Essays on Queueing Systems with Endogenous Service Times

by

Mohammad Delasay Sorkhab

A thesis submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

in

Operations and Information Systems

Faculty of Business

University of Alberta

ABSTRACT

Service rates in many real queueing systems, e.g., call centers and emergency departments, change with the system conditions. We investigate and model load-dependant service rates in this dissertation. First, we propose a general framework that explains different mechanisms that cause service rates to change in response to the system load. We use the framework to categorize and explain the results of published empirical papers that document dependence of service times on load. We employ the framework to analyze the effect of load on service times of an Emergency Medical Services (EMS) system based on a data set for emergency calls received by the Calgary EMS system in 2009.

Second, we propose a state-dependent queueing model in which servers speed up in response to the system "load," but eventually slow down as a result of "overwork," a situation where the system has been under a heavy load for an extended time period. We quantify load as the fraction of occupied servers and we operationalize overwork as the number of users served so far in the current high-load period. Our model is a quasi-birth-and-death process with a special structure that we exploit to develop efficient algorithms to compute system performance measures. We use the model and simulation to demonstrate how using models that ignore adaptive server behavior can result in inconsistencies between planned and realized performance and can lead to suboptimal, unstable, or oscillatory staffing decisions.

PREFACE

Chapters 2 and 3 of this thesis are parts of a working paper co-authored with Dr. Armann Ingolfsson, Dr. Bora Kolfal, and Dr. Kenneth Schultz. The LEST framework proposed in Chapter 2 was designed with the assistance from the co-authors. The literature review in Chapter 2 and data analysis in Chapter 3 are my original work.

Chapter 4 of this thesis was co-authored with Dr. Armann Ingolfsson and Dr. Bora Kolfal. The paper was submitted to the *Operations Research* journal for publication in August 2013 and has received a revise-and-resubmit decision.

ACKNOWLEDGEMENTS

My sincere gratitude to my supervisors, Dr. Armann Ingolfsson and Dr. Bora Kolfal, for their motivation, patience, and continuous support of my PhD study and research. I would like to thank Dr. Reidar Hagtvedt, the other member of the supervisory committee, and members of the candidacy exam committee, Dr. Yonghua Ji, Dr. Trish Reay, and Dr. John Doucette for their insightful feedback and comments. I would also like to thank the external examiner, Dr. Gad Allon from Northwestern University, for his time to review this dissertation.

My sincere thanks goes to Dr. Vedat Verter, McGill University, for financial support from the NSERC CREATE program in healthcare operations and information management that he directs. My sincere gratitude also goes to Dr. Itai Gurvich, Northwestern University, for providing me with the opportunity to visit the Kellogg School of Management and to work with him during the of summer 2014.

Many thanks to Dr. David Deephouse, Business PhD Program Chair, Dr. Karim Jamal, Department of Accounting, Operations and Information Systems Chair, Jeanette Gosine, Debbie Giesbrecht, and Kathy Harvey, PhD Program Office staff, and Deb Picken and Karmeni Govender, Department of Accounting, Operations and Information Systems staff, for their support.

Last but not the least, thank you especially to my wife, Leila, my parents, Mahmoud and Maliheh, and my brother and sister, Ali and Nafiseh, for their support and encouragement. Finally, thanks to my fellows and friends, Fernanda Campello, Amir Rastpour, Hooman Hidaji, and Mohamad Soltani.

TABLE OF CONTENTS

CHAPTERS

LIST OF TABLES

Table

LIST OF FIGURES

Figure

CHAPTER 1

Introduction

In this dissertation, we study queueing systems in which the exogenous service rate assumption does not hold. Most traditional capacity planning and queueing models, which are mainly based on the classical Erlang C and B models, assume that service rates are exogenous, that is, service rates are independent of the system state. Recent empirical research has increasingly called into question the validity of the exogeneity assumption. In particular, an important stream of empirical research has found evidence for the dependence of service times on system load in various service and production systems, e.g., call centers (Gans et al. 2010), healthcare systems (Kc and Terwiesch 2012), and serial production lines (Schultz et al. 1998).

In this dissertation, we study the dependence of service times on system load from three perspectives. (1) Empirical: we strive to establish fundamental knowledge about how and why service rates adapt to system load. (2) Analytical: we develop queueing models that incorporate some aspects of server behavior in response to load that cause service rates to vary. (3) Prescriptive: we investigate the impact on solution quality of accounting for adaptive service rates in models used to generate capacity planning and staffing solutions. We focus on the first perspective in Chapters 2 and 3. We focus on the second and third perspectives in Chapter 4.

In Chapter 2, we develop a general framework to help both empirical and analytical researchers to investigate and model how load impacts service times. We examine interactions among "load characteristics," "system components," and "service time determinants" while studying the effect of load on service times. We characterize load in terms of three dimensions: “changeover,” “load,” and “extended load.” We distinguish between three system components: “server,” “customer,” and “network.” We decompose service time into “work content” and the “service speed.” We use the framework to categorize and explain the results of published empirical papers that document dependence of service times on load.

In Chapter 3, we illustrate the use of the framework to generate hypotheses about service times in an emergency medical services (EMS) system. We employ the framework to identify mechanisms that cause EMS service times to change with the EMS load. The framework helps us to identify new mechanisms that have not been studied in previous papers. We test the hypotheses based on a data set from the EMS system of Calgary, Canada.

In Chapter 4, we develop a two-dimensional Markov chain for a multi-server queueing system in which the service rate depends on the system “load” and “overwork”—a model which generalizes the Erlang C model. Overwork refers to a situation where the system has been under a heavy load for an extended time period, which might result in fatigue. We quantify load as the fraction of occupied servers and we operationalize overwork as the number of users served so far in the current high-load period. Our model is a quasi-birth-and-death process with a special structure that we exploit to develop efficient and easy-to-implement algorithms to compute system performance measures. We use the analytical model and simulation to demonstrate how using models that ignore state-dependent service rates can result in inconsistencies between planned and realized performance and can lead to suboptimal, unstable, or oscillatory staffing decisions.

CHAPTER 2

LEST: A General Framework for the Influence of Load on Service Times

2.1 Introduction

What is the relationship between load and service times? This question has been a focus of recent empirical research, but the conclusions are not clear. On the one hand, Kc and Terwiesch find that "for two vastly different services ... workers can adapt to system needs by [increasing] the service rate" (Kc and Terwiesch 2009) and that an intensive care unit "rations its capacity during busy periods by discharging patients earlier" (Kc and Terwiesch 2012)—in other words, they find that service times decrease with load. On the other hand, the same authors caution that "increases in productivity cannot be sustained over a long period of time" (Kc and Terwiesch 2009) and that "high utilization results in ... [a] decrease in productivity." In a similar vein, Batt and Terwiesch (2012) find "evidence of both Speedup and Slowdown mechanisms" and Tan and Netessine (2012) observe that "as workload increases, the meal duration first increases and then decreases." In other words, Tan and Netessine find that service times have an inverted U-shaped relationship with load. Hasija et al. (2010) report ambiguous results when studying whether call center agents speed up in response to load. The relationship between load and service time remains an unresolved question.

