
\]

(6.18)

where \( E[N_{M_G}^{cy}] \) is the average number of cycles per unit time when gated-limited policy with parameter \( M_G \) is applied.
6.5.1 Average Holding Cost

Similar to a static policy, when a greedy and exhaustive policy downstream of stage 2 is applied, the average number of customers in stage 1 can be determined using the ETQ model with gated-limited policy. When a gated-limited policy in stage 1 and a greedy and exhaustive policy downstream of stage 2 are applied, \( m \) customers are served in each cycle, where \( m = 1, 2, \ldots, M_G \). Thus,

\[
L_i^{(N)}(M_G) = \frac{\sum_{m=1}^{M_G} \mathcal{L}_i^{(N)}(m) P\{N_{M_G} = m\}}{E[T_{M_G}^{\text{cycles}}]} \quad : \quad i = 2, 3, \ldots, N .
\]

(6.19)

where \( \mathcal{L}_i^{(N)}(m) \) is the total accumulated delay in stage \( i \) during a cycle which serves \( m \) customers. \( P\{N_{M_G} = m\} \) is the probability of having such a cycle, and \( E[T_{M_G}^{\text{cycles}}] \) is the average cycle time when gated-limited policy with parameter \( M_G \) is implemented. Therefore,

\[
E[T_{M_G}^{\text{cycles}}] = \sum_{m=1}^{M_G} E[T_{M_G}^{\text{cycles}} \mid N_{M_G} = m] P\{N_{M_G} = m\} .
\]

(6.20)

On the other hand, \( \mathcal{L}_i^{(N)}(m) \) is actually the area under a curve similar to Figure 6.2. Thus, for a cycle which serves \( m \) customers,

\[
\mathcal{L}_2^{(N)}(m) = \sum_{i=1}^{m-1} iE[R_{i,i+1}] + mE[D_{12}] + \frac{m(m + 1)}{2}E[S_2] .
\]

(6.21)

Since the server is never idle in stage 1 when there is at least one customer in the system, we will have

\[
E[R_{i,i+1}^{(1)}] = E[S_1] \quad : \quad i = 1, 2, \ldots, M_G - 1.
\]

(6.22)

and again considering the Figure 6.2 analogy,

\[
E[T_{M_G}^{\text{cycles}} \mid N_{M_G} = m] = \sum_{k=0}^{m-1} E[R_{k,k+1}^{(1)}] + \sum_{j=1}^{N} E[D_{j,j+1}] + m \sum_{i=2}^{N} E[S_i] .
\]

(6.23)

When a cycle serves more than one customer \( (m > 1) \), then

\[
E[R_{01}^{(1)}] = E[S_1].
\]

(6.24)

However, for cycles in which only one customer is served \( (m = 1) \), and considering the imbedded Markov chain for two-stage tandem queues with a gated-limited policy
as in Chapter four \(^2\), we have

\[
E[R_{01}^{(1)}] = E[\tau_{1}^{(1)}|I = 2, X_2 = 0, m = 1] = E[\tau_{1}^{(1)}|I = 2, X_2 = 0, X_1^+ = 0, m = 1]P\{X_1^+ = 0|I = 2, X_2 = 0, m = 1\}
+ E[\tau_{1}^{(1)}|I = 2, X_2 = 0, X_1^+ = 1, m = 1]P\{X_1^+ = 1|I = 2, X_2 = 0, m = 1\}.
\]

(6.25)

where \(X_1^+\) is the number of customers in stage 1 after switchover time \(D_{21}^0\) in the ETQ model. Hence

\[
E[R_{01}^{(1)}] = \frac{1}{\lambda} + E[S_1] \left( \frac{\pi_{000}^{(2)}d_0^{(2)}}{\pi_{000}^{(2)}d_0^{(2)} + \pi_{100}^{(2)}d_0^{(2)} + \pi_{000}^{(2)}d_1^{(2)}} \right) + E[S_1] \left( \frac{\pi_{000}^{(2)}d_0^{(2)}}{\pi_{000}^{(2)}d_0^{(2)} + \pi_{100}^{(2)}d_0^{(2)} + \pi_{000}^{(2)}d_1^{(2)}} \right) - E[S_1] + \frac{1}{\lambda} \left( \pi_{000}^{(2)}d_0^{(2)} \right).
\]

(6.26)

where \(d_i^{(2)}\) is the probability that in the ETQ model \(l\) customers arrive at stage 1 during the switchover time from stage 2 to 1, so that

\[
d_i^{(2)} = \int_0^\infty \frac{e^{-\lambda t}(\lambda t)^i}{i!} dB_{N_1}(t).
\]

\(\Phi_i(y)\) is defined in (4.24) and therefore \(\Phi_1(1)\) the probability that at the arrival epoch to stage 1 the server finds at most one customer there.

Using (6.26) and (6.24) in (6.23) we get

\[
E[T_{M_2} | N_{M_2} = m] = \left\{ \begin{array}{ll}
\sum_{j=1}^N E[D_{j,j+1}] + \sum_{i=1}^N E[S_i] + \frac{1}{\lambda} \left( \sum_{j=1}^N d_j^{(2)} \right) & : m = 1 \\
\sum_{j=1}^N E[D_{j,j+1}] + m \sum_{i=1}^N E[S_i] & : m = 2, 3, \ldots , M_2
\end{array} \right.
\]

(6.27)

and, therefore, it only remains to obtain \(P\{N_{M_2} = m\}\) for equations (6.19) and (6.20). By expansion of (4.63) for the ETQ model we have

\[
\sum_{r=0}^n \pi_{r00}^{(2)}d_{n-r}^{(2)} = \Phi_n(1) - \Phi_{n-1}(1).
\]
Thus, using (4.96), we obtain

\[
P\{N_{M_G}^G = m\} = \begin{cases}
\frac{\Phi_1(1)}{\Pi_2(1,0,0)} & : m = 1 \\
\frac{\Phi_m(1) - \Phi_{m-1}(1)}{\Pi_2(1,0,0)} & : m = 2, 3, \ldots, M_G - 1 \\
1 - \frac{\Phi_{M_G-1}(1)}{\Pi_2(1,0,0)} & : m = M_G.
\end{cases}
\tag{6.28}
\]

For the average numbers of customers in stages 3, 4, \ldots, N,

\[
\mathcal{L}_j^{(N)}(m) = \frac{m(m-1)}{2} E[S_{j-1}] + mE[D_{j-1,j}] + \frac{m(m+1)}{2} E[S_j] .
\tag{6.29}
\]

and \(P\{N_{M_G}^G = m\}\) is determined from (6.28).

### 6.5.2 Average Switching Costs

Considering \(E[N_{M_G}^G]\) as the average number of customers served during a cycle when gated-limited policy is applied, then the average number of cycles per unit time is

\[
E[N_{M_G}^G] = \frac{\lambda}{E[N_{M_G}^G]} .
\tag{6.30}
\]

Therefore, the average switching costs per unit time is \(E[N_{M_G}^G] \sum_{i=1}^{N} K_{i,i+1} .\) where, using the ETQ model, according to (4.68),

\[
E[N_{M_G}^G] = \eta + \frac{2\pi_0^{(2)} d_0^{(2)}}{2\Pi_2(1,0,0)} + \rho.
\tag{6.31}
\]

### 6.6 Optimal Dynamic Policy

When a dynamic (double-threshold) policy is applied, the server continues serving in stage 1 until either \(M_n\) services have been completed without interruption, or until the first stage becomes empty and there are at least \(M_e\) customers in stage 2 \((M_e \leq M_n)\). Considering a greedy and exhaustive policy downstream of stage 2 and a double-threshold policy in stage 1, the optimal limits \(M_e^*\) and \(M_n^*\) which minimize
the total long-run average cost per unit time can be found by solving the following optimization problem:

\[
\text{Min } E[T_{(M_e, M_n)}] = \sum_{i=1}^{N} h_i L_i^{(N)}(M_e, M_n) + E[N_{(M_e, M_n)}^{\text{cy}}] \sum_{i=1}^{N} K_i, i+1
\]

subject to

\[
\rho + \eta M_n \leq 1
\]

\[
M_e \in \{1, 2, 3, \ldots\}
\]

\[
M_n = M_e M_e + 1, \ldots
\]

(6.32)

where \(E[N_{(M_e, M_n)}^{\text{cy}}]\) is the average number of cycles per unit time when double-threshold policy with parameters \((M_e, M_n)\) is applied.

### 6.6.1 Average Holding Costs

The ETQ model can be used to obtain the average holding cost in stage 1 in the \(N\)-stage tandem queue, since \(L_1^{(N)}(M_e, M_n) = L_1^{(Q)}(M_e, M_n)\). For the average number of customers in stages 2, 3, \ldots, \(N\), we have

\[
L_i^{(N)}(M_G) = \frac{\sum_{m=M_e}^{M_n} L_i^{(N)}(m) P\{N_{(M_e, M_n)}^{\text{cu}} = m\}}{E[T_{(M_e, M_n)}^{\text{cy}}]}
\]

: \(i = 2, 3, \ldots, N\). (6.33)

where

\[
E[T_{(M_e, M_n)}^{\text{cy}}] = \sum_{m=M_e}^{M_n} E[T_{(M_e, M_n)}^{\text{cy}}] \cdot N_{(M_e, M_n)}^{\text{cu}} = m P\{N_{(M_e, M_n)}^{\text{cu}} = m\} .
\]

(6.34)

For stage 2, \(L_2^{(N)}(m)\) is the same as (6.21), but the average interdeparture times from stage 1, \(R_{k,k+1}^{(1)} (k = 1, 2, \ldots, M_e - 1)\), can be found using the ETQ model with double-threshold policy as follows:

\[
E[R_{k,k+1}^{(1)}] = E[\tau_{k+1}^{(m)} - \tau_k^{(m)} | X_2 = k, I = 1]
\]

\[
= \sum_{n=0}^{\infty} E[\tau_{k+1}^{(m)} - \tau_k^{(m)} | X_1 = n, X_2 = k, I = 1] P\{X_1 = n | X_2 = k, I = 1\} .
\]

(6.35)
where
\[
P\{X_1 = n \mid X_2 = k, I = 1\} = \frac{P\{X_1 = n, X_2 = k, I = 1\}}{P\{X_2 = k, I = 1\}} = \frac{\pi_{nk}^{(1)}}{\sum_{n=0}^{\infty} \pi_{nk}^{(1)}} \quad (6.36)
\]

Thus,
\[
E[R_{k,k+1}^{(1)}] = \left[\frac{1}{\lambda} + E[S_1]\right] \frac{\pi_{nk}^{(1)}}{\sum_{n=0}^{\infty} \pi_{nk}^{(1)}} + E[S_1] \left(1 - \frac{\pi_{nk}^{(1)}}{\sum_{n=0}^{\infty} \pi_{nk}^{(1)}}\right)
\]
\[
= \left(\frac{1}{\lambda}\right) \frac{\pi_{nk}^{(1)}}{\sum_{n=0}^{\infty} \pi_{nk}^{(1)}} + E[S_1] \quad k = 1, 2, \ldots, M_e - 1. \quad (6.37)
\]

On the other hand, equation (6.9) can also be used for systems with double-threshold policies. Therefore, considering equation (5.14) for the ETQ model with a double-threshold policy, we get
\[
\Pi_1(1, y) = \frac{y}{1 - y}\{\Pi_2(1, 0) - \Theta_{M_e,M_n}(y) - y^M_n[\Pi_2(1, 0) - \Theta_{M_e,M_n}(1)]\}. \quad (6.38)
\]

Now, suppose
\[
g(y) = \Pi_2(1, 0) - \Theta_{M_e,M_n}(y) - y^M_n[\Pi_2(1, 0) - \Theta_{M_e,M_n}(1)] . \quad (6.39)
\]

Then,
\[
(1 - y)\Pi_1(1, y) = yg(y), \quad (6.40)
\]

and it can be shown that
\[
\Pi_1^{(k)}(1, 0) = \left\{\begin{array}{cl}
g(0) & : k = 1 \\
k!g^{(k-1)}(0) + \Pi_1^{(k-1)}(1, 0) & : k = 2, 3, \ldots, M_e - 1. \quad (6.41)
\end{array}\right.
\]

where \(g^{(r)}(y)\) is the \(r\)th derivation of \(g(y)\) before setting \(y = 0\). Using (6.41) iteratively, we obtain
\[
\Pi_1^{(k)}(1, 0) = \frac{k!}{(k - 1)!}g^{(k-1)}(0) + \frac{k!}{(k - 2)!}g^{(k-2)}(0) + \cdots + \frac{k!}{2!}g^{(2)}(0) + \frac{k!}{1!}[g^{(1)}(0) + g(0)],
\]
or
\[
\Pi_1^{(k)}(1, 0) = \sum_{j=1}^{k-1} \frac{k!}{j!}g^{(j)}(0) + k!g(0) \quad ; \quad k = 1, 2, \ldots, M_e - 1. \quad (6.42)
\]
However, \( g(0) = \Pi_2(1.0) \), and \( g^{(j)}(0) = 0 \) for \( j = 1, 2, \ldots, M - 1 \); hence.

\[
\Pi^{(k)}_1(1.0) = k!\Pi_2(1.0). \tag{6.43}
\]

Substituting (6.43) into (6.41) and considering (6.9), we obtain

\[
\sum_{n=0}^{\infty} \pi^{(1)}_{nk} = \Pi_2(1.0) \quad : \quad k = 1, 2, \ldots, M - 1. \tag{6.44}
\]

and finally, using (6.44) in (6.37) yields

\[
E[R^{(1)}_{k,k+1}] = \left( \frac{1}{\lambda} \frac{\pi^{(1)}_{0k}}{\Pi_2(1.0)} + E[S_1]. \right. \tag{6.45}
\]

Since the server never waits for a customer in an empty stage 1 when there are at least \( M_e \) customers in stage 2, the average interdeparture time between the \( k \)th and \((k+1)\)th customers when \( k = M_e, M_e + 1, \ldots, M_n - 1 \) is always \( E[S_1] \), so that

\[
E[R^{(1)}_{k,k+1}] = \begin{cases} 
\frac{1}{\lambda} \frac{\pi^{(1)}_{0k}}{\Pi_2(1.0)} + E[S_1] & : \quad k = 1, 2, \ldots, M - 1 \\
E[S_1] & : \quad k = M_e, M_e + 1, \ldots, M_n - 1.
\end{cases} \tag{6.46}
\]

To complete the determination of \( L_2^{(V)}(M_e, M_n) \), it only remains to obtain

\[
E[T^{(c)}_{(M_e, M_n)} \mid N^{cu}_{(M_e, M_N)} = m] \quad \text{which is given by}

\[
E[T^{(c)}_{(M_e, M_n)} \mid N^{cu}_{(M_e, M_N)} = m] = E[R^{(1)}_{01}] + \sum_{k=1}^{m-1} E[R^{(1)}_{k,k+1}] + \sum_{j=1}^{N} E[D_{j,j+1}] + m \sum_{i=2}^{N} E[S_i]. \tag{6.47}
\]

where

\[
E[R^{(1)}_{01}] = E[r^{(1)}_1 \mid X_2 = 0, I = 2]
\]

\[
= \sum_{n=0}^{\infty} E[r^{(1)}_1 \mid X_1^+ = n, X_2 = 0, I = 2]P\{X_1^+ = n \mid X_2 = 0, I = 2\}
\]

\[
= \left( \frac{1}{\lambda} + E[S_1]\right)P\{X_1^+ = 0 \mid X_2 = 0, I = 2\}
\]

\[
+ E[S_1](1 - P\{X_1^+ = 0 \mid X_2 = 0, I = 2\}) . \tag{6.48}
\]

and

\[
P\{X_1^+ = 0 \mid X_2 = 0, I = 2\} = \frac{P\{X_1^+ = 0, X_2 = 0, I = 2\}}{P\{X_2 = 0, I = 2\}}
\]

\[
= \frac{\pi^{(2)}_{00} d^{(2)}_0}{\Pi_2(1, 0)} . \tag{6.49}
\]
Therefore, substituting (6.49) into (6.48), we get
\[ E[R_{01}^{(1)}] = \frac{1}{\Pi_1(1.0)} \left( \frac{\tau_{00}^{(1)} \cdot d_0^{(2)}}{\lambda} \right) + E[S_1]. \] (6.50)

Now using (6.50) and (6.46) in (6.47), the average cycle time in which \( m \) customers are served, \( E[T_{Mcu}^{(v)} \mid N_{Mcu}^{(v)} = m] \), is determined.

For stages 3, 4, ..., \( N \), analogously with the other policies, we have
\[ \mathcal{L}_j^{(N)}(m) = \frac{m(m - 1)}{2} E[S_{j-1}] + mE[D_{j-1,j}] + \frac{m(m + 1)}{2} E[S_j]. \] (6.51)

Considering equation (5.72) for the ETQ model with double-threshold policy, we obtain
\[ P\{N_{Mcu}^{(v)} = m\} = \begin{cases} \frac{\pi_{0m}^{(1)}}{\sum_{j=M_e}^{M_n} \pi_{0j}^{(1)} + \Psi_{M_n}(1,1)} & : m = M_e, M_e + 1, \ldots, M_n - 1 \\ \frac{\pi_{0m}^{(1)} + \Psi_{M_n}(1,1)}{\sum_{j=M_e}^{M_n} \pi_{0j}^{(1)} + \Psi_{M_n}(1,1)} & : m = M_n. \end{cases} \] (6.52)

Therefore, the average holding cost per unit time in the \( N \)-stage tandem queue attended by a moving server with double-threshold policy in the first stage and a greedy and exhaustive policy downstream of stage 2, is determined by inserting (6.47) and (6.52) into (6.34), and then (6.34) and (6.51) into (6.33).

### 6.6.2 Average Switching Cost

Using equation (5.73) for the ETQ model with double-threshold policy, the average number of customers served in each cycle, \( E[N_{Mcu}^{(v)}] \), is
\[ E[N_{Mcu}^{(v)}] = \frac{\Theta_{M_e,M_n}^{'} + M_n \Psi_{M_n}(1,1)}{\Theta_{M_e,M_n}^{'} + \Psi_{M_n}(1,1)}. \] (6.53)

Therefore, the average number of cycles per unit time, \( E[N_{Mcu}^{(v)}] \), is
\[ E[N_{Mcu}^{(v)}] = \frac{\lambda}{E[N_{Mcu}^{(v)}]} , \]
and the average switching cost per unit time will be \( E[N_{Mcu}^{(v)}] \sum_{i=1}^{N} K_{i,i+1} \).
6.7 Numerical Study

To find the optimal static (S), gated-limited (G), double-threshold (D) and limited (L) policies which minimize the long-run average holding and switching costs, $E[TC(\gamma)]$ ($\gamma \in \{S, G, D, L\}$), we must minimize

$$\text{Min} \ E[TC(\gamma)] = \sum_{i=1}^{N} h_i L_i^{(N)}(\gamma) + E[N(\gamma)] \sum_{i=1}^{N} K_{i,i+1}$$

subject to stationary conditions for policy $\gamma$. As an example, consider an $M/M_1 - M_2 - \cdots - M_5/1$ queue with $\lambda = 0.7/11$. $E[S_1] = 4$, $E[S_2] = 3$, $E[S_3] = 2$, $E[S_4] = 1$, $E[S_5] = 1$, exponential switchover times $D_{i,i+1}$ with average $E[D_{12}] = 0.25$, $E[D_{23}] = 0.5$, $E[D_{34}] = 0.75$, $E[D_{45}] = 1$, $E[D_{51}] = 0$, holding costs $h = (10, 20, 30, 40, 50)$ and switching costs $K = 3000$. Therefore, the equivalent two-stage tandem queue has exponential service times with mean 4 in stage 1, and in stage 2, the service times have distribution $F_2 * F_3 * \cdots * F_5$ with mean 7. On the other hand, switchover times to stage 1 are zero and the switchover time to stage 2 is a random variable with distribution $B_{12} * B_{23} * \cdots * B_{45}$ and mean 2.5. Table 6.1 presents the average number of customers in the system and the total average cost when a limited policy is applied.

<table>
<thead>
<tr>
<th>$M_L$</th>
<th>$L_1$</th>
<th>$L_2$</th>
<th>$L_3$</th>
<th>$L_4$</th>
<th>$L_5$</th>
<th>L</th>
<th>$E[N_c]$</th>
<th>$E[TC_{M_L}]$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3.3458</td>
<td>0.2068</td>
<td>0.1591</td>
<td>0.1114</td>
<td>0.1273</td>
<td>3.9503</td>
<td>1.0000</td>
<td>244.1</td>
</tr>
<tr>
<td>2</td>
<td>1.7971</td>
<td>0.3669</td>
<td>0.2735</td>
<td>0.1800</td>
<td>0.1731</td>
<td>2.7905</td>
<td>1.5612</td>
<td>171.6</td>
</tr>
<tr>
<td>3</td>
<td>1.4481</td>
<td>0.4809</td>
<td>0.3548</td>
<td>0.2288</td>
<td>0.2056</td>
<td>2.7182</td>
<td>1.7935</td>
<td>160.6</td>
</tr>
<tr>
<td>4</td>
<td>1.2793</td>
<td>0.5636</td>
<td>0.4139</td>
<td>0.2643</td>
<td>0.2292</td>
<td>2.7503</td>
<td>1.8935</td>
<td>159.3</td>
</tr>
<tr>
<td>5</td>
<td>1.1790</td>
<td>0.6238</td>
<td>0.4569</td>
<td>0.2901</td>
<td>0.2464</td>
<td>2.7962</td>
<td>1.9397</td>
<td>160.3</td>
</tr>
<tr>
<td>6</td>
<td>1.1129</td>
<td>0.6686</td>
<td>0.4889</td>
<td>0.3093</td>
<td>0.2683</td>
<td>2.8728</td>
<td>1.9630</td>
<td>161.7</td>
</tr>
</tbody>
</table>

Table 6.1. Average number of customers and total average costs for the $M/M_1 - M_2 - \cdots - M_5/1$ queue under limited policy.

Table 6.1 shows that the optimal limited policy has the limit $M_L^* = 4$. To investigate the relationships among the parameters of the different policies, several sets of 5-stage tandem queues were analyzed and the parameters of the optimal policies determined. Table 6.2 shows one of these sets which was introduced in Duenyas et al [16]. The switching cost $K = 3000$ is added to this set of problems, and the parameters of the optimal policies for these problems are shown in Table 6.3. The service
times and switchover times are all mutually independent exponential random variables. Holding costs are \((h_1, h_2, h_3, h_4, h_5) = (10, 20, 30, 40, 50)\). We assume \(D_1 = D_{51}\) and \(D_i = D_{i-1},\ (i = 2, 3, 4, 5)\), for simplification. Also \(M_S^*, M_G^*\) and \(M_L^*\) are assumed to be the optimal limits of static, semi-dynamic (gated-limited) and limited policies, respectively. As expected, the same result as section 5.5.2 in Chapter 5 about the parameters and the average costs of static, gated-limited and double-threshold policies were observed during our numerical studies of these 5-stage tandem queues, and therefore they are not presented again.
Table 6.2. Parameters of problems 1 to 18

<table>
<thead>
<tr>
<th>Problem</th>
<th>( \rho )</th>
<th>( (S_1, S_2, S_3, S_4, S_5) )</th>
<th>( (S_1, S_2, S_3, S_4, S_5) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.7</td>
<td>(4.3, 2, 1.1)</td>
<td>(0.0, 0.0, 0.0)</td>
</tr>
<tr>
<td>2</td>
<td>0.7</td>
<td>(4.3, 2, 1.1)</td>
<td>(0.5, 0.5, 0.0)</td>
</tr>
<tr>
<td>3</td>
<td>0.7</td>
<td>(4.3, 2, 1.1)</td>
<td>(1.1, 1, 1)</td>
</tr>
<tr>
<td>4</td>
<td>0.7</td>
<td>(4.3, 2, 1.1)</td>
<td>(2.2, 0.5, 0.0)</td>
</tr>
<tr>
<td>5</td>
<td>0.7</td>
<td>(4.3, 2, 1.1)</td>
<td>(0.25, 0.5, 0.75)</td>
</tr>
<tr>
<td>6</td>
<td>0.7</td>
<td>(4.3, 2, 1.1)</td>
<td>(1.0, 0.5, 0.0)</td>
</tr>
<tr>
<td>7</td>
<td>0.8</td>
<td>(1.1, 1, 1)</td>
<td>(0.0, 0.0, 0.0)</td>
</tr>
<tr>
<td>8</td>
<td>0.8</td>
<td>(1.1, 1, 1)</td>
<td>(0.5, 0.5, 0.0)</td>
</tr>
<tr>
<td>9</td>
<td>0.8</td>
<td>(1.1, 1, 1)</td>
<td>(1.1, 1, 1)</td>
</tr>
<tr>
<td>10</td>
<td>0.8</td>
<td>(1.1, 1, 1)</td>
<td>(2.2, 0.5, 0.0)</td>
</tr>
<tr>
<td>11</td>
<td>0.8</td>
<td>(1.1, 1, 1)</td>
<td>(0.25, 0.5, 0.75)</td>
</tr>
<tr>
<td>12</td>
<td>0.8</td>
<td>(1.1, 1, 1)</td>
<td>(1.0, 0.5, 0.0)</td>
</tr>
<tr>
<td>13</td>
<td>0.9</td>
<td>(1.1, 5, 2, 2.5, 3)</td>
<td>(0.0, 0.0, 0.0)</td>
</tr>
<tr>
<td>14</td>
<td>0.9</td>
<td>(1.1, 5, 2, 2.5, 3)</td>
<td>(0.5, 0.5, 0.0)</td>
</tr>
<tr>
<td>15</td>
<td>0.9</td>
<td>(1.1, 5, 2, 2.5, 3)</td>
<td>(1.1, 1, 1)</td>
</tr>
<tr>
<td>16</td>
<td>0.9</td>
<td>(1.1, 5, 2, 2.5, 3)</td>
<td>(2.2, 0.5, 0.0)</td>
</tr>
<tr>
<td>17</td>
<td>0.9</td>
<td>(1.1, 5, 2, 2.5, 3)</td>
<td>(0.25, 0.5, 0.75)</td>
</tr>
<tr>
<td>18</td>
<td>0.9</td>
<td>(1.1, 5, 2, 2.5, 3)</td>
<td>(1.0, 0.5, 0.0)</td>
</tr>
</tbody>
</table>

Table 6.3. Comparison of optimal policies for problems 1 to 18

<table>
<thead>
<tr>
<th>Problem</th>
<th>Static Policy</th>
<th>Gated-limited Policy</th>
<th>Double-threshold Policy</th>
<th>Limited Policy</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( M_2^* )</td>
<td>( \text{Cost} )</td>
<td>( M_2^*' )</td>
<td>( \text{Cost} )</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>131.6</td>
<td>4</td>
<td>171.2</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>133.2</td>
<td>4</td>
<td>168.5</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>148.5</td>
<td>4</td>
<td>167.1</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>141.9</td>
<td>4</td>
<td>162.7</td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td>140.8</td>
<td>4</td>
<td>169.9</td>
</tr>
<tr>
<td>6</td>
<td>3</td>
<td>133.8</td>
<td>4</td>
<td>167.2</td>
</tr>
<tr>
<td>7</td>
<td>4</td>
<td>243.5</td>
<td>5</td>
<td>347.2</td>
</tr>
<tr>
<td>8</td>
<td>4</td>
<td>250.9</td>
<td>5</td>
<td>321.1</td>
</tr>
<tr>
<td>9</td>
<td>7</td>
<td>327.9</td>
<td>8</td>
<td>318.8</td>
</tr>
<tr>
<td>10</td>
<td>6</td>
<td>304.8</td>
<td>7</td>
<td>303.0</td>
</tr>
<tr>
<td>11</td>
<td>5</td>
<td>278.4</td>
<td>6</td>
<td>311.4</td>
</tr>
<tr>
<td>12</td>
<td>5</td>
<td>255.4</td>
<td>5</td>
<td>310.4</td>
</tr>
<tr>
<td>13</td>
<td>3</td>
<td>233.0</td>
<td>4</td>
<td>246.3</td>
</tr>
<tr>
<td>14</td>
<td>3</td>
<td>258.5</td>
<td>4</td>
<td>258.0</td>
</tr>
<tr>
<td>15</td>
<td>8</td>
<td>420.2</td>
<td>9</td>
<td>367.2</td>
</tr>
<tr>
<td>16</td>
<td>7</td>
<td>391.6</td>
<td>9</td>
<td>346.2</td>
</tr>
<tr>
<td>17</td>
<td>5</td>
<td>315.6</td>
<td>6</td>
<td>297.5</td>
</tr>
<tr>
<td>18</td>
<td>4</td>
<td>271.9</td>
<td>5</td>
<td>267.8</td>
</tr>
</tbody>
</table>
Chapter 7

Optimal Policy in
$G^{(x)}/G_1 - G_2 - \cdots - G_N/1$ Queues

7.1 Introduction

When a greedy and exhaustive service policy downstream of stage 2 is not considered in an $M/G_1 - G_2 - \cdots - G_N/1$ queue, the number of applicable policies which can be used in stages 2, 3, $\ldots$, $N$ increases as $N$ and the number of customers in these stages increases. Therefore, it is hard to find the optimal policy for relatively large problems. In this chapter, greedy and exhaustive policies in stages $i$ to $j$ ($j \geq i$) are introduced and the properties of these policies are analyzed in an $N$-stage tandem queue attended by a moving server with zero switchover times and batch arrivals. First, we explain that this model is actually a single multi-functional stochastic scheduling problem with the goal of minimizing the total average holding and switching costs. Then the optimal control problem is formulated and analyzed. Some preliminaries are defined and used as an efficient operational tool to establish the basis for our heuristic algorithms. Finally, two heuristic algorithms are presented and their performances are evaluated using numerical results.
7.2 Single Multi-Functional Machine Stochastic Scheduling or \( G^{(x)}/G_1 - G_2 - \cdots - G_N/1 \)

A single machine (resource) is available to process a set of \( \mathcal{M} = \{1, 2, \ldots, M\} \) jobs. The processing time for job \( j \in \mathcal{M} \) are independent random variables with distribution function \( f_j(.) \). The allocation of the machine to the jobs must be according to a precedence relation specification \( \mathcal{Y} \) on \( \mathcal{M} \). Costs are incurred as processing takes place and whenever the machine switches from one job to another. This single machine stochastic scheduling problem with precedence relations has been analyzed by Glazebrook [18, 19], Sidney [61] and Benkherouf. Glazebrook and Owen [7].

Now suppose that the machine is able to perform a set \( \mathcal{N} = \{1, 2, \ldots, N\} \) of different operations and the precedence relation \( \mathcal{Y} \) on \( \mathcal{M} \) is relaxed: and instead, the precedence relation \( \mathcal{Y}_j \) is defined for job \( j \in \mathcal{M} \) so that if \( (j_u, j_v) \in \mathcal{Y}_j \), then the machine must complete operation \( u \) on job \( j \) before it can begin processing operation \( v \) on this job. \( (u, v \in \mathcal{N}) \). If we now let

\[
\mathcal{Y}_j = \{(u, v) \mid v = u + 1, \ u \in \mathcal{N}\} \quad \forall j \in \mathcal{M}
\]

where \( (N, N + 1) \in \mathcal{Y}_j \) means that job \( j \) is completed after operation \( N \), then this is actually a single multi-functional machine stochastic scheduling problem with \( M \) jobs in which each job requires the sequence of operations 1, 2, \ldots, \( N \), and the machine is capable of performing all these operations. Cost are incurred as processing takes place and also when the machine switches from one operation to another.

From a queueing system perspective, this problem is the stochastic scheduling of a moving server in a system of \( N \)-stage tandem queues with batch arrivals to the first stage. Jobs in each batch require the same sequence of operations and must go through stages 1, 2, \ldots, \( N \). The server (machine) does not start processing a new batch until he completes the jobs in the previous batch.
7.3 Problem Formulation

Consider an $N$-stage tandem queue attended by a moving server with batch arrivals to the first stage (Figure 7.1). Service times in stage $i \in \mathcal{N} = \{1, 2, \ldots , N\}$ are independent random variables $S_i$, with distribution function $F_i(.)$. Batches (orders) arrive according to some stochastic process and the server must complete one order before starting to process the next order. The $k$th batch (order) contains $M_k$ customers (jobs), ($M_k = 1, 2, \ldots$), and all customers in this batch must progress through the same sequence of stages $\mathcal{N}_k = \{n^k_1, n^k_2, \ldots , n^k_N\} \subset \mathcal{N}$. The holding cost $h_{n^k_i}$ per unit time is considered for each customer in batch $k$ during its waiting time in stage $n^k_i$.