Classical queueing models assume that service times do not depend on load. By classical queueing models we mean the Erlang C ($M/M/c$) and B ($M/M/c/c$) models that students learn about in operations management (OM) and engineering courses; that are used for capacity planning in manufacturing, telecommunication, and service systems; and that are used extensively in research on production systems. The voluminous body of research in queueing theory since the days of Erlang has extended the classical models in many ways but typically retaining the assumption that service times do not depend on load. There are exceptions, dating back as far as Jackson (1963). Although some work has continued on the modeling of such state-dependent queues, we postulate three possible reasons that have limited progress in this area:

1. The mathematical modeling of dependence between load and service times is challenging, especially if the dependence is more complicated than the one modeled by Jackson (1963), where the instantaneous service rates of all servers at a node in a queueing network depend (only) on the instantaneous number of customers at that node.

2. It is not clear that the effect of load on service times is economically or statistically significant.

3. The nature of the dependence of service times on load is not clear.

In this chapter, we propose a framework to analyze the influence of system load on service times in queueing systems. The proposed framework is general in the sense that it is applicable to any type of manufacturing or service system. The framework provides a comprehensive and systematic basis to investigate and explain how system components react and interact in response to system load and how those reactions and interactions cause variations in service times. We justify the generality of the framework, in part, by scrutinizing published OM empirical studies and using the framework to explain the observed relationships between service times and system load. In Chapter 3, we demonstrate application of the framework to analyze systematically the effect of the emergency medical system (EMS) load on service times.

Our framework has important implications for both empirical and analytical research. The framework conceptualizes a thinking process that an empirical researcher can use by provoking stylized questions: What are the system components? How is load characterized in the system? Which system components react to which load characteristics? What are the mechanisms that relate load changes to system component reactions? Which parts of the service time increase or decrease with which mechanism? Our framework can also help analytical researchers to answer two fundamental questions: What are the factors on which service times depend? How can these factors be translated into state variables? The proposed framework also emphasizes the importance of some of the queueing modeling characteristics, including: single-queue systems vs. queue networks, human vs. inanimate servers or customers, dedicated vs. shared servers, and single vs. multiple customer types.

This chapter is organized as follows. In Section 2.2, we discuss the common assumption that service times are exogenous, list examples of systems where this assumption is not valid, and review state-dependent queueing models. In Sections 2.3 and 2.4, we propose the framework and show how previous empirical research fits within the framework.

2.2 Literature Review

A. K. Erlang developed the classical Erlang C and B queueing models in the 1910s, to quantify traffic congestion in telephone systems (Brockmeyer et al. 1948). The Erlang C and B models are characterized by the assumption that service time distribution parameters are exogenous, that is, independent of the system state. This exogeneity assumption continues to be common in research and practice. The exogeneity assumptions leads to simpler models and it simplifies the data collection process by eliminating the need for tracking correlations between variables of interest (Inman 1999).

Empirical research on queueing systems gained momentum in the 1990s (Scudder and Hill 1998, Gupta et al. 2006). Empirical research involves analysis of real data collected by field research, from archival records, or from a laboratory experiment. Empirical research has increasingly called into question the validity of the exogeneity assumption. (e.g., Inman 1999, Robbins et al. 2010).

An important stream of empirical research has found evidence for the dependence of service times on system load. An early field study of toll collection processes for the Port Authority of New York (Edie 1954) found, for example, that drivers who wait longer in line are more likely to have change ready, leading to shorter average payment times. A laboratory experiment of a low-inventory serial line (Schultz et al. 2003) found that subjects worked at a slower pace during a warm-up period after an unintended break caused by a job shortage. Regression analysis of archival data from several hospitals (Kuntz et al. 2011) suggested a concave relation between bed occupancy and hospital length of stay (LOS): The LOS increases with occupancy up to a tipping point as patients wait longer for diagnosis and the LOS drops after the tipping point because doctors discharge patients earlier to accommodate incoming patients. We discuss other empirical studies in detail in Section 2.4.

The findings of these empirical studies represent some fundamental differences. For example, it is the behavior of the driver (the customer) in response to load that affects the payment time in Edie (1954), whereas in Schultz et al. (2003), it is the worker (the server) who behaves adaptively. Another example is the way in which system load is characterized: Edie (1954) characterizes load as the queue length (number of cars in line) and Schultz et al. (2003) characterizes load based on whether the amount of work-in-process is zero (idle period) or positive (busy period).

To point out another difference, some studies show positive, e.g., Edie (1954), some negative, e.g., Schultz et al. (2003), and some both positive and negative, e.g., Kuntz et al. (2011), relationship between service time and system load. In this paper, we propose a general framework that incorporates these and some other types of controversies.

Several queueing theorists have attempted to relax the exogeneity assumption by developing state-dependent queueing models, including Jackson (1963), Welch (1964), and Harris (1967). In state-dependent models, the mean service rate typically depends on the state of the system, which could either be the queue length or the amount of unfinished work (Dshalalow 1997). Others have developed vacation queueing models, which capture the type of load characterization that Schultz et al. (1998) observed; that is, lower service rates after break (vacation) due to setup, e.g., Levy and Yechiali (1975).

State-dependent models often disregard some important characteristics of queueing systems, mostly for the sake of model tractability. Our proposed framework highlights some of these characteristics. For example, most of the state-dependent models are single-server models and overlook behaviors like “free riding” (Karau and Williams 1993) in multi-server systems. Another example is that queues are usually parts of a network. Performance of a queue might affect service times in other parts of the network. For example, occupancy of a hospital affects the LOS of patients in the hospital emergency department (ED) (Hillier et al. 2009).

The growing evidence-based knowledge about queueing systems and advances in numerical techniques provide the opportunity for queueing modelers to include important characteristics of a real system and allow for more flexible interactions between different system components. For example, phase-type distributions facilitate viewing service times as the outcome of a dynamic process of customer-server interaction (Khudyakov et al. 2010, as reported in Gans et al. 2010). Or, quasi-birth-and-death modelling (Neuts 1981) allows for capturing different load characteristics that affect service times simultaneously (e.g., Delasay et al. 2013). In this respect, OM research is starting to achieve the kind of fertile interplay between experiment and theory that one sees in other sciences, e.g., in Physics (Fisher 2007).

2.3 Framework to Link System Load to Service Time

Figure 2.1 illustrates our framework as a chain of effects that connects system load to service time. We name the framework as LEST (Load Effect on Service Times). In the LEST framework, we identify three load characteristics named as: “changeover,” “load,” and “extended load.” The load characteristics induces behaviours, or “mechanisms,” in at least one of the system components; either the “server,” “network,” or “customer.” The induced mechanism influence the service time determinants of the “work content” or “service speed.” In subsections 2.3.1-2.3.3, we explain each box in the framework and define the used terminology.

2.3.1 Load Characteristics

Load characteristics are the indices, measures, or conditions by which system load is characterized. We have identified three different system load characteristics as follows:

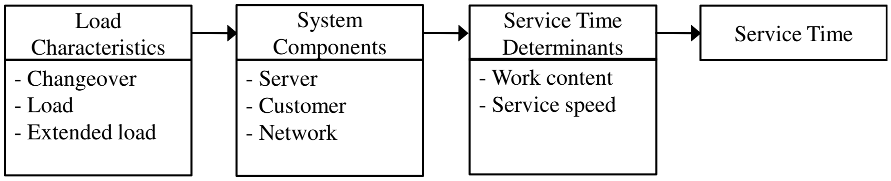


Figure 2.1: The LEST framework

- **Changeover**: Changeover refers to a change in system load from zero to a positive value or vice versa. In other words, situations where the system switches from an idle state to a busy state or from a busy state to an idle state. Changeover captures the type of system load effects that was observed in Schultz et al. (2003); slower service pace after breaks. Changeover also includes switching from one service type to another.