We assume that $h_{n^k_i}$ is nondecreasing in $i$, because of the increase in the value of jobs after completion of an operation in each stage. Also, switching cost $K_j$ is incurred whenever the server switches to operation $j$. The goal is to find the optimal control (schedule) of the server in processing each batch, which indicates the index of the stage at which server must serve the next customer in order to minimize the total average holding and switching cost (for each batch).

Let us consider an order of size $m$ in which all jobs must be processed according to the sequence of stages $\{1, 2, \ldots , n\}$. Then there are $mn$ decision epochs where a decision must be made about the next stage for processing a job. Let

$$N_{ij} = \text{Number of jobs waiting in stage } j \text{ at } i\text{th decision epoch}$$

$$X_{ij} = \begin{cases} 
1 & \text{if } i\text{th decision is to process a job in stage } j \\
0 & \text{otherwise}
\end{cases}$$
\[ Y_{ij} = \begin{cases} 1 & \text{if } (i+1)\text{th decision is to process a job in stage } j \smallskip 0 & \text{otherwise} \end{cases} \]

Then, the problem of minimizing the total average holding and switching costs, \( E[TC] \), for the order of size \( m \) is the following nonlinear mixed-integer programming model:

\[
\begin{align*}
\min \ E[TC] &= \sum_{i=1}^{m} \sum_{j=1}^{n} K_j Y_{ij} + \sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=1}^{m} h_j \bar{S}_k N_{ij} X_{ik} \\
\text{subject to:} & \\
\sum_{j=1}^{n} X_{ij} &= 1 \quad i = 1, 2, \ldots, mn \quad (7.1) \\
X_{ij} - N_{ij} &\leq 0 \quad i = 1, 2, \ldots, mn, \quad j = 1, 2, \ldots, n \quad (7.2) \\
N_{i1} - N_{i-1,1} + X_{i-1,1} &= 0 \quad i = 2, 3, \ldots, mn \quad (7.3) \\
N_{ij} - N_{i-1,j} + X_{i-1,j} - X_{i-1,j-1} &= 0 \quad i = 2, 3, \ldots, mn, \quad j = 2, 3, \ldots, n \quad (7.4) \\
Y_{ij} + U_{ij} &\geq 1 \quad i = 1, 2, \ldots, mn - 1, \quad j = 1, 2, \ldots, n \quad (7.5) \\
X_{i+1,j} - X_{ij} + U_{ij} &\leq 1 \quad i = 1, 2, \ldots, mn - 1, \quad j = 1, 2, \ldots, n \quad (7.6) 
\end{align*}
\]

\( X_{ij}, U_{ij}, Y_{ij} = (0, 1), \quad N_i = (m, 0, 0, \ldots, 0), \quad N_{ij} \geq 0 \text{ integer}. \)

Relations (7.1) and (7.2) guarantee that at each decision epoch only one stage is selected for the next action and this stage is selected from among the nonempty stages. Equations (7.3) and (7.4) update the variable \( N_{ij} \) (state of the system) after processing a job. Relations (7.5) and (7.6) establish a relation between \( X_{ij} \) and \( Y_{ij} \) so that, whenever there is a switch to stage \( j, \) \((X_{i+1,j} - X_{ij} = 1)\), switching cost \( K_j \) is incurred \((Y_{ij} = 1)\).

If we define \( Z_{ijk} = N_{ij} X_{ik} \), this model can be linearized as follows:

\[
\begin{align*}
\min \ E[TC] &= \sum_{i=1}^{m} \sum_{j=1}^{m} K_j Y_{ij} + \sum_{i=1}^{m} \sum_{j=1}^{m} \sum_{k=1}^{m} h_j \bar{S}_k Z_{ijk} 
\end{align*}
\]
Subject To:

\[ \sum_{j=1}^{n} X_{ij} = 1 ; \quad i = 1,2,\ldots,mn \]

\[ X_{ij} - N_{ij} \leq 0 ; \quad i = 1,2,\ldots,mn, \quad j = 1,2,\ldots,n \]

\[ N_{i1} - N_{i-1,1} + X_{i-1,1} = 0 ; \quad i = 2,3,\ldots,mn \]

\[ N_{ij} - N_{i-1,j} + X_{i-1,j} - X_{i-1,j-1} = 0 ; \quad i = 2,3,\ldots,mn, \quad j = 2,3,\ldots,n \]

\[ Y_{ij} + U_{ij} \geq 1 ; \quad i = 1,2,\ldots,mn-1, \quad j = 1,2,\ldots,n \]

\[ X_{i+1,j} - X_{ij} + U_{ij} \leq 1 ; \quad i = 1,2,\ldots,mn-1, \quad j = 1,2,\ldots,n \]

\[ \begin{cases} 
-Z_{ijk} + N_{ij} - \tilde{M} W_{ijk} \leq 0 & \quad i = 1,2,\ldots,mn, \quad j = 1,2,\ldots,m \\
Z_{ijk} - N_{ij} - \tilde{M} W_{ijk} \leq 0 & \quad i = 1,2,\ldots,mn, \quad j = 1,2,\ldots,m \\
X_{ik} + \tilde{M} W_{ijk} \leq \tilde{M} & \quad k = 1,2,\ldots,m 
\end{cases} \tag{7.7} \]

\[ X_{ij}, \quad U_{ij}, \quad Y_{ij}, \quad W_{ijk} = (0,1), \quad N_1 = (m,0,0,\ldots,0), \quad N_{ij}, \quad Z_{ijk} \geq 0 \text{ integer} \]

where \( \tilde{M} \) is an arbitrary but large number. This new linear mixed-integer programming model has \( 2n(2mn + m^3 - 1) \) variables and \( mn(1 + 4n + 3m^2) - 3n \) constraints; therefore, for a relatively small problem with \( m = 20 \) jobs and \( n = 10 \) operations we will have a rather large mixed-integer programming problem with \( 187980 \) variables and \( 248170 \) constraints.

The control of the server in an \( N \)-stage tandem queue with batch arrivals can also be formulated as a dynamic programming model, or specifically, a shortest route problem. Consider the state of the system (the nodes in a shortest route problem) as vector \( (y^r, x_1^r, x_2^r, \ldots x_n^r) \), where \( x_j^r \) is the number of jobs waiting in stage \( j \) after the \( r \)th task (\( r \)th decision epoch), \( r = 1,2,\ldots,mn, \ x_j^r = 0,1,2,\ldots,m \), and \( y^r \) is the index of the stage at which the server performed his \( r \)th task. The distance between two nodes \( (y^r, x_1^r, x_2^r, \ldots x_n^r) \) and \( (y^{r+1}, x_1^{r+1}, x_2^{r+1}, \ldots x_n^{r+1}) \) in the shortest route problem, or, alternately, the least cost path problem, is \( \sum_{i=1}^{n} h_i x_i^r \tilde{S}_{y^r+1} \) when \( y^{r+1} = y^r \), and it is \( \sum_{i=1}^{n} h_i x_i^r \tilde{S}_{y^{r+1}} + K_{y^{r+1}} \) when \( y^{r+1} \neq y^r \). In this least cost path problem, the objective is to find the least cost path between nodes \((1,m,0,0,\ldots,0)\) and \((n,0,0,\ldots,0)\).

To find the number of nodes for the problem with batches of size \( m \) jobs and \( n \) stages when the least cost path problem is used, we first consider the case \( m \geq n \). For states \((y^r, x_1^r, x_2^r, \ldots x_n^r)\) in which \( x_1^r + x_2^r + \cdots + x_n^r = u \) and \( 1 \leq u \leq m \), these \( u \)
jobs could be waiting in \( v \) stages (\( v = 1, 2, \ldots, u \)). Since there are \( n \) stages, there will therefore be \( C^n_v \) different sets of \( v \) stages (\( C^n_v \) is the number of combinations of \( v \) from \( n \)). On the other hand, there are \( C^{v-1}_{v-1} \) different ways that \( u \) jobs can be distributed among \( v \) stages, where each stage has at least one job. When the jobs are distributed into \( v \) stages, it means that there are \( v \) possibilities for the server to choose the next stage to process a job. Consequently, the number of nodes in which \( x_1' + x_2' + \cdots + x'_n = u \) is \( \sum_{v=1}^u vC^n_v C^{u-1}_{v-1} \). Since \( u = 1, 2, \ldots, m \), then the total number of nodes for the least cost path problem will be \( \sum_{u=1}^n \sum_{v=1}^u vC^n_v C^{u-1}_{v-1} + \sum_{u=n+1}^m \sum_{v=1}^n vC^n_v C^{u-1}_{v-1} \). By expansion we will have

\[
\sum_{u=1}^n \sum_{v=1}^u vC^n_v C^{u-1}_{v-1} + \sum_{u=n+1}^m \sum_{v=1}^n vC^n_v C^{u-1}_{v-1} = \sum_{j=1}^n jC^n_j \sum_{k=j}^m C^{k-1}_{j-1}. \tag{7.8}
\]

Using the same approach for \( m < n \), we determine that the number of nodes for the least cost path problem is \( \sum_{u=1}^m \sum_{v=1}^u vC^n_v C^{u-1}_{v-1} \), and by expansion, it can be shown that

\[
\sum_{u=1}^m \sum_{v=1}^u vC^n_v C^{u-1}_{v-1} = \sum_{j=1}^m jC^n_j \sum_{k=j}^m C^{k-1}_{j-1}. \tag{7.9}
\]

The shortest route problem with \( n \) nodes can be solved in polynomial time \( O(n^2) \). In the present case with \( m \geq n \), since \( \sum_{j=1}^n jC^n_j = n2^{n-1} \) and \( \sum_{k=j}^m C^{k-1}_{j-1} > 1 \), we can consider \( n2^{n-1} \) as a lower bound for the number of nodes. In other words, for \( n = 2^{n-1} \), to solve our least cost path problem we need an algorithm with complexity \( O(n^22^n) \), which is no longer a polynomial time algorithm. Using the same argument when \( m < n \), an algorithm with complexity \( O(m^22^{m-n}) \) is required. For our example with \( m = 20 \) jobs and \( n = 10 \) stages, and using (7.8), the number of nodes is over \( 2 \times 10^8 \).

These considerations lead us to the obvious need to develop simple heuristics to find a suitable suboptimal policy. We will show that our heuristics drastically reduce the number of required computations. For instance, one such heuristic will approximate the least cost path model for \( m = 20 \) and \( n = 10 \), using a shortest route problem with 11 nodes. Numerical results will show that these heuristics are highly efficient and accurate.
Before the introduction of these heuristics, we first need to establish some preliminaries which will be used to construct the basis of the heuristics.

### 7.4 Preliminaries

Since our heuristics are based on the properties of greedy and exhaustive policies, the following definitions must be introduced:

**Definition 7.1:** A greedy policy in stages $i$ to $j$ ($i < j$), is a policy in which the server applies a greedy policy in stages $i, i+1, \ldots, j$, and never switches to stages $1, 2, \ldots, i-1, j+1, \ldots, N$, until stages $i$ to $j$ all become empty. For $j = N$, it is called a greedy policy downstream of stage $i$. Let $\Gamma_{ij}$ be the set of all greedy policies in stages $i$ to $j$, and $\Gamma = \bigcup_{i<j} \Gamma_{ij}$.

**Definition 7.2:** A greedy and exhaustive policy in stages $i$ to $j$ ($i < j$), is a policy in which the server applies a greedy and exhaustive policy in stages $i, i+1, \ldots, j$, and switches from stage $i$ to $i+1$ after each exhaustion epoch. When the last customer departs from stage $j$, and stages $i$ to $j$ all become empty, the server switches to one of the stages $1, 2, \ldots, i-1, j+1, \ldots, N$. For $j = N$, this policy is called a greedy and exhaustive policy downstream of stage $i$.

To simplify the analytic procedure, some new concepts, terminology and notation will also be introduced:

- $i \uparrow j$ indicates a $(j - i + 1)$-stage tandem queue ($j \geq i$) consisting of stages $i, i+1, \ldots, j-1, j$, and buffer of stage $j+1$ sequentially. Some examples are shown in Figure 7.2.

- $(i \triangleright m, j)$ indicates a policy which serves $m$ customers from among the $x_i \geq m$ customers waiting in stage $i$ according to a greedy and exhaustive policy in stages $i$ to $j$ ($j \geq i$).

- Notation $\circ$ shows the order of a sequence of different policies from left to right. For instance, $\gamma \circ \alpha$ indicates that the server applies policy $\alpha$ immediately after
completing policy $\gamma$.

- Notation $\equiv$ is used to show that two policies are equivalent. As an example, it is clear that

$$(i \overset{m}{\triangleright} j) \equiv (i \overset{m}{\triangleright} k) \circ (k + 1 \overset{m}{\triangleright} j) ; i \leq k < j$$

- Notation $\circ$ is used to show a policy which is repeated $r$ times. So, for example

$$\overset{3}{\circ} (i \overset{m}{\triangleright} j) \equiv (i \overset{m}{\triangleright} j) \circ (i \overset{m}{\triangleright} j) \circ (i \overset{m}{\triangleright} j)$$

- Suppose policy $\gamma$ is applied in stage $i$; then $||\gamma||$ indicates the number of customers served in stage $i$ when policy $\gamma$ is applied. If policy $\gamma$ is defined for stages $i$ to $j$, then $||\gamma||$ is a vector with $j - i + 1$ elements in which the $k$th element indicates the number of customers served in stage $i + k - 1$ during the time that policy $\gamma$ is applied. As examples

$$||(i \overset{m}{\triangleright} k) \circ (k + 1 \overset{m}{\triangleright} j)|| = (m, m, \ldots, m)$$

$$||(i \overset{m}{\triangleright} j) \circ (i \overset{m}{\triangleright} j)|| = (m + n, m + n, \ldots, m + n)$$

$$|| \overset{r}{\circ} (i \overset{m}{\triangleright} j)|| = (rm, rm, \ldots, rm)$$

To simplify $||\gamma||$ when all its elements are equal to, say, $v$, we use $[\gamma]$ to show the scalar value $v$. Therefore

$$[\overset{m}{(i \overset{m}{\triangleright} k) \circ (k + 1 \overset{m}{\triangleright} j)}] = m$$
\[(i \overset{m}{\triangleright} j) \circ (i \overset{m}{\triangleright} j) = m + n\]

\[\circ (i \overset{m}{\triangleright} j) = rm\]

Example 7.1

Complex policies in a system of connected queues with a moving server such as a polling system are usually represented in tabular form. However, our new notation can be used effectively to express different policies in these systems. As an example, suppose in a 10-stage tandem queue the server applies policy \(\gamma\) in stages 6 to 10 according to Table 7.1.

<table>
<thead>
<tr>
<th>Stage</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>9</th>
<th>10</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Served Customers</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>4</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>

Table 7.1. Policy \(\gamma\) in 6 \(\triangleright\) 10 queue.

Policy \(\gamma\) can be shown using our notations as follows:

\[\gamma \equiv (6 \overset{4}{\triangleright} 8) \circ (9 \overset{2}{\triangleright} 10) \circ (9 \overset{2}{\triangleright} 10) \circ (6 \overset{4}{\triangleright} 8) \circ (9 \overset{2}{\triangleright} 10) \circ (9 \overset{2}{\triangleright} 10)\]

\[\equiv (6 \overset{4}{\triangleright} 8) \overset{2}{\triangleright} (9 \overset{2}{\triangleright} 10) \circ (6 \overset{4}{\triangleright} 8) \overset{2}{\triangleright} (9 \overset{2}{\triangleright} 10)\]

\[\equiv \overset{2}{\triangleright} [(6 \overset{4}{\triangleright} 8) \overset{2}{\triangleright} (9 \overset{2}{\triangleright} 10)]\]

The new expression is much more comprehensive, compact and flexible. Also, by using these notations, policies can be expressed in terms of numbers and symbols. We therefore now have the capability of constructing an efficient operational tool for optimization analysis of systems with connected queues and moving servers.

- Let \(C_h[\gamma]\) (\(C_K[\gamma]\)) be the total average holding (switching) cost associated with the customers who receive service during the time that policy \(\gamma\) is applied. As an example, in a 4-stage tandem queue with state \((2, x_1, x_2, x_3, x_4)\) for \(m \leq x_2\), we have

\[C_h[2 \overset{m}{\triangleright} 3] = \frac{m(m+1)}{2} h_2 \bar{S}_2 + \frac{m(m-1)}{2} h_3 \bar{S}_2 + \frac{m(m+1)}{2} h_3 \bar{S}_3 + \frac{m(m-1)}{2} h_4 \bar{S}_3\]

\[C_K[2 \overset{m}{\triangleright} 3] = K_3\]
in which \( \overline{S}_i = E[S_i] \). If \( C[\gamma] \) is the total average holding and switching cost, we will have

\[
C[\gamma] = C_h[\gamma] + C_K[\gamma]. \tag{7.12}
\]

**Example 7.2**

Consider a 4-stage tandem queue attended by a moving server with holding costs \( h = (2, 4, 10, 12) \), switching costs \( K = (10, 20, 5, 15) \) and average service times \( \overline{S} = (2, 3, 1, 5) \). If policy \( \gamma \) serves 4 customers according to a greedy and exhaustive policy in stages 2 to 3, then \( \gamma \equiv (2 \downarrow 3) \) and

\[
C_h(2 \downarrow 3) = \frac{4(4 + 1)}{2}((4)(3) + (10)(1)) + \frac{4(4 - 1)}{2}((10)(3) + (12)(1)) = 472
\]

\[
C_K(2 \downarrow 3) = 5.
\]

and therefore

\[
C(2 \downarrow 3) = 477 \quad \square
\]

**Lemma 7.1.**

For an \( N \)-stage tandem queue with state \((i, x_1, x_2, \ldots, x_N)\) and \( j > i, m \leq x_i, \)

\[
(i) \quad C_h[i \supset j] = \frac{m(m + 1)}{2} \sum_{k=i}^{j} h_k \overline{S}_k + \frac{m(m - 1)}{2} \sum_{k=i+1}^{j+1} h_k \overline{S}_{k-1} \tag{7.13}
\]

\[
(ii) \quad C_K[i \supset j] = \sum_{k=i+1}^{j} K_k. \tag{7.14}
\]

\[
(iii) \text{If } \gamma_m \equiv (i \supset m) \text{ and } \gamma_n \equiv (i \supset n), \text{ then}
\]

\[
C[\gamma_m \circ \gamma_n] = C[\gamma_m] + C[\gamma_n] + K_i + [\gamma_m] \sum_{k=i}^{j} \overline{S}_k, \tag{7.15}
\]

and, therefore, \( C[\gamma_m \circ \gamma_n] = C[\gamma_n \circ \gamma_m]. \)

\[
(iv) \quad C[\supset (i \supset j)] = rC[i \supset j] + (r - 1)K_i + m^2 \frac{r(r - 1)}{2} (h_i + h_{j+1}) \sum_{k=i}^{j} \overline{S}_k. \tag{7.16}
\]
Proof

(i) Since $C_h[i \succ j]$ is actually the average holding cost of the $m$ customers who receive their service according to the greedy and exhaustive policy in stages $i$ to $j$, therefore

$$C_h[i \succ j] = \frac{m(m+1)}{2}h_i\bar{S}_i + \frac{m(m-1)}{2}h_{i+1}\bar{S}_i + \frac{m(m+1)}{2}h_{i+1}\bar{S}_{i+1} + \frac{m(m-1)}{2}h_{i+2}\bar{S}_{i+1} + \cdots + \frac{m(m+1)}{2}h_j\bar{S}_j + \frac{m(m-1)}{2}h_{j+1}\bar{S}_j$$

$$= \frac{m(m+1)}{2}j \sum_{k=i}^{j} h_k\bar{S}_k + \frac{m(m-1)}{2} \sum_{k=i+1}^{j+1} h_k\bar{S}_{k-1}.$$ 

(ii) Since the server switches from stage $i$ to $i+1$ during policy $(i \succ j)$ and the last switch is from stage $j-1$ to $j$, then (7.14) follows directly.

(iii) Customers who will be served when policy $\gamma_n$ is applied must wait in stage $i$ during the services of $m$ customers in stages $i$ to $j$. On the other hand, customers who are served by policy policy $\gamma_m$ must wait in stage $j+1$ during the service of $n$ customers in stages $i$ to $j$. Therefore, considering $[\gamma_m] = m$ and $[\gamma_n] = n$, we have

$$C[\gamma_m \circ \gamma_n] = C[\gamma_m] + C[\gamma_n] + K_i + nh_i(m \sum_{k=i}^{j} \bar{S}_k)$$

$$+ mh_i(n \sum_{k=i}^{j} \bar{S}_k)$$

$$= C[\gamma_m] + C[\gamma_n] + K_i + [\gamma_m] [\gamma_n] (h_i + h_{j+1}) \sum_{k=i}^{j} \bar{S}_k.$$ 

(iv) Setting $\gamma_m \equiv (i \succ j)$, and then using (7.15),

$$C[\bar{O} \gamma_m] = C[\gamma_m \circ \gamma_m] = 2C[\gamma_m] + K_i + m^2(h_i + h_{j+1}) \sum_{k=i}^{j} \bar{S}_k$$

$$C[\bar{O} \gamma_m] = C[\bar{O} \gamma_m \circ \gamma_m] = 3C[\gamma_m] + K_i + (m^2 + 2m^2)(h_i + h_{j+1}) \sum_{k=i}^{j} \bar{S}_k.$$ 

and continuing this approach, we get

$$C[\bar{O} \gamma_m] = rC[\gamma_m] + (r-1)K_i + m^2(1 + 2 + \cdots + r - 1)(h_i + h_{j+1}) \sum_{k=i}^{j} \bar{S}_k.$$
leading to (7.16). □

- **Substitute Policy:** Suppose $\gamma, \pi \in \Gamma$. Then policy $\pi$ is called a substitute policy for $\gamma$ (or vice versa) if $\|\gamma\| = \|\pi\|$. For example, if $\gamma \equiv (i \overset{m}{\rightarrow} j)$, $\pi \equiv (i \overset{m}{\rightarrow} k) \circ (k + 1 \overset{m^*}{\rightarrow} j)$ and $\beta \equiv (i \overset{m^*}{\rightarrow} j) \circ (i \overset{1}{\rightarrow} j)$, then $\pi$, $\gamma$ and $\beta$ are substitute policies for one another. Let $\mathcal{S}(\gamma)$ be the set of all possible substitute policies for policy $\gamma$.

**Example 7.3**

If $\gamma \equiv (3 \overset{3}{\rightarrow} 5)$, then $\pi$ and $\beta$ are two possible substitute policies for $\gamma$, where

$$
\beta \equiv (3 \overset{2}{\rightarrow} 4) \circ (3 \overset{1}{\rightarrow} 3) \circ (5 \overset{2}{\rightarrow} 5) \circ (4 \overset{1}{\rightarrow} 5),
\pi \equiv \overset{3}{\beta_3} (3 \overset{1}{\rightarrow} 4) \circ (5 \overset{3}{\rightarrow} 5). \quad \square
$$

- **Greedy and Exhaustive Substitutes:** A greedy and exhaustive policy $(i \overset{m}{\rightarrow} j)$ has substitutes which are a sequence of greedy and exhaustive policies $(i \overset{m_i}{\rightarrow} j)$, where $\sum_{i=1}^{m} m_i = m$. The set of these substitute policies constructs set $\mathcal{E}(i \overset{m}{\rightarrow} j) \subset \mathcal{S}(i \overset{m}{\rightarrow} j)$ and is called the set of greedy and exhaustive substitutes for policy $(i \overset{m}{\rightarrow} j)$. We assume $(i \overset{m}{\rightarrow} j) \in \mathcal{E}(i \overset{m}{\rightarrow} j)$.

**Example 7.4**

For policy $\gamma$ in Example 7.3, the set of greedy and exhaustive substitute policies, $\mathcal{E}(\gamma)$, is

$$
\mathcal{E}(3 \overset{3}{\rightarrow} 5) = \{ \overset{3}{\beta} (3 \overset{1}{\rightarrow} 5), (3 \overset{2}{\rightarrow} 5) \circ (3 \overset{1}{\rightarrow} 5), (3 \overset{1}{\rightarrow} 5) \circ (3 \overset{2}{\rightarrow} 5), (3 \overset{3}{\rightarrow} 5) \}. 
$$

We use $\gamma \leftarrow \xi$ to show that $\xi \in \mathcal{E}(\gamma)$. Therefore, we can write

$$
(3 \overset{3}{\rightarrow} 5) \leftarrow \overset{3}{\beta} (3 \overset{1}{\rightarrow} 5) \quad \square
$$
**Order:** Suppose $\gamma \in \Gamma_{ij}$. If $\xi^{(i)} \in \mathcal{E}(\gamma)$, and $\xi^{(i)}$ consists of $l$ greedy and exhaustive policies in stages $i$ to $j$, then $\xi^{(i)}$ is called a substitute policy of order $l$. The subset of $\mathcal{E}(\gamma)$ which consists of all substitutes of order $l$ is shown by $\mathcal{E}_l(\gamma)$. If $\xi^{(i)} \in \mathcal{E}_l(i \gg j)$, then

$$\xi^{(i)} \equiv (i \gg j) \circ (i \gg j) \circ \cdots \circ (i \gg j)$$

and $\xi^{(i)}$ is fully determined by $m_1, m_2, \ldots, m_l$. These values are called parameters of policy $\xi^{(i)} \in \mathcal{E}_l(i \gg j)$. It is clear that

$$\bigcup_{l=1}^{m} \mathcal{E}_l(i \gg j) = \mathcal{E}(i \gg j). \quad (7.17)$$

If $\xi^{(i)} \in \mathcal{E}_l(\gamma)$, then we use notation $\gamma \xrightarrow{l} \xi^{(i)}$ to show that $\xi^{(i)}$ is a substitute policy of order $l$ for policy $\gamma$.

**Example 7.5**

For policy $\gamma$ in Example 7.3, we will have

$$\mathcal{E}_1(3 \gg 5) = \{(3 \gg 5)\}$$

$$\mathcal{E}_2(3 \gg 5) = \{(3 \gg 5) \circ (2 \gg 3), (3 \gg 5) \circ (3 \gg 5)\}$$

$$\mathcal{E}_3(3 \gg 5) = \{3 \circ (3 \gg 5)\}$$

and $(3 \gg 5) \circ (3 \gg 5) \in \mathcal{E}_2(3 \gg 5)$ has parameters $(2, 1) \quad \square$.

**Least Cost Policy of Order $l$:** Suppose $\mathcal{E}_l(\gamma)$ consists of the $n_l$ policies $\xi_1^{(i)}, \xi_2^{(i)} - - - , \xi_{n_l}^{(i)}$, and

$$\xi_{LC}^{(i)} \equiv \{\xi_a^{(i)} \mid \xi_a^{(i)} \in \mathcal{E}_l(\gamma), \mathcal{C}[\xi_a^{(i)}] \leq \mathcal{C}[\xi_u^{(i)}] ; u = 1, 2, \ldots, n_l\} ;$$

then $\xi_{LC}^{(i)}$ is called the Least Cost policy in $\mathcal{E}_l(\gamma)$. If $\mathcal{C}[\xi_{LC}^{(i)}] \leq \mathcal{C}[\gamma]$, then $\xi_{LC}^{(i)}$ is called the best greedy and exhaustive policy of order $l$ for policy $\gamma$, and it is shown by $\gamma \xleftarrow{l} \xi_{LC}^{(i)}$. In cases for which $\mathcal{C}[\xi_{LC}^{(i)}] > \mathcal{C}[\gamma]$, there is no better policy than $\gamma$ in $\mathcal{E}_l(\gamma)$, and this can be shown by $\gamma \xleftarrow{l} \emptyset$. 


Example 7.6

Consider policy $\gamma \equiv (1 \rightarrow 2)$ in a two-stage tandem queue with holding costs $h = (2.6)$ switching costs $K = (10.7)$ and average service times $\overline{S} = (1.3)$. The set of greedy and exhaustive substitute policy of order 2 for policy $\gamma$ is

$$\mathcal{J}_2(1 \rightarrow 2) = \{ \frac{2}{3} (1 \rightarrow 2), \ (1 \rightarrow 2) \circ (1 \rightarrow 2), \ (1 \rightarrow 2) \circ (1 \rightarrow 2) \}$$

To find the least cost policy of order 2 in $\mathcal{J}_2(1 \rightarrow 2)$ we have to compare the total average cost of all policies in that set. Using Lemma 7.1, we get

$$C[\frac{2}{3} (1 \rightarrow 2)] = 2C[1 \rightarrow 2] + K_1 + 4h_1(\overline{S}_1 + \overline{S}_2)$$

$$= 188.$$  

and also

$$C[(1 \rightarrow 2) \circ (1 \rightarrow 2)] = C[1 \rightarrow 2] + C[1 \rightarrow 2] + K_1 + 3h_1(\overline{S}_1 + \overline{S}_2)$$

$$= 206.$$  

Since $C[(1 \rightarrow 2) \circ (1 \rightarrow 2)] = C[(1 \rightarrow 2) \circ (1 \rightarrow 2)]$, the least cost policy of order 2 for policy $(1 \rightarrow 2)$ is $\frac{2}{3} (1 \rightarrow 2)$, or in other words.

$$(1 \rightarrow 2) \xleftarrow{\frac{2}{3}} (1 \rightarrow 2). \quad \Box$$

- **The Best Greedy and Exhaustive Substitute Policy:** Suppose $\gamma \equiv (i \rightarrow j)$, and set

$$\psi_\gamma \equiv \{ \xi_{\mathcal{E}_b}^{(b)} \mid \xi_{\mathcal{E}_b}^{(b)} \in \mathcal{E}_b(\gamma), b \in \mathbb{Z}^+, 1 \leq b \leq m, C[\xi_{\mathcal{E}_b}^{(b)}] \leq C[\xi_{\mathcal{E}_u}^{(u)}] ; \forall u = 1, 2, \ldots, m \}.$$  

Then, $\psi_\gamma$ is called the best greedy and exhaustive substitute policy for policy $\gamma$, and this is shown by writing $\gamma \xleftarrow{\psi_\gamma}$. However, $\Omega^*(\gamma) = b$ indicates that the order of the best greedy and exhaustive substitute policy for $\gamma$ is $b$. As an example, $\Omega^*(i \rightarrow j) = 1$ means that the best greedy and exhaustive substitute policy for $(i \rightarrow j)$ is itself.
Example 7.7

It can be shown that for policy $\gamma \equiv (1 \rightarrow 2)$ in Example 7.6,

$$(1 \rightarrow 2) \leftarrow^1 (1 \rightarrow 2)$$

$$(1 \rightarrow 2) \leftarrow^2 \frac{2}{3} (1 \rightarrow 2)$$

$$(1 \rightarrow 2) \leftarrow^3 (1 \rightarrow 2) \frac{2}{5} (1 \rightarrow 2)$$

$$(1 \rightarrow 2) \leftarrow^4 \frac{4}{5} (1 \rightarrow 2) .$$

Since

$$C[1 \rightarrow 2] = 243$$

$$C[\frac{2}{3} (1 \rightarrow 2)] = 188$$

$$C[(1 \rightarrow 2) \frac{2}{5} (1 \rightarrow 2)] = 187$$

$$C[\frac{4}{5} (1 \rightarrow 2)] = 186 .$$

the best greedy and exhaustive substitute policy for $1 \rightarrow 2$ is policy $\frac{4}{5} (1 \rightarrow 2)$, or

$$(1 \rightarrow 2) \leftarrow^4 \frac{4}{5} (1 \rightarrow 2) .$$

Policy $\frac{4}{5} (1 \rightarrow 2)$ has parameters $(1.1.1.1)$, and since $\frac{4}{5} (1 \rightarrow 2)$ is a greedy and exhaustive substitute policy of order 4, we have $\Omega^*(1 \rightarrow 2) = 4 . \quad \Box$

7.5 Basis of Heuristics

Our heuristics are based on the properties of the greedy and exhaustive policy in stages $i$ to $j$. These properties are analyzed in Lemmas 7.2, 7.3 and 7.4.