- **Load**: Load refers to a measure or a set of measures that identify how busy or congested a system is. Load is usually measured as the number of jobs in system, multitasking level or the number of jobs assigned to a server, amount of unfinished work, and occupancy rates or occupied capacity. For example, the number of patients waiting in an ED is a way to measure ED load.

- **Extended load**: Extended load tracks the history of how the system load has changed. It usually refers to a situation where the system has been under a heavy load for an extended time period. For example, fatigue is the direct symptom of the extended load.

2.3.2 System Components

We use the term "mechanism" to denote a link between load characteristics and service time due to a specific cause. Changes to the three load characteristics invoke different behaviours or induce mechanisms in three system components: the server, the customer, or the network. An example is that the extended load may cause fatigue in servers, which results in lower service speed and longer service time (Kc and Terwiesch 2009). Here, "fatigue" is the mechanism.

- **Server**: We use the term "server" generically, without necessarily implying that servers are human. The server is the person, the resource, or the bundle of people and other resources that provides service. Some systems have shared resources that do not belong exclusively to any single server—diagnostic imaging for hospital physicians or computer and telecommunication infrastructure for a call center (Aksin and Harker 2003).

- **Customer**: The "customer" is the person or thing that receives service. Like a server, a customer can be human or inanimate. For example, patients are customers in an ED and unfinished products are customers in a manufacturing line.
- **Network**: A system may consist of multiple subsystems. When we analyze a subsystem or a "node," consisting of a queue or multiple queues and a single set of servers, we consider any mechanism that originate from outside of the node of interest but impacts service times in the node of interest as a "network" mechanism.

To illustrate the above definitions, consider a call center: servers are agents with associated resources (computers, desks, cubicles), customers are callers, and the network could include an interactive voice response unit that callers interact with prior to entering a queue of callers waiting to talk to an agent. In an EMS system, as another example, servers are ambulances with crews, customers are patients, and the network could include the road network or the ED to which ambulances transport patients.

2.3.3 Service Time Determinants

Mechanisms that originate from a system component in response to a load characteristic either increase or decrease one of the service time determinants of "work content" or "service speed." We view each customer entering service as having a random amount of work, W, that needs to be completed. The work content W can include set-ups, in-process delays, and customer-server interactions. If the service speed is S, measured in units of work per time unit, then the service time is $T = W/S$. It is often useful to decompose a service into either stages (single or multi-stage, as in Gross et al. (2008)) or phases (access, check-in, diagnosis, service delivery, and check-out, as in Bitran and Lojo (1993)). Denoting the work content and the service speed for stage or phase i by W_i and S_i, respectively, we represent the total service time as:

$$T = \sum_i \frac{W_i}{S_i}. \tag{2.1}$$

In Section 2.4, we illustrate how the proposed framework explains and classifies the findings in the empirical OM literature.

2.4 Classification of Previous Work

In this section, we review empirical papers that document dependency of service time on system load. We classify the identified mechanisms in these papers in Table 2.1, which is tabulated according to the LEST framework. In Table 2.1, we illustrate that the findings of published studies can be explained by the framework. In our literature review, we were interested in papers that pass two conditions: (1) Those that use data analysis to test the dependency of service times to system load and (2) discuss the mechanisms that cause service times to change with load. Although a paper must include some sort of data analysis to satisfy the first condition, it does not necessarily require data analysis for the second condition. The second condition can be based on intuition, judgment, observation, interviews, past knowledge, or data analysis. We tried to be inclusive in our literature review, though we do not claim that we have covered all related papers.

The nine cells of Table 2.1 correspond to all combinations of the three load characteristics and the three system components. In each cell, we classify the identified mechanisms in each paper based on the two service time determinants. Inside the first parentheses in front of a mechanism, we identify whether the corresponding mechanism increases or decreases the service time: (↑) for increase and (↓) for decrease. Then, we list the related papers in the second parentheses. A paper has either no superscript, a "$+$" superscript, or a "$-$" superscript. A paper is listed with no superscript if it hypothesizes the involved mechanism based on intuition, judgment, observation, interviews, or past knowledge but it does not involve data analysis to test the hypothesized mechanism. If a paper includes data analysis to test the hypothesized mechanism and the result of the data analysis supports the mechanism, we list it with a "$+$" superscript. On the other hand, if the result of the data analysis does not

Table 2.1: Mechanisms

		System components		
		Server Section 2.4.1	Network Section 2.4.2	Customer Section 2.4.3
Load characteristics	Changeover Section 2.4.X.1	Work content - Setup (↑)(This paper) - Forgetting (↑) (Kc 2011, Schultz et al. 2003^{-}) Service speed - Loss of rhythm (↑) (Schultz et al. 2003^{+})	Work content - Network arrangement (↓) (This paper)	Work content - Early task initiation (↓) (Edie 1954)
	Load Section 2.4.X.2	Work content - Task reduction/service cancellation (↓) (Kc and Terwiesch 2009, 2012, Kuntz et al. 2011, Kc 2011^{+}, Batt and Terwiesch 2012^{+}, Forster et al. 2003^{-}, Mæstad et al. 2010^{-}, This paper) - Task increase (↑) (Tan and Netessine 2012^{+}, This paper) - Early task initiation (↓) (Batt and Terwiesch 2012^{+}, This paper) - Multitasking - Time sharing and interruptions (↑) (Kc 2011, Tan and Netessine 2012, Chisholm et al. 2000^{+}, Lu 2013^{+}) - Workload smoothing (↓) (Jaeker and Tucker 2012^{+}, This paper) Service speed - Social pressure - Speedup (↓) (Edie 1954, Mas and Moretti 2009, Kc and Terwiesch 2009, Staats and Gino 2012, Lu 2013, Schultz et al. 1998^{+}, Tan and Netessine 2012^{+}, This paper) - Social loafing - Slowdown (↑) (Mas and Moretti 2009, Jaeker and Tucker 2012)	Work content - Downstream system congestion (↑) (Asaro et al. 2007^{+}, Forster et al. 2003, Hillier et al. 2009, This paper) - Resource sharing (↑) (Hillier et al. 2009) - Geographical dispersion (↑) (This paper) Service speed - Geographical speedup (↓) (This paper)	Work content - Service complication (↑) (Kc 2011^{+}, Kc and Terwiesch 2012^{+}, This paper)
	Extended Load Section 2.4.X.3	Work content - Task reduction/service cancellation (↓) (Brown et al. 2005, This paper) Service speed - Overwork - Slowdown (↑) (Kc and Terwiesch 2009, Gans et al. 2010, Staats and Gino 2012, Lu 2013, This paper) - Learning-by-doing (↓) (Lu 2013)	Work content - Network chaos (↑) (This paper)	Work content - Service complication (↑) (Kc and Terwiesch 2009)

support the hypothesized mechanism, then we list the paper with a "$-$" superscript. We first list papers with no superscript, then papers with "$+$" superscript, and finally, papers with "$-$" superscript. We use the wording "This paper" if we hypothesize possible existence of a mechanism in the studied EMS system in Chapter 3. In the remainder of this section, we discuss the mechanisms and papers listed in each cell of Table 2.1.

2.4.1 Server Mechanisms

In this section, we review the mechanisms in the server column of Table 2.1, which is the most populated column of the table: changeover mechanisms in 2.4.1.1, load mechanisms in 2.4.1.2, and extended load mechanism in 2.4.1.3.

2.4.1.1 Server - Changeover Mechanisms

Question: How does server's reaction to changeover change work content or service speed?