Lemma 7.2

For each tandem queue $i \uparrow j$, the holding cost factor, $A_{ij}$, and the switching cost factor, $K_{ij}$, are defined as follows

$$A_{ij} = \sum_{k=i}^{j} (h_k - h_i - h_{j+1})S_k + \sum_{k=i+1}^{j+1} h_k S_{k-1} \quad (7.18)$$

$$K_{ij} = \sum_{k=i}^{j} K_k . \quad (7.19)$$
Then,

(i) considering policies $\xi^{(l)} \in \mathcal{E}_l(i \triangleright j)$ and $\xi^{(l-1)} \in \mathcal{E}_{l-1}(i \triangleright j)$ with parameters $(m_1, m_2, m_3, \ldots, m_l)$ and $(m_1 + m_2, m_3, \ldots, m_l)$, respectively, we will have

$$C[\xi^{(l)}] = C[\xi^{(l-1)}] - m_1 m_2 A_{ij} + K_{ij} .$$  \hspace{1cm} (7.20)

(ii) if $\xi^{(l)} \in \mathcal{E}_l(i \triangleright j)$ has parameters $(m_1, m_2, \ldots, m_l)$, we will have

$$C[\xi^{(l)}] = C[i \triangleright j] - \varphi_l(m) A_{ij} + (l - 1) K_{ij}$$  \hspace{1cm} (7.21)

where $\varphi_l(m) = \sum_{k=1}^{l-1} m_k \sum_{u=k+1}^{l} m_u$.

(iii) if $\xi^{(l)} \in \mathcal{E}_l(i \triangleright j)$ and $\xi^{(r)} \in \mathcal{E}_r(i \triangleright j)$ have parameters $(m_1, m_2, \ldots, m_l)$ and $(n_1, n_2, \ldots, n_r)$, respectively, we will have

$$C[\xi^{(l)}] = C[\xi^{(r)}] + [\varphi_r(n) - \varphi_l(m)] A_{ij} + (l - r) K_{ij} .$$  \hspace{1cm} (7.22)

Proof

(i) Suppose $\gamma = (i \triangleright^3 j) \circ (i \triangleright^4 j) \circ \cdots \circ (i \triangleright^l j)$; then $[\gamma] = \sum_{k=3}^{l} m_k$. and

$$\xi^{(l)} = (i \triangleright^1 j) \circ (i \triangleright^2 j) \circ \gamma$$

$$\xi^{(l-1)} = (i \triangleright^{m_1 + m_2} j) \circ \gamma .$$

Now, using part (iii) of Lemma 7.1, we get

$$C[\xi^{(l)}] = C[(i \triangleright^1 j) \circ (i \triangleright^2 j) ] + C[\gamma]$$

$$+ K_i + (m_1 + m_2)(\sum_{k=3}^{l} m_k)(h_i + h_{j+1}) \sum_{k=i}^{j} S_k ,$$  \hspace{1cm} (7.23)

and

$$C[\xi^{(l-1)}] = C[i \triangleright^{m_1 + m_2} j] + C[\gamma]$$

$$+ K_i + (m_1 + m_2)(\sum_{k=3}^{l} m_k)(h_i + h_{j+1}) \sum_{k=i}^{j} S_k .$$  \hspace{1cm} (7.24)

Hence,

$$C[\xi^{(l)}] - C[\xi^{(l-1)}] = C[i \triangleright^1 j] \circ (i \triangleright^2 j) ] - C[i \triangleright^{m_1 + m_2} j] .$$  \hspace{1cm} (7.25)
On the other hand,

\[
C[(i \vartriangleright j) \circ (i \vartriangleright j)] = C[i \vartriangleright j] + C[i \vartriangleright_2 j] + K_i + m_1 m_2 (h_i + h_{j+1}) \sum_{k=i}^{j} \overline{s}_k .
\] (7.26)

and, using part (i) of Lemma 7.1 for \( C[i \vartriangleright j] \), \( C[i \vartriangleright_2 j] \) and \( C[i \vartriangleright m_1 \vartriangleright m_2 j] \), and then substituting the results into (7.26) and (7.25), we get, after some algebra, that

\[
C[\xi^{(l)}] - C[\xi^{(l-1)}] = -m_1 m_2 \left( \sum_{k=i}^{j} h_k \overline{s}_k + \sum_{k=i+1}^{j+1} h_k \overline{s}_{k-1} \right) + m_1 m_2 (h_i + h_{j+1}) \sum_{k=i}^{j} \overline{s}_k + \sum_{k=i+1}^{j+1} K_k + K_i
\]

\[
= -m_1 m_2 \left( \sum_{k=i}^{j} (h_k - h_i - h_{j+1}) \overline{s}_k + \sum_{k=i+1}^{j+1} h_k \overline{s}_{k-1} \right) + K_{ij}
\]

\[
= -m_1 m_2 A_{ij} + K_{ij} .
\]

which is (7.20).

(ii) Since \( \xi^{(l)} \in E_l(i \vartriangleright j) \) has parameters \((m_1, m_2, \ldots, m_l)\), then

\[
\xi^{(l)} = (i \vartriangleright j) \circ (i \vartriangleright j) \circ \cdots \circ (i \vartriangleright j) .
\]

Suppose

\[
\gamma_l = (i \vartriangleright \gamma_{l+1}) \circ (i \vartriangleright \gamma_{l+1}) \circ \cdots \circ (i \vartriangleright j) .
\] (7.27)

then \( \gamma_l = (i \vartriangleright \gamma_{l+1}) \circ \gamma_{l+1} \) with \( |\gamma_l| = |\gamma_{l+1}| + m_{k+1} = \sum_{u=k+1}^{l} m_u \). Using (7.27) we have the relations set

\[
\begin{align*}
C[(i \vartriangleright j) \circ \gamma_1] &= C[i \vartriangleright j] - m_1 \left[ \gamma_1 \right] A_{ij} + K_{ij} \\
C[(i \vartriangleright j) \circ (i \vartriangleright j) \circ \gamma_2] &= C[(i \vartriangleright j) \circ \gamma_1] - m_2 \left[ \gamma_2 \right] A_{ij} + K_{ij} \\
&\vdots \\
C[(i \vartriangleright j) \circ \cdots \circ (i \vartriangleright j) \circ \gamma_{l-1}] &= C[(i \vartriangleright j) \circ \cdots \circ (i \vartriangleright j) \circ \gamma_{l-1}] - m_{l-1} \left[ \gamma_{l-1} \right] A_{ij} + K_{ij} .
\end{align*}
\] (7.28)

Summing all the equations in (7.28) yields

\[
C[\xi^{(l)}] = C[i \vartriangleright j] - \sum_{k=1}^{l-1} m_k \left[ \gamma_k \right] A_{ij} + (l - 1)K_{ij}
\]

\[
= C[i \vartriangleright j] - \sum_{k=1}^{l-1} m_k \sum_{u=k+1}^{l} m_u A_{ij} + (l - 1)K_{ij}
\]

\[
= C[i \vartriangleright j] - \varphi_l(m) A_{ij} + (l - 1)K_{ij} .
\]
(iii) Using (7.21) for \( \xi^{(l)} \) and \( \xi^{(r)} \), gives

\[
C[\xi^{(l)}] = C\left[ i \overset{m}{\longrightarrow} j \right] - \varphi_i(m)A_{ij} + (l - 1)K_{ij} \tag{7.29}
\]

\[
C[\xi^{(r)}] = C\left[ i \overset{n}{\rightarrow} j \right] - \varphi_r(n)A_{ij} + (r - 1)K_{ij} \tag{7.30}
\]

Subtracting (7.30) from (7.29) yields (7.22). \( \square \)

Equations (7.20) and (7.21) can be used to compare the total average cost of the greedy and exhaustive substitutes of a policy. This comparison can be done using the holding and switching cost factors and the parameters of the policies.

**Example 7.8**

Consider the two-stage tandem queue in Example 7.6; then.

\[
A_{12} = (h_2 - h_1)\bar{S}_2 + h_2\bar{S}_1
\]

\[
= 18
\]

\[
K_{12} = K_1 + K_2
\]

\[
= 17
\]

and using (7.20), we obtain

\[
C[\overset{2}{\circ} (1 \overset{2}{\rightarrow} 2)] - C[\overset{3}{\circ} (1 \overset{4}{\rightarrow} 2)] = -4(18) + 1(17)
\]

\[
= -55
\]

\[
C[(1 \overset{2}{\rightarrow} 2) \overset{3}{\circ} (1 \overset{1}{\rightarrow} 2)] - C[\overset{4}{\circ} (1 \overset{4}{\rightarrow} 2)] = -5(18) + 2(17)
\]

\[
= -56
\]

\[
C[\overset{4}{\circ} (1 \overset{1}{\rightarrow} 2)] - C[\overset{4}{\circ} (1 \overset{4}{\rightarrow} 2)] = -6(18) + 3(17)
\]

\[
= -57.
\]

Now, comparing these results, it is clear that policy \( \overset{4}{\circ} (1 \overset{1}{\rightarrow} 2) \) is the best greedy and exhaustive substitute policy for \( (1 \overset{4}{\rightarrow} 2) \). \( \square \)
Lemma 7.3

Consider policy \( \gamma \equiv (i \triangleright j) \) and the least cost policy of order \( l \), \( \xi_{LC}^{(i)} \in \mathcal{E}_l(\gamma) \), with parameters \((m_1^*, m_2^*, \ldots, m_l^*)\). If \( A_{ij} > 0 \) and \( m = \mathfrak{M}[\frac{m}{l}] l + u \). then.

\[
m_i^* = \begin{cases} 
\mathfrak{M}[\frac{m}{l}] + 1 & : i = 1, 2, \ldots, u \\
\mathfrak{M}[\frac{m}{l}] & : i = u + 1, u + 2, \ldots, l 
\end{cases}
\]  \hspace{1cm} (7.31)

where \( \mathfrak{M}[\frac{m}{l}] \) is the greatest integer less than \( \frac{m}{l} \).

**Proof**

Suppose policy \( \xi^{(i)} \in \mathcal{E}_l(\gamma) \) has parameters \((n_1, n_2, \ldots, n_l)\), where \( n_i = m_i^* : i = 1, 2, \ldots, l - 2 \) and \( n_{l-1} + n_l = m_{l-1}^* + m_l^* \). Then, using (7.22), we will have,

\[
C[\xi_{LC}^{(i)}] - C[\xi^{(i)}] = [\varphi_l(n) - \varphi_l(m)]A_{ij}
= n_{l-1}n_l - m_{l-1}^*m_l^*.
\]

Since \( A_{ij} > 0 \) and \( \xi_{LC}^{(i)} \) is the least cost policy of order \( l \), then for all \( n_{l-1}, n_l \) we must have

\[
n_{l-1}n_l - m_{l-1}^*m_l^* \leq 0  \hspace{1cm} (7.32)
\]

If \( m_{l-1}^* + m_l^* \) is an even number, then \( m_{l-1}^* = m_l^* = \frac{m_{l-1}^* + m_l^*}{2} \) guarantee (7.32), and if \( m_{l-1}^* + m_l^* \) is an odd number, then \( m_{l-1}^* = \mathfrak{M}[\frac{m_{l-1}^* + m_l^*}{2}] \) and \( m_l^* = \mathfrak{M}[\frac{m_{l-1}^* + m_l^*}{2}] + 1 \) guarantee that \( m_{l-1}^* \) and \( m_l^* \) are the parameters of the least cost policy of order \( l \). In other words, \( m_{l-1}^* \) and \( m_l^* \) are the parameters of the least cost policy if the difference between them is 1 or 0.

The actual sequencing of the parameters \((m_1^*, m_2^*, \ldots, m_l^*)\) is not important in \( C[\xi_{LC}^{(i)}] \). Hence, for \( \xi_{LC}^{(i)} \) with parameters \((m_1^*, m_2^*, \ldots, m_{l-3}^*, m_{l-1}^*, m_{l-2}^*, m_l^*)\), and \( \xi^{(i)} \), which now has the parameter \((n_1, n_2, \ldots, n_{l-3}, n_{l-1}, n_{l-2}, n_l)\), where \( n_i = m_i^* : i = 1, 2, \ldots, l - 3 \), \( n_{l-1} = m_{l-1}^* \) and \( n_{l-2} + n_l = m_{l-2}^* + m_l^* \), we can conclude that in \( \xi_{LC}^{(i)} \) the difference between \( m_{l-2}^* \) and \( m_l^* \) must be zero (if \( m_{l-2}^* + m_l^* \) is even) or one (if \( m_{l-2}^* + m_l^* \) is odd). Iterating this procedure we find that the difference between \( m_j^* \) and \( m_j^* \) (\( \forall i, j = 1, 2, \ldots, l \)) must be 0 (when \( m_j^* + m_j^* \) is even) or 1 (when \( m_j^* + m_j^* \) is odd). Thus, \( m_j^* \) (\( j = 1, 2, \ldots, l \)) can be obtained from (7.31), because (7.31) is
the only way to guarantee that \( m_i^* \) and \( m_j^* \) have this property, while maintaining \( \sum_{i=1}^{l} m_i^* = m \). \( \Box \)

Lemma 7.3 actually introduces an easy approach to find the least cost policy of a specific order instead of computing the total average cost of all greedy and exhaustive substitute policies of that order.

**Example 7.9**

Consider policy \( \gamma \equiv (2 \uparrow 6) \) in a 2 \( \uparrow 6 \) queue. Policy \( \gamma \) has 36 greedy and exhaustive substitute policies of order 3 with parameters (1.1.8), (1.2.7), (1,3,6)\( \ldots \). Instead of comparing the total average costs of these policies, using Lemma 7.3, since 10 = \( \exists [10/3]3 + 1 \), then the least cost policy of order 3 has parameter (4.3.3). \( \Box \)

Lemma 7.4 introduces a simple approach to find the order of the best greedy and exhaustive substitute policy for any policy \( (i \uparrow m \uparrow j) \), using holding and switching cost factors.

**Lemma 7.4**

Let \( (n_1^*, n_2^*) \) be the parameters of \( \xi^{(2)}_{LC} \), and let cost ratio factor \( R_{ij} \) be defined by

\[
R_{ij} = \frac{\kappa_{ij}}{A_{ij}}.
\]  

Then, the value of \( \Omega^*(i \uparrow m \uparrow j) \) is

(i) 1, if \( A_{ij} \leq 0 \).

(ii) 1, if \( A_{ij} > 0 \) and \( n_1^* n_2^* \leq R_{ij} \),

(iii) obtained from Table 7.2, if \( A_{ij} > 0 \) and \( n_1^* n_2^* > R_{ij} \). Table 7.2 is for \( 1 \leq m \leq 10 \) and \( 0 \leq R_{ij} \leq 13 \) and \( R_k \) indicates the real numbers in interval \([k, k + 1]\).
Table 7.2. $\Omega^*(i \succ j)$ for $R_i \in R_c$.

<table>
<thead>
<tr>
<th>$R_i$</th>
<th>$R_0$</th>
<th>$R_1$</th>
<th>$R_2$</th>
<th>$R_3$</th>
<th>$R_4$</th>
<th>$R_5$</th>
<th>$R_6$</th>
<th>$R_7$</th>
<th>$R_8$</th>
<th>$R_9$</th>
<th>$R_{10}$</th>
<th>$R_{11}$</th>
<th>$R_{12}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>5</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>6</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>7</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>8</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>9</td>
<td>9</td>
<td>9</td>
<td>5</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>10</td>
<td>5</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>

Proof

(i) Consider (7.22) for $\forall \xi^{(i)} \in E_i((i \succ j)$ for $l = 2, 3, \ldots, m$. We then have

$$C[\xi^{(i)}] - C[i \succ j] = -\phi_l(m)A_{ij} + (l - 1)K_{ij}.$$ 

Therefore, if $A_{ij} \leq 0$, and since $\phi_l(m) > 0$ and $K_{ij} \geq 0$, it follows that.

$$C[\xi^{(i)}] - C[i \succ j] \geq 0,$$

and consequently, policy $(i \succ j)$ has the least total average cost among its greedy and exhaustive substitute policies of any order. In other words, $\Omega^*(i \succ j) = 1$.

(ii) To start the proof, we first need to show that if $(i \succ j) \xleftarrow{\xi_L^{(l)}} \xi^{(l)}_{LC}$ and $n_1^*n_2^* \leq R_{ij}$, then $(i \succ j) \xleftarrow{\xi^{(l-1)}} \xi^{(l-1)}_{LC}$. In other words, we want to show that if there is a better substitute of order $l$ for $(i \succ j)$ in $E_i(i \succ j)$, then there will also be a better substitute of order $(l - 1)$ for $(i \succ j)$ in $E_{i-1}(i \succ j)$. Since $(i \succ j) \xleftarrow{\xi^{(l)}} \xi^{(l)}_{LC}$, then for $\xi^{(l)}_{LC}$ with parameters $(m_1^*, m_2^*, \ldots, m_l^*)$, we have

$$C[\xi^{(l)}_{LC}] - C[i \succ j] \leq 0.$$  \hspace{1cm} (7.34)

Suppose policy $\xi^{(l-1)} \in E_{i-1}(i \succ j)$ has parameters $(m_1^*, m_2^*, \ldots, m_{l-2}^*, m_{l-1}^* + m_1^*)$; then, using (7.22) we get

$$C[\xi^{(l-1)}_{LC}] = C[\xi^{(l-1)}] - m_{l-1}^*A_{ij} + K_{ij},$$ \hspace{1cm} (7.35)

which upon substitution into (7.34) yields

$$C[\xi^{(l-1)}] - C[i \succ j] \leq m_{l-1}^*A_{ij} - K_{ij}. $$ \hspace{1cm} (7.36)
On the other hand, since $n^*_1 n^*_2 \leq R_{ij}$, and considering (7.33), we will have

\[ n^*_1 n^*_2 A_{ij} - \kappa_{ij} \leq 0. \tag{7.37} \]

Since $m_{i-1}^* m_i^* < n^*_1 n^*_2$ and $A_{ij} > 0$, then (7.37) also holds when $m_{i-1}^* m_i^*$ is used instead of $n^*_1 n^*_2$. Therefore,

\[ m_{i-1}^* m_i^* A_{ij} - \kappa_{ij} \leq 0. \tag{7.38} \]

and from (7.38) and (7.36) we deduce that

\[ C[\xi^{(l-1)}] - C[i \triangleright j] \leq 0. \tag{7.39} \]

Since $C[\xi^{(l-1)}] \leq C[\xi^{(l-1)}]$, therefore, using (7.39), we will have $C[\xi^{(l-1)}] - C[i \triangleright j] \leq 0$. or $(i \triangleright j) \leftarrow^{(l-1)} \xi^{(l-1)}_{LC}$.

To complete the proof of part (ii), we use a method of contradiction. Suppose $A_{ij} > 0$ and $n^*_1 n^*_2 \leq R_{ij}$, but there is a better substitute policy of order $r$, $\xi^{(r)}_{LC} \in E_r(i \triangleright j)$, for policy $(i \triangleright j)$. Therefore, since $(i \triangleright j) \leftarrow^{(r-1)} \xi^{(r)}_{LC}$, we can conclude that $(i \triangleright j) \leftarrow^{(r-1)} \xi^{(r-1)}_{LC}$. Iterating this procedure, we conclude that $(i \triangleright j) \leftarrow^2 \xi^{(2)}_{LC}$. However, the parameters of $\xi^{(2)}_{LC}$ are $(n^*_1, n^*_2)$, and according to $(i \triangleright j) \leftarrow^2 \xi^{(2)}_{LC}$, we will have

\[ C[\xi^{(2)}_{LC}] - C[i \triangleright j] = n^*_1 n^*_2 A_{ij} - \kappa_{ij} \leq 0. \]

or $n^*_1 n^*_2 \geq R_{ij}$, which violates the initial assumption. Hence, the proof of part (ii) is now complete.

(iii) If $A_{ij} > 0$ and $n^*_1 n^*_2 > R_{ij}$, then it is clear that there is at least one better greedy and exhaustive substitute policy for policy $(i \triangleright j)$ in $E(i \triangleright j)$ (which is $\xi^{(2)}_{LC}$). Therefore, to obtain Table 7.2, the least cost substitute of all orders must be compared to find the best substitute greedy and exhaustive policy for policy $(i \triangleright j)$ related to each interval $R_k$. As an example, let us obtain the row of Table 7.2 for $m = 6$. Using Lemma 7.3 for $\xi^{(1)}_{LC} \in E(i \triangleright j)$, we see that $\xi^{(2)}_{LC}, \xi^{(3)}_{LC}$ and $\xi^{(4)}_{LC}$ have parameters $(3,3), (2,2,2)$ and $(2,2,1,1)$, respectively. Considering part (ii) of Lemma 7.4, we
will have for $R_{ij} \geq 3(3) = 9$ that $\Omega^*(i \overset{6}{\triangleright} j) = 1$. Comparing $\xi^{(2)}_{LC}$ and $\xi^{(3)}_{LC}$.

$$C[\xi]^{(2)}_{LC} - C[\xi]^{(3)}_{LC} = [\varphi^*_3(6) - \varphi^*_2(6)]A_{ij} - K_{ij}$$

$$= (12 - 9)A_{ij} - K_{ij}.$$ 

Then, if $A_{ij} - K_{ij} \leq 0$ [i.e. if $R_{ij} \geq 3$], policy $\xi^{(2)}_{LC}$ is better than $\xi^{(3)}_{LC}$. To compare $\xi^{(3)}_{LC}$ and $\xi^{(4)}_{LC}$, we have

$$C[\xi]^{(3)}_{LC} - C[\xi]^{(4)}_{LC} = [\varphi^*_4(6) - \varphi^*_3(6)]A_{ij} - K_{ij}$$

$$= (13 - 12)A_{ij} - K_{ij}.$$ 

and therefore, if $A_{ij} - K_{ij} \leq 0$ [or $R_{ij} \geq 1$], then policy $\xi^{(3)}_{LC}$ is better than $\xi^{(4)}_{LC}$. Thus,

$$\Omega^*(i \overset{6}{\triangleright} j) = \begin{cases} 
3 & : 1 \leq R_{ij} \leq 3 \\
2 & : 3 \leq R_{ij} \leq 9 \\
1 & : 9 \leq R_{ij} \end{cases} \quad (7.40)$$

It is not necessary to compare $\xi^{(4)}_{LC}$, $\xi^{(5)}_{LC}$ and $\xi^{(6)}_{LC}$ with parameters $(2, 2, 1, 1)$, $(2, 1, 1, 1, 1)$ and $(1, 1, 1, 1, 1, 1)$, respectively, because, using Table 7.2 for $m = 2$ and $R_{ij} \leq 1$, it is clear that $(i \overset{2}{\triangleright} j) \overset{3}{\leftarrow} (i \overset{1}{\triangleright} j)$. Therefore, it is also clear that policy $\xi^{(6)}_{LC}$ which does not have any parameter value equal to 2 is better than $\xi^{(4)}_{LC}$ and $\xi^{(5)}_{LC}$ in region $R_{ij} \leq 1$. In other words, $\Omega^*(i \overset{1}{\triangleright} j) = 6$ when $0 \leq R_{ij} \leq 1$. 

Lemma 7.4 actually introduces a simple approach to find the best greedy and exhaustive substitute policy for any policy $(i \overset{m}{\triangleright} j)$, using the ratio cost factors. The following example shows how efficient and time saving this approach can be.

**Example 7.10**

For policy $\gamma$ in Example 7.9, the least cost policy of order $l = 1, 2, \ldots, 10$ has parameters

$$(1, 1, 1, 1, 1, 1, 1, 1, 1, 1) \quad (2, 1, 1, 1, 1, 1, 1, 1, 1)$$

$$(2, 2, 1, 1, 1, 1, 1) \quad (2, 2, 2, 1, 1, 1) \quad (2, 2, 2, 2, 1, 1)$$

$$(2, 2, 2, 2) \quad (3, 3, 2, 2) \quad (3, 3, 4) \quad (5, 5) \quad (10).$$
If cost ratio factor $R_{26} = 4.36$, then using Table 7.2, $\Omega^*(2 ^{10} 6) = 3$, and therefore

$$(2 ^{10} 6) \iff (2 ^{4} 6) \iff (2 ^{3} 6).$$

Based on the results of Lemmas 7.2, 7.3, and 7.4, two heuristic algorithms $AND$ and $SHORA$ are presented in the next sections. Both algorithms use the properties of greedy and exhaustive policies in stages $i$ to $j$, but the difference is the way that these algorithms decompose the $N$-stage tandem queue into tandem queues $i \uparrow j$.

### 7.6 $AND$ Heuristic Algorithm

The *Adjointed N-stage Decomposition* (or $AND$) algorithm decomposes the $N$-stage tandem queue attended by a moving server into $N$ multi-stage tandem queues $1 \uparrow N$, $2 \uparrow N$, ..., and $(N - 1) \uparrow N$, so that, stage $i$ appears in $i$ decomposed systems $i = 1, 2, \ldots, N - 1$ (Figure 7.3).

For each batch of $m$ customers (jobs), the $AND$ algorithm starts with the first decomposed system and finds $\Omega^*(1 ^{m} N)$. Suppose $(1 ^{m} N) \iff \gamma_{1N}$ and $\gamma_{1N}$ has parameters $(m_1^{(1)}, m_1^{(1)}, \ldots, m_i^{(1)})$; then the $AND$ algorithm applies policy $(1 ^{m_1^{(1)}} 1)$ and uses the second decomposed system to find the parameters of $\gamma_{2N}$, where $(2 ^{m_1^{(1)}} N) \iff \gamma_{2N}$. Writing $(m_1^{(2)}, m_1^{(2)}, \ldots, m_i^{(2)})$ as the parameters of $\gamma_{2N}$, the $AND$
algorithm applies policy \(2 \uparrow 2\) and uses the third decomposed system to find \(\gamma_{3N}\), where \(3 \uparrow N \leftarrow \gamma_{3N}\). The AND algorithm continues this process until \(m^{(N-1)}_1\) customers are served exhaustively in stage \(N\), and then it switches to the nearest nonempty stage to stage \(N\) to repeat this process for the remaining customers until stages \(1, 2, \ldots, N\) all become empty. According to the AND algorithm, if \(\Omega^*(i \uparrow N) = 1\) for every \(i = 1, 2, \ldots, N - 1\), then the AND algorithm yields the greedy and exhaustive policy downstream of stage 1.

The AND algorithm can be summarized in the following steps:

**AND Algorithm For**

**an N-stage Tandem Queue with Input Batch of Size \(m\)**

- **Step 1:** If the number of customers (jobs) in the batch is \(m = 1\), then the only policy which can be used is \(\beta^1 \equiv (2 \uparrow N)\). Terminate the algorithm.

- **Step 2:** Set \(EC_0 = 0\) and find \(A_{iN}, K_{iN}\) and also compute \(R_{iN}\) (if \(A_{iN} \geq 0\)), for \(i = 1, 2, \ldots, N - 1\). from

\[
A_{iN} = \sum_{k=i+1}^{N} (h_k - h_i) \bar{S}_k + \sum_{k=i+1}^{N} h_k \bar{S}_{k-1}
\]

\[
K_{iN} = \sum_{k=i}^{N} K_k
\]

\[
R_{iN} = \frac{K_{iN}}{A_{iN}}
\]

- **Step 3:** If \(m \geq 2\), set \(i = 1, j = 1\), and also set vector \(N_j = (n_1, n_2, \ldots, n_N)\) as \(N_j = (m, 0, 0, \ldots, 0)\), where \(n_i\) is the number of customers in stage \(i\) (\(i = 1, 2, \ldots, N\)). Also set \(U = (u_1, u_2, \ldots, u_N)\), where \(u_i = 0; \forall i = 1, 2, \ldots, N\) and \(U^c = (u_1^c, u_2^c, \ldots, u_N^c)\), where \(u_i^c = 0; \forall i = 1, 2, \ldots, N\).

- **Step 4:** If \(A_{iN} > 0\), go to step 5; otherwise, set \(\beta_j \equiv (i \uparrow m_i)\), \(U = 0\), \(u_i = -m\), \(u_{i+1} = m\), \(U^c = 0\), \(u_i^c = \frac{m-1}{2}\), \(u_{i+1}^c = \frac{m-1}{2}\) and set

\[
N_{j+1} = N_j + U
\]
and go to step 6.

- **Step 5:** Considering \( \mathcal{R}_{i,N} \) and \( n_i \), find \( \Omega^*(i^\mathcal{R}_{i,N}) = l^* \) from Table 7.2. Then, considering the largest parameters of policy \( \xi_{LC}^{(i^*)} \in \mathcal{E}_l(i^\mathcal{R}_{i,N}) \) as \( m^* \), set \( \beta_j \equiv (i^\mathcal{R}_{i,N} \triangleright i) \). \( U' = 0 \). \( u_i = -m^* \). \( u_{i+1} = m^* \). \( U^c = 0 \). \( u_i^c = -\frac{m^* - 1}{2} \). \( u_{i+1}^c = \frac{m^* - 1}{2} \) and set

\[
N_{j+1}^c = N_j + U
\]
\[
N_{j+1} = N_j + U^c
\]
\[
EC_j = EC_{j-1} + m^* N_{j+1}^c h^T S_j + K_{i+1}
\]

- **Step 6:** Set \( i = i + 1 \) and \( j = j + 1 \). If \( i < N \) return to step 4; however, if \( i = N \), then set \( \beta_j \equiv (N^\triangleright N) \). \( U = 0 \). \( u_N = -n_N \). \( U^c = 0 \). \( u_{N^c} = -\frac{n_N - 1}{2} \) and

\[
N_{j+1} = N_j + U
\]
\[
N_{j+1}^c = N_j + U^c
\]
\[
EC_j = EC_{j-1} + n_N N_{j+1}^c h^T S_j
\]

- **Step 7:** If \( N_j^m = (0,0,\ldots,0) \), terminate the algorithm. the suboptimal policy is \( \beta \equiv \beta_1 \cap \beta_2 \cap \ldots \cap \beta_j \), with the average cost \( EC_j \). If \( N_j \neq 0 \), then considering \( N_j \), set \( i \) equal to the index of the closest nonempty stage to stage \( N \), and set \( EC_j = EC_j + K_i \) and \( j = j + 1 \); return to step 4.

**Example 7.11**

Consider a 4-stage tandem queue with input batch of size \( m = 4 \) and

\[
h = (72, 77, 87, 94)
\]
\[
S = (2, 3, 8, 4)
\]
\[
K = (496, 771, 467, 205)
\]

The \( AND \) algorithm can be applied as follows:
Step 2:

\[ \mathcal{A}_{14} = 1390 \quad \kappa_{14} = 1939 \quad \mathcal{R}_{14} = 1.39 \]
\[ \mathcal{A}_{24} = 1161 \quad \kappa_{24} = 1443 \quad \mathcal{R}_{24} = 1.24 \]
\[ \mathcal{A}_{34} = 780 \quad \kappa_{34} = 672 \quad \mathcal{R}_{34} = 0.86 \]

Step 3: \( i = 1, \ j = 1 \). \( N_1 = (4.0.0.0) \), \( \mathcal{U} = (0.0.0.0) \), \( \xi^c = (0.0.0.0) \).

Step 4: Since \( \mathcal{A}_{14} > 0 \), go to step 5.

Step 5: Since \( \mathcal{R}_{14} = 1.39 \) and \( n_1 = 4 \), then using Table 7.2, \( \Omega^*(1 \searrow 4) = 2 \). The largest parameter of \( \xi_{LC}^{(2)} \in \xi(1 \searrow 4) \) is 2. therefore, \( J_1 \equiv (1 \searrow 1) \) and \( u_1 = -2, u_2 = 2, u_1^c = -\frac{1}{2}, u_2^c = \frac{1}{2} \) and

\[ N_2 = (4.0.0.0) + (-2.0.0.0) = (2.2.0.0) \]
\[ N_2^c = (4.0.0.0) + (-\frac{1}{2}, \frac{1}{2}, 0.0) = (3.5.0.5.0.0) \]
\[ EC_1 = 0 + 2(3.5.0.5.0.0)(72.77.87.94)^T = 2 + 771 = 1933 \]

Step 6: \( i = 2 \) and \( j = 2 \). Since \( i < 4 \), return to step 4.

Step 4: Since \( \mathcal{A}_{24} > 0 \), go to step 5.

Step 5: Since \( \mathcal{R}_{24} = 1.24 \) and \( n_2 = 2 \), then using Table 7.2, \( \Omega^*(2 \searrow 4) = 1 \); therefore, \( J_2 \equiv (2 \searrow 2) \) and

\[ N_3 = (2.2.0.0) + (0, -2.0.0) = (2.0.2.0) \]
\[ N_3^c = (2.2.0.0) + (0, -\frac{1}{2}, \frac{1}{2}, 0.0) = (2.1.5.0.5.0) \]
\[ EC_2 = 1933 + 2(2.1.5.0.5.0)(72.77.87.94)^T3 + 467 = 4218 \]

Step 6: \( i = 3 \) and \( j = 3 \). Since \( i < 4 \), return to step 4.