(W) Setup: When system load becomes zero, servers are forced either to take a break or switch to another task. In both cases, the changeover characteristic is in effect. The most obvious mechanism that increases the work content in case of a changeover is setup. Researchers have long argued for the productivity benefits of reducing setups involved in changeovers by strategies like *specialization* and *mass production* (Cellier and Eyrolle 1992, Schultz et al. 2003). There are two types of setups: *physical setups* and *cognitive setup*. In the next two mechanisms, *forgetting* and *loss of rhythm*, we consider two mechanisms involved mostly in the cognitive setup. In Chapter 3, we investigate physical setup in an EMS system.

(W) Forgetting: When servers take break from their main duty, they may forget the routine of the operation. Time to remember the operation incurs a cognitive setup that leads to increase in the work content and processing time (Steedman 1970). In the *forgetting* mechanism, longer breaks cause longer processing time penalty (Carlson and Rowe 1976, Bailey 1989). Kc (2011) shows that patients' LOS increases with physician's multitasking level and argues that this is partly because of cognitive setups involved in task switching, e.g., going through medical notes to recall patient's situation. Schultz et al. (2003)$^{-}$ test the *forgetting* mechanism in a laboratory setting of a low-inventory serial production line. Although the experiments show that breaks lead to significantly longer processing times, they do not support the association between the time penalty and the length of the break.

(S) Loss of rhythm: Another explanation for longer processing times after a changeover is the *loss of rhythm* mechanism (Schultz et al. 2003)$^+$. In repetitive tasks, servers adapt a rhythm of performing their task. Breaks interrupt the rhythm and lowers service speed for a short period until the rhythm is regained (Rubinstein et al. 2001). Staats and Gino (2012) analyze loan processing times in a Japanese bank and find that assigning new tasks to employees causes higher average completion times. Schultz et al. (2003)$^+$, discussed in the *forgetting* mechanism, show evidence for the *loss of rhythm* mechanism in a low-inventory serial line noting that the time penalty is independent of the break length in the *loss of rhythm* mechanism.

2.4.1.2 Server - Load Mechanisms

Question: How does server's reaction to variation in load change work content or service speed?

(W) Task reduction/service cancellation: By *task reduction*, we refer to situations where servers terminate a service stage, before it is completely accomplished, or eliminate one or more stages of a service, usually to manage workload. We borrow the term *task reduction* from Batt and Terwiesch (2012). Another term for this mechanism is *cutting corners*. Service cancellation is the extreme case of task reduction. Obviously, the *task reduction/service cancellation* mechanism shrinks the work content. This mechanism is mostly observed in complex professional tasks with discretionary task completion criteria; that is, completion of tasks are determined by server's subjective criteria, e.g., engineers and physicians (Hopp et al. 2007).

Based on an analytical model of a system with discretionary tasks, Hopp et al. (2007) prove that *task reduction* can be the optimal service policy if service value is concave-increasing with the service time and cost is increasing by the amount of time a customer spends in the system. In this setting, the optimal service policy is to set a service cutoff time that is monotone decreasing in queue length. Stidham and Weber (1989) and George and Harrison (2001) confirm similar policies.

Several empirical papers document *task reduction* in healthcare systems. This is because of the discretionary characteristic of healthcare related tasks. Early discharge is a common manifestation of the *task reduction* mechanism in healthcare systems. Kc and Terwiesch (2009) relate the shorter LOS of cardiothoracic surgery patients in high occupancy levels to early patient discharges to increase bed availability for future surgeries. Kuntz et al. (2011) observe an inverted U-shape relation between bed occupancy and hospital LOS. They posit early discharges for decreasing LOS when occupancy exceeds the tipping point. Interviews with personnel of an intensive care unit (ICU) also confirm that doctors ration ICU capacity during busy periods by discharging patients earlier (Kc and Terwiesch 2012). The interviews also acknowledge that there are rare instances of elective surgery cancellations as a result of full ICU capacity.

Batt and Terwiesch (2012)$^{+}$ find statistical evidence that the number of tests ordered by doctors decreases with ED load, measured as the waiting room census. Kc (2011)$^{+}$ measures load as the number of patients multitasked by a physician. His findings support the hypothesis that multitasking has inverse effect on the ED care quality as physicians spend less time on patient diagnosis when they are treating several patients at the same time.

Forster et al. (2003)$^{-}$ is a contradicting example. They view an ED and a hospital as nodes of a network. They study the effect of hospital occupancy on ED throughput but they find no evidence of the *task reduction* mechanism. Their regression analysis do no support reduction in the proportion of ED patients who are referred to hospital consultants when the hospital is experiencing high load. The analysis do not even provide evidence of early discharges in the ED. Mæstad et al. (2010)$^{-}$ also find no association between physicians multitasking level and the level of effort per patient in the diagnostic process measured by the number of relevant questions asked and examinations performed.

(W) Task increase: In contrast to the *task reduction* mechanism, in the *task increase* mechanism servers put more time and effort for a service to improve service quality or earn more income. Tan and Netessine (2012)$^{+}$ report the *task increase* mechanism in restaurant waiters. They observe that increasing waiter's load when he is serving few diners prolongs diners' meal duration. They hypothesize that increasing the waiter-level load engages the waiter to exert more upselling effort when restaurant is not full. They validate this by showing that hourly waiter's sales increases with load. However, restaurant load limits waiter's sales effectiveness after a certain threshold, which we discuss later in the *speedup* mechanism.

(W) Early task initiation: Early task initiation is to perform some stages or tasks of a service in an earlier time to reduce workload of the system bottleneck. This includes initiatives undertaken by servers of a preceding stage while the customer has to wait for the attention of the next stage's server. An example is when ED triage nurses are allowed to order diagnostic tests while a patient is waiting to be seen by a physician. Batt and Terwiesch (2012)$^+$ confirm that nurses order more tests when the waiting room is more crowded in order to shorten the LOS by making test results ready when a physician visits the patient. High degrees of early task initiation may be undesirable as over-testing by triage nurses, before it is fully known that the tests are required, increases financial costs, medical risks, and load on diagnostic resources.

(W) Multitasking - Time sharing and interruptions: The term "multitasking" was first used in computing science to describe the sharing of computing resources among many users or programs (Brown 2006). Higher multitasking level is an indication of higher load. Humans tend to multitask when they confront workload pressure to be able to utilize idle time between tasks (Pennebaker 2009). For example, an ED physician who treats several patients at the same time can check another patient while waiting for test results of a patient. Despite presumed benefits of multitasking, phycological studies are against multitasking mostly because of productivity loss caused by mental setups to refocus on switching tasks (Gladstones at al. 1989, Pashler 1994, Rubinstein et al. 2001). We discussed some implications of multitasking in the *forgetting* mechanism and the *task reduction/service cancelation* mechanism. Here, we focus on other implications of multitasking.

We are aware of four operational papers that discuss other implications of the *multitasking* mechanism, *time-sharing* and *interruptions*, that lead to in-process wait and work content increase. Kc (2011) finds that productivity of physicians, measured as the overall patient throughput per unit time, increases with multitasking up to a level of five patients.

Productivity losses of multitasking dominate its gains beyond that level. Despite the partial productively gains, experiments reveal that multitasking at any level extends an individual patient's LOS. One mentioned reason is the division of physician's time over a larger number of patients. For example, a patient's test results are ready but since the physician is taking care of another patient, the patient needs to wait. Tan and Netessine (2012) also mention time-sharing as a possible reason for prolonged meal duration of diners assigned to a waiter serving several table at the same time.