Step 4: Since \( \mathcal{A}_{34} > 0 \), go to step 5.

Step 5: Since \( \mathcal{R}_{34} = 0.86 \) and \( n_3 = 2 \), then using Table 7.2, \( \Omega^*(3 \searrow 4) = 2 \); therefore, \( J_3 \equiv (3 \searrow 3) \) and

\[ N_4 = (2.0.2.0) + (0.0, -1.1) = (2.0, 1, 1) \]
\[ N_4^c = (2.0.2.0) + (0.0, 0.0) = (2.0, 2.0) \]
\[ EC_3 = 4218 + 1(2.0.2.0)(72.77.87.94)^T8 + 205 = 6967 \]
Step 6: $i = 4$ and $j = 4$. Since $i = 4$, then $J_4 \equiv (4 \triangleright 4)$ and

\[
    \begin{align*}
    \mathcal{N}_5 &= (2.0.1.1) + (0.0.0.0) = (2.0.1.0) \\
    \mathcal{N}_5^x &= (2.0.1.1) + (0.0.0.0) = (2.0.1.1) \\
    EC_4 &= 6967 + 1 (2.0.1.1)(72.77.87.94)^T 4 = 8267
    \end{align*}
\]

Step 7: Since $\mathcal{N}_5 \neq 0$, then $i = 3$. $EC_4 = 8267 + 467 = 8734$. $j = 5$ and return to step 4.

Step 4: Since $\mathcal{N}_{34} > 0$, go to step 5.

Step 5: Since $\mathcal{R}_{34} = 0.86$ and $n_3 = 1$, then $J_3 \equiv (3 \triangleright 3)$ and

\[
    \begin{align*}
    \mathcal{N}_6 &= (2.0.1.0) + (0.0.0.0) = (2.0.1.0) \\
    \mathcal{N}_6^x &= (2.0.1.0) + (0.0.0.0) = (2.0.1.0) \\
    EC_5 &= 8734 + 1 (2.0.1.0)(72.77.87.94)^T 8 + 205 = 10787
    \end{align*}
\]

Step 6: $i = 4$ and $j = 6$. Since $i = 4$, then $J_6 \equiv (4 \triangleright 4)$ and

\[
    \begin{align*}
    \mathcal{N}_7 &= (2.0.0.1) + (0.0.0.0) = (2.0.0.0) \\
    \mathcal{N}_7^x &= (2.0.0.1) + (0.0.0.0) = (2.0.0.1) \\
    EC_6 &= 10787 + 1 (2.0.1.1)(72.77.87.94)^T 4 = 11739
    \end{align*}
\]

Step 7: Since $\mathcal{N}_7 \neq 0$, then $i = 1$. $EC_6 = 11739 + 496 = 12235$. $j = 7$ and return to step 4.

The remaining part of the algorithm is omitted and only the results are given as follows:

\[
    \begin{align*}
    \beta_7 &\equiv (1 \triangleright 1) & \beta_8 &\equiv (2 \triangleright 2) & \beta_9 &\equiv (3 \triangleright 3) \\
    \beta_{10} &\equiv (4 \triangleright 4) & \beta_{11} &\equiv (3 \triangleright 3) & \beta_{12} &\equiv (4 \triangleright 4)
    \end{align*}
\]

with the total average cost $EC_{12} = 19078$. Hence the policy that the AND algorithm yields is

\[
    \mathcal{Z} \equiv \bigcirc_{i=1}^{12} \beta_i
\]

\[
    \equiv \bigcirc \left[ (1 \triangleright 2) \bigcirc (3 \triangleright 4) \right]
\]

which is actually the optimal policy. \qed
7.7 SHORA Heuristic Algorithm

The shortest Route Approach algorithm decomposes the $N$-stage tandem queue into all possible multi-stage tandem queues in which a greedy and exhaustive policy can be used. The set of all possible multi-stage tandem queues consists of: $N$ one-stage tandem queues, $(N-1)$ two-stage tandem queues, $(N-2)$ three-stage tandem queues, .... and 1 N-stage tandem queue. Matrix $Dec(N)$ shows all possible decomposed tandem queues for the $N$-stage tandem queue.

\[
Dec(N) = \begin{bmatrix}
1 & 1 & 1 & 1 & 1 & 1 & \ldots & 1 & 1 & N-1 & N \\
0 & 2 & 2 & 2 & 3 & 2 & \ldots & 2 & N-1 & 2 & N \\
0 & 0 & 3 & 3 & 3 & 4 & \ldots & 3 & N-1 & 3 & N \\
0 & 0 & 0 & 4 & 4 & 4 & \ldots & 4 & N-1 & 4 & N \\
0 & 0 & 0 & 0 & 5 & 5 & \ldots & 5 & N-1 & 5 & N \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & \ldots & N-1 & N-1 & N-1 & N-1 & N & N \\
0 & 0 & 0 & 0 & \ldots & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{bmatrix}
\] (7.41)

Since $Dec(N)$ is an upper triangular matrix, it is clear that the number of decomposed systems (the nonzero elements of $Dec(N)$) is $\frac{N(N+1)}{2}$.

For each $m$, the SHORA algorithm finds the best greedy and exhaustive substitute policy, $\beta_{ij}^m$, for policy $(i \geq j)$ in the $i \uparrow j$ tandem queue. Considering $\Omega^*(\beta_{ij}^m) = I_{ij}^m$, the SHORA algorithm generates matrices $Pol(m)$ and $Eco(m)$ as follows:

\[
Pol(m) = \begin{bmatrix}
1 & I_{12} & I_{13} & I_{14} & \ldots & I_{1,N-1} & I_{1,N} \\
0 & I_{23} & I_{24} & \ldots & I_{2,N-1} & I_{2,N} \\
0 & 0 & 1 & I_{34} & \ldots & I_{3,N-1} & I_{3,N} \\
0 & 0 & 0 & 1 & \ldots & I_{4,N-1} & I_{4,N} \\
0 & 0 & 0 & 0 & \ldots & I_{5,N-1} & I_{5,N} \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & 0 & 0 & \ldots & 1 & I_{N-1,N} \\
0 & 0 & 0 & 0 & \ldots & 0 & 1 \\
\end{bmatrix}
\] (7.42)

\[
Eco(m) = \begin{bmatrix}
0 & C'[\beta_{12}^m] & C'[\beta_{13}^m] & C'[\beta_{14}^m] & \ldots & C'[\beta_{1,N-1}^m] & C'[\beta_{1,N+1}^m] \\
0 & 0 & C'[\beta_{23}^m] & C'[\beta_{24}^m] & \ldots & C'[\beta_{2,N-1}^m] & C'[\beta_{2,N+1}^m] \\
0 & 0 & 0 & C'[\beta_{34}^m] & \ldots & C'[\beta_{3,N-1}^m] & C'[\beta_{3,N+1}^m] \\
0 & 0 & 0 & 0 & \ldots & C'[\beta_{4,N-1}^m] & C'[\beta_{4,N+1}^m] \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & 0 & 0 & \ldots & 0 & C'[\beta_{N,N+1}^m] \\
\end{bmatrix}
\] (7.43)
where $C'[\beta_{i,j}^m] = C[\beta_{i,j}^m] + K_{j+1}$ and $K_{N+1} = 0$. Thus, $C'[\beta_{i,j}^m]$ is the total average holding and switching cost of serving $m$ customers according to policy $\beta_{i,j}^m$ in tandem queue $i \uparrow j$ and then switching from stage $j$ to $j+1$. In other words, $C'[\beta_{i,j}^m]$ is the average holding and switching cost during the time that server starts from stage $i$ to the time that he completes the services of all $m$ customers in stage $j$ and switches to stage $j+1$. Therefore, we can assume that when the server moves from stage $i$ to $j$ and then to $j+1$ according to policy $\beta_{i,j}^m$, the total average cost $C'[\beta_{i,j}^m]$ is incurred. From this perspective, the minimization of the total average cost of serving $m$ customers in stages 1 to $N$ using greedy and exhaustive substitute policies, is equivalent to the minimization of the total average cost of the server’s trip from stage 1 to $N+1$. This is actually a shortest-route problem in which the origin is stage 1, the destination is stage $N+1$ and the distance between nodes (stages) $i$ and $j$ is $C'[\beta_{i,j}^m]$ (Figure 7.4).

Dynamic programming techniques can be used to find the least cost path from node 1 to node $N+1$. To find the best greedy and exhaustive substitute policy $\beta_{i,j}^m$ for policy $(i \rightarrow j)$ in each tandem queue $i \uparrow j$, Lemma 7.4 can be used. The SHORA algorithm can be summarized in the following steps:
SHORA Algorithm For
an N-stage Tandem Queue with Input Batch of Size \( m \)

- **Step 1** Construct matrices \( A' \) and \( K' \) with elements \( A'_{ij} \) and \( K'_{ij} \) (\( i, j = 1, 2, \ldots, N \)). respectively. where

\[
A'_{ij} = \begin{cases} A_{ij} & : i < j \\ 0 & : i \geq j \end{cases} \tag{7.44}
\]

\[
K'_{ij} = \begin{cases} K_{ij} & : i < j \\ 0 & : i \geq j \end{cases} \tag{7.45}
\]

- **Step 2** Construct matrix \( X^{(m)} \) with elements \( x^{(m)}_{ij} \) (\( i, j = 1, 2, \ldots, N \)), where

\[
x^{(m)}_{ij} = \begin{cases} \mathcal{C}[i \triangleright j] & : i \leq j \\ 0 & : i > j \end{cases} \tag{7.46}
\]

- **Step 3** For each tandem queue \( i \triangleleft j \) (\( i < j \)) in Dec(\( N \)), find the best greedy and exhaustive substitute policy \( \beta^{m}_{ij} \) with order \( l^{(m)}_{ij} = \Omega^{*}(i \triangleright j) \), and then construct matrix \( Pol^{(m)} \) with elements \( pol^{(m)}_{ij} \), (\( i, j = 1, 2, \ldots, N \)), where

\[
pol^{(m)}_{ij} = \begin{cases} 1 & : i = j \\ l^{(m)}_{ij} & : i < j \\ 0 & : i > j \end{cases} \tag{7.47}
\]

- **Step 4** Construct matrix \( Eco^{(m)} \) with elements \( e^{(m)}_{ij} \) (\( i = 1, 2, \ldots, N \); \( j = 2, 3, \ldots, N + 1 \)), where

\[
e^{(m)}_{ij} = \begin{cases} x^{(m)}_{ij-1} + K_j & : l^{(m)}_{i,j-1} = 1 \\ x^{(m)}_{ij-1} - \Phi^{*}_{i,j-1}(m)A_{i,j-1} + K_j & : l^{(m)}_{i,j-1} \geq 2 \\ 0 & : l^{(m)}_{i,j-1} = 0 \end{cases} \tag{7.48}
\]

and \( \Phi^{*}_{i,j-1}(m) \) is the value of function \( \varphi_{i,j-1}(m) \) for the parameters of policy \( \pi \), for which \( i \triangleright j - 1 \) \( \frac{w^{(m)}_{i,j-1}}{\pi} \). for

- **Step 5** Consider element \( e^{(m)}_{ij} \) as the distance between nodes \( i \) and \( j \) in a shortest route problem, and find the shortest route between nodes \( i \) and \( N + 1 \). Suppose that the shortest route passes through nodes \( 1, i_1, i_2, \ldots, i_r, N + 1 \) with total distance \( TD^{(m)} \). Then the suboptimal policy is \( \beta^{m} = \beta_{1,i_1-1} \bigcirc \beta_{i_1,i_2-1} \bigcirc \cdots \bigcirc \beta_{i_r,N} \) with total average holding and switching cost \( TD^{(m)} \).
Example 7.12

Consider the 4-stage tandem queue on Example 7.11. The _SHORA_ algorithm can be applied for this problem as follows:

Step 1: Matrices $\mathbf{A}'$ and $\mathbf{K}'$ are

$$
\mathbf{A}' = \begin{pmatrix}
0 & -5 & 80 & 1390 \\
0 & 0 & 59 & 1161 \\
0 & 0 & 0 & 780 \\
0 & 0 & 0 & 0
\end{pmatrix}
\mathbf{K}' = \begin{pmatrix}
0 & 1267 & 1734 & 1939 \\
0 & 0 & 1238 & 1443 \\
0 & 0 & 0 & 672 \\
0 & 0 & 0 & 0
\end{pmatrix}
$$

Step 2: Matrix $\mathbf{X}^{(4)}$ is

$$
\mathbf{X}^{(4)} = \begin{pmatrix}
2364 & 7011 & 18950 & 22915 \\
0 & 3876 & 15815 & 19780 \\
0 & 0 & 11472 & 15437 \\
0 & 0 & 0 & 3760
\end{pmatrix}
$$

Step 3: Matrix $\mathbf{Pol}^{(4)}$ is

$$
\mathbf{Pol}^{(4)} = \begin{pmatrix}
1 & 1 & 1 & 2 \\
0 & 1 & 1 & 2 \\
0 & 0 & 1 & 4 \\
0 & 0 & 0 & 1
\end{pmatrix}
$$

Step 4: Matrix $\mathbf{Eco}^{(4)}$ is constructed as follows:

$$
\mathbf{Eco}^{(4)} = \begin{pmatrix}
0 & 3135 & 7478 & 19155 & 19294 \\
0 & 0 & 4343 & 16020 & 16579 \\
0 & 0 & 0 & 11677 & 12773 \\
0 & 0 & 0 & 0 & 3760
\end{pmatrix}
$$

Step 5: Considering element $e_{ij}^{(4)}$ of matrix $\mathbf{Eco}^{(4)}$ as the distance between nodes $i$ and $j$, and applying dynamic programming for the shortest route problem (Figure 7.5), the optimal route is found which passes through nodes 1 to $N + 1$ directly. Therefore, the suboptimal policy $\beta \equiv \beta_{1,N} \equiv \odot (1 \odot 4)$ with total average cost 19294. □

In this example, the _SHORA_ algorithm actually approximates a shortest route problem with 140 nodes by a shortest route problem with 5 nodes. Policy $\beta_{1,N}$ is not the optimal policy, but its cost is only around 1% more than the optimal cost.
Remark 7.1

Suppose that the N-stage tandem queue has a buffer after stage N with holding cost \( h_{N+1} \geq h_N \), where finished jobs of a batch are stored until the completion of the last job in that batch. In other words, a batch (order) is considered to be complete when all jobs in that batch (order) are completed. Now, if we decompose the N-stage tandem queue into \((N-1)\) two-stage tandem queues \( i \mapsto i+1 \) \((i = 1, 2, \ldots, N-1)\), then we have

\[
A_{i,i+1} = (h_{i+1} - h_i)\setminus_{i+1} - (h_{i+2} - h_{i+1})\setminus_i ,
\]

and, if \( A_{i,i+1} \leq 0 \), or

\[
(h_i - h_{i+1})\setminus_{i+1} \geq (h_{i+1} - h_{i+2})\setminus_i ,
\]

then, according to Lemma 7.4, the exhaustive policy is the least cost policy in stages \( i \) and \( i+1 \). Let service rate \( \mu_i = \frac{1}{s_i} \); then the exhaustive policy is the least cost policy in stages \( i \) and \( i+1 \), if

\[
(h_i - h_{i+1})\mu_i \geq (h_{i+1} - h_{i+2})\mu_{i+1} .
\]

Considering condition (7.50) for \( i = 1, 2, \ldots, N-1 \), we reach the main theorem in Van Oyen and Teneketzis [69] in which (7.50) is presented as the optimality condition for an exhaustive service policy.
Remark 7.2

Consider the $N$-stage tandem queues in Chapter 6 in which greedy and exhaustive policies downstream of stage 2 are applied. In other words, using static, semi-dynamic or dynamic policies in stage 1, when the server switches to stage 2 in which $m$ customers are waiting, he applies policy $(2 \uparrow N)$ to serve these customers in queue $2 \uparrow N$. However, the optimal schedule of the server to serve these $m$ customers is actually the optimal control of a server in tandem queue $2 \uparrow N$ with the input batch of size $m$. Hence, when switchover times are zero, the AND and SHORA algorithms can be applied to find a better policy than $(2 \uparrow N)$ for each value of $m$. □

7.8 Numerical Study

In order to compare the policies obtained using the AND and SHORA algorithms with the optimal policy, we considered 4-stage, 5-stage and 6-stage tandem queues with $m = 4$, $m = 5$ and $m = 6$ jobs. These cases create shortest route problems with 140 nodes (for $n = 4$ and $m = 4$) to 1260 nodes (for $n = 6$ and $m = 5$). We examined over 300 different problems and obtained the relative error $\delta_A$ for any algorithm $A$ from

$$\delta_A = \frac{TC_A - TC^*}{TC^*},$$

(7.51)

where $TC_A$ is the average holding and switching cost of the policy obtained by algorithm $A$, and $TC^*$ is the optimal average holding and switching cost. We found that the average relative errors for the AND and SHORA algorithms are 0.008 and 0.006, and they obtain the optimal policy 72% and 75% of the time, respectively. The corresponding maximum relative errors are 0.10 and 0.09, respectively. Some of the numerical results are presented in Tables 7.3, 7.4 and 7.5.

If each algorithm is applied to a specific problem and the better result chosen, the aggregate average and maximum relative errors decrease to 0.004 and 0.09, respectively, and there is an 80% chance that the optimal policy is obtained. In other words, these algorithms are almost complements of each other.
<table>
<thead>
<tr>
<th>Problem</th>
<th>Policy</th>
<th>Cost</th>
<th>δ</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Optimal</td>
<td>9810</td>
<td>0.0000</td>
</tr>
<tr>
<td>h = (35.40, 42.51)</td>
<td>(1 \gg 1) \frac{2}{3} (2 \gg 4)</td>
<td>9810</td>
<td>0.0000</td>
</tr>
<tr>
<td>S = (3.2, 6.4)</td>
<td>AND</td>
<td>(1 \gg 4)</td>
<td>9932</td>
</tr>
<tr>
<td>K = (842.50, 831.181)</td>
<td>SHORA</td>
<td>(1 \gg 4) \frac{2}{3} (2 \gg 4)</td>
<td>9810</td>
</tr>
<tr>
<td>2</td>
<td>Optimal</td>
<td>20335</td>
<td>0.0000</td>
</tr>
<tr>
<td>h = (48.49, 53.57)</td>
<td>(1 \gg 1) \frac{2}{3} (2 \gg 4)</td>
<td>20335</td>
<td>0.0000</td>
</tr>
<tr>
<td>S = (4, 13, 9, 3)</td>
<td>AND</td>
<td>(1 \gg 1) \frac{2}{3} (2 \gg 4)</td>
<td>20335</td>
</tr>
<tr>
<td>K = (833.33, 923.429)</td>
<td>SHORA</td>
<td>(1 \gg 4) \frac{2}{3} (2 \gg 4)</td>
<td>20386</td>
</tr>
<tr>
<td>3</td>
<td>Optimal</td>
<td>6038</td>
<td>0.0000</td>
</tr>
<tr>
<td>h = (6.6.8.10)</td>
<td>(1 \gg 2) \frac{2}{3} (3 \gg 4)</td>
<td>6038</td>
<td>0.0000</td>
</tr>
<tr>
<td>S = (15.3, 17.12)</td>
<td>AND</td>
<td>(1 \gg 2) \frac{2}{3} (3 \gg 4)</td>
<td>6038</td>
</tr>
<tr>
<td>K = (416.76, 297.55)</td>
<td>SHORA</td>
<td>(1 \gg 2) \frac{2}{3} (3 \gg 4)</td>
<td>6038</td>
</tr>
<tr>
<td>4</td>
<td>Optimal</td>
<td>6618</td>
<td>0.0000</td>
</tr>
<tr>
<td>h = (2.6.13.13)</td>
<td>(1 \gg 4)</td>
<td>6618</td>
<td>0.0000</td>
</tr>
<tr>
<td>S = (15.17, 17.2)</td>
<td>AND</td>
<td>(1 \gg 4)</td>
<td>6618</td>
</tr>
<tr>
<td>K = (160.58, 279.495)</td>
<td>SHORA</td>
<td>(1 \gg 4)</td>
<td>6618</td>
</tr>
<tr>
<td>5</td>
<td>Optimal</td>
<td>26741</td>
<td>0.0000</td>
</tr>
<tr>
<td>h = (34.34, 43.52)</td>
<td>(1 \gg 1) \frac{2}{3} (2 \gg 4)</td>
<td>26741</td>
<td>0.0000</td>
</tr>
<tr>
<td>S = (11.14, 21.10)</td>
<td>AND</td>
<td>(1 \gg 1) \frac{2}{3} (2 \gg 4)</td>
<td>27075</td>
</tr>
<tr>
<td>K = (541.27, 266.694)</td>
<td>SHORA</td>
<td>(1 \gg 1) \frac{2}{3} (2 \gg 4)</td>
<td>27075</td>
</tr>
<tr>
<td>6</td>
<td>Optimal</td>
<td>14032</td>
<td>0.0000</td>
</tr>
<tr>
<td>h = (21.24, 31.34)</td>
<td>(1 \gg 1) \frac{2}{3} (2 \gg 4)</td>
<td>14032</td>
<td>0.0000</td>
</tr>
<tr>
<td>S = (3.10, 6.14)</td>
<td>AND</td>
<td>(1 \gg 1) \frac{2}{3} (2 \gg 4)</td>
<td>14098</td>
</tr>
<tr>
<td>K = (714.29, 937.886)</td>
<td>SHORA</td>
<td>(1 \gg 1) \frac{2}{3} (2 \gg 4)</td>
<td>14032</td>
</tr>
<tr>
<td>7</td>
<td>Optimal</td>
<td>4760</td>
<td>0.0000</td>
</tr>
<tr>
<td>h = (18.25, 25.30)</td>
<td>(1 \gg 1) \frac{2}{3} (2 \gg 4)</td>
<td>4760</td>
<td>0.0000</td>
</tr>
<tr>
<td>S = (3.3, 11.1)</td>
<td>AND</td>
<td>(1 \gg 1) \frac{2}{3} (2 \gg 4)</td>
<td>4848</td>
</tr>
<tr>
<td>K = (224.0, 124.0)</td>
<td>SHORA</td>
<td>(1 \gg 1) \frac{2}{3} (2 \gg 4)</td>
<td>4848</td>
</tr>
<tr>
<td>8</td>
<td>Optimal</td>
<td>3970</td>
<td>0.0000</td>
</tr>
<tr>
<td>h = (10.17, 20.22)</td>
<td>(1 \gg 1) \frac{2}{3} (2 \gg 4)</td>
<td>3970</td>
<td>0.0000</td>
</tr>
<tr>
<td>S = (6.9, 6.1)</td>
<td>AND</td>
<td>(1 \gg 1) \frac{2}{3} (2 \gg 4)</td>
<td>3970</td>
</tr>
<tr>
<td>K = (802.0, 0.00)</td>
<td>SHORA</td>
<td>(1 \gg 1) \frac{2}{3} (2 \gg 4)</td>
<td>4024</td>
</tr>
<tr>
<td>9</td>
<td>Optimal</td>
<td>9632</td>
<td>0.0000</td>
</tr>
<tr>
<td>h = (28.35, 50.76)</td>
<td>(1 \gg 1)</td>
<td>9632</td>
<td>0.0000</td>
</tr>
<tr>
<td>S = (6.9, 7.6)</td>
<td>AND</td>
<td>(1 \gg 1)</td>
<td>9632</td>
</tr>
<tr>
<td>K = (0.97, 0.0)</td>
<td>SHORA</td>
<td>(1 \gg 1)</td>
<td>9632</td>
</tr>
<tr>
<td>10</td>
<td>Optimal</td>
<td>20184</td>
<td>0.0000</td>
</tr>
<tr>
<td>h = (46, 51, 58, 72)</td>
<td>(1 \gg 1)</td>
<td>20184</td>
<td>0.0000</td>
</tr>
<tr>
<td>S = (9, 10, 11, 10)</td>
<td>AND</td>
<td>(1 \gg 1)</td>
<td>20184</td>
</tr>
<tr>
<td>K = (0.0, 4.0)</td>
<td>SHORA</td>
<td>(1 \gg 1)</td>
<td>20184</td>
</tr>
</tbody>
</table>
Table 7.4. Comparison of \( \text{AND} \) and \( \text{SHORA} \) for systems with \( n = 5 \) and \( m = 4 \).

<table>
<thead>
<tr>
<th>Problem</th>
<th>Policy</th>
<th>Cost</th>
<th>( \delta )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 ( h = (41, 50, 58, 64, 69) )</td>
<td>Optimal ( 1 )</td>
<td>21383</td>
<td>0.0000</td>
</tr>
<tr>
<td>( S = (1.9, 5.9.4) )</td>
<td>AND ( 2 )</td>
<td>21383</td>
<td>0.0000</td>
</tr>
<tr>
<td>( K = (125, 736, 306, 412, 467) )</td>
<td>SHORA ( 3 )</td>
<td>21383</td>
<td>0.0000</td>
</tr>
<tr>
<td>2 ( h = (28, 33, 39, 41, 41) )</td>
<td>Optimal ( 1 )</td>
<td>11334</td>
<td>0.0000</td>
</tr>
<tr>
<td>( S = (1.7, 7.4.1.4) )</td>
<td>AND ( 2 )</td>
<td>11334</td>
<td>0.0000</td>
</tr>
<tr>
<td>( K = (802, 748, 159, 274, 937) )</td>
<td>SHORA ( 3 )</td>
<td>11334</td>
<td>0.0000</td>
</tr>
<tr>
<td>3 ( h = (41, 46, 53, 55, 61) )</td>
<td>Optimal ( 1 )</td>
<td>17830</td>
<td>0.0000</td>
</tr>
<tr>
<td>( S = (3.4.8.1.8) )</td>
<td>AND ( 2 )</td>
<td>17832</td>
<td>0.0000</td>
</tr>
<tr>
<td>( K = (974, 897, 263, 546, 82) )</td>
<td>SHORA ( 3 )</td>
<td>17830</td>
<td>0.0000</td>
</tr>
<tr>
<td>4 ( h = (6.11, 18, 26, 32) )</td>
<td>Optimal ( 1 )</td>
<td>8420</td>
<td>0.0000</td>
</tr>
<tr>
<td>( S = (1.4.8, 1.4) )</td>
<td>AND ( 2 )</td>
<td>8420</td>
<td>0.0000</td>
</tr>
<tr>
<td>( K = (973, 934, 749, 230, 371) )</td>
<td>SHORA ( 3 )</td>
<td>8420</td>
<td>0.0000</td>
</tr>
<tr>
<td>5 ( h = (10, 20, 20, 21, 29) )</td>
<td>Optimal ( 1 )</td>
<td>8391</td>
<td>0.0000</td>
</tr>
<tr>
<td>( S = (7.10, 2.6.8) )</td>
<td>AND ( 2 )</td>
<td>8391</td>
<td>0.0000</td>
</tr>
<tr>
<td>( K = (47, 519, 671, 0, 0) )</td>
<td>SHORA ( 3 )</td>
<td>8867</td>
<td>0.0567</td>
</tr>
<tr>
<td>6 ( h = (41, 50, 58, 64, 69) )</td>
<td>Optimal ( 1 )</td>
<td>17963</td>
<td>0.0000</td>
</tr>
<tr>
<td>( S = (1.9.5.9.4) )</td>
<td>AND ( 2 )</td>
<td>17963</td>
<td>0.0000</td>
</tr>
<tr>
<td>( K = (125, 736, 306, 0, 0) )</td>
<td>SHORA ( 3 )</td>
<td>17963</td>
<td>0.0000</td>
</tr>
<tr>
<td>7 ( h = (28, 33, 39, 41, 41) )</td>
<td>Optimal ( 1 )</td>
<td>8800</td>
<td>0.0000</td>
</tr>
<tr>
<td>( S = (1.7.4.1.4) )</td>
<td>AND ( 2 )</td>
<td>9150</td>
<td>0.0398</td>
</tr>
<tr>
<td>( K = (802, 748, 159, 0, 0) )</td>
<td>SHORA ( 3 )</td>
<td>8912</td>
<td>0.0127</td>
</tr>
<tr>
<td>8 ( h = (31, 39, 39, 43, 44) )</td>
<td>Optimal ( 1 )</td>
<td>9698</td>
<td>0.0000</td>
</tr>
<tr>
<td>( S = (7.5.3.1.3) )</td>
<td>AND ( 2 )</td>
<td>9698</td>
<td>0.0000</td>
</tr>
<tr>
<td>( K = (126, 476, 426, 0, 0) )</td>
<td>SHORA ( 3 )</td>
<td>9792</td>
<td>0.0097</td>
</tr>
<tr>
<td>9 ( h = (34, 39, 44, 44, 50) )</td>
<td>Optimal ( 1 )</td>
<td>11313</td>
<td>0.0000</td>
</tr>
<tr>
<td>( S = (1.2, 8, 2.7) )</td>
<td>AND ( 2 )</td>
<td>11313</td>
<td>0.0000</td>
</tr>
<tr>
<td>( K = (331, 0, 738, 0, 125) )</td>
<td>SHORA ( 3 )</td>
<td>11347</td>
<td>0.0030</td>
</tr>
<tr>
<td>10 ( h = (41, 46, 53, 55, 61) )</td>
<td>Optimal ( 1 )</td>
<td>13838</td>
<td>0.0000</td>
</tr>
<tr>
<td>( S = (3.4.8, 1.8) )</td>
<td>AND ( 2 )</td>
<td>13840</td>
<td>0.0001</td>
</tr>
<tr>
<td>( K = (974, 0, 263, 0, 82) )</td>
<td>SHORA ( 3 )</td>
<td>13838</td>
<td>0.0000</td>
</tr>
<tr>
<td>11 ( h = (11, 21, 22, 29, 39) )</td>
<td>Optimal ( 1 )</td>
<td>9418</td>
<td>0.0000</td>
</tr>
<tr>
<td>( S = (5, 10, 6.6, 2) )</td>
<td>AND ( 2 )</td>
<td>9418</td>
<td>0.0000</td>
</tr>
<tr>
<td>( K = (0, 850, 0.541, 0) )</td>
<td>SHORA ( 3 )</td>
<td>9418</td>
<td>0.0000</td>
</tr>
<tr>
<td>Problem</td>
<td>Policy</td>
<td>Cost</td>
<td>$\delta$</td>
</tr>
<tr>
<td>---------</td>
<td>--------</td>
<td>------</td>
<td>------</td>
</tr>
<tr>
<td>1</td>
<td>Optimal</td>
<td>20142</td>
<td>0.0000</td>
</tr>
<tr>
<td>$h = (10, 14, 21, 27, 31, 38)$</td>
<td>AND</td>
<td>20142</td>
<td>0.0000</td>
</tr>
<tr>
<td>$S = (7, 2, 6, 9, 8)$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$K = (47, 830, 383, 486, 910, 632)$</td>
<td>SHORA</td>
<td>20142</td>
<td>0.0000</td>
</tr>
<tr>
<td>2</td>
<td>Optimal</td>
<td>40351</td>
<td>0.0000</td>
</tr>
<tr>
<td>$h = (49, 55, 57, 66, 76, 81)$</td>
<td>AND</td>
<td>40351</td>
<td>0.0000</td>
</tr>
<tr>
<td>$S = (8, 6, 3, 6, 2, 10)$</td>
<td>SHORA</td>
<td>40351</td>
<td>0.0000</td>
</tr>
<tr>
<td>$K = (772, 272, 359, 904, 266, 277)$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Optimal</td>
<td>11530</td>
<td>0.0000</td>
</tr>
<tr>
<td>$h = (2, 2, 12, 18, 23, 31)$</td>
<td>AND</td>
<td>11295</td>
<td>0.0577</td>
</tr>
<tr>
<td>$S = (8, 8, 5, 9, 3, 5)$</td>
<td>SHORA</td>
<td>11530</td>
<td>0.0000</td>
</tr>
<tr>
<td>$K = (761, 868, 233, 412, 287, 534)$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Optimal</td>
<td>38332</td>
<td>0.0000</td>
</tr>
<tr>
<td>$h = (24, 27, 30, 34, 35, 43)$</td>
<td>AND</td>
<td>38406</td>
<td>0.0019</td>
</tr>
<tr>
<td>$S = (20, 15, 6, 9, 9, 10)$</td>
<td>SHORA</td>
<td>38406</td>
<td>0.0019</td>
</tr>
<tr>
<td>$K = (464, 126, 389, 947, 71, 349)$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Optimal</td>
<td>28796</td>
<td>0.0000</td>
</tr>
<tr>
<td>$h = (12, 19, 20, 28, 29, 36)$</td>
<td>AND</td>
<td>28822</td>
<td>0.0009</td>
</tr>
<tr>
<td>$S = (3, 17, 8, 12, 19, 11)$</td>
<td>SHORA</td>
<td>28822</td>
<td>0.0009</td>
</tr>
<tr>
<td>$K = (233, 696, 227, 553, 485, 135)$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Optimal</td>
<td>31296</td>
<td>0.0000</td>
</tr>
<tr>
<td>$h = (24, 27, 30, 34, 35, 43)$</td>
<td>AND</td>
<td>31296</td>
<td>0.0000</td>
</tr>
<tr>
<td>$S = (20, 15, 6, 9, 9, 10)$</td>
<td>SHORA</td>
<td>31296</td>
<td>0.0000</td>
</tr>
<tr>
<td>$K = (464, 0, 389, 0, 71, 0)$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Optimal</td>
<td>29368</td>
<td>0.0000</td>
</tr>
<tr>
<td>$h = (10, 15, 24, 34, 35, 44)$</td>
<td>AND</td>
<td>29368</td>
<td>0.0009</td>
</tr>
<tr>
<td>$S = (19, 18, 18, 6, 19, 15)$</td>
<td>SHORA</td>
<td>29368</td>
<td>0.0009</td>
</tr>
<tr>
<td>$K = (679, 0, 439, 0, 576, 0)$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Optimal</td>
<td>33512</td>
<td>0.0000</td>
</tr>
<tr>
<td>$h = (25, 28, 34, 36, 39, 45)$</td>
<td>AND</td>
<td>33554</td>
<td>0.0012</td>
</tr>
<tr>
<td>$S = (4, 15, 13, 1, 4, 16)$</td>
<td>SHORA</td>
<td>33554</td>
<td>0.0012</td>
</tr>
<tr>
<td>$K = (0, 0, 495, 731, 245, 935)$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Optimal</td>
<td>23115</td>
<td>0.0000</td>
</tr>
<tr>
<td>$h = (8, 17, 22, 25, 32, 33)$</td>
<td>AND</td>
<td>23115</td>
<td>0.0000</td>
</tr>
<tr>
<td>$S = (1, 13, 13, 15, 3, 17)$</td>
<td>SHORA</td>
<td>23115</td>
<td>0.0000</td>
</tr>
<tr>
<td>$K = (0, 0, 420, 743, 359, 562)$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>Optimal</td>
<td>32734</td>
<td>0.0000</td>
</tr>
<tr>
<td>$h = (28, 30, 36, 43, 49, 59)$</td>
<td>AND</td>
<td>32734</td>
<td>0.0000</td>
</tr>
<tr>
<td>$S = (9, 13, 3, 3, 2, 15)$</td>
<td>SHORA</td>
<td>32734</td>
<td>0.0000</td>
</tr>
<tr>
<td>$K = (0, 934, 405, 131, 982, 0)$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Optimal</td>
<td>23234</td>
<td>0.0000</td>
</tr>
<tr>
<td>$h = (43, 49, 53, 57, 58, 65)$</td>
<td>AND</td>
<td>23330</td>
<td>0.0127</td>
</tr>
<tr>
<td>$S = (2, 2, 2, 9, 7, 3)$</td>
<td>SHORA</td>
<td>23530</td>
<td>0.0127</td>
</tr>
<tr>
<td>$K = (0, 574, 0, 91, 487, 0)$</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Chapter 8

U-Shaped Production Lines

8.1 Introduction

Traditional production or assembly lines are based on establishing a processing sequence for the parts being produced on the line. Parts move in a smooth, simple, logical and direct path through a sequence of work stations each comprised of special purpose equipment and single-functional workers. Although characterized by relatively high production rates, these lines have their own limitations. For example; machine stoppages halt the line, the slowest station paces the line and the line requires general supervision. However, the most important limitation of traditional production or assembly lines is the inherent inflexibility in changing the production rate. One way of increasing the flexibility of these lines in adapting to demand changes is through attaining flexibility in the number of workers. This is called Shojinka in Japanese (Monden [47]). Shojinka in the Toyota production system means to increase or decrease the number of workers at a shop when the production demand changes. Shojinka actually increases productivity by adjusting and rescheduling human resources. This concept has created new flexible lines in which there are fewer workers than stations in the line, and the workers walk to adjacent stations to continue work
Two main factors in Shojinka are:

- multi-functional workers, and
- U-shaped layout.

Multi-functional workers have the proper skills to work in different work stations in the line. and under a U-shaped layout, the walking times of workers among stations are reduced: and furthermore, it is easier to broaden or narrow the range of jobs for which each worker is responsible.

It should be noted that in some U-shaped lines the number of workers is the same as the number of work stations. These lines, which we prefer to call traditional U-shaped lines, are designed in a U shape for better material flow (entry and exit are located in the same side, close to each other), layout design considerations, or for better supervision and control. In this chapter, by U-shaped lines we only mean lines in which there are fewer worker than stations, whether the line is U-shaped or not.

Recently, U-shaped lines have become very popular. Some reasons for their popularity over traditional lines are as follows (Monden [47], Miltenburg [46], Bartholdi and Eisenstein [6] and Japanese Management Association [24]):

- Flexibility to increase or decrease the necessary number of workers when adapting to changes in production quantities (changes in demand).
- Having multi-functional workers rotating through stations in the line allows more workers to participate in efforts to improve the process.
- Workers stay more alert by rotating through a variety of tasks as compared to repeating a single short cycle task.
- Number of stations in a U-shaped line is always less or equal to that required on a traditional line, because there are more possibilities for grouping tasks into stations on a U-shaped line.
The worker moves the material automatically as a part of the task, so that usually no special material-handling equipment is necessary.

Also, the advantages of U-shaped lines over traditional batch production in shops with functional layout can be summarized as: lower inventories, simpler material handling, easier production planning and control, opportunities of team work and problem solving, better control of quality, and so on.

These advantages have increased the number of U-shaped lines in manufacturing companies. However, based on the nature of products and activities, different versions of U-shaped lines with different features have been designed. In this chapter a general definition for U-shaped lines is presented and different versions of U-shaped lines are introduced. Then, the literature on U-shaped lines is studied, and finally, considering the fact that a U-shaped line is actually a tandem queue attended by moving servers, the effect of switching costs and walking times are examined by decomposing a U-shaped line into a number of tandem queues each attended by a moving server.

8.2 U-Shaped Lines

Based on the production resources and the level of technology which were required in manufacturing units, different types of U-shaped lines have been designed to satisfy the requirements of those units. Before introducing these lines, we present our general definition for a U-shaped line which is used and referred to in this chapter as follows:

A U-shaped line is a production or assembly system in which there are machines or tools available to perform \( N \) different operations on an item, and all items that enter the system require the sequence of operations \( 1, 2, \ldots, N \). There are \( M \) workers (operators), \( M < N \), who use the machines and tools to perform operations on items. Each different operation is usually performed at a different place which is called a work station. Therefore, U-shaped lines actually consist of \( N \) work stations sequenced from 1 to \( N \) in a line which is usually configured in a U shape. Workers move among work stations to process items according to operational rules. The operational rules determine each worker's next action when an operation is completed. According to
the operational rules. worker $m \ (m \in \mathcal{M}_U \ where \ \mathcal{M}_U = \{1, 2, \ldots, M\})$ may perform operations at a specific zone or the set of work stations $N_m$, where $\bigcup_{m=1}^{M} N_m = \mathcal{N}_U$ where $\mathcal{N}_U = \{1, 2, \ldots, N\}$. In other words, worker $m$ is restricted to work in a specific zone in the line which contains work stations in set $N_m$. This zone is called the working zone of worker $m$. It should be noted that for two workers $i$ and $j$, we may have $N_i \cap N_j = \emptyset$. This means that the working zones of workers $i$ and $j$ may overlap.” Figure 8.1 shows a typical U-shaped line with $N = 12$ work stations and $M = 3$ workers.

Our definition characterizes different aspects of U-shaped lines for better understanding of the elements involved in these lines. In the next sections we present a new classification of U-shaped lines with the objective of organizing the previous research and providing a framework for further studies on these lines.

8.2.1 Worker-Oriented U-Shaped Lines

In a worker-oriented U-shaped line the operations on items require the presence of a worker during the whole operation. In other words, one item at a time can be processed in each work station and exactly one worker is required during the operation in that station. Typically, in these types of U-shaped lines, the machines or
tools which are used are not highly automated or advanced, and the line is actually established based on the workers' skills and team work rather than the capabilities of the equipment in the line.

Bucket brigade production systems are one type of worker-oriented U-shaped lines. In a bucket brigade production system, exactly one worker is required during the operation on an item of a batch. Each worker processes his batch from station to station until he reaches a busy station where his successor is working on his batch. In this case, the worker must wait until the next station becomes available, and then he continues to work on his batch. When the last worker completes his batch on the last work station, the line resets. The reset process is actually a takeover process as follows: the last worker returns and takes over the batch of his predecessor, who in turn returns and takes over the batch of his predecessor, and so on, until the first worker starts a new batch at the first station. This idea was first commercialized in the apparel industry by "Aisin Seiki Co. Ltd." a subsidiary of Toyota, and named Toyota Sewn Products Management System or TSS (Bartholdi and Eisenstein [5]).

TSS lines were first developed in Japan in 1970's and are widely used in apparel and textile industries. However, the first implementation of TSS in the US was in 1986. ¹

TSS is actually a bucket brigade production system in which the bucket size is 1. The operational rule in TSS lines consists of two separate rules (forward and backward rules), and if the workers are numbered from 1 to $M$ in the direction of product flow, then each worker must independently follows these rules (adapted from Bartholdi and Eisenstien [5]):

**Forward rule:** Process your item at successive work stations taking into account that at any station the worker with the higher index has priority. If your successor takes over your item, or if you are the last worker and you complete processing your item in the last station, follow the backward rule.

**Backward rule:** Walk back and begin to work on the item of your predecessor, or if you are the first worker, pick up raw material and start a new item in the first

¹In Riverside Fashions of Norris, South Carolina (Bartholdi and Eisenstien [5]).
station. Follow the forward rule.

It is assumed that there is always enough raw material in front of the first work station. In TSS lines (bucket brigades) there is always a possibility for workers to be blocked in forward movements. In this case, they are not allowed to pass their successor or return to take over the work of their predecessor.

8.2.2 Machine-Oriented U-Shaped Lines

In each work station $n$ of a machine-oriented U-shaped line, there is a machine which performs operation $n$ ($n \in N_{U}$) on the items. These machines are advanced and are able to perform the main operation automatically on an item after being set up by an operator. Here, the workers are machine operators, and the number of operators, $M$, is less than the number of machines, $N$. Therefore, each operator is in charge of at least one machine. The difference between worker-oriented and machine-oriented U-shaped lines is that in worker-oriented U-shaped lines one worker is required during the whole processing time of an item in a station; however, in machine-oriented U-shaped lines, when an item is attached to a machine and the machine is turned on by an operator, the item can be processed automatically without the operator. Therefore, the operator can switch to another machine to continue his job and the item will be detached by the same or another operator later. In a worker-oriented U-shaped line, the number of busy work stations is at most equal to the number of workers; but, in a machine-oriented U-shaped line this number can be more than the number of operators. One example of the machine-oriented U-shaped line is the *single unit production and conveyance* ("Ikko-Nagashi" in Japanese), which is applied to a production line without conveyors to manufacture different kinds of relatively small parts (Monden [47]). In this line, one operator is in charge of $N$ machines. When the operator visits a machine, his job is to wait for the processing of the preceding item if it is not complete, detach the processed item from the machine, attach the item which he has brought, turn the machine on and transfer the detached item to the next machine. According to this operational rule and considering that the line has only one operator, the new item enters the system only after one completed product exits.
The work in process in the system is constant and to increase the production rate more operators may be allocated to the system and zones assigned to each operator.

8.2.3 Static Working Zones Vs Dynamic Working Zones

Suppose in a machine-oriented U-shaped line that \( N_m \) is the set of machines which are assigned to operator \( m \) (\( m \in \mathcal{M}_U \)), and \( N_i \cap N_j = \emptyset \) for \( \forall i \neq j \in \mathcal{M}_U \). Then, each machine is actually assigned to one of the operators. In other words, the set of machines which are assigned to operators is completely different and therefore, the operators' working zones have no common area. These working zones which remain fixed and unchanged during the operation of the line are called static working zones. In worker-oriented U-shaped lines, the static working zones appear when only one specific worker is allowed to work in each working station. This means that operation \( n \) is always performed by worker \( m \).

Sometimes static working zones are defined based on the fraction of work completed on an item instead of the number of operations performed on an item. In other words, the boundaries of a static working zone may not be the end of an operation. If a boundary covers a fraction of an operation in a work station, then the remaining fraction belongs to another working zone. This means that there exists a work station which is used by two workers; but these workers perform separate fractions of an operation on an item in that station.

In U-shaped lines with dynamic working zones, set \( N_m \) is not a fixed set and may change in time. This means that each operator (worker) may be in charge of different machines (may work in different work stations) in each cycle. Therefore, for each worker different working zones are created in each cycle. Bucket brigades are a typical example of U-shaped lines with dynamic working zones. In bucket brigades each operation is not always performed by the same worker. U-shaped lines with dynamic working zones such as bucket brigades actually eliminate the possibility of starvation in a work station and create a self-organized line. On the other hand, workers in U-shaped lines with dynamic zones must be more skillful, because they must be able to work in more stations compared with the case of static zones designed.
for the same number of workers in a given U-shaped line.

8.2.4 Sequenced Working Zones Vs Mixed Working Zones

Consider a U-shaped line in which each worker is in charge of consecutive work stations. This usually means that the servers are not allowed to pass each other in the line. Thus, the U-shaped line will consist of $M$ working zones which are located in a prescribed sequence. We call these zones as *sequenced working zones* (Figure 8.2.a). In U-shaped lines with sequenced working zones, each zone is a multi-stage tandem queue attended by a moving server. The sequenced zones may either change in different cycles (dynamic working zones) or remain the same all the time (static working zones). However, in both cases, the working zones can be always numbered from 1 to $M$ in the direction of production flow. A TSS line is an example of a U-shaped line with sequenced dynamic working zones.

In the U-shaped lines with *mixed working zones*, the work stations assigned to
at least one worker are not necessarily consecutive work stations. In other words, at least one worker is allowed to skip some work stations and go from work station $i$ to $j$ ($j > n > i$). i.e., $n \in N_U$. where another worker is in charge of work station $n$ (Figure 8.3.b).

### 8.2.5 U-Shaped Lines with Multi-Functional Machines

Consider the single multi-functional machine scheduling problem in which a single machine is able to perform $N$ different operations, one at a time. If this machine is used to process items which require the sequence of operations $1, 2, \ldots, N$, then this system is indeed a U-shaped line with $N$ work stations and 1 worker. The setup time required after operation $i$ to start operation $j$ can be considered as the walking time between work stations $i$ and $j$. Now, if $M$ ($M < N$) similar machines are used and the set of operations $N_m$ is assigned to machine $m$, where $\bigcup_{m=1}^{M} N_m = N_U$, then for one operator per machine, this system may be considered a U-shaped line with $N$ work stations and $M$ workers.

U-shaped lines with multi-functional machines are not usually configured as a U-shaped line or at least the number of work stations is not quite clear to an observer. The walking times in these lines, which are actually the setup times, are typically greater than walking times in worker-oriented and machine-oriented U-shaped lines. In worker-oriented U-shaped lines, there are usually no setup times, and walking times are relatively small, while in machine-oriented U-shaped lines, the setup times are the times required to set the machine to perform the same operation on the next item. This time is usually less than the time required to set the machine to perform another operation on the next item (the setup time in a multi-functional machine).

### 8.3 Literature Review

In the category of worker-oriented U-shaped lines, Schoer, Wang and Ziemke [59] published the first paper on TSS lines in 1991. They analyzed a particular TSS line through simulation to achieve some statistics they needed, and did not reach
any general conclusions about TSS lines. However, the first comprehensive paper on TSS lines (bucket brigades) was presented by Bartholdi and Eisenstein [5]. They described TSS lines in the apparel industry and introduced a sufficient condition to achieve the maximum production rate. They assumed that all items are identical so require the same total processing time, and work station $j$ performs a fraction $p_j$ of the total processing time of an item. They define the state of the system as vector $X = (x_1, x_2, \ldots, x_M)$, where $x_m$, the position of worker $m$ in the line, is determined by the fraction of the cumulated work completed so far on an item being processed by worker $m$ to the total work required for that item in the line. Therefore, $0 \leq x_1 \leq x_2 \leq \cdots \leq x_M \leq 1$, because workers are not allowed to pass one another.

They modeled worker $m$ by work velocity $v_m$ which can be interpreted as the number of complete items that worker $m$ can produce per unit time, while working alone in the TSS line. For deterministic processing times and almost zero walking times, they showed that if workers are sequenced from slowest to fastest ($v_1 < v_2 < \cdots < v_M$), then the dynamic working zones in the line converge to static working zones, and if each worker is never blocked, then the working zone of worker $m$ is bounded in the interval of work content $[l_m^{(w)}, u_m^{(w)}]$, where

$$l_m^{(w)} = \frac{\sum_{i=1}^{m-1} v_i}{\sum_{i=1}^{M} v_j},$$

$$u_m^{(w)} = \frac{\sum_{i=1}^{m} v_i}{\sum_{i=1}^{M} v_j},$$

and the production rate of the line reaches to its maximum rate $\sum_{i=1}^{M} v_j$. In other words, by sequencing workers from slowest to fastest, the TSS line balances itself. On the other hand, they considered that if workers are not sequenced from slowest to fastest, then: (i) the TSS line can fail to balance itself, (ii) adding a worker to the line can decrease the production rate, and (iii) increasing the velocity of a worker may decrease the production rate. However, in TSS lines where workers are sequenced from slowest to fastest (a balanced TSS line), adding or speeding up a worker never decreases the production rate.

In their next paper with Bunimovich [3], Bartholdi and Eisenstein analyzed the behaviour of bucket brigade production lines with 2 and 3 workers. They simplified
the model by assuming that the work content is spread continuously through the line rather than clumped in discrete amounts at work stations. Thus, when a faster worker follows a slower one, he will not be blocked because the next work station is occupied by the slower one. He will be blocked when his item reaches the same state of completion as that of the slower worker, whereupon he remains blocked continuously with his work velocity decreasing to the velocity of the slower worker. For bucket brigades with two workers, they concluded that if \( v_1 < v_2 \), then the system reaches to its maximum production rate \( v_1 + v_2 \). However, in bucket brigades where \( v_1 > v_2 \), the faster worker is continually blocked and the production rate is \( 2v_2 \). In bucket brigades with three workers, they labeled workers so that \( v_{\min} < v_{\text{mid}} < v_{\text{max}} \) and they concluded the following:

- If the last worker in the line is the fastest one, then the line will achieve the maximum production rate \( v_{\min} + v_{\text{mid}} + v_{\text{max}} \) (irrespective of sequence of \( v_{\min} \) and \( v_{\text{mid}} \)).

- If the last worker in the line is the slowest one, then the line will achieve the smallest production rate \( 3v_{\min} \) (irrespective of sequence of \( v_{\text{max}} \) and \( v_{\text{mid}} \)).

- If the workers are sequenced as (slowest, fastest, mid), then the line will achieve either maximum production rate \( v_{\min} + v_{\text{mid}} + v_{\text{max}} \) or production rate \( 2(v_{\min} + v_{\text{mid}}) \).

- If the workers are sequenced as (fastest, slowest, mid), then the system displays a complex behaviour and will achieve a suboptimal production rate.

Bartholdi et al [3] also established the necessary condition \( v_1 < v_M \) for a bucket brigade to balance itself.

The application of bucket brigades in order-picking systems in warehouses is described in Bartholdi, Bunimovich and Eisenstein [4]. In the order-picking version of bucket brigades, the workers are pickers, the items are orders and the working stations are different bays of the flowracks. The papers which describes orders are picked up by the first picker, who opens a box, and then slides it along the line as he
moves, picking up different items to put into the box. This is the same job that other pickers do, except that each picker receives his box from his predecessor according to the bucket brigade operational rule. The main issue in the order-picking version of bucket brigades is that the order picking times are not deterministic, because the amount of work varies from order to order. However, Bartholdi, Bunimovich and Eisenstein claimed that although they have not been able to establish the proof for the stochastic case, nevertheless an array of evidence, including plausible explanation, simulation and field experiments have confirmed that if workers are sequenced from slowest to fastest, the bucket brigade will continuously and spontaneously (re)balance the work with the result that the average pick-up (production) rate is maximized. All results presented in Bartholdi et al [5, 3, 4] are based on the assumption that in bucket brigades the walking times between work stations are significantly less than processing times in work stations.

In their last paper, Bartholdi and Eisenstein [6] examined 150 TSS lines and presented some useful comments to help managers design bucket brigades. Some of these results are summarized as follows:

- The number of workers in a bucket brigade line must be less than \(1/p_{\text{max}}\), where \(p_{\text{max}}\) is the largest fraction of total work to be done in a work station.

- Bucket brigades with a small number of workers perform better than bucket brigades with large numbers of workers. In the apparel industry, experience has shown that team effectiveness is reduced if the team has more than ten members; while three to six members is most common.

- Processing in large buckets reduces the variance of the work at each station and the chance of blocking.

- The bucket brigade production lines are mostly recommended when

  1. there are significant changes in demand,

  2. work stations are inexpensive relative to labor costs,
3. the work in different work stations mostly need a single skill.

4. the workers can move easily among work stations and the work takeover process can be done without difficulties.

Miltenburg and Wijngaard [46] considered the line balancing problem of the worker-oriented U-shaped line with zero walking times and deterministic processing times. They introduced an optimization problem to find the optimal balance which defines the optimal static working zones for the U-shaped line with a minimum number of workers for a given set of tasks and cycle time. A dynamic programming procedure for computing the optimal balance was presented along with two heuristic procedures which represent extensions of well-known heuristics for the traditional line balancing problem. Finally, these heuristics were evaluated through their performances on well-known line balancing problems in the literature.

In the category of machine-oriented U-shaped lines, Ohno and Nakade [57] considered the single unit production and conveyance system with \( N \) machines, single operator and deterministic and constant processing, operation and walking times. Operation times are actually considered as the time required for detaching the processed item, putting it on a chute, and attaching a new item. They derived the operator's waiting time at each machine in each cycle and the cycle times. They also studied the same problem with \( M \) operators and obtained the overall cycle time of the line for given static working zones. To find the optimal static working zones for the operators which minimize the overall cycle time of the line, they formulated the problem as a combinatorial optimization problem, and then analyzed the optimal static working zones for the problem with two workers. Finally, they derived the average throughput of the line for the stochastic version of this problem with a single operator.

In another paper by Nakade and Ohno [52], two systems of machine-oriented U-shaped lines with a single operator were considered, in which the stochastic processing, operation and walking times are comparable in the sense of an increasing convex order. It is shown that as the operator is more skillful in the operation, in the sense of this
order, the cycle time is shorter. They derived the expected cycle time for the line in which the processing times are Erlang random variables, and also obtained an upper and lower bound for the expected cycle time for the line with generally distributed processing times.

Most of the literature on U-shaped lines has not considered a switching cost when the worker moves (switches) from one work station to another, or if walking time is significant compared to the processing times. Therefore, the only cost involved is the holding cost in the work station, and this can be minimized if a worker process a single item instead of a batch, and completes that item moving from station to station. In the next sections we examine the effect of switching costs and significant walking times on the optimal batch size in some classes of U-shape lines.

8.4 U-Shaped Lines with Multi-Functional Machines and Switching Costs

Consider a worker-oriented U-shaped line with $N$ work stations and $M$ workers who work in static and sequenced working zones. Consider $i_m$ and $j_m$ as the first and the last work stations in set $N_m$, sequenced according to the increasing order of the work station number. Also, suppose a processed item is called a type $j$ item when it is processed in work station $j - 1$ and then requires processing at work station $j$. Hence, type 1 items are actually the raw materials and type $N + 1$ items are final products.

In a U-shaped line with static and sequenced working zones, there is always a chance for work station $i_m$ to be starved. One way to eliminate this possibility is to have enough type $i_m$ items in the buffer of work station $i_m$ ($m \in M_U$). Applying this policy, the U-shaped line can be decomposed into $M$ mutually independent multi-stage tandem queues, each attended by a moving server. This usually occurs in U-shaped lines with multi-functional machines, in which machine $m$ performs all operations in set $N_m$ and one operator is in charge of each machine. Switching costs and walking times from work stations $i$ to $j$ are actually the setup costs and setup
times, respectively, when operation \( j \) must be done after operation \( i \). Since setup costs and setup times are involved, it may be optimal for operator \( m \) to complete a batch of items before switching to the next operation. To find the optimal batch size for operator \( m \) which minimizes the total average holding and switching costs, consider the multi-stage tandem queue \( i_m \uparrow j_m \) with random service times \( S_j \) in work station \( j \), random walking times \( D_{ij} \) from work station \( i \) to \( j \), holding cost \( h_j \) in work station \( j \) and switching cost \( K_{ij} \) whenever the operator switches from operation \( i \) to \( j \) (\( i \neq j; \ i, j \in N_m \)). If operator \( m \) completes a batch of size \( k \) in a cycle by applying a greedy and exhaustive policy in stages \( i_m \) to \( j_m \), then using (7.13) and (7.14), the total average holding and switching costs in a cycle is

\[
C[i_m \uparrow j_m] = \frac{k(k+1)}{2} \sum_{r=1}^{j_m} h_r \bar{S}_r + \frac{k(k-1)}{2} \sum_{r=1}^{j_m} h_{r+1} \bar{S}_r + \sum_{r=i_m}^{j_m-1} K_{r,r+1} + K_{j_m,i_m} + k \sum_{r=i_m}^{j_m} h_{r+1} \bar{D}_{r,r+1}.
\]  

(8.3)

Since it is assumed that there are always enough type \( i_m \) items available in the buffer of work station \( i_m \), therefore, the holding costs of these items are not considered in (8.3). Suppose \( TC_m(1|k) \) is the total average holding and switching cost per produced item in queue \( i_m \uparrow j_m \) when batch of size \( k \) is completed in each cycle; then,

\[
TC_m(1|k) = \frac{C[i_m \uparrow j_m]}{k} = \frac{C[i_m \uparrow j_m]}{k} = kH_{i_m,j_m} + \frac{1}{k} \tilde{K}_{i_m,j_m} + \mathcal{H}D_{i_m,j_m}.
\]  

(8.4)

where

\[
H_{i_m,j_m} = \frac{1}{2} \sum_{r=i_m}^{j_m} (h_r + h_{r+1}) \bar{S}_r
\]  

(8.5)

\[
\tilde{K}_{i_m,j_m} = \sum_{r=i_m}^{j_m-1} K_{r,r+1} + K_{j_m,i_m}
\]  

(8.6)

\[
\mathcal{H}D_{i_m,j_m} = \frac{1}{2} \sum_{r=i_m}^{j_m} (h_r - h_{r+1}) \bar{S}_r + h_{r+1} \bar{D}_{r,r+1}.
\]  

(8.7)

Therefore, \( k^* \) will be the optimal batch size in queue \( i_m \uparrow j_m \), if

\[
TC_m(1|k^*) - TC_m(1|k^* + 1) < 0
\]
Combining (8.8) and (8.9), we conclude that the optimal batch size which minimizes the total average holding and switching costs per produced item in queue $i_m \uparrow j_m$ is the integer $k^*$ satisfying

$$k^*(k^* - 1) < \frac{\tilde{K}_{im,jm}}{H_{im,jm}} < k^*(k^* + 1).$$

(8.10)

It is clear that the optimality condition (8.10) is independent of the walking times. This is so because the total average holding cost during the walking times is constant for any single item of a batch, and it is equal to $\sum_{r=i_m}^{j_m} h_{r+1} D_{r,r+1}$. 

**Example 8.1**

Consider a U-shaped line with three multifunctional machines where $N_1 = \{1,2\}$, $N_2 = \{3,4,5\}$, $N_3 = \{6,7,8,9,10\}$ and

$$h = [2, 3, 5, 7, 10, 10, 11, 14, 15, 18],$$

$$\overline{S} = [9, 5, 1, 2, 3, 8, 2, 1, 3, 1],$$

$$K = [250, 200, 45, 80, 75, 90, 60, 70, 20, 100],$$

in which $K_i$ is the switching cost to operation (stage) $i$ ($i = 1, 2, \ldots, 10$). Assuming that there are always enough items available in the buffer of each machine, the optimal batch size $k^*_1$ for machine 1 (queue 1 $\uparrow$ 2) can be obtained using (8.10) as follows:

$$\tilde{\tilde{K}}_{1,2} = K_1 + K_2 = 450,$$

$$H_{1,2} = \frac{1}{2} \sum_{r=1}^{2} (h_r + h_{r+1}) \overline{S}_r = 42.5.$$
Since \( k_1^* \) must satisfy

\[
\frac{450}{42.5} < K_1^*(K_1^* - 1) < K_1^*(K_1^* + 1),
\]

therefore, \( k_1^* = 3 \).

Using the same approach for machine 2 (queue 3 \( \uparrow 5 \)) and machine 3 (queue 6 \( \uparrow 10 \)), we get

\[
K_2^*(K_2^* - 1) < \frac{200}{53} < K_2^*(K_2^* + 1)
\]

\[
K_3^*(K_3^* - 1) < \frac{340}{182} < K_3^*(K_3^* + 1)
\]

which leads to \( k_2^* = 2 \) and \( k_3^* = 1 \).

\[\square\]

## 8.5 Worker-Oriented U-Shaped Lines with Sequenced Static Working Zones

Consider a worker-oriented U-shaped line with \( \mathcal{W} \) work stations and \( M \) workers in which:

- Working zones are static and sequenced, so that worker \( m (m \in \mathcal{M}_U) \) is in charge of tandem work stations \( i_m \) to \( j_m \) (tandem queue \( i_m \uparrow j_m \)).

- Walking times are insignificant compared with the processing times in work stations.

- Items arrive at the buffer of work station 1 according to a random process, and the buffer of work station \( i_m \) is supplied by work station \( j_{m-1}, m = 2, 3, \ldots, M \).

- the holding cost rate is \( h_i \) per unit time in work station \( i \), and the switching cost \( K_{ij} \) is charged whenever a worker switches from work station \( i \) to \( j \).

The differences between this model and the model in section 8.4 are: (i) in this model walking times are actually considered to be zero, (ii) the items in the buffer of work station \( i_m \) are received from work station \( j_{m-1} \); thus, there exists a chance for work station \( i_m \) to be starved during the operation. Since static and sequenced
working zones are considered, the U-shaped line can be decomposed into $M$ multi-stage tandem queues, each attended by a moving server. Suppose that worker $m$ applies a limited policy in work station $i_m$ and a greedy and exhaustive policy in work stations $i_m + 1$ to $j_m$. Then the optimal limit $M_{Lm}^*$ which minimizes the total average holding and switching cost in work stations $i_m$ to $j_m$ (queue $i_m$ to $j_m$) can be approximated as follows:

1. For $m = 1$, since there are always enough items (raw materials) available in work station 1, the multi-stage tandem queue $i_1 \uparrow j_1$ with a limited policy behaves like the multi-stage tandem queue in Section 8.4, and so the optimal limit $M_{L1}^*$ is actually the optimal batch size $k^*$ which can be obtained using (8.10).

2. Multi-stage tandem queues $i_m \uparrow j_m$, $2 \leq m \leq M$ are $G/G_1 - G_2 - \cdots - G_{jm-1_{m+1}}/1$ queues with zero switchover times, a limited policy in stage $i_m$ and a greedy and exhaustive policy in stages $i_m + 1$ to $j_m$. These models are the same as those introduced in Chapter 6, except that here the arrival process is not Poisson. The arrival process to queue $i_m \uparrow j_m$ is actually the departure process from queue $i_{m-1} \uparrow j_{m-1}$. Nevertheless, the optimal limits of the limited policy in high traffic intensity can be used as an approximator for the optimal limits in lower traffic intensities, even for systems with non-Poissonian arrival process for the following reasons:

- According to Proposition 4.1 and 4.2, the sufficient and necessary conditions for optimality of limits 1 and 2, respectively, are independent of the arrival process.