Trough time-motion analysis, Chisholm et al. $(2000)^+$ find a positive correlation between multitasking level and number of interruptions that require attention of a multitasking physician. Lu $(2013)^+$ analysis on productivity of agents of an information technology services provider reveals that although higher multitasking level is not associated with number of interruptions, it prolongs the revisit time for suspended services.

(W) Workload smoothing: The ability to predict future load induces different mechanisms in servers to avoid periods of high congestion. For example, Green at al. (2013) observe that workers react to predictable overloaded periods by not showing up at work. *Workload smoothing* is another reaction to predictable load, which balances system load over time and prevents over-utilized and under-utilized periods without hurting service quality. If servers can predict future high load periods, they have the opportunity to smooth their workload by serving their current work content before high load periods start. Jaeker and Tucker (2012) argue that load predictability plays a significant role in discharging decisions in a hospital. They find that medical teams react to high volume of incoming scheduled patients from surgery units to hospital by early discharging current patients, if the hospital is congested.

(S) Social pressure - Speedup: *Speeding up* or *rushing* is a common phenomenon in queueing systems where server's performance is visible to other servers, customers, or the manager. For example, slower workers work faster when performance feedback is available—workers can see how other workers perform (Schultz et al. 2003, Bandiera et al. 2012). Edie (1954) is one of the first empirical studies that reports the *speedup* behavior. He demonstrates that toll collectors at George Washington Bridge tend to expedite the operation under the pressure of backed-up traffic by limiting the conversation with drivers.

Mas and Moretti (2009) show that a slower cashier in a supermarket speeds up when customers are waiting in line and he/she is in the line of vision of other faster cashiers. Regression models by Kc and Terwiesch (2009) support the hypothesis that patient transporters in a hospital, whose performance is evaluated through a patient tracking system, speed up in response to load, defined as the fraction of busy transporters. However, their further analysis, which we will discuss in Section 2.4.1.3, demonstrates that speeding up is not a sustainable behavior. Staats and Gino (2012) observe the *speedup* behavior in home loan processing employees of a Japanese bank. Although employees are not aware of the system load, they speedup when manager encourage them to speed up. Lu (2013) find evidence that agents of an information technology services provider speed up with the number of assigned requests, but the increase of speed diminishes as load exceeds a threshold.

Schultz et al. (1998)$^+$ observe the *speedup* mechanism in a serial production line where worker's performance is tractable by the amount of work-in-process accumulated in the preceding and succeeding buffers. Using a laboratory experiment, they show that workers in a low-inventory serial line react to pressure from workers of the neighboring workstations and the buffer inventory level; they work faster when they are causing blockage of the preceding station or starvation of the succeeding station. In a restaurant, Tan and Netessine (2012)$^+$ hypothesize that a high level of load encourages restaurant waiters to accelerate service to reduce the costs of customer waiting. Their hypothesis is consistent with their regression results, which reveal reduced upselling effectiveness of waiters when their workload is very high.

(S) Social loafing - Slowdown: *Social loafing* is the counterpart of the *social pressure* mechanism. *Social loafing*, a.k.a. *free riding*, occurs when servers exert less effort to avoid pulling the weight of a fellow team member (Karau and Williams 1993, Krumm 2000). *Social loafing* is more prevalent in congested systems in which individual effort is difficult to monitor (Latane et al. 1979).

Mas and Moretti (2009) argue that as the number of customers waiting in a supermarket increases, cashiers find more incentive to free ride and let other cashiers handle the additional workload. Viewing an ED and a hospital as a series queueing network, Jaeker and Tucker (2012) find that an in-hospital patient stays longer if there is a high load of incoming non-acute patients from the ED. They associate this with nurses' social loafing behavior:

nurses work slower intentionally to avoid being assigned new patients when it is difficult to being recognized as the bottleneck.

2.4.1.3 Server - Extended Load Mechanisms

Question: How does server's reaction to the past history of load change work content or service speed?

(W) Overwork - Service cancellation: *Overwork* and the consequent productivity deterioration is the natural outcome of working for long periods (Cakir et al. 1980, Setyawati 1995). Though not documented widely, overworked servers may simply refuse to serve a customer to obtain extra rest. Brown et al. (2005) find evidence for this mechanism when they encountered call times of less than 10 seconds while analyzing data of a call center. Short service times were primarily caused by overworked agents who simply hung up on customers to reduce workload and obtain extra rest.

(S) Overwork - Slowdown: The *social pressure-speedup* cannot be sustained indefinitely; when servers are overworked they start to slow down (Sze 1984, Dietz 2011). In lab experiments, Tanabe and Nishihara (2004) find that reaction times are longer when servers work over a long time period. Kc and Terwiesch (2009) demonstrate that hospital transporters' speedup behavior in response to load is not sustainable and they slow down after experiencing extended periods of high load. They also argue that overwork slows down physicians in making discharge decisions for cardiothoracic surgery patients. Gans et al. (2010) measure overwork of agents of a call center by *run length*—the number of services an agent has performed since the last gap of longer than one hour. They find that higher run length is associated with longer average call times for some agents. Staats and Gino (2012) observe the same behaviour by loan processors in a bank.

(S) Learning-by-doing: In contrast to the *overwork* mechanism, extended load can bring productivity gains through the *learning-by-doing* mechanism, first recognized by Wright (1936). Learning can occur over long-term horizons (weeks or months, for instance) or short-term horizons (within a shift, for instance). Pisano et al. (2001) and Gans et al. (2010) are among those that have studied long-term learning, as a function of the cumulative number of service completions for a medical team and for a call center agent, respectively. Lu

(2013) studied short-term learning, as a function of the elapsed duration of a call center agent's current shift and found evidence for shorter processing times later in a shift.

2.4.2 Network Mechanisms

In this section, we review the mechanisms in the network column of Table 2.1: changeover mechanisms in 2.4.2.1, load mechanisms in 2.4.2.2, and extended load mechanism in 2.4.2.3.

2.4.2.1 Network - Changeover Mechanisms

(W) Network arrangement: We explain this mechanism in Section 3.2.

2.4.2.2 Network - Load Mechanisms

(W) Downstream system congestion: When different services are interrelated nodes of a network, a change in load of a node can have uncontrollable impact on service times in other nodes. One possible scenario, which we call *downstream queue congestion* mechanism, is when resources have to be engaged for a longer time to serve customers who cannot be admitted by a full downstream service. For example, Forster et al. (2003) and Hillier et al. (2009) view an ED and a hospital as a network. They find that patients who are admitted to hospitals need to stay longer in ED when the hospital occupancy level is above 80%. Asaro et al. (2007)$^+$ confirm the existence of the *downstream system congestion* mechanism by finding positive effect of hospital occupancy on boarding times of ED patients.

(W) Resource sharing: Another network effect of load on service time happens when different service nodes share common resources, which usually prolongs in-process delays. One other possible reason for the effects observed by Forster et al. (2003) and Hillier et al. (2009), discussed in the *downstream system congestion* mechanism, is shared resources between the ED and the hospital, two nodes of the network. Hillier et al. (2009) find that high hospital occupancy not only prolongs ED LOS of admitted patients to the hospital but also increases LOS of patients discharged from the ED. This finding indicates ED and hospital share resources, including treatment areas and care providers.

(W) Geographical dispersion: We explain this mechanism in Section 3.2.

(S) Geographical speedup: We explain this mechanism in Section3.2.

2.4.2.3 Network - Extended Load Mechanisms

(W) **Network chaos**: We explain this mechanism in Section 3.2.

2.4.3 Customer Mechanisms

In this section, we review the mechanisms in the customer column of Table 2.1: changeover mechanisms in 2.4.3.1, load mechanisms in 2.4.3.2, and extended load mechanism in 2.4.3.3.

2.4.3.1 Customer - Changeover Mechanisms

(W) **Early task initiation**: Like server's *early task initiation* mechanism, customers can also help to shorten in-process delays by performing tasks that are under their control before the service encounter starts. To clarify the reason for shorter holding times at higher volumes of traffic per lane at George Washington Bridge toll booths, Edie (1954) speculates that drivers have the opportunity to get their tolls ready, while waiting, when there is a waiting line at a toll booth; whereas when there is no line, they have to search to find their tolls when they drive right up to the booth.

2.4.3.2 Customer - Load Mechanisms

(W) **Service complication**: As discussed in the *task reduction/service cancellation* and *workload smoothing* mechanisms, servers might respond to high load by cutting a service encounter prematurely. In some services, this may endanger service quality and cause complications in customer needs that require him/her to bounce back to the system at a later time. This causes longer total service times. Kc (2011)$^+$ and Kc and Terwiesch (2012)$^+$ document this mechanism by showing that the likelihood of patients revisits to medical units (an ICU and an ED) increases with load. In Kc (2011)$^+$, lower care quality due to excessive multi-tasking is the reason for revisits, while in Kc and Terwiesch (2012)$^+$ early discharge decisions cause revisits.

2.4.3.3 Customer - Extended Load Mechanisms

(W) Service complication: When overwork is associated with a reduction in service quality, additional reprocessing and rework is required to achieve desirable service quality. After showing that system-wide overwork increases the LOS for cardiothrocic surgery patients, Kc and Terwiesch (2009) argue that fatigued care providers are more prone to making medical errors, which leads to complications that call for additional rework.

CHAPTER 3

Using the LEST Framework to Analyze EMS Service Times

3.1 Introduction

In Chapter 2, we used the LEST framework to explain and categorize the findings of the empirical studies that document the effect of system load on service times. In this section, we describe how to employ the framework to analyze EMS service times. An EMS response to a medical emergency begins when a patient or bystander calls 911.

An emergency medical dispatcher (EMD) answers and triages the call trough a systematic medical interrogation to determine the patient's condition acuity. After gathering the required information, including call address and the type of required equipment, the EMD dispatches an appropriately equipped ambulance close to the incident scene. We refer to a service as "regular" if the ambulance receives a dispatch notification in the standby mode. It is also possible that an ambulance receives a dispatch notification just after finishing a service, while returning from that service. We call this situation "extended service."

The EMS service time begins when the ambulance receives the dispatch notification and it includes the five time intervals shown in Figure 3.1:

- Chute time (T^{Chute}): The preparation and boarding time for the ambulance crew after receiving the dispatch notification.

- Travel time (T^{Travel}): The travel time from the dispatch location to the incident scene.

- Scene time (T^{Scene}): The time that ambulance crew are on scene providing medical care to a patient.

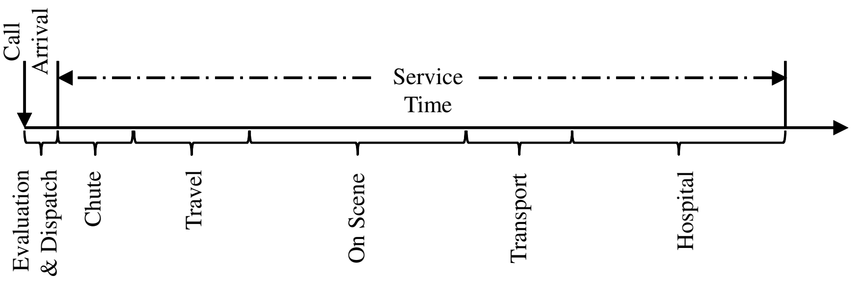


Figure 3.1: EMS Service Time

- Transportation time ($T^{\text{Transport}}$): The travel time from the scene to a hospital, if the patient requires hospital transportation.

- Hospital time (T^{Hospital}): The offload time to transfer the patient to the ED after arriving to the hospital.

In this chapter, we use the LEST framework for a systematic investigation of mechanisms that cause EMS service times to depend on EMS load. We consider each cell of the LEST framework and we try to understand how the corresponding system component and load characteristic are manifested in different EMS service time intervals. Then, we propose possible mechanisms that relate the system component and the load characteristic to a service time interval.

We investigate server mechanisms in Section 3.2.1, network mechanisms in Section 3.2.2, and customer mechanisms in Section 3.2.3. In Section 3.3, we aggregate the effect of the server, network, and customer mechanisms on each time interval and we investigate the effect of load on the total EMS service time. We test our hypotheses by analyzing service time data for the EMS system of the city of Calgary, Canada, in Section 3.4.

3.2 Identifying EMS Mechanisms Based on the LEST Framework

In general, "load" of an EMS system is measured as the number of patients who require medical care from the EMS system, which is often equal to the number of busy ambulances. A system-wide "changeover," when system load resets to zero, might occur rarely for an EMS system as large as Calgary EMS (typically around 40 ambulance on duty) but we can look for changeover mechanisms when an ambulances becomes idle. When most of the

Table 3.1: Mechanisms for the effect of EMS load on service time

		System components		
		Server	Network	Customer
Load characteristics	Changeover	Work content - Setup (M 1.1)	Work content - Network arrangement (M 2.5)	
	Load	Work content - Early task initiation (M 1.2) - Task increase (M 1.3.a) - Workload smoothing (M 1.3.b) - Task reduction (M 1.4, M 1.5) Service speed - Speedup (M 1.6)	Work content - Downstream system congestion (M 2.1) - Geographical dispersion (M 2.2) Service speed - Geographical speedup (M 2.3)	Work content - Service complication (M 3.1, M 3.2)
	Extended load	Work content - Slowdown (M 1.7) - Task reduction (M 1.8)	Work content - Network chaos (M 2.4)	

ambulances are busy for a long period, we expect to observe the “extended load” related mechanisms.

Table 3.1 previews the identified mechanisms that cause EMS service times to depend on different load characteristics. We explain mechanisms of each cell in the following sections.

3.2.1 EMS Server Mechanisms

In the EMS system, “servers” are ambulances with paramedics. In this section, we first investigate server - changeover mechanisms, then server - load mechanisms, and finally server - extended load mechanisms.

3.2.1.1 EMS Server - Changeover Mechanisms

The *forgetting* and *loss of rhythm* mechanisms do not seem to influence the performance of EMS servers as these mechanisms are more prevalent in repetitive and routine tasks.

Chute time, the first time interval, can be conceived as the setup time of the EMS service and thus, a point of interest to investigate the *setup* mechanism. It is reasonable to believe that chute time of regular services is longer than chute time of extended services. In most cases, chute time for extended services is equal zero as ambulance crew are already in the vehicle (Aehlert and Vroman 2011, page 654).

M 1.1 *Setup (server, changeover, work content): The setup involved in regular services is longer than the setup involved in extended services.*

3.2.1.2 EMS Server - Load Mechanisms.

We rule out the *multitasking* mechanism for the EMS system as each response unit handles one patient at a time. We expect to observe the effect of the *early task initiation* mechanism on the chute time, as the first time interval. We speculate that ambulance crew's information about the EMS load forms an expectation about the likelihood of being dispatched in the near future. When the EMS load is high, it is more predictable for ambulance crew that they will receive a dispatch notification soon. So, they can start perpetration tasks before receiving the notification.