- In $G/G_1 - G_2 - \cdots - G_N/1$ under low traffic intensity, limited policies with limit $M_L \geq 2$ behave almost the same, and therefore, the total average cost $TC(M_L)$ is very flat at the bottom and the difference in total average cost between the optimal limit and the limit which is optimal in high traffic intensity is insignificant. On the other hand, as $\rho$ increases, the optimal limit in systems with lower traffic intensities approach the optimal limit in the same system with high traffic intensity.
Therefore, when \( 2 \leq m \leq M \), the optimal limit \( M_L^{\text{m}} \) can be approximated for queue \( i_m \uparrow j_m \) using Proposition 8.1 which yields the optimal limits of the limited policy in high traffic intensity.

**Proposition 8.1**

Consider a multi-stage tandem queue \( i_m \uparrow j_m \) with zero switchover times in which a limited policy with limit \( M_L^{\text{m}} \) in stage \( i_m \) and a greedy and exhaustive policy downstream of stage \( i_m + 1 \) are applied. Then assuming a high traffic intensity, the optimal limit \( M_L^{\text{m}} \) satisfies

\[
\frac{M_L^{\text{m}}(M_L^{\text{m}} - 1)}{2} \leq \frac{\tilde{K}_{i_m,j_m}}{\sum_{r=i_m+1}^{j_m} (h_r - h_{i_m})S_r + \sum_{r=i_m+1}^{j_m} h_r S_{r-1}} \leq \frac{M_L^{\text{m}}(M_L^{\text{m}} + 1)}{2}.
\]

**(8.11)**

**Proof**

Since in high traffic the limited policy with limit \( M_L^{\text{m}} \) actually serves \( M_L^{\text{m}} \) customers in each cycle. therefore to conclude (8.11) we now compare two limited policies with limits \( k \) and \( k + 1 \) over the time during which \( k(k + 1) \) customers are served. Let \( h_{j_m + 1} = 0 \). then using (7.16), for \( M_L^{\text{m}} = k \) during \( k + 1 \) cycles, we have

\[
C[\overset{k+1}{\circ} (i_m \uparrow j_m)] = (k + 1)C[i_m \uparrow j_m] + kK_{j_m,i_m} + \frac{k^2k(k + 1)}{2}h_{i_m} \sum_{r=i_m}^{j_m} S_r.
\]

and similarly, for \( M_L^{\text{m}} = k + 1 \) during \( k \) cycles, we have

\[
C[\overset{k}{\circ} (i_m \uparrow j_m)] = kC[i_m \uparrow j_m] + (k - 1)K_{j_m,i_m} + \frac{(k + 1)^2k(k - 1)}{2}h_{i_m} \sum_{r=i_m}^{j_m} S_r.
\]

so that

\[
C[\overset{\circ}{\circ} (i_m \uparrow j_m)] - C[\overset{\circ}{\circ} (i_m \uparrow j_m)] = kC[i_m \uparrow j_m] - (k + 1)C[i_m \uparrow j_m] - K_{j_m,i_m} - \frac{k(k + 1)}{2}h_{i_m} \sum_{r=i_m}^{j_m} S_r.
\]

On the other hand, using (7.13) and (7.14) we get

\[
C[i_m \overset{k+1}{\circ} j_m] = \frac{(k + 1)(k + 2)}{2} \sum_{r=i_m}^{j_m} h_r S_r + \frac{k(k + 1)}{2} \sum_{r=i_m+1}^{j_m} h_r S_{r-1} + \tilde{K}_{i_m,j_m}.
\]

\[(8.15)\]
and
\[
C[i_m \uparrow j_m] = \frac{k(k+1)}{2} \sum_{r=i_m}^{j_m} h_r \overline{S}_r + \frac{k(k-1)}{2} \sum_{r=i_m+1}^{j_m} h_r \overline{S}_{r-1} + \hat{K}_{i_m,j_m}. \tag{8.16}
\]

Therefore,
\[
kC[i_m \uparrow^{k+1} j_m] - (k+1)C[i_m \uparrow j_m] = \frac{k(k+1)}{2} \left( \sum_{r=i_m}^{j_m} h_r \overline{S}_r + \sum_{r=i_m+1}^{j_m} h_r \overline{S}_{r-1} \right) - \hat{K}_{i_m,j_m}. \tag{8.17}
\]

and by substituting (8.17) into (8.14), we obtain
\[
C[i_m \uparrow^{k+1} j_m] - C[i_m \uparrow j_m] = \frac{k(k+1)}{2} \left[ \sum_{r=i_m}^{j_m} (h_r - h_{i_m}) \overline{S}_r + \sum_{r=i_m+1}^{j_m} h_r \overline{S}_{r-1} \right] - \hat{K}_{i_m,j_m}. \tag{8.18}
\]

Since \( C[i_m \uparrow^{k+1} j_m] \leq C[i_m \uparrow j_m] \), thus, in high traffic, systems with limit \( M_L^U = k \) have lower average cost than systems with limit \( M_L^U = k + 1 \), provided
\[
C[i_m \uparrow^{k+1} j_m] - C[i_m \uparrow j_m] \geq 0
\]
or, equivalently,
\[
\frac{k(k+1)}{2} \geq \frac{\hat{K}_{i_m,j_m}}{\sum_{r=i_m+1}^{j_m} (h_r - h_{i_m}) \overline{S}_r + \sum_{r=i_m+1}^{j_m} h_r \overline{S}_{r-1}}. \tag{8.19}
\]

Using the same approach and comparing \( C[i_m \uparrow^{k-1} j_m] \) and \( C[i_m \uparrow^{k+1} j_m] \), it can be concluded that if
\[
\frac{k(k-1)}{2} \leq \frac{\hat{K}_{i_m,j_m}}{\sum_{r=i_m+1}^{j_m} (h_r - h_{i_m}) \overline{S}_r + \sum_{r=i_m+1}^{j_m} h_r \overline{S}_{r-1}}. \tag{8.20}
\]
then the system with limit \( M_L^U = k \) has lower average cost than a system with limit \( M_L^U = k - 1 \). Combining (8.19) and (8.20), the proof is complete. \( \square \)

It should be noted that applying an optimal limited policy by worker \( m \) in tandem queue \( i_m \uparrow j_m \) \( (m \in M_U) \) does not mean that the U-shape line is optimized. Applying an optimal limited policy only guarantees the local optimization in static working zones given that limited policies must be implemented in working zones.
Example 8.2

Consider a worker-oriented U-shaped line with \( N = 10 \) work stations and \( M = 3 \) workers who work in sequenced and static working zones \( V_1 = \{1, 2\} \), \( V_2 = \{3, 4, 5\} \) and \( V_3 = \{6, 7, 8, 9, 10\} \). Also, suppose that the switching costs, holding costs and average service times in this line are the same as for Example 8.1. If each worker applies a limited policy in his working zone and items in the buffer of work stations 3 and 6 are supplied by work stations 2 and 5, respectively, then using (8.11), the optimal limit \( M^{*m} \) for working zone \( m \) \((m = 1, 2, 3)\) must satisfy the following:

\[
\begin{align*}
\frac{M^1(M^1 - 1)}{2} & \leq \frac{450}{32} \leq \frac{M^1(M^1 + 1)}{2} \\
\frac{M^2(M^2 - 1)}{2} & \leq \frac{200}{46} \leq \frac{M^2(M^2 + 1)}{2} \\
\frac{M^3(M^3 - 1)}{2} & \leq \frac{340}{214} \leq \frac{M^3(M^3 + 1)}{2}
\end{align*}
\]

which lead to \( M^1 = 5 \), \( M^2 = 3 \) and \( M^3 = 2 \). \( \square \)

8.6 Bucket Brigades with Switching Costs

In the model introduced in Bartholdi and Eisenstein [5] for bucket brigades with \( M \) workers and \( N \) work stations, the position of worker \( m \) is expressed as the cumulative fraction \( x_m \) of work completed on her item (batch). Thus, the position of worker \( m \) is actually a real number between zero and one. On the other hand, the total processing time of an item (batch) in the line is normalized to one unit time so that the processing time requirement in work station \( n \) is \( p_n \), which is a fixed percentage of the total standard work content of the product. Therefore, if the standard processing time in work station \( n \) is \( S^{(st)}_n \) \((n \in V_L)\), then the interval of work content \([l^{(st)}_n, u^{(st)}_n]\),

\[
\begin{align*}
l^{(st)}_n &= \frac{\sum_{r=1}^{n-1} S^{(st)}_j}{\sum_{r=1}^{N} S^{(st)}_j} \\
u^{(st)}_n &= \frac{\sum_{r=1}^{n} S^{(st)}_j}{\sum_{r=1}^{N} S^{(st)}_j}
\end{align*}
\]

actually refers to work station \( n \).
Suppose the work velocity of a standard worker who completes processing of an item in its standard time is set at 1, and the velocity of the slower and faster workers are scaled according to this standard worker. Then a worker with work velocity \( v = 0.5 \) is half as fast as the standard worker. The standard worker needs exactly time \( S_n^{(st)} \) to complete processing of an item in work station \( n \); however, a worker with work velocity \( \nu \neq 1 \) does the same operation in \( S_n^{(st)}/\nu \).

According to Bartholdi and Eisenstein [5], if the workers of a bucket brigade with deterministic processing times are sequenced from slowest to fastest, then dynamic working zones converge to static working zones, and if workers are never blocked, then the static working zone of worker \( m \) is bounded in the interval of work content \([l_m^{(w)}, u_m^{(w)}]\). A bucket brigade under these assumption is called a balanced bucket brigade. Corollary 8.1 describes how a balanced bucket brigade can be analyzed in terms of tandem queues attended by a moving server.

**Corollary 8.1**

A balanced bucket brigade with \( N \) work stations and \( M \) workers behaves like \( M \) identical parallel \( N \)-stage tandem queues, each attended by a moving server who applies a greedy and exhaustive policy in stages 1 to \( N \); and as each server moves from stage 1 to \( N \), his work velocity increases in consecutive intervals of work content. \( \square \)

Corollary 8.1 actually decomposes a balanced bucket brigade regarding to the number of workers (number of batches being processed in the line), \( M \), rather than the working zones of workers. Each batch is processed in stages 1 to \( N \) independent of the other \( M - 1 \) batches which are processed simultaneously in the line. In other words, it can be considered that \( M \) batches are processed in \( M \) parallel \( N \)-stage tandem queues. These \( M \) parallel tandem queues are identical and all servers have work velocity \( v_m \) in the interval of work content \([l_m^{(w)}, u_m^{(w)}]\). As described in Section 8.2.3, the boundaries of working zones, \( l_m^{(w)} \) and \( u_m^{(w)} \) (\( m \in \mathcal{M}_U \)), do not necessarily refer to the boundaries of some work stations, \( l_i^{(s)} \) and \( u_j^{(s)} \) (\( i, j \in \mathcal{N}_U \)). In other words, worker \( m + 1 \) may take over the item of worker \( m \) in the middle of its processing in
one of the work stations. This means that an item may be processed in two (or more) work velocities in one working station. Figure 8.3 presents a typical example showing how interval \([l_m^{(w)}, u_m^{(w)}]\) describes the static working zone of worker \(m\).

Suppose Figure 8.3 refers to a TSS line; then worker \(m\) is actually in charge of work stations \(i + 1, i + 2, \ldots, j - 1\). He also takes over the work of his predecessor (worker \(m - 1\)) in work station \(i\) when fraction \(l_m^{(w)}\) of work (compared to the total work required for an item in the line) on his item is complete. On the other hand, his item is taken over by his successor (worker \(m + 1\)) in work station \(j\) when fraction \(u_m^{(w)}\) of work on the item is complete. Since all items in work stations \(i + 1, i + 2, \ldots, j - 1\) are processed by worker \(m\) at work velocity \(v_m\), the actual processing time on an item in work stations \(n\), \(S_n\), is

\[ S_n = \frac{S_n^{(st)}}{v_n} \quad ; \quad n = i + 1, i + 2, \ldots, j - 1. \quad (8.23) \]

However, the actual processing times \(S_i\) and \(S_j\) of an item in work stations \(i\) and \(j\), respectively, are obtained from

\[ S_i = \left( \frac{l_m^{(w)} - l_i^{(s)}}{u_i^{(s)} - l_i^{(s)}} \right) \frac{S_i^{(st)}}{v_{m-1}} + \left( 1 - \frac{l_m^{(w)} - l_i^{(s)}}{u_i^{(s)} - l_i^{(s)}} \right) \frac{S_i^{(st)}}{v_m} \]

\[ S_j = \left( \frac{u_m^{(w)} - l_j^{(s)}}{u_j^{(s)} - l_j^{(s)}} \right) \frac{S_j^{(st)}}{v_m} + \left( 1 - \frac{u_m^{(w)} - l_j^{(s)}}{u_j^{(s)} - l_j^{(s)}} \right) \frac{S_j^{(st)}}{v_{m+1}}. \quad (8.25) \]

Considering different static working zones, the same approach can be used to obtain the actual processing time \(S_n\) in work station \(n\) \((n \in \mathcal{N}_{(\ell)})\). Therefore, based on Corollary 8.1, the TSS line with deterministic processing times in each work station can be decomposed into \(M\) parallel \(G/D_1 - D_2 - \cdots - D_N/1\) queues in which the service time in stage \(i\) is \(S_i\) \((i = 1, 2, \ldots, N)\).

Now suppose that Figure 8.3 refers to a bucket brigade. Since worker \(m\) is in charge of work stations \(i + 1, i + 2, \ldots, j - 1\), all items in a batch will be processed
at the same work velocity $v_m$ in these stations. and equation (8.23) is now also true for bucket brigades. However equations (8.24) and (8.25) will only be true if $\overline{S}_i$ and $\overline{S}_j$ are considered the actual processing times of a batch in work stations $i$ and $j$, respectively. To find the actual processing times of items in a batch in work station $j$, let $q_j$ be the fraction of the job in stage $j$ (Figure 8.3) which is done at work velocity $v_m$: then.

$$q_j = \frac{u_m^{(w)} - t_i^{(s)}}{u_j^{(s)} - t_i^{(s)}}. \tag{8.26}$$

If a batch consisting of $k$ items must be processed in work station $j$, then $kq_jS_j^{(st)}$ is that portion of the total standard work $kS_j^{(st)}$ in stage $j$ which is completed at work velocity $v_m$, and the remaining part, $k(1-q_j)S_j^{(st)}$, is completed at work velocity $v_{m+1}$. This means that fraction $kq_j$ of items in a batch are processed in stage $j$ in actual time $kq_jS_j^{(st)}/v_m$ and the remaining fraction $k(1-q_j)$ of items are processed in actual time $k(1-q_j)S_j^{(st)}/v_{m+1}$.

Let $J_i$ be the work station in which the work velocity changes from $v_i$ to $v_{i+1}$ during the operation in that stage. and let $J_i (J = \{J_1, J_2, \ldots, N+1\})$ be the set of these work stations. Here, $N+1$ is considered to be a hypothetical stage where the server changes work velocity from $v_M$ to $v_1$.

Corollary 8.2 describes the changes in the work velocity in the work stations of a balanced bucket brigade.

**Corollary 8.2**

Let $N_m$ be the set of work stations in the working zone of worker $m$. Then in a balanced bucket brigade the work velocity in work stations $n \in N_m \cap J^c$ ($J^c$ is the complement of set $J$) is the constant $v_m$. Now let $\mathbb{Z}[u]$ be the largest integer less than $u$. Then, in a balanced bucket brigade with bucket size $k$, the work velocity in work station $J_m \in J$, $m \in M_U$ is:

- $v_m$ for the first $\mathbb{Z}[kq_{J_m}]$ items in the batch.
- $v_m$ for the fraction $kq_{J_m} - \mathbb{Z}[kq_{J_m}]$ of work on the $(\mathbb{Z}[kq_{J_m}] + 1)$th item.
- \( v_{m+1} \) for the remaining fraction \( 1 - (kq_{Jm} - \geq[kq_{Jm}]) \) of work on the \( (\geq[kq_{Jm}] + 1) \)th item, and

- \( v_{m+1} \) for the last \( k - \geq[kq_{Jm}] - 1 \) items.

**Remark 8.1**

Corollary 8.2 only considers bucket brigades in which at most one change in work velocity may occur in each work station. In other words, each worker is in charge of at least one operation (work station) in the line, which is a realistic assumption.

According to Corollary 8.1, the balanced bucket brigades with deterministic processing times and bucket size \( k \) can be considered as \( M \) parallel \( G/D_1 - D_2 - \cdots - D_N/1 \) queues in which servers apply greedy and exhaustive policies in stages 1 to \( N \) to process buckets of size \( k \) and change their work velocities according to Corollary 8.2.

Thus, finding the optimal batch size for the balanced bucket brigade when switching costs are involved is equivalent to finding the optimal batch size in one of these identical queues. Since there are always enough items available in stage 1, obtaining the optimal batch size is almost similar to the model in Section 8.4. However, the difference is that here in each queue the server changes his work velocity in some stages while processing an item of a batch. Therefore, for the batch of size \( k \) in each parallel \( N \)-stage tandem queue, we have

\[
C[1 \overset{k}{\rightarrow} N] = C_h[1 \overset{k}{\rightarrow} N] + \sum_{r=1}^{N-1} K_{r,r+1} + K_{N1},
\]  

(8.27)

where

\[
C_h[1 \overset{k}{\rightarrow} N] = C_h[1 \overset{k}{\rightarrow} J_1 - 1] + \sum_{J_i \in \mathcal{J}} C_h[J_i \overset{k}{\rightarrow} J_i + 1]
+ \sum_{J_i \in \mathcal{J}} C_h[J_i + 2 \overset{k}{\rightarrow} J_{i+1} - 1],
\]  

(8.28)

and

\[
C_h[1 \overset{k}{\rightarrow} J_1 - 1] = \frac{k(k+1)}{2} \frac{J_i-1}{2} \sum_{r=1}^{J_i-1} h_r \overline{s}_r + \frac{k(k-1)}{2} \frac{J_i-1}{2} \sum_{r=1}^{J_i-1} h_{r+1} \overline{s}_r,
\]  

(8.29)

\[
C_h[J_i + 2 \overset{k}{\rightarrow} J_{i+1} - 1] = \frac{k(k+1)}{2} \frac{J_{i+1}-1}{2} \sum_{r=J_i+2}^{J_{i+1}-1} h_r \overline{s}_r + \frac{k(k-1)}{2} \frac{J_{i+1}-1}{2} \sum_{r=J_i+2}^{J_{i+1}-1} h_{r+1} \overline{s}_r.
\]  

(8.30)
However, $C_h[J_i \rhd J_i + 1]$ is different because the server changes his work velocity during the operation in stage $J_i$. ($J_i \in J$). Therefore, considering Corollary 8.2 and defining $a_{J_i} = \mathbb{N}[k q_{J_i}]$, we will have

$$C_h[J_i \rhd J_i + 1] = h_{J_i} \left[ \sum_{r = k - a_{J_i} + 1}^{k} \left( \sum_{r = 1}^{k - a_{J_i} - 1} \frac{S_{J_i}^{(st)}}{v_i} \right) + \sum_{r = 1}^{k - a_{J_i} - 1} \frac{S_{J_i}^{(st)}}{v_i + 1} \right]$$

$$+ h_{J_i+1} \left[ ( k - a_{J_i} ) \left( \frac{k q_{J_i} - a_{J_i}}{v_i} \right) + ( 1 - k q_{J_i} + a_{J_i} ) \frac{S_{J_i}^{(st)}}{v_i + 1} \right]$$

$$+ h_{J_i+1} \left[ ( k - a_{J_i} ) \left( \frac{k q_{J_i} - a_{J_i}}{v_i} \right) + ( 1 - k q_{J_i} + a_{J_i} ) \frac{S_{J_i}^{(st)}}{v_i + 1} \right]$$

$$+ h_{J_i+1} \left[ ( k - a_{J_i} ) \left( \frac{k q_{J_i} - a_{J_i}}{v_i} \right) + ( 1 - k q_{J_i} + a_{J_i} ) \frac{S_{J_i}^{(st)}}{v_i + 1} \right]$$

which, after some algebra, yields

$$C_h[J_i \rhd J_i + 1] = h_{J_i} \left[ \frac{S_{J_i}^{(st)}}{v_i} \left( \frac{a_{J_i} a_{J_i} + 1}{2} + k q_{J_i} ( k - a_{J_i} ) \right) \right]$$

$$+ h_{J_i} \left[ \frac{S_{J_i}^{(st)}}{v_i} \left( \frac{k + 1}{2} - \frac{a_{J_i} a_{J_i} + 1}{2} - k q_{J_i} ( k - a_{J_i} ) \right) \right]$$

$$+ h_{J_i+1} \left[ \frac{S_{J_i}^{(st)}}{v_i} \left( \frac{a_{J_i} a_{J_i} + 1}{2} + a_{J_i} ( k q_{J_i} - a_{J_i} ) \right) \right]$$

$$+ h_{J_i+1} \left[ \frac{S_{J_i}^{(st)}}{v_i} \left( \frac{k + 1}{2} - \frac{a_{J_i} a_{J_i} + 1}{2} - a_{J_i} ( k q_{J_i} - a_{J_i} ) \right) \right]$$

$$+ h_{J_i+1} \left[ \frac{S_{J_i}^{(st)}}{v_i} \left( \frac{k + 1}{2} \right) \right] + h_{J_i+2} \left[ \frac{S_{J_i+1}^{(st)}}{v_{i+1}} \left( \frac{k + 1}{2} \right) \right]. \quad (8.31)$$

Substituting (8.29) – (8.31) into (8.28), and (8.28) into (8.27), the total costs for batch of size $k$ in each of the parallel queues are obtained. Therefore, the total cost per produced item when the batch size is $k$, $TC(1|k)$, is

$$TC(1|k) = \frac{C_h[1 \rhd N]}{k},$$

and the optimal batch size in each of the $M$ parallel queues, or the optimal bucket size in the balanced bucket brigade will be the integer $k^*$ which satisfies

$$\begin{cases} 
TC(1|k^*) - TC(1|k^* + 1) < 0 \\
TC(1|k^*) - TC(1|k^* - 1) < 0.
\end{cases}$$
Remark 8.2

A simpler approach to approximate the optimal bucket size $k^*$ can be used by assuming a constant actual processing time $\bar{S}_J$ for all items of a batch in work station $J_i \in \mathcal{J}$ as following

$$\bar{S}_J = q_J \frac{S_J^{(st)}}{v_i} + (1 - q_J) \frac{S_J^{(st)}}{v_{i+1}}.$$ 

Thus optimality condition (8.10) can be used to find the optimal bucket size $k^*$. □

Remark 8.3

Bartholdi and Eisenstein [4] claimed that even though they don't have the proof, they believe that if workers are sequenced from slowest to fastest, a bucket brigade with stochastic processing times balances itself and the production rate reaches to maximum. If this is true, then the optimal bucket size which minimizes the total average holding and switching costs can also be obtained for the stochastic version of bucket brigades using the same approach as section 8.6 by considering $\bar{S}_n$ as the average actual processing time of an item in work station $n$ ($n \in \mathcal{N}_U$). □

Example 8.3

Consider a bucket brigade with $N = 10$ work stations and $M = 3$ workers with work velocities $v_1 = 0.9$, $v_2 = 1$, and $v_3 = 1.2$. Also, consider that the holding costs, switching costs and standard processing times of an item in each work station are the same as $h$, $K$ and $\bar{S}$ in Example 8.1. Suppose that the workers were sequenced from slowest to fastest and the bucket brigade balanced itself with no blocking; then the working zone of worker $m$, namely the interval of work content $[l_m^{(w)}, u_m^{(w)}]$, will be, for $m = 1, 2, 3$,

$$[l_1^{(w)}, u_1^{(w)}] = [0.00, \frac{0.9}{3.1}] = [0.00, 0.29]$$

$$[l_2^{(w)}, u_2^{(w)}] = [0.29, \frac{1.9}{3.1}] = [0.29, 0.61]$$

$$[l_3^{(w)}, u_3^{(w)}] = [0.61, \frac{3.1}{3.1}] = [0.61, 1.00].$$

However, work station $n$ can be represented by interval of work content $[l_n^{(st)}, u_n^{(st)}]$ as follows:
Figure 8.4. Working zones of balanced bucket brigade in example 8.3.

\[ [l_n^{(s)}, u_n^{(s)}] = \left[ \frac{\sum_{r=1}^{n-1} S_r^{(st)}}{\sum_{r=1}^{10} S_r^{(st)}}, \frac{\sum_{r=1}^{n} S_r^{(st)}}{\sum_{r=1}^{10} S_r^{(st)}} \right] = \left[ \frac{\sum_{r=1}^{n-1} S_r^{(st)}}{35}, \frac{\sum_{r=1}^{n} S_r^{(st)}}{35} \right]. \]

Therefore,

\[ [l_1^{(s)}, u_1^{(s)}] = [0.00, 0.26] \quad [l_2^{(s)}, u_2^{(s)}] = [0.26, 0.40] \quad [l_3^{(s)}, u_3^{(s)}] = [0.40, 0.43] \]

\[ [l_4^{(s)}, u_4^{(s)}] = [0.43, 0.46] \quad [l_5^{(s)}, u_5^{(s)}] = [0.46, 0.57] \quad [l_6^{(s)}, u_6^{(s)}] = [0.57, 0.80] \]

\[ [l_7^{(s)}, u_7^{(s)}] = [0.80, 0.86] \quad [l_8^{(s)}, u_8^{(s)}] = [0.86, 0.89] \quad [l_9^{(s)}, u_9^{(s)}] = [0.89, 0.97] \]

\[ [l_{10}^{(s)}, u_{10}^{(s)}] = [0.97, 1.00] \]

Figure 8.4 shows the working zones of the workers and work stations in terms of the interval of work contents. According to Figure 8.4, \( \mathcal{J} = \{2, 6, 11\} \), which means that the work velocities are changed in work stations 2, 6 and 11 from 0.9 to 1.1 to 1.2 and 1.2 to 0.9, respectively. Hence

\[ q_2 = \frac{u_1^{(w)} - l_2^{(s)}}{u_2^{(s)} - l_2^{(s)}} = \frac{0.29 - 0.26}{0.40 - 0.26} = 0.21 \]

\[ q_6 = \frac{u_2^{(w)} - l_6^{(s)}}{u_6^{(s)} - l_6^{(s)}} = \frac{0.61 - 0.57}{0.80 - 0.57} = 0.17 \]

Using Corollary 8.1, the actual processing times in work stations \( n \in \mathcal{J}^c \), \( \overline{s}_n \), are

\( \overline{s}_1 = \frac{9}{0.9} = 10 \quad \overline{s}_3 = \frac{1}{1} = 1 \quad \overline{s}_4 = \frac{2}{1} = 2 \quad \overline{s}_5 = \frac{3}{1} = 3 \)
Considering (8.27), we will have
\[
C_h[1 \triangleright 10] = C_h[1 \triangleright 10] + \sum_{r=1}^{9} K_{r,r+1} + K_{10,1} = C_h[1 \triangleright 10] + 990. \quad (8.32)
\]

where
\[
C_h[1 \triangleright 10] = C_h[1 \triangleright 1] + C_h[2 \triangleright 3] + C_h[4 \triangleright 5] + C_h[6 \triangleright 7] + C_h[8 \triangleright 10]. \quad (8.33)
\]

On the other hand.
\[
C_h[1 \triangleright 1] = \frac{k(k+1)}{2} h_1 \overline{S}_1 + \frac{k(k-1)}{2} h_2 \overline{S}_1 = 25k^2 - 5k. \quad (8.34)
\]
\[
C_h[4 \triangleright 5] = \frac{k(k+1)}{2} (h_4 \overline{S}_4 + h_5 \overline{S}_5) + \frac{k(k-1)}{2} (h_4 \overline{S}_4 + h_6 \overline{S}_5) = 47k^2 - 3k. \quad (8.35)
\]
\[
C_h[8 \triangleright 10] = \frac{k(k+1)}{2} (h_8 \overline{S}_8 + h_9 \overline{S}_9 + h_{10} \overline{S}_{10}) + \frac{k(k-1)}{2} (h_9 \overline{S}_8 + h_{10} \overline{S}_9) = 60.76k^2 + 3.31k. \quad (8.36)
\]

However, if \( a_2 = \Re[kq_2] \) and \( a_6 = \Re[kq_6] \), then according to (8.32) we have
\[
C_h[2 \triangleright 3] = h_2 \left\{ \frac{5}{0.9} \left[ a_2(a_2 + 1) \right] + 0.21k(k - a_2) \right\} + h_2 \left\{ \frac{5}{0.9} \left[ \frac{k(k+1)}{2} - a_2(a_2 + 1) - 0.21k(k - a_2) \right] \right\} + h_2 \left\{ \frac{5}{0.9} \left[ a_2(a_2 - 1) + a_2(0.21k - a_2) \right] \right\} + h_2 \left\{ \frac{5}{0.9} \left[ \frac{k(k-1)}{2} - a_2(a_2 - 1) + a_2(0.21k - a_2) \right] \right\} + h_2 \left\{ \frac{1}{1} \left[ \frac{k(k+1)}{2} \right] \right\} + h_4 \left\{ \frac{1}{1} \left[ \frac{k(k-1)}{2} \right] \right\} = 26.35k^2 - 6k + 0.23a_2k - 0.56a_2(a_2 - 1). \quad (8.37)
\]

and
\[
C_h[6 \triangleright 7] = h_6 \left\{ \frac{8}{1} \left[ a_6(a_6 + 1) \right] + 0.17k(k - a_6) \right\}
\]
Substituting (8.34) - (8.38) into (8.33), after some algebra we get

\[
C_h[1 \leq 10] = 252.21k^2 - 16.52k + 0.23k(a_2 + a_6) - 0.56a_2(a_2 + 1) - 0.67a_6(a_6 + 1) + 990.
\]

Therefore,

\[
TC[1/k] = 252.21k - \frac{1}{k}(0.56a_2(a_2 + 1) + 0.67a_6(a_6 + 1) - 990) + 0.23(a_2 + a_6) - 16.52.
\]

which implies that the optimal bucket size is \( k^* = 2 \) with \( TC[1/k^* = 2] = 982.9. \)

As we have shown in this chapter, U-shaped production lines can be analyzed by decomposing them into tandem queues each attended by a moving server. These lines may be decomposed into completely disjointed tandem queues as in Sections 8.4 and 8.5, or exactly similar tandem queues as in Section 8.6. Therefore, it is clear that the characteristics or optimization analysis of tandem queues attended by a moving server can be used as an efficient tool in the analysis of U-shaped production lines.
Chapter 9

Summary

The serial routing of customers in polling systems creates a new feature in those systems in which a job is generated at queue $i$ upon completion of a job in queue $i - 1$. This job generation feature mostly appears in production or assembly lines where raw materials enter the line at work station 1 and each work station $i$ in the line receives its job from the previous work station $i - 1$. Therefore, the version of polling systems which can be used to model the production or assembly line, will be tandem queues attended by moving servers. These models are not able to be analyzed in terms of traditional production lines, because in traditional lines the number of workers in the line is at least equal to the number of work stations. However, a new type of production line in which the number of workers is less than the number of work stations has recently become very popular. These lines are called U-shaped production lines, and the main reason for their popularity is that they are highly flexible and can adapt to changes in demand. U-shaped production lines with $N$ work stations and $M$ workers ($M < N$) are actually $N$ multi-stage tandem queues each attended by a moving server, where the behaviour of each queue and interactions among these queues determine the behaviour of the line. This fact, along with the other applications of tandem queues attended by a moving server in manufacturing and communications,
and also the lack of literature on these type of queues have motivated this research to
deal with the optimization analysis of these systems especially when switching costs
and switchover times are involved.

This research starts with the optimization analysis of a two-stage tandem queue
attended by a moving server as a basic model, and then the problem is extended to
an N-stage tandem queue attended by a moving server with holding and switching
costs and switchover times. Finally, by introducing a general definition and classifi-
cation of U-shaped lines as tandem queues attended by moving servers, the effect of
switching costs and switchover times on these lines are studied using the results from
the optimization analysis of tandem queues attended by a moving server. The whole
procedure is organized in eight chapters.