M 1.2 *Early task initiation (server, load, work content): Service predictability in EMS higher load enables ambulance crew to start preparation tasks before receiving a dispatch notification.*

We justify the *task reduction*, *task increase*, and *workload smoothing* mechanisms in the EMS system based on Hopp et al. (2007) discretionary task completion model, discussed in Section 2.4.1.2. Failure to meet response and service time targets has negative impact on the EMS system and even might result in ambulance shortage to cover new calls. Scene time and hospital time are the two time intervals with discretionary task completion criteria.

We believe that paramedics' information about the EMS load, besides other factors like urgency, influences the amount of time they spend on scene to cure a patient before deciding about whether to transport the patient to hospital. Our speculation is that as load increases up to a threshold, paramedics spend more time on the scene to treat the patient as they tend to avoid transporting the patient to hospital due to longer hospital wait in higher EMS load (due to the *downstream system congestion* mechanism, which we will discuss in Section 3.2.2). When EMS load is very high, paramedics prefer to shorten the scene time and instead continue the care process inside the ambulance while transferring the patient to hospital, probably in response to some protocols. This can be an example of the *workload smoothing* mechanism. We hypothesize:

M 1.3.a *Task increase (server, load, work content): When EMS load is below a critical threshold, paramedics spend more time on scene as load increases in order to stabilize patient condition on scene and avoid hospital transportation.*

M 1.3.b *Workload smoothing (server, load, work content): When EMS load is beyond a critical threshold, paramedics spend less time on scene as load increases and instead perform the care process inside the ambulance while transferring the patient to hospital.*

We also believe paramedics' information about the EMS load affects their discretion about whether to transfer the patient to hospital. We think that paramedics' expectation of longer hospital times when load is high makes them less inclined to transfer the patient to hospital. Therefore, we hypothesize:

M 1.4 *Task reduction (server, load, work content): The probability of hospital transportation decreases with load.*

The other time interval that involves discretionary task completion criteria is the hospital time. We can consider ED's staff, including nurses and physicians, in addition to ambulance crew as the servers involved in the hospital time interval. We believe paramedics and ED staff feel pressure to shorten hospital time when the EMS load is close to its limit. For example in Alberta, Canada, "ED surge capacity protocols" force EDs to accelerate the admission process of patients transferred by ambulances when fewer than 7 ambulances are available for service in the city (Alberta Health Services 2010). A suggested strategy in the protocol is freeing up capacity by moving current ED patients out of the ED or hospital beds. Such protocols encourage early discharges.

M 1.5 *Task reduction (server, load, work content): In very high EMS load and in order to shorten ambulance delays in hospitals, ED staff discharge ED patients early for faster accommodation of the patients arrived to the ED by ambulance.*

The *social pressure - speedup* mechanism is another potential server - load mechanism that affects EMS time intervals. This is mainly due to the importance of service speed on the EMS performance and meeting response and service time targets. Also, ambulance crew's actions are mostly monitored and tracked by computer aided dispatching systems in the dispatching center. Disentangling the service speed and the work content is not clear for chute, scene, and hospital time intervals. Whereas, it is possible to disentangle the work content and service speed for travel and transportation times by considering distance as the work content and the cruising speed as the service speed. The most plausible manifestation of the *social pressure - speedup* is increasing the cruising speed by ambulance drivers.

M 1.6 *Speedup (server, load, service speed): Ambulance drivers increase cruising speed as EMS load increases.*

3.2.1.3 EMS Server - Extended Load Mechanisms

As discussed previously, there are times that ambulance crew are required to perform *extended services* due to their proximity to a high volume demand region. Extended services are susceptible to extended load mechanisms, like the *overwork - slowdown* mechanism. Chute, travel, and transportation times do not involve demanding physical or mental tasks and do not seem to be influenced significantly by extended load mechanisms.

Scene time involves both physical and mental tasks. Therefore, we speculate that the overwork caused by performing extended service slows down paramedics and makes scene time longer. We also hypothesize that those paramedics that are more overworked by longer extended services are less inclined to transfer patients to hospital.

M 1.7 *Slowdown (server, extended load, work content): Extended load increases the amount of overwork and slows down paramedics performance on the scene.*

M 1.8 *Task reduction (server, extended load, work content): The probability of hospital transportation is lower for longer extended services.*

3.2.2 EMS Network Mechanisms

We can view the EMS system and the ED to which an ambulance transports patients as two nodes of a network. In this setting, ED is the downstream system of the EMS system and is a point of interest to look for network mechanisms. We also introduce a new perspective for network mechanisms of the EMS system: One can model the city road network that connects the dispatch location to the scene location and the scene location to the hospital as an infinite-server virtual queue that delays medical care delivery. This perspective allows us to identify new network mechanisms.

3.2.2.1 EMS Network - Changeover Mechanisms

We introduce a new mechanism for the effect of changeover on travel and transportation distances, which are the parameters of the EMS network configuration based on the new network perspective explained in the previous paragraph. We call the new mechanism *network arrangement*. This mechanism is interrelated to the *geographical dispersion* and *network chaos* mechanisms, which we will introduce in the next two subsections. For better clarification, we opt to introduce the *network arrangement* mechanism later, after we introduce the *network chaos* mechanism in the network - extended load subsection.

3.2.2.2 EMS Network - Load mechanisms

Like Forster et al. (2003), Asaro et al. (2007), and Hillier et al. (2009), the most apparent load mechanism to speculate here is the effect of *downstream system congestion* mechanism, ED congestion, on the hospital time. Assuming that EMS load is positively correlated with ED load, hospitals are crowded when EMS load is high. ED congestion causes ambulances to back up to offload patients, which results in longer hospital time. Early discharges in the ED in response to capacity management protocols, as discussed in mechanism M 1.5 (the *task reduction* mechanism), mitigate the effect of *downstream system congestion* mechanism when EMS load is above a critical threshold.

M 2.1 *Downstream system congestion (network, load, work content): ED congestion causes hospital time to increase with load when load is below a critical threshold.*

Now, we focus on the other perspective of the EMS network. Travel time causes delay in providing medical care to patients. For sure, the amount of delay depends on distance. The distance itself depends on the EMD's decision on dispatching the closest possible ambulance to the scene and also, ambulance crew's selection of the shortest route to the scene.

The travel distance and travel time are affected by two new network mechanisms of *geographical dispersion* and *geographical speedup*.

In mechanism M 2.2, we speculate that the *geographical dispersion* mechanism causes longer travel distance and time in high EMS system load. *Geographic dispersion* occurs because fewer ambulances cover a fixed geographic area (a city) when the EMS system load is high. As a result, ambulances need to travel further to a scene. This mechanism might also be relevant to other services, for example, repair and tow truck services, porters in hospitals, other emergency services (fire, police), and taxi and delivery services. The *geographical speedup* mechanism mitigates the *geographical dispersion* mechanism by enabling ambulance drivers to drive faster as it is likely that long trips involve at least some highway travel (Budge et al. 2010), as hypothesized in M 2.3.

M 2.2 *Geographical dispersion (network, load, work content): Distance increases with load because of the geographical dispersion mechanism.*

M 2.3 *Geographical speedup (network, load, service speed): Ambulance cruising speed increases with distance due to the geographical speedup mechanism.*

3.2.2.3 EMS Network - Extended Load Mechanisms

Ambulance locations in a city are picked so that they can cover all calls in a target response time. We argue that when EMS load is high for a long period, it causes more disarrangement of ambulances from their original planned positions, which leads to supoptimal dispatches and longer travel times. We call this the *network chaos* mechanism.