In Chapter 1 tandem queues attended by a moving server are introduced and the
literature on performance and optimization analysis of these systems are described.
Then the basic two-stage tandem queue, an $M/G_1 - G_2/1$ queue, and the basic semi-
Markov decision model are presented to be used as the framework of the analysis of
two-stage tandem queues in the rest of the thesis.

The optimal policy in the second stage of a $G/G_1 - G_2/1$ queue is characterized in
Chapter 2. It is shown that the optimal policy in the second stage which minimizes
the total discounted and long-run average holding and switching costs is a greedy
policy, and if the holding cost rate in the second stage is greater than the rate in the
first stage, then the optimal policy is also an exhaustive policy.

Three different policies for the first stage of an $M/G_1 - G_2/1$ queue are introduced
in this research based on the capability of the server to adapt to changes in the system
and the availability of information about the number of customers in the system after
each service completion in stage 1. For cases in which no information about the
number of customers in the first stage is available, or the server is not able to adapt
to different states of the system, the static policy is defined. Using the static policy
the server always serves a predetermined number $M_5$ of customers in stage 1 before
switching to stage 2. For situations in which the server has partial information about
the number of customers in the first stage, the semi-dynamic policy is introduced.
In a partial information environment, the server has information about the number of customers in stage 1 only when he arrives there. Therefore, according to a semi-dynamic policy, he must choose some customers from among those present to serve upon his arrival. When the server has complete information about the number of customers in the system at each service completion epoch, then he has the opportunity to decide about his next action (i) continue to serve at the present stage. (ii) switch to the other stage, or (iii) be idle. This is called a dynamic policy. It should be noted that in static, semi-dynamic and dynamic policies it is assumed that the server applies a greedy and exhaustive policy in stage 2.

Two models are developed in Chapter 3 to find the optimal static policy which minimizes the total long run average holding and switching costs per unit time in an \( M/G_1 - G_2/1 \) queue. The second model is an \( M/G/1 \) queue with cyclic service times which has a broad range of applications other than the ability to analyze tandem queues attended by a moving server with a static policy. This model can be used to study a type of queueing system with vacations, where the vacation starts after serving a predetermined number of service rather than at the end of a busy period.

Chapter 4 is devoted to a class of semi-dynamic policy called the gated-limited policy which has the monotonicity property related to the number of customers in stage 1 at each decision epoch. Using a gated-limited policy with parameter \( M_G \), when the server arrives at stage 1, he chooses \( a = \min\{n, M_G\} \) customers to serve in the next cycle, where \( n \) is the number of customers in stage 1 at his arrival epoch and \( M_G \) is a predetermined number. For systems with switchover times, a model is developed to obtain the optimal limit \( M^*_G \) for an \( M/G_1 - G_2/1 \) queue to minimize the total long run average holding and switching costs. However, when switchover times are zero, it is shown that the optimal limit of the gated-limited policy is almost independent of the arrival rate, and therefore it can be obtained by analyzing the two-stage tandem queue in high traffic intensity.

Double-threshold policies are introduced and studied as a class of dynamic policies in Chapter 4. Applying a double-threshold policy with parameters \( M_L \) and \( M_U \), the server continues serving in stage 1 either until \( M_U \) services have been completed.
without interruption, or until the first stage becomes empty and there are at least $M_e$ customers in stage 2, whichever comes first. Double-threshold policies are highly flexible because they present (i) sequential policy when $M_e = M_n = 1$, (ii) limited policy when $M_e = 1$, (iii) static policy when $M_e = M_n$, and (iv) exhaustive policy when $M_e = 1$, $M_n \to \infty$. A model is developed in Chapter 5 to find the optimal parameters $M_e^*$ and $M_n^*$ for systems with switchover times to minimize the total long run average holding and switching costs. Then, these parameters are compared to optimal parameters of limited, gated-limited and static policies using some numerical examples.

In Chapter 6, greedy and exhaustive service policies downstream of stage 2 are defined for N-stage tandem queues attended by a moving server. Then, based on the results which were obtained for the $M/G_1 - G_2/1$ queue, three models are developed to obtain the optimal static, gated-limited and double-threshold policies for an $M/G_1 - G_2 - \cdots - G_N/1$ queue.

An N-stage tandem queue attended by moving server with batch arrivals is analyzed in Chapter 7 to find the optimal policy which minimizes the total average holding and switching cost. First, an efficient operational tool is developed which leads to the properties of greedy and exhaustive policies in tandem queues attended by a moving server. Then, two heuristic algorithms are presented to approximate the optimal policy; and finally, using numerical examples, it is shown that these algorithms are efficient and accurate.

Tandem queues attended by moving servers or U-shaped production lines are studied in Chapter 8. A general definition for U-shaped production lines is presented in this chapter which covers different aspects of these lines. Also, a classification of U-shaped lines is introduced which organizes the previous studies and leads to open problems which have not yet been analyzed. Finally, by decomposing U-shaped lines into tandem queues attended by a moving server, the effects of switching costs and switchover times on some types of these lines are studied.

Areas for further research on tandem queues attended by moving servers are wide open. The optimization analysis of tandem queues with non-Poissonian arrivals under
different policies are still an open problem.

As was described, the optimal limits of limited and gated-limited policies are almost independent of the arrival rate in a two-stage tandem queues with zero switchover times. This independence property can be further investigated in a more general case as an independence between the parameters of the optimal nonidling policy and the arrival process. Because, when nonidling policies such as limited and gated-limited policies are applied, the server never waits for a customer in stage 1 unless the system is empty. This behaviour actually decreases the correlation between interarrival times and the parameters of optimal nonidling policies.

Another area for further research is to characterize the optimal policy in the \( G^{(x)} / G_1 - G_2 - \ldots - G_N / 1 \) queue with holding and switching costs, or to extend the algorithms to obtain the optimal or suboptimal policy for the system with switchover times.

Obviously, the more interesting and attractive field for further research is U-shaped production lines. The amount of literature on U-shaped production lines started to increase in 1991 in an attempt to analyze the behaviour the TSS lines. The advantages of U-shaped production lines over traditional production lines encourages more manufacturing companies to employ these lines every day. Therefore, further investigations on the performance of these lines will be required in the near future. Further studies on U-shaped lines can be carried out on the issue of their design. These would attempt to find optimal characteristics of the line such as the optimal number of work stations, optimal number of workers, optimal working zones and optimal operational rules.
Appendix A

Average Number of Customers in the $M/G_1 - G_2/1$ Queue with Static Policy

A.1 Average Number of Customers in Stage 1

Considering (3.48) and $L_1 = \lambda E[W_1]$, the average number of customers in stage 1, $L_1$, is

$$L_1 = \frac{d}{dx} \Pi_1(x,1) \bigg| _{x=1}. \quad (A.1)$$

On the other hand, using (3.12)

$$\frac{d^2}{dx^2} \{\Pi_1(x,1)[x - P_1(x)]\} \bigg| _{x=1} = \frac{d^2}{dx^2} \{P_1(x)D_2(x)\Pi_2(x,0) - P_1(x)\Pi_1(0,1)$$

$$- P_1(x)\Psi_{M_1}(x,1) - (1 - x)\Pi_1(x)\eta_1^{(1)}d_0^{(1)}$$

$$+ xP_1(x)\Theta_{M_1-1}(1) \} \bigg| _{x=1}. \quad (A.2)$$

For the left hand side of (A.2), we have

$$\frac{d^2}{dx^2} \Pi_1(x,1)[x - P_1(x)] \bigg| _{x=1} = (1 - \rho_1)2P_1''(1,1) - \frac{1}{2}P_1''(1)$$

$$= (1 - \rho_1)L_1 - \frac{P_1''(1)}{2}. \quad (A.3)$$

For the first term of the right-hand side of (A.2), we have

$$\frac{d^2}{dx^2} \{P_1(x)D_2(x)\Pi_2(x,0)\} \bigg| _{x=1} = \Pi_2(1,0)[P_1''(1) + D_1''(1) + 2\rho_1\eta_2]$$

$$+ 2(\rho_1 + \eta_2)\Pi_2''(1,0) + \Pi_2''(1,0). \quad (A.4)$$
Also, for the other terms of the right-hand side of (A.2), we get

\[
\frac{d^2}{dz^2}[-P_1(x)\Pi_1(0,1)] \bigg|_{x=1} = -P_1''(1)\Pi_1(0,1) \tag{A.5}
\]

\[
\frac{d^2}{dz^2}[-P_1(x)\Psi_M(x,1)] \bigg|_{x=1} = -P_1''(1)\Psi_M(1,1) - 2\rho_1\Psi_M'(1,1) - \Psi_M''(1,1) \tag{A.6}
\]

\[
\frac{d^2}{dz^2}[-(1-x)P_1(x)\pi_0^2(1)\omega] \bigg|_{x=1} = 2\rho_1\pi_0^2(1)\omega \tag{A.7}
\]

\[
\frac{d^2}{dz^2}[xP_1(x)\Theta_{M-1}(1)] \bigg|_{x=1} = \Theta_{M-1}(1)[P_1''(1) + 2\rho_1]. \tag{A.8}
\]

Substituting (A.3) to (A.8) into (A.2), after some algebra we get

\[
L_1 = \frac{1}{1 - \rho_1}\{\Pi_2(1,0)[P_1''(1) + D_1''(1) + 2\rho_1\eta_2] - P_1''(1)[\Pi_1(0,1) + \Psi_M'(1,1)] + \Theta_{M-1}(1)[P_1''(1) + 2\rho_1] + 2\Pi_2'(1,0)[\rho_1 + \eta_2] - 2\rho_1\Psi_M'(1,1) + 2\rho_1\pi_0^2(1)\omega - \frac{P_1''(1)}{2}\}. \tag{A.9}
\]

where \( P_i''(1) = \lambda^2E[S_i^2] \), \( D_i''(1) = \lambda^2E[D_i^2] \) and \( \omega = \Psi_M''(1,1) - \Pi_2''(1,0) \). To complete the calculation of \( L_1 \), the unknowns \( \omega \), \( \Psi_M'(1,1) \) and \( \Pi_2'(1,0) \) should be found.

### A.1.1 Determination of \( \Psi_M'(1,1) \)

Considering (3.27), we have

\[
\frac{d}{dx}[\Psi_M(x,1)x^{(M)}] \bigg|_{x=1} = \frac{d}{dx}\{D_1(x)[P_1(x)]^{M}\Pi_2(x,0) - (1-x)[P_1(x)]^{M}\pi_0^2(1)\omega - [P_1(x)]^{M}\Theta_{M-1}(1) + x[P_1(x)]^{M}\Theta_{M-1}(1)\} \bigg|_{x=1}. \tag{A.10}
\]

The left-hand side of (A.10) can be written as

\[
\frac{d}{dx}[\Psi_M(x,1)x^{(M)}] \bigg|_{x=1} = M\Psi_M(1,1) + \Psi_M'(1,1). \tag{A.11}
\]
For the right hand side of (A.10), we get

\[
\frac{d}{dx} [D_2(x)[P_1(x)]^M_2 \Pi_2(x, 0)] \bigg|_{x=1} = (M_2 \rho_1 + \eta_2) \Pi_2(1, 0) + \Pi'_2(1.0) \quad (A.12)
\]

\[
\frac{d}{dx} [-(1-x)[P_1(x)]^M_2 \pi_0^2 d_0(2)] \bigg|_{x=1} = \pi_0^2 d_0(2) \quad (A.13)
\]

\[
\frac{d}{dx} [-[P_1(x)]^M_2 \Theta_{M_2}(\frac{x}{P_1(x)})] \bigg|_{x=1} = -M_2 \rho_1 \Theta_{M_2}(1) - (1 - \rho_1) \Theta'_{M_2}(1) \quad (A.14)
\]

\[
\frac{d}{dx} [x[P_1(x)]^M_2 \Theta_{M_2-1}(\frac{x}{P_1(x)})] \bigg|_{x=1} = (1 + M_2 \rho_1) \Theta_{M_2-1}(1) + (1 - \rho_1) \Theta'_{M_2-1}(1)
\]

(A.15)

Substituting (A.11) - (A.15) into (A.10), after some algebra we obtain

\[
\Psi'_{M_2}(1,1) = (M_2 \rho_1 + \eta_2) \Pi_2(1,0) - M_2 \Psi_{M_2}(1,1) + \pi_0^2 d_0(2)
\]

\[+ \Theta_{M_2-1}(1) - M_2 \pi_0^1 + \Pi'_2(1.0). \quad (A.16)
\]

A.1.2 Determination of \( \omega \)

Considering the second derivative of the both sides of (3.27), we have

\[
\frac{d^2}{dx^2} [\Psi_{M_2}(x,1)x^{M_2}] \bigg|_{x=1} = \frac{d^2}{dx^2} \{D_2(x)[P_1(x)]^M_2 \Pi_2(x,0)
\]

\[-(1-x)[P_1(x)]^M_2 \pi_0^2 d_0(2)
\]

\[-[P_1(x)]^M_2 \Theta_{M_2}(\frac{x}{P_1(x)})
\]

\[+ x[P_1(x)]^M_2 \Theta_{M_2-1}(\frac{x}{P_1(x)}) \bigg|_{x=1}. \quad (A.17)
\]

For the left-hand side we get

\[
\frac{d^2}{dx^2} [\Psi_{M_2}(x,1)x^{M_2}] \bigg|_{x=1} = M_2(M_2 - 1) \Psi_{M_2}(1,1) + 2M_2 \Psi'_{M_2}(1,1) + \Psi''_{M_2}(1,1)
\]

(A.18)

For the first term of the right-hand side of (A.17), we get

\[
\frac{d^2}{dx^2} [D_2(x)[P_1(x)]^M_2 \Pi_2(x,0)] \bigg|_{x=1} = [M_2 P_1''(1) + D_2''(1) + 2M_2 \rho_1 \eta_2
\]

\[+ M_2(M_2 - 1) \rho_1^2 \Pi_2(1,0)
\]

\[+ [2M_2 \rho_1 + 2 \eta_2] \Pi'_2(1,0) + \Pi''_2(1,0), \quad (A.19)
\]
and for the second term,

\[
\frac{d^2}{dx^2}[-(1-x)[P_1(x)]^{M_s\pi_0^{(2)}d_0^{(2)}]} \bigg|_{x=1} = 2M_s\rho_1\pi_0^{(2)}d_0^{(2)}. \tag{A.20}
\]

Finally, for the last two terms of the right hand side of (A.17), we obtain

\[
\frac{d^2}{dx^2}[-[P_1(x)]^{M_s}\Theta_{M_s}(\frac{x}{P_1(x)})] \bigg|_{x=1} = [-M_sP_1''(1) - M_s(M_s - 1)\rho_1^2]\Theta_{M_s}(1)
+ [P_1''(1) - 2\rho_1(1 - \rho_1)(M_s - 1)]\Theta'_{M_s}(1)
- (1 - \rho_1)^2\Theta''_{M_s}(1) \tag{A.21}
\]

\[
\frac{d^2}{dx^2}[x[P_1(x)]^{M_s}\Theta_{M_s-1}(\frac{x}{P_1(x)})] \bigg|_{x=1} = [M_sP_1''(1) + 2M_s\rho_1
+ M_s(M_s - 1)\rho_1^2]\Theta_{M_s-1}(1)
+ [2(1 + M_s\rho_1)(1 - \rho_1) - P_1''(1)
- 2\rho_1(1 - \rho_1)]\Theta'_{M_s-1}(1)
+ (1 - \rho_1)^2\Theta''_{M_s-1}(1). \tag{A.22}
\]

Substituting (A.18) - (A.22) into (A.17), after some algebra we obtain

\[
\omega = [M_sP_1''(1) + D_2''(1) + 2M_s\rho_1\eta_2 + M_s(M_s - 1)\rho_1^2]\Pi_2(1, 0)
- M_s(M_s - 1)\Psi_{M_s}(1, 1) + 2M_s\rho_1[\pi_0^{(2)}d_0^{(2)} + \Theta_{M_s-1}(1)] - M_s(M_s - 1)\pi_0^{(1)}M_s
+ 2(1 - \rho_1)\Theta'_{M_s-1}(1) + 2(M_s\rho_1 + \eta_2)\Pi_2'(1, 0) - 2M_s\Psi_{M_s}'(1, 1). \tag{A.23}
\]

### A.1.3 Determination of $\Pi_2'(1, 0)$

Using (3.30) yields

\[
\frac{d^2}{dx^2}[\Pi_2(x, 0)f(x)] \bigg|_{x=1} = \frac{d^2}{dx^2}[(x - 1)D_1(x)[P_1(x)P_2(x)]^{M_s\pi_0^{(2)}d_0^{(2)}}
+ (x - 1)D_1(x)[P_1(x)P_2(x)]^{M_s\Theta_{M_s-1}(\frac{x}{P_1(x)})}] \bigg|_{x=1}. \tag{A.24}
\]

where $f(x) = x^{M_s} - D_1(x)D_2(x)[P_1(x)P_2(x)]^{M_s}$. Considering the left-hand side of (A.24), we will have

\[
\frac{d^2}{dx^2}[\Pi_2(x, 0)f(x)] \bigg|_{x=1} = f(1)\Pi_2''(1, 0) + 2f'(1)\Pi_2'(1, 0) + f''(1)\Pi_2(1, 0),
\]
and, since \( f(1) = 0 \) and \( f'(1) = M_S(1 - \bar{\rho}) \), then

\[
\left. \frac{d^2}{dx^2} \{ \Pi_2(x, 0) f(x) \} \right|_{x=1} = f''(1) \Pi_2(1, 0) + 2 M_S(1 - \bar{\rho}) \Pi_2'(1, 0). \tag{A.25}
\]

where

\[
f''(1) = M_S(M_S - 1)(1 - \rho^2) - [D_1''(1) + D_2''(1) + M_S(P_1''(1) + P_2''(1))] - 2 M_S(\eta \rho + \rho_1 \rho_2) - 2 \eta_1 \eta_2.
\]

On the other hand, for the terms in the right-hand side of (A.24) we have

\[
\left. \frac{d^2}{dx^2} \left[ (x - 1) D_1(x) [P_1(x) P_2(x)] M_S \pi_0^{(2)} d_0^{(2)} \right] \right|_{x=1} = 2(\eta_1 + M_S \rho) \pi_0^{(2)} d_0^{(2)}. \tag{A.26}
\]

and

\[
\left. \frac{d^2}{dx^2} \left[ (x - 1) D_1(x) [P_1(x) P_2(x)] M_S \Theta_{M_S-1} \left( \frac{x}{P_1(x)} \right) \right] \right|_{x=1} = 2(\eta_1 + M_S \rho) \Theta_{M_S-1}(1) + 2(1 - \rho_1) \Theta'_{M_S-1}(1). \tag{A.27}
\]

Substituting (A.25) - (A.27) into (A.24), after some algebra we get

\[
\Pi_2'(1, 0) = \frac{1}{2 M_S(1 - \bar{\rho})} \left\{ 2(\eta_1 + M_S \rho) \left[ \pi_0^{(2)} d_0^{(2)} + \Theta_{M_S-1}(1) \right] + 2(1 - \rho_1) \Theta'_{M_S-1}(1) - \Pi_2(1, 0) f''(1), \right\} \tag{A.28}
\]

and this completes all the requirements to obtain the average number of customers in the first stage.

**A.2 Average Number of Customers in Stage 2**

For the average number of customers in the second stage, \( L_2 \), we have

\[
L_2 = L_{1,2} - L_1 = 2 \Pi_2'(x, x) \bigg|_{x=1} - L_1. \tag{A.29}
\]
Using (3.15) yields
\[
\frac{d^2}{dx^2}[\Pi_2(x,x)(x - P_2(x))] \bigg|_{x=1} = \frac{d^2}{dx^2}\left\{P_2(x)D_1(x)\Psi_{M_5}(x,x) + P_2(x)D_1(x)\pi^{(1)}_{0M_5}x^{M_5} - P_2(x)\Pi_2(x,0)\right\} \bigg|_{x=1}. \tag{A.30}
\]

For the left-hand side of (A.30) we obtain
\[
\frac{d^2}{dx^2}[\Pi_2(x,x)(x - P_2(x))] \bigg|_{x=1} = (1 - \rho_2) L_{1,2} - \frac{P_2''(1)}{2}. \tag{A.31}
\]

For the terms in the right-hand side of (A.30), we have
\[
\frac{d^2}{dx^2}[P_2(x)D_1(x)\Psi_{M_5}(x,x)] \bigg|_{x=1} = [P_2''(1) + D_1''(1) + 2\rho_2\eta_1]\Psi_{M_5}(1,1) + 2[\rho_2 + \eta_1]\Psi_{M_5}'(x = 1, x = 1) + \Psi_{M_5}''(x = 1, x = 1) \tag{A.32}
\]
\[
\frac{d^2}{dx^2}[P_2(x)D_1(x)\pi^{(1)}_{0M_5}x^{M_5}] \bigg|_{x=1} = [P_2''(1) + D_1''(1) + 2\rho_2\eta_1 + 2M_5(\rho_2 + \eta_1) + M_5(M_5 - 1)]\pi_{0M_5}^{(1)}. \tag{A.33}
\]

and
\[
\frac{d^2}{dx^2}[-P_2(x)\Pi_2(x,0)] \bigg|_{x=1} = -P_2''(1)\Pi_2(1,0) - 2\rho_2\Pi_2'(1,0) - \Pi_2''(1,0). \tag{A.34}
\]

Substituting (A.31) – (A.34) into (A.30), after some algebra we get
\[
L_{1,2} = \frac{1}{1 - \rho_2}\left\{[P_2''(1) + D_1''(1) + 2\rho_2\eta_1]\Psi_{M_5}(1,1) - P_2''(1)\Pi_2(1,0)
- 2\rho_2\Pi_2'(1,0) + 2(\rho_2 + \eta_1)\Psi_{M_5}'(x = 1, x = 1)
+ [P_2''(1) + D_1''(1) + 2\rho_2\eta_1 + 2M_5(\rho_2 + \eta_1) + M_5(M_5 - 1)]\pi_{0M_5}^{(1)} - \delta + \frac{P_2''(1)}{2}\right\}, \tag{A.35}
\]

where
\[
\delta = \Pi_2''(1,0) - \Psi_{M_5}''(x = 1, x = 1)
\]

To complete the calculation of \(L_{1,2}\), the unknowns \(\delta\) and \(\Psi_{M_5}'(x = 1, x = 1)\) must be found.
A.2.1 Determination of $\Psi'_{M_S}(x = 1, x = 1)$

Considering (3.27) it is clear that

$$\Psi_{M_S}(x, x) = x^{M_S} \Psi_{M_S}(x, 1). \quad (A.36)$$

Therefore

$$\Psi'_{M_S}(x = 1, x = 1) = M_S \Psi_{M_S}(1, 1) + \Psi'_{M_S}(1, 1).$$

where $\Psi'_{M_S}(1, 1)$ can be found using (A.16).

A.2.2 Determination of $\delta$

Using (A.36) we have

$$\Psi''_{M_S}(x = 1, x = 1) = M_S(M_S - 1) \Psi_{M_S}(1, 1) + 2M_S \Psi'_{M_S}(1, 1) + \Psi''_{M_S}(1, 1).$$

or

$$\Psi''_{M_S}(1, 1) = \Psi''_{M_S}(x = 1, x = 1) - M_S(M_S - 1) \Psi_{M_S}(1, 1) - 2M_S \Psi'_{M_S}(1, 1), \quad (A.37)$$

and considering the definition of $\omega$, using (A.37), we get

$$\omega = \Psi''_{M_S}(1, 1) - \Pi''(1, 0)$$

$$= \Psi''_{M_S}(x = 1, x = 1) - M_S(M_S - 1) \Psi_{M_S}(1, 1) - 2M_S \Psi'_{M_S}(1, 1) - \Pi''(1, 0).$$

and so

$$\delta = -[\omega + M_S(M_S - 1) \Psi_{M_S}(1, 1) + 2M_S \Psi'_{M_S}(1, 1)]. \quad (A.38)$$
Appendix B

Average Number of Customers in the $M/G_1 - G_2/1$ Queue with Gated-Limited Policy

B.1 Average Number of Customers in Stage 1

Suppose $L_1$ is the average number of customers in $Q_0$ and stage 1. Then considering that

$$E[L_1] = 2 \frac{d}{dx} \Pi_1(x, x, 1) \bigg|_{x=1}. \quad (B.1)$$

and using (4.27), we have

$$\Pi_1(x, x, 1)[x - P_1(x)] = P_1(x)D_2(x)\Pi_2(x, 0, 0) - P_1(x)\Pi_1(x, 0, 1) - (1 - x)\pi_0^{(2)}d_0^{(2)}$$

and therefore,

$$\frac{d^2}{dx^2}[\Pi_1(x, x, 1)[x - P_1(x)] \bigg|_{x=1} = \frac{d^2}{dx^2}[P_1(x)D_2(x)\Pi_2(x, 0, 0) - P_1(x)\Pi_1(x, 0, 1)$$

$$- (1 - x)\pi_0^{(2)}d_0^{(2)}] \bigg|_{x=1}. \quad (B.2)$$

For the left-hand side of (B.2), we have

$$\frac{d^2}{dx^2}[\Pi_1(x, x, 1)[x - P_1(x)] \bigg|_{x=1} = (1 - \rho_1)\Pi_1 + \frac{P''_1(1)}{2}, \quad (B.3)$$

and for the first term of the right-hand side of (B.2), we get

$$\frac{d^2}{dx^2}[P_1(x)D_2(x)\Pi_2(x, 0, 0)] \bigg|_{x=1} = [P''_1(1) + 2\rho_1\eta_2 + D''_2(1)]\Pi_2(1, 0, 0)$$

232
+2(\rho_1 + \eta_2)\Pi_2^{\prime} (1.0, 0) + \Pi_2^{\prime\prime} (1.0, 0).

(B.4)

Considering the second term of the right-hand side of (B.2).

\[
\left. \frac{d^2}{dx^2}[-P_1(x)\Pi_1(x, 0.1)] \right|_{x=1} = -P'_{1}(1)\Pi_{1}(1.0, 1) - 2\rho_1 \Pi_1^{\prime} (1.0, 1) - \Pi_1^{\prime\prime} (1.0, 1).
\]

(B.5)

and finally for the last term of the right-hand side of (B.2), we obtain

\[
\left. \frac{d^2}{dx^2}[-(1-x)P_1(x)\pi_{000}^{(2)}d_0^{(2)}] \right|_{x=1} = 2\rho_1 \pi_{000}^{(2)}d_0^{(2)}.
\]

(B.6)

Substituting (B.3) – (B.6) into (B.2), after some algebra we get

\[
L_1 = \frac{1}{1 - \rho_1} \{\Pi_2(1.0, 0)[D_2^{\prime\prime} (1) + 2\rho_1 \eta_2] + 2(\rho_1 + \eta_2)\Pi_2^{\prime} (1.0, 0) - \Delta \\
-2\rho_1 \Pi_1^{\prime} (1.0, 1) + 2\rho_1 \pi_{000}^{(2)}d_0^{(2)} + \frac{P''_{1}(1)}{2} \}.
\]

(B.7)

where

\[
\Delta = \Pi_1^{\prime\prime} (1.0, 1) - \Pi_2^{\prime\prime} (1.0, 0).
\]

To complete the calculations of \(L_1\), unknowns \(\Pi_2^{\prime} (1.0, 0)\), \(\Pi_1^{\prime} (1.0, 1)\) and \(\Delta\) must be found.

### B.1.1 Determination of \(\Pi_2^{\prime} (1.0, 0)\) and \(\Pi_1^{\prime} (1.0, 1)\)

Let \(f(x) = x^{M_0} - D_1(x)D_2(x)[P_1(x)P_2(x)]^{M_0}\). Then considering (4.30), we have

\[
\left. \frac{d^2}{dx^2}[\Pi_2(x, 0, 0)f(x)] \right|_{x=1} = \left. \frac{d^2}{dx^2}[-D_1(x)[P_1(x)P_2(x)]^{M_0}\Phi_{M_0-1}(x) \\
+ x^{M_0}D_1(x)\Phi_{M_0-1}(P_1(x)P_2(x)) \\
- x^{M_0}D_1(x)[1 - P_1(x)P_2(x)]\pi_{000}^{(2)}d_0^{(2)}] \right|_{x=1}.
\]

(B.9)

For the left-hand side of (B.9), we have

\[
\left. \frac{d^2}{dx^2}[\Pi_2(x, 0, 0)f(x)] \right|_{x=1} = \Pi_2(1, 0, 0)f''(1) + 2f'(1)\Pi_2^{\prime} (1, 0, 0),
\]

(B.10)
where it can be shown that \( f'(1) = M_G(1 - \bar{p}) \) and

\[
f''(1) = M_G(M_G - 1)(1 - p^2) - (D''_1(1) + D''_2(1) + M_G[P''_1(1) + P''_2(1)]) - 2M_G(\eta \rho + \rho_1 \rho_2) - 2\eta_1 \eta_2.
\]

However, for the first term of the right-hand side of (B.9), we obtain

\[
\frac{d^2}{dx^2}[-D_1(x)[P_1(x)P_2(x)]^{M_G} \Phi_{M_G-1}(x)] \bigg|_{x=1} = \Phi_{M_G-1}(1)\{-D''_1(1) - M_G[P''_1(1) + P''_2(1)] - M_G(M_G - 1) \rho^2 - 2M_G(\eta_1 \rho + \rho_1 \rho_2)\} - 2\Phi'M_{M_G-1}(1)[\eta_1 + M_G \rho] - \Phi''_{M_G-1}(1).
\]

(B.11)

and for the second term.

\[
\frac{d^2}{dx^2}[x^{M_G}D_1(x) \Phi_{M_G-1}(P_1(x)P_2(x))] \bigg|_{x=1} = \Phi_{M_G-1}(1)[D''_1(1) + 2M_G \eta_1 + M_G(M_G - 1)] + \Phi'M_{M_G-1}(1)[P''_1(1) + P''_2(1)] + 2(\rho_1 \rho_2 + M_G \rho + \eta_1 \rho) + \rho^2 \Phi''_{M_G-1}(1).
\]

(B.12)

Finally, for the last term of the right-hand side of (B.9), we have

\[
\frac{d^2}{dx^2}[-x^{M_G}D_1(x)[1 - P_1(x)P_2(x)]\pi^{(2)}_{000d_0^{(2)}}] \bigg|_{x=1} = [P''_1(1) + P''_2(1)] + 2(\rho_1 \rho_2 + M_G \rho + \eta_1 \rho)]\pi^{(2)}_{000d_0^{(2)}}.
\]

(B.13)

Substituting (B.10) - (B.13) into (B.9), after some algebra,

\[
\Pi'_2(1, 0, 0) = \frac{-1}{2M_G(1 - \bar{p})} \{ \Phi_{M_G-1}(1)[M_G P''_1(1) + M_G P''_2(1)]
\]
\(-M_G(M_G - 1)(1 - \rho^2)\)
\(-2M_G\eta_1(1 - \rho) + 2M_G \rho_1 \rho_2 \]
\(+\Phi''_{M_G-1}(1) \left[ 2\eta_1(1 - \rho) - P_1''(1) - P_2''(1) - 2\rho_1 \rho_2 \right] \]
\(+\Phi''_{M_G-1}(1) \left[ 1 - \rho^2 \right] + f''(1)\Pi_2(1,0,0) \]
\(-[P_1''(1) + P_2''(1) + 2(M_G + \eta_1)\rho + 2\rho_1 \rho_2 \] \pi_{000d_0}^{(2)} \} \right). \]

(B.14)

Now, \(\Pi_1''(1,0,1)\) can be found using equation (4.38) from

\[
\Pi_1''(1,0,1) = -\frac{\rho_2^2}{2} - \eta_1 \Pi_2(1,0,0) + \Pi_2''(1,0,0). \tag{B.15}
\]

**B.1.2 Determination of \(\Delta\)**

To find \(\Delta\), the second derivative of equation (4.28) should first be taken:

\[
\frac{d^2}{dx^2}[x^{M_G}\Pi_1(x,0,1)] \bigg|_{x=1} = \frac{d^2}{dx^2}\{[P_1(x)]^{M_G}D_2(x)\Pi_2(x,0,0) \]
\(-[P_1(x)]^{M_G}\Phi_{M_G-1}(x) \]
\(+x^{M_G}\Phi_{M_G-1}(P_1(x)) \]
\(-x^{M_G}(1 - P_1(x))\pi_{000d_0}^{(2)} \} \bigg|_{x=1}. \tag{B.16}
\]

For the left-hand side of (B.16) we have

\[
\frac{d^2}{dx^2}[x^{M_G}\Pi_1(x,0,1)] = \Pi_1''(1,0,1) + 2M_G\Pi_1''(1,0,1) + M_G(M_G - 1)\Pi_1(1,0,1). \tag{B.17}
\]

However, for the first term of the right-hand side of (B.16), we obtain

\[
\frac{d^2}{dx^2}[D_2(x)[P_1(x)]^{M_G}\Pi_2(x,0,0)] \bigg|_{x=1} = \Pi_2(1,0,0)[M_GP_1''(1) + D_2''(1) \]
\(+M_G(M_G - 1)\rho_1^2 + 2M_G \rho_1 \eta_2 \]
\(+\Pi_2''(1,0,0)2M_G \rho_1 + 2\eta_2 \]
\(+\Pi_2''(1,0,0), \tag{B.18}
\]
and also for the second term, we obtain
\[
\frac{d^2}{dx^2}[-[P_1(x)]^{MG} \Phi_{MG-1}(x)] \bigg|_{x=1} = -\Phi_{MG-1}'(1)[MG P''_1(1) \\
+ MG(M_G - 1) \rho_1^2] - \Phi_{MG-1}''(1)[2MG \rho_1] \\
- \Phi_{MG-1}''(1). 
\] (B.19)

Considering that
\[
\Phi_{MG-1}'(P_1(x)) \bigg|_{x=1} = \rho_1 \Phi_{MG-1}'(1) \\
\Phi_{MG-1}''(P_1(x)) \bigg|_{x=1} = \rho_1^2 \Phi_{MG-1}''(1) + P_1''(1) \Phi_{MG-1}'(1),
\]
for the third term of the right-hand side of (B.16), we get
\[
\frac{d^2}{dx^2}[x^{MG} \Phi_{MG-1}(P_1(x))] \bigg|_{x=1} = MG(M_G - 1) \Phi_{MG-1}(1) \\
+ [2MG \rho_1 + P_1''(1)] \Phi_{MG-1}'(1) \\
+ \rho_1^2 \Phi_{MG-1}''(1). 
\] (B.20)

Substituting (B.17) - (B.21) into (B.16), and noting that \( \Pi_1(1,0,1) = \Pi_2(1,0,0) \), after some algebra, we obtain
\[
\Delta = \Pi_2(1,0,0) [MG P''_1 + D''_2(1) + 2MG \rho_1 \eta_2 - MG(M_G - 1)(1 - \rho_1^2)] \\
+ \Pi_2'(1,0,0)[2MG \rho_1 + \eta_2] - 2MG \Pi_2'(1,0,1) \\
- \Phi_{MG-1}(1) [MG P''_1(1) - MG(M_G - 1)(1 - \rho_1^2)] + P_1''(1) \Phi_{MG-1}'(1) \\
- \Phi_{MG-1}''(1) [1 - \rho_1^2] + [2MG \rho_1 + P_1''(1)] \pi_{000d_0}^{(2)}d_{d_0}^{(2)}.
\] (B.21)

and this completes the requirements to calculate the average number of customers in \( Q_0 \) and the first stage.

**B.2 Average Number of Customers in Stage 2**

The average number of customers in the second stage, \( L_2 \), is
\[
L_2 = L_{1,2} - L_1 \\
= 2 \frac{d}{dx} [\Pi_2(x,0,x)] \bigg|_{x=1} - L_1, 
\] (B.22)
where $L_{1,2}$ is the average number of customers in queue $Q_0$, stage 1 and stage 2. Using (4.12), we have

$$
\frac{d^2}{dx^2} \left\{ [x - P_2(x)] \Pi_2(x, 0, x) \right\} \bigg|_{x=1} = \frac{d^2}{dx^2} \left\{ P_2(x) D_1(x) \Pi_1(x, 0, x) \right. \left. - P_2(x) \Pi_2(x, 0, 0) \right\} \bigg|_{x=1} . \quad (B.23)
$$

The left-hand side of (B.23) is

$$
\frac{d^2}{dx^2} \left\{ [x - P_2(x)] \Pi_2(x, 0, x) \right\} \bigg|_{x=1} = (1 - \rho_2) L_{1,2} - P_2''(1) \Pi_2(1, 0, 1). \quad (B.24)
$$

On the other hand,

$$
\frac{d^2}{dx^2} \left\{ P_2(x) D_1(x) \Pi_1(x, 0, x) \right\} \bigg|_{x=1} = \Pi_1(1, 0, 1) \left[ P_2''(1) + D_1''(1) + 2 \rho_2 \eta_1 \right]
+ 2(\rho_2 + \eta_1) \Pi_1''(x = 1, 0, x = 1)
+ \Pi_1''(x = 1, 0, x = 1). \quad (B.25)
$$

Also,

$$
\frac{d^2}{dx^2} \left\{ - P_2(x) \Pi_2(x, 0, 0) \right\} \bigg|_{x=1} = -P_2''(1) \Pi_2(1, 0, 0) - 2 \rho_1 \Pi_2'(1, 0, 0) - \Pi_2''(1, 0, 0) \quad (B.26)
$$

and finally, substituting (B.24) to (B.26) into (B.23), after some algebra we get

$$
L_{1,2} = \frac{1}{1 - \rho_2} \left\{ \Pi_1(1, 0, 1) \left[ D_1'(1) + 2 \rho_2 \eta_1 \right] + 2(\rho_2 + \eta_1) \Pi_1'(x = 1, 0, x = 1)
- 2 \rho_2 \Pi_2'(1, 0, 0) + \Upsilon + \frac{P_2''(1)}{2} \right\} , \quad (B.27)
$$

where

$$
\Upsilon = \Pi_1''(x, 0, x) - \Pi_2''(x, 0, 0) \bigg|_{x=1} .
$$

To complete calculation for $L_{1,2}$, it is necessary to find $\Pi_1'(x = 1, 0, x = 1)$ and $\Upsilon$.

**B.2.1 Determination of $\Pi_1'(x = 1, 0, x = 1)$**

From (4.28) we have

$$
\Pi_1'(x, 0, x) \bigg|_{x=1} = \frac{d}{dx} \left[ D_2(x)[P_1(x)]^{M\alpha} \Pi_2(x, 0, 0) - [P_1(x)]^{M\alpha} \Phi_{M\alpha-1}(x) \right.
+ \Phi_{M\alpha-1}(x) P_1(x)) - (1 - x P_1(x)) \pi^{(2)}(2) d_0^{(2)} \bigg] \bigg|_{x=1} . \quad (B.28)
$$
But,
\[
\frac{d}{dx}[D_2(x)[P_1(x)]^{M_G} \Pi_2(x, 0, 0)] \Bigg|_{x=1} = (M_G \rho + \eta_2) \Pi_2(1, 0, 0) + \Pi_2'(1, 0, 0), \quad (B.29)
\]
and
\[
\frac{d}{dx}[-[P_1(x)]^{M_G} \Phi_{M_G-1}(x)] \Bigg|_{x=1} = -M_G \rho_1 \Phi_{M_G-1}(1) - \Phi_{M_G-1}'(1) \quad (B.30)
\]
\[
\frac{d}{dx}[\Phi_{M_G-1}(x P_1(x))] \Bigg|_{x=1} = (1 + \rho_1) \Phi_{M_G-1}'(1). \quad (B.31)
\]
and finally
\[
\frac{d}{dx}[-(1 - x P_1(x))\pi_0^{(2)} d_{0}^{(2)}] \Bigg|_{x=1} = (1 + \rho_1)\pi_0^{(2)} d_{0}^{(2)}. \quad (B.32)
\]

Using (B.29) - (B.32) in (B.28), yields
\[
\Pi_t'(x = 1, 0, x = 1) = \Pi_2'(1, 0, 0) + (M_G \rho_1 + \eta_2) \Pi_2(1, 0, 0)
\]
\[
- \rho_1 [M_G \Phi_{M_G-1}(1) - \Phi_{M_G-1}'(1)] + (1 + \rho_1)\pi_0^{(2)} d_{0}^{(2)}. \quad (B.33)
\]

**B.2.2 Determination of \( \Upsilon \)**

Consider the second derivative of equation (4.28):
\[
\frac{d^2}{dx^2} [\Pi_1(x, 0, x)] \Bigg|_{x=1} = \frac{d^2}{dx^2} \{D_2(x)[P_1(x)]^{M_G} \Pi_2(x, 0, 0) - P_1^{M_G}(x) \Phi_{M_G-1}(x)
\]
\[
+ \Phi_{M_G-1}(x P_1(x)) - (1 - x P_1(x))\pi_0^{(2)} d_{0}^{(2)} \} \Bigg|_{x=1} . \quad (B.34)
\]

The first term of the right-hand side of (B.34) will be
\[
\frac{d^2}{dx^2} [D_2(x)[P_1(x)]^{M_G} \Pi_2(x, 0, 0)] \Bigg|_{x=1} = [M_G P_1''(1) + M_G (M_G - 1) \rho_1^2
\]
\[
+ D_2''(1) + 2M_G \rho_1 \eta_2] \Pi_2(1, 0, 0)
\]
\[
+ 2(M_G \rho_1 + \eta_2) \Pi_2'(1, 0, 0)
\]
\[
+ \Pi_2''(1, 0, 0). \quad (B.35)
\]
and
\[
\frac{d^2}{dx^2}[-[P_1(x)]^{M_G} \Phi_{M_G-1}(x)] \bigg|_{x=1} = \left[-M_G P_1''(1) - M_G(M_G - 1)\rho_1^2\right] \Phi_{M_G-1}(1) \\
-2M_G \rho_1 \Phi'_{M_G-1}(1) - \Phi''_{M_G-1}(1). 
\] 
(B.36)

Also,
\[
\frac{d^2}{dx^2}\left[\Phi_{M_G-1}(x P_1(x))\right] \bigg|_{x=1} = \left[P_1''(1) + 2\rho_1\right] \Phi'_{M_G-1}(1) + (1 + \rho_1)^2 \Phi''_{M_G-1}(1), 
\] 
(B.37)

and for the last term of the right-hand side of (B.34), we have
\[
\frac{d^2}{dx^2}[-(1 - x P_1(x))\pi^{(2)}_{000}d^{(2)}_0] \bigg|_{x=1} = \left[P_1''(1) + 2\rho_1\right] \pi^{(2)}_{000}d^{(2)}_0. 
\] 
(B.38)

Therefore, by substituting (B.35) – (B.38) into (B.34), after some algebra we get
\[
\Upsilon = \Pi_2(1.0.0) \left[M_G P_1''(1) + D_2''(1) + 2M_G \rho_1 \eta_2 + M_G(M_G - 1)\rho_1^2\right] \\
+ 2\Pi_2'(1.0.0)[M_G \rho_1 + \eta_2] - \Phi_{M_G-1}(1) \left[M_G P_1''(1) + M_G(M_G - 1)\rho_1^2\right] \\
+ \Phi'_{M_G-1}(1)[P_1''(1) - 2(M_G - 1)\rho_1] + \Phi''_{M_G-1}(1) [\rho_1(2 + \rho_1)] \\
+ \left[P_1''(1) + 2\rho_1\right] \pi^{(2)}_{000}d^{(2)}_0. 
\] 
(B.39)
Appendix C

Average Number of Customers in the $M/G_1 - G_2/1$ Queue with Double-Threshold Policy

C.1 Average Number of Customers in Stage 1

Considering (5.62) and $L_1 = \lambda E[W_1]$, the average number of customers in stage 1, $L_1$, is

$$L_1 = 2 \frac{d}{dx} \Pi_1(x, 1) \bigg|_{x=1}. \quad (C.1)$$

On the other hand, using (C.2),

$$\frac{d^2}{dx^2} \Pi_1(x, 1)[x - P_1(x)] = \frac{d^2}{dx^2} [P_1(x)D_2(x)\Pi_2(x, 0) - P_1(x)\Pi_1(0, 1) - P_1(x)\Psi_{\lambda_n}(x, 1) - (1 - x)P_1(x)\pi^{(3)}_{00}d_0^{(2)} + x P_1(x)\Theta_{i, \lambda_{e-1}}(1)]. \quad (C.2)$$

For the left-hand side of (C.2), we have,

$$\frac{d^2}{dx^2} \Pi_1(x, 1)[x - P_1(x)] \bigg|_{x=1} = (1 - \rho_1)\Lambda_1 - \frac{P''_1(1)}{2}. \quad (C.3)$$

For the first term of the right-hand side of (C.2), we have

$$\frac{d^2}{dx^2} [P_1(x)D_2(x)\Pi_2(x, 0)] \bigg|_{x=1} = \Pi_2(1, 0)[P''_1(1) + D''_1(1) + 2\rho_1\eta_2] + 2(\rho_1 + \eta_2)\Pi_2^*(1, 0) + \Pi_2^{**}(1, 0). \quad (C.4)$$
Also, for the other terms of the right-hand side of (C.2), we get

\[
\frac{d^2}{dx^2}[-P_1(x)\Pi_1(0,1)] \bigg|_{x=1} = -P''_1(1)\Pi_1(0,1) \tag{C.5}
\]

\[
\frac{d^2}{dx^2}[-P_1(x)\Psi_{M_n}(x,1)] \bigg|_{x=1} = -P''_1(1)\Psi_{M_n}(1,1) - 2\rho_1\Psi^{' \prime}_{M_n}(1,1) \tag{C.6}
\]

\[
\frac{d^2}{dx^2}[-(1-x)P_1(x)\pi^2_{00}d_0^2] \bigg|_{x=1} = 2\rho_1\pi^2_{00}d_0^2 \tag{C.7}
\]

\[
\frac{d^2}{dx^2}[xP_1(x)\Theta_{1,M_{n-1}}(1)] \bigg|_{x=1} = \Theta_{1,M_{n-1}}(1)[P''_1(1) + 2\rho_1]. \tag{C.8}
\]

Substituting (C.3) - (C.8) into (C.2), after some algebra we get

\[
L_1 = \frac{1}{1 - \rho_1} \{ \Pi_2(1,0)[P''_1(1) + D''_1(1) + 2\rho_1\eta_2] - P''_1(1)[\Pi_1(0,1) + \Psi_{M_n}(1,1)] 
+ \Theta_{1,M_{n-1}}(1)[P''_1(1) + 2\rho_1] + 2\Pi^{' \prime}_2(1,0)[\rho_1 + \eta_2] - 2\rho_1\Psi^{' \prime}_{M_n}(1,1) 
+ 2\rho_1\pi^2_{00}d_0^2 - 3 + \frac{P''_1(1)}{2} \} \tag{C.9}
\]

where \( \beta = \Psi^{' \prime}_{M_n}(1,1) - \Pi^{' \prime}_2(1,0) \). To complete the calculation of \( L_1 \), the unknowns \( \beta, \Psi^{' \prime}_{M_n}(1,1) \) and \( \Pi^{' \prime}_2(1,0) \) must be found.

**C.1.1 Determination of \( \Psi^{' \prime}_{M_n}(1,1) \)**

Considering (5.32), we have

\[
\frac{d}{dx} [\Psi_{M_n}(x,1)x^{M_n}] \bigg|_{x=1} = \frac{d}{dx} \left\{ D_2(x)[P_1(x)]^{M_n}\Pi_2(x,0) - (1-x)[P_1(x)]^{M_n}\pi^2_{00}d_0^2 \right\} 
- \left[\frac{P_1(x)}{x}\right]^M_n\Theta_{1,M_{n-1}}(\frac{x}{P_1(x)}) 
+ x[P_1(x)]^{M_n}\Theta_{1,M_{n-1}}(\frac{x}{P_1(x)}) \bigg|_{x=1}. \tag{C.10}
\]

The left-hand side of (C.10) can be written as

\[
\frac{d}{dx} [\Psi_{M_n}(x,1)x^{M_n}] \bigg|_{x=1} = M_n\Psi_{M_n}(1,1) + \Psi^{' \prime}_{M_n}(1,1). \tag{C.11}
\]

For the right-hand side of (C.10), we get

\[
\frac{d}{dx} [D_2(x)[P_1(x)]^{M_n}\Pi_2(x,0)] \bigg|_{x=1} = (M_n\rho_1 + \eta_2)\Pi_2(1,0) + \Pi^{' \prime}_2(1,0). \tag{C.12}
\]
\[
\frac{d}{dx} \left[ -(1-x)[P_1(x)]^{M_n n_0^{(2)} d_0^{(2)}} \right] \bigg|_{x=1} = \pi_0^{(2)} d_0^{(2)}
\] (C.13)

\[
\frac{d}{dx} \left[ -P_1(x)^{M_n \Theta_{1,M_n}(\frac{x}{P_1(x)})} \right] \bigg|_{x=1} = -M_n \rho_1 \Theta_{1,M_n}(1) - (1 - \rho_1) \Theta'_{1,M_n}(1)
\] (C.14)

\[
\frac{d}{dx} \left[ x[P_1(x)]^{M_n \Theta_{1,M_{n-1}}(\frac{x}{P_1(x)})} \right] \bigg|_{x=1} = (1 + M_n \rho_1) \Theta_{1,M_{n-1}}(1) + (1 - \rho_1) \Theta'_{1,M_{n-1}}(1)
\] (C.15)

Substituting (C.11) – (C.15) into (C.10). after some algebra we obtain

\[
\Psi_{M_n}(1,1) = (M_n \rho_1 + \eta_2) \Pi_2(1,0) - M_n \Psi_{M_n}(1,1) + \pi_0^{(2)} d_0^{(2)} - M_n \rho_1 \Theta_{M_n,M_n}(1) \]
\[
+ \Theta_{1,M_{n-1}}(1) - (1 - \rho_1) \Theta'_{M_n,M_n}(1) + \Pi_2''(1,0).
\] (C.16)

### C.1.2 Determination of 3

Considering the second derivative of both sides of (5.32) we have

\[
\frac{d^2}{dx^2} [\Psi_{M_n}(x,1)x^{M_n}] \bigg|_{x=1} = \frac{d^2}{dx^2} \left\{ D_2(x)[P_1(x)]^{M_n \Pi_2(x,0)} \right. \]
\[
- (1-x)[P_1(x)]^{M_n n_0^{(2)} d_0^{(2)}} \]
\[
- [P_1(x)]^{M_n \Theta_{1,M_n}(\frac{x}{P_1(x)})} \]
\[
+ x[P_1(x)]^{M_n \Theta_{1,M_{n-1}}(\frac{x}{P_1(x)})} \bigg\} \bigg|_{x=1} \quad \text{(C.17)}
\]

For the left-hand side. we obtain

\[
\frac{d^2}{dx^2} [\Psi_{M_n}(x,1)x^{M_n}] \bigg|_{x=1} = M_n(M_n - 1) \Psi_{M_n}(1,1) + 2M_n \Psi'_{M_n}(1,1) + \Psi''_{M_n}(1,1).
\] (C.18)

For the right-hand side of (C.17), we get

\[
\frac{d^2}{dx^2} \left\{ D_2(x)[P_1(x)]^{M_n \Pi_2(x,0)} \right\} \bigg|_{x=1} = [M_n \rho_1''(1) + D_2''(1) + 2M_n \rho_1 \eta_2 \]
\[
+ M_n(M_n - 1) \rho_1^2] \Pi_2(1,0) \]
\[
+ 2[M_n \rho_1 + \eta_2] \Pi_2'(1,0)
\]
\[
+ \Pi_2''(1,0)
\] (C.19)

\[
\frac{d^2}{dx^2} \left[ -(1-x)[P_1(x)]^{M_n n_0^{(2)} d_0^{(2)}} \right] \bigg|_{x=1} = 2M_n \rho_1 n_0^{(2)} d_0^{(2)}
\] (C.20)
Substituting \((2.18)-(C.2)\) into \((C.17)\), after some algebra we obtain

\[
\begin{align*}
\frac{d^2}{dx^2} \left[ -[P_1(x)]^M \Theta_{1,M_n} \left( \frac{x}{P_1(x)} \right) \right]_{x=1} = & \left[ -M_n P_1''(1) - M_n (M_n - 1) \rho_1^2 \right] \Theta_{1,M_n}(1) \\
& -[2 \rho_1 (1 - \rho_1) (M_n - 1)] \\
& -P_1''(1) \Theta_{1,M_n}'(1) \\
& -[1 - \rho_1]^2 \Theta_{1,M_n}''(1) \\
\end{align*}
\]

\((C.21)\)

\[
\begin{align*}
\frac{d^2}{dx^2} \left[ x [P_1(x)]^M \Theta_{1,M_n-1} \left( \frac{x}{P_1(x)} \right) \right]_{x=1} = & \left[ M_n P_1''(1) + 2 M_n \rho_1 \\
& + M_n (M_n - 1) \rho_1^2 \right] \Theta_{1,M_n-1}(1) \\
& + [1 - \rho_1]^2 \Theta_{1,M_n-1}''(1) \\
& + [2 (1 + M_n \rho_1) (1 - \rho_1) - P_1''(1) \\
& - 2 \rho_1 (1 - \rho_1)] \Theta_{1,M_n-1}'(1). \\
\end{align*}
\]

\((C.22)\)

Substituting \((C.18)-(C.22)\) into \((C.17)\), after some algebra we obtain

\[
\begin{align*}
3 = & \left[ M_n P_1''(1) + D_2''(1) + 2 M_n \rho_1 \eta_2 + M_n (M_n - 1) \rho_1^2 \right] \Pi_2(1,0) \\
& - M_n (M_n - 1) \Psi_M(1,1) + 2 M_n \rho_1 \left[ \pi_{00}^{(2)} d_0^{(2)} + \Theta_{1,M_n-1}(1) \right] \\
& - \left[ M_n P_1''(1) + M_n (M_n - 1) \rho_1^2 \right] \Theta_{M_n,M_n}(1) \\
& + \left[ P_1''(1) - 2 (M_n - 1) \rho_1 (1 - \rho_1) \right] \Theta_{M_n,M_n}'(1) \\
& + 2 (1 - \rho_1) \Theta_{1,M_n-1}'(1) - (1 - \rho_1)^2 \Theta_{M_n,M_n}''(1) \\
& + 2 (M_n \rho_1 + \eta_2) \Pi_2(1,0) - 2 M_n \psi_M(1,1). \\
\end{align*}
\]

\((C.23)\)

\section*{C.1.3 Determination of $\Pi_2'(1,0)$}

Using \((5.35)\) we have

\[
\begin{align*}
\frac{d^2}{dx^2} \left[ \Pi_2(x,0) f(x) \right]_{x=1} = & \frac{d^2}{dx^2} \left\{ x^M D_1(x) \Theta_{M_n-1}(P_2(x)) \\
& - (1-x) D_1(x) [P_1(x) P_2(x)]^{M_n} \Pi_{00}^{(2)} d_0^{(2)} \\
& - D_1(x) [P_1(x) P_2(x)]^{M_n} \Theta_{1,M_n-1}(\frac{x}{P_1(x)}) \\
& + x D_1(x) [P_1(x) P_2(x)]^{M_n} \Theta_{1,M_n-1}(\frac{x}{P_1(x)}) \right\}_{x=1}, \\
\end{align*}
\]

\((C.24)\)
where \( f(x) = x^{M_n} - D_1(x)D_2(x)[P_1(x)P_2(x)]^{M_n} \). From the left-hand side of (C.24), we have

\[
\left. \frac{d^2}{dx^2} \left[ \Pi_2(x, 0) f(x) \right] \right|_{x=1} = f(1) \Pi_2''(1, 0) + 2f'(1) \Pi_2'(1, 0) + f''(1) \Pi_2(1, 0).
\]

Since \( f(1) = 0 \), and \( f'(1) = M_n(1 - \bar{p}) \), then

\[
\left. \frac{d^2}{dx^2} \left[ \Pi_2(x, 0) f(x) \right] \right|_{x=1} = f''(1) \Pi_2(1, 0) + 2M_n(1 - \bar{p}) \Pi_2'(1, 0), \tag{C.25}
\]

where

\[
f''(1) = M_n(M_n - 1)(1 - \rho^2) - \left[ D_1''(1) + D_2''(1) + M_n(P_1''(1) + P_2''(1)) \right] - 2M_n(\eta \rho + \rho_1 \rho_2) - 2\eta_1 \eta_2.
\]

On the other hand, for the terms in the right-hand side of (C.24), we have

\[
\left. \frac{d^2}{dx^2} \left[ x^{M_n} D_1(x)\Theta_{M_n,M_{n-1}}(P_2(x)) \right] \right|_{x=1} = \left[ M_n(M_n - 1) + 2M_n \eta_1 \right. \\
\left. + D_1''(1) \Theta_{M_n,M_{n-1}}(1) \right] \\
\left. + 2(M_n + \eta_1) \rho_2 + P_2''(1) \Theta'_{M_n,M_{n-1}}(1) \right) \\
\left. + \rho_2^2 \Theta''_{M_n,M_{n-1}}(1) \right] \tag{C.26}
\]

\[
\left. \frac{d^2}{dx^2} \left[ -(1 - x)D_1(x)[P_1(x)P_2(x)]^{M_n} \pi^{(3)}_{00} d^{(2)}_0 \right] \right|_{x=1} = 2(\eta_1 + M_n \rho) \pi^{(2)}_{00} d^{(2)}_0 \tag{C.27}
\]

\[
\left. \frac{d^2}{dx^2} \left[ -D_1(x)[P_1(x)P_2(x)]^{M_n} \Theta_{M_n,M_{n-1}}(\frac{x}{P_1(x)}) \right] \right|_{x=1} = -\left[ D_1''(1) + M_n(P_1''(1) + P_2''(1)) \right. \\
\left. + M_n(M_n - 1)(\rho_1^2 + \rho_2^2) \right. \\
\left. + 2M_n \eta_1 \rho + 2M_n^2 \rho_1 \rho_2 \Theta'_{M_n,M_{n-1}}(1) \right) \\
\left. + [P_1''(1) - 2(\eta_1 + M_n \rho)(1 - \rho_1) \right. \\
\left. + 2\rho_1(1 - \rho_1) \Theta''_{M_n,M_{n-1}}(1) \right) \\
\left. - (1 - \rho_1)^2 \Theta''_{M_n,M_{n-1}}(1) \right] \tag{C.28}
\]

\[
\left. \frac{d^2}{dx^2} \left[ xD_1(x)[P_1(x)P_2(x)]^{M_n} \Theta_{M_n,M_{n-1}}(\frac{x}{P_1(x)}) \right] \right|_{x=1} = \left[ D_1''(1) + 2\eta_1 + 2M_n^2 \rho_1 \rho_2 \right. \\
\left. + M_n(P_1''(1) + P_2''(1)) + 2M_n \eta_1 \rho \right]
\]
Substituting (C.25) – (C.29) into (C.24), after some algebra we get

\[
\Pi_2'(1, 0) = \frac{1}{2M_n(1-\rho)} \left\{ M_n[-P_1''(1) - P_2''(1) + 2\eta_1(1-\rho) - 2M_n\rho_1\rho_2 + (M_n - 1)(1-\rho_1^2 - \rho_2^2)]\Theta_{\ell_M, M_n}\right) \\
- [2(1-\rho)(\eta_1 + M_n\rho_1) - P_1''(1) - P_2''(1) - 2\rho_1(1 - \rho_1)]\Theta_{\ell_M, M_n}' \right) \\
- [(1-\rho_1)^2 - \rho_2^2]\Theta_{\ell_M, M_n}' \right) \\
+ 2[1 - \rho_1]\Theta_{\ell_M, M_n}' \right) \\
+ 2(\eta_1 + M_n\rho)\pi^{(2)}_0 d^{(2)}_0 - f''(1)\Pi_2(1, 0). \right) \}
\] (C.30)

and this completes all the requirements to obtain the average number of customers in the first stage.

### C.2 Average Number of Customers in Stage 2

For the average number of customers in the second stage, \( L_2 \), we have.

\[
L_2 = L_{1, 2} - L_1 \\
= 2\Pi_2'(x, x) \bigg|_{x=1} - L_1. \tag{C.31}
\]

Using (5.20), we have

\[
\frac{d^2}{dx^2} [\Pi_2(x, x)(x - P_2(x))] \bigg|_{x=1} = \frac{d^2}{dx^2} \left\{ P_2(x)D_1(x)\Psi_{\ell_M}(x, x) + P_2(x)D_1(x)\Theta_{\ell_M, M_n}(x) - P_2(x)\Pi_2(x, 0) \right\} \bigg|_{x=1}. \tag{C.32}
\]

For the left-hand side of (C.32) we obtain

\[
\frac{d^2}{dx^2} [\Pi_2(x, x)(x - P_2(x))] \bigg|_{x=1} = (1 - \rho_2)L_{1, 2} - \frac{P_2''(1)}{2}. \tag{C.33}
\]
For the terms in the right-hand side of (C.32), we have
\[
\frac{d^2}{dx^2} [P_2(x) D_1(x) \Psi_{Mn}(x,x)] \bigg|_{x=1} = \left[ P_2''(1) + D_1''(1) + 2\rho_2 \eta_1 \right] \Psi_{Mn}(1,1) \\
+ 2[\rho_2 + \eta_1] \Psi_{Mn}'(x=1, x=1) \\
+ \Psi_{Mn}''(x=1, x=1)
\]
(C.34)

\[
\frac{d^2}{dx^2} [P_2(x) D_1(x) \Theta_{Me, Mn}(x)] \bigg|_{x=1} = \left[ P_2''(1) + D_1''(1) + 2\rho_2 \eta_1 \right] \Theta_{Me, Mn}(1) \\
+ 2[\rho_2 + \eta_1] \Theta_{Me, Mn}'(1) + \Theta_{Me, Mn}''(1)
\]
(C.35)

and
\[
\frac{d^2}{dx^2} [-P_2(x) \Pi_2(x, 0)] \bigg|_{x=1} = -P_2''(1) \Pi_2(1,0) - 2\rho_2 \Pi_2'(1,0) - \Pi_2''(1,0)
\]
(C.36)

Substituting (C.33) to (C.36) into (C.32), after some algebra we get
\[
L_{1.2} = \frac{1}{1 - \rho_2} \left\{ [P_2''(1) + D_1''(1) + 2\rho_2 \eta_1] \Psi_{Mn}(1,1) + \Theta_{Me, Mn}(1) \\
- P_2''(1) \Pi_2(1,0) - 2\rho_2 \Pi_2'(1,0) \\
+ 2[\rho_2 + \eta_1] \Psi_{Mn}'(x=1, x=1) + \Theta_{Me, Mn}'(1) \\
+ \Theta_{Me, Mn}''(1) - \xi + \frac{P_2''(1)}{2} \right\}.
\]
(C.37)

where
\[
\xi = \Pi_2''(1,0) - \Psi_{Mn}''(x=1, x=1).
\]

To complete the calculation of $L_{1.2}$, the unknowns $\xi$ and $\Psi_{Mn}'(x=1, x=1)$ must be found.

**C.2.1 Determination of $\Psi_{Mn}'(x=1, x=1)$**

From (5.32) it is clear that
\[
\Psi_{Mn}(x, x) = x^{Mn} \Psi_{Mn}(x, 1).
\]
(C.38)

Therefore
\[
\Psi_{Mn}'(x, x) \bigg|_{x=1} = M_n \Psi_{Mn}(1,1) + \Psi_{Mn}'(1,1),
\]
(C.39)

where $\Psi_{Mn}'(1,1)$ can be found using (C.16).
C.2.2 Determination of $\xi$

Using (C.38) we get

$$
\Psi_{M_n}^x(x = 1, x = 1) = M_n(M_n - 1)\Psi_{M_n}(1, 1) + 2M_n\Psi_{M_n}'(1, 1) + \Psi_{M_n}''(1, 1).
$$

Therefore

$$
\Psi_{M_n}^x''(1, 1) = \Psi_{M_n}^x(x = 1, x = 1) - M_n(M_n - 1)\Psi_{M_n}(1, 1) - 2M_n\Psi_{M_n}'(1, 1). \quad (C.40)
$$

Considering the definition of $\beta$, we have

$$
\beta = \Psi_{M_n}^x''(1, 1) - \Pi_{M_n}^x''(1, 0)
$$

$$
\beta = \Psi_{M_n}^x''(x = 1, x = 1) - M_n(M_n - 1)\Psi_{M_n}(1, 1) - 2M_n\Psi_{M_n}'(1, 1) - \Pi_{M_n}^x''(1, 0).
$$

Upon using (C.40). Thus

$$
\xi = -[\beta + M_n(M_n - 1)\Psi_{M_n}(1, 1) + 2M_n\Psi_{M_n}'(1, 1)]. \quad (C.41)
$$
Bibliography