M 2.4 *Network chaos (network, extended load, work content): An ambulance needs to travel longer to a scene location as extended load continues for longer periods.*

When EMS load gets back to its normal situation and more ambulances become idle, ambulances have the chance to return to their original planned dispatch locations. So, changeover brings the EMS network arrangement. This is the *network arrangement* mechanism that we mentioned in the *EMS Network - Changeover Mechanisms* subsection and we postponed its explanation to here.

M 2.5 *Network arrangement (network, changeover, work content): Travel distances are shorter for regular (not extended) services.*

3.2.3 EMS Customer Mechanisms

In the EMS system, "customers" are callers or patients. Changeover and extended load characteristics do not seem to result in patient driven mechanisms.

3.2.3.1 EMS Customer - Load Mechanisms

Patients do not have direct knowledge about the EMS load. So, they cannot react directly to load. However, patients experience the consequences of high system load by a delayed response due to longer travel distances (M 2.2 - the *geographical dispersion* mechanism). As documented widely in the EMS research literature, long response times result in inferior patient condition (e.g., Feero et al. 1995, Blackwell and Kaufman 2002), which may induce the *service complication* mechanism. We hypothesize that the inferior patient condition results in complications that cause longer scene time and also, higher chance that the patient requires hospital transportation.

M 3.1 *Service complication (customer, load, work content): Longer response time in higher load due to longer travel times causes complications in patient health condition and increases scene time.*

M 3.2 *service complication (customer, load, work content): Longer response time in higher load due to longer travel times causes complications in patient health condition and increases the probability that the patient requires hospital transportation.*

3.3 Generating Hypotheses about EMS Service Time Intervals: Aggregating the Effects of the Identified Mechanisms

In this section, we aggregate the effects of the mechanisms identified in Sections 3.2.1-3.2.3 on each time interval and the total EMS service time.

3.3.1 Chute Time Hypotheses

We base our hypotheses about chute time on mechanisms M 1.1 (the *setup* mechanism) and M 1.2 (the *early task initiation* mechanism). Based on the shorter setup time involved in chute time of extended services and early task initiation of ambulance crew owing to the higher predictability of dispatching notifications in higher EMS load, we hypothesize:

H 1 *Chute time increases with changeover.*

H 2 *Chute time decreases with load.*

3.3.2 Travel and Transportation Times Hypotheses

We base hypothesis H 3 for the effect of changeover on travel time on M 2.5 (the *network arrangement* mechanism).

H 3 *Travel time decreases with changeover.*

M 1.5 (the *social pressure - speedup* mechanism), M 2.2 (the *geographical dispersion* mechanism), and M 2.3 (the *geographical speedup* mechanism) form our hypothesis for the effect of load on travel time. Ambulances travel in higher speed when load increases because of the *social pressure - speedup* mechanism and the *geographical speedup* mechanism. On the other hand, the travel distance increases with system load because of the *geographical dispersion* mechanism. We believe that the *geographical dispersion* mechanism dominates the other two mechanisms. We make the same argument about the transportation time. So,

H 4 *Travel time increases with load.*

H 5 *Transportation time increases with load.*

Finally, M 2.4 (the *network chaos* mechanism) form our hypothesis for the effect of extended load on travel time:

H 6 *Travel time increases as high load periods last longer.*

3.3.3 Scene Time Hypotheses

We do not speculate any changeover mechanism that affects the scene time. Our hypothesis for the aggregated effect of load on scene time results from M 1.3.a (the *task increase* mechanism), M 1.3.b (the *workload smoothing* mechanism), and M 3.1 (the *service complication* mechanism). Based on the three mentioned mechanisms, we expect to find a concave relation between scene time and load as expressed in H 7.

H 7 *scene time increases with load below a critical threshold and decreases with load above the threshold.*

M 1.6 (the *overwork - slowdown* mechanism) is the only hypothesis that relates extended load to scene time. We express hypothesis H 8 based on M 1.6.

H 8 *Scene time increases with extended load.*

3.3.4 Hospital Time Hypotheses

Like scene time, we do not speculate any changeover mechanism that affects hospital time. The network - load mechanism of *downstream system congestion* increases hospital time up to a critical load threshold (M 2.1), but the ED surge capacity planning protocols force early discharges in the ED after the critical EMS load threshold to accommodate patients transferred by ambulances faster (M 1.5). So, we predict a concave relation between the hospital time and the EMS load.

H 9 *Hospital time increases with load up to a threshold but decreases with load after the threshold.*

3.3.5 EMS Total Service Time Hypotheses

Chute, travel, and scene times are the three time intervals for without-hospital transportation services. Changeover and extended load increase chute and scene times, respectively (H 1 and H 8). Load reduces chute time (H 2), increases travel time (H 4), and increases scene time up to a threshold and reduces it beyond the threshold (H 7). Transportation time and hospital time are added to the service time for services with hospital transportation. Changeover and extended load do not affect these two intervals. Load's effect on transportation time is similar to it's effect on travel time (H 5). Like scene time, hospital time varies concavely with load (H 9). Based on the above discussion, we express our hypotheses for the total EMS service time as follows:

H 10 *Service time of without and with-hospital transportation services increases with changeover.*

H 11 *Service time of without and with-hospital transportation services increases with load below a threshold and decreases with load above the threshold.*

H 12 *Service time of without and with-hospital transportation services increases with extended load.*

When we discussed about the discretionary task completion criteria of the scene time, we argued that paramedics' decision on transporting a patient to a hospital or spending more time on scene to cure the patient is influenced by EMS load (M 1.4 - *task reduction* mechanism). On the other hand, load increases the probability of hospital transportation due to *service complication* mechanism because of longer response time (M 3.2). If so, patients with almost similar urgency might experience different service times in different EMS load situations because one has been transported to hospital while the other has not been transported to hospital. We expect the *task reduction* mechanism dominates the *service complication* mechanism. knowing that services with hospital transportation are generally longer than services without hospital transportation, we hypothesize:

H 13 *Service time of a random call decreases with load.*

3.4 Testing Hypotheses: Calgary EMS System

We test our proposed hypotheses by analyzing a 2009 data set for the EMS system of the city of Calgary, Canada. The data set contains $108,423$ call records. We focus on $92,893$ calls for which an ambulance was dispatched. The information for a call includes: (1) time stamps for different events generated by the EMD and paramedics, (2) coordinates for the ambulance dispatch location, call address, and hospital location, (3) call priority numbers assigned by the EMD (a number from 1 to 7 with 1 being assigned to Delta/Echo or the most critical priority calls), and (4) the number of busy and scheduled ambulances at the moment of call arrival. We use this information to extract the following variables for each call:

- The length of the EMS time intervals: T^{Chute}, T^{Travel}, T^{Scene}, $T^{\text{Transport}}$, and T^{Hospital}
- Response and service times for without-hospital transportation (T^{SWOT}) and with-hospital transportation (T^{SWT}) services:

$$\begin{aligned}
&T^{\text{Response}} = T^{\text{Chute}} + T^{\text{Travel}}\\
&T^{\text{SWOT}} = T^{\text{Chute}} + T^{\text{Travel}} + T^{\text{Scene}}\\
&T^{\text{SWT}} = T^{\text{Chute}} + T^{\text{Travel}} + T^{\text{Scene}} + T^{\text{Transport}} + T^{\text{Hospital}}
\end{aligned}$$

- Travel (D) and transportation (D') distances
- Travel (S) and transportation (S') speed. We assume a uniform speed in the entire length of a trip and we compute the average ambulance speed by $S = D/T^{\text{Travel}}$ and $S' = D'/T^{\text{Transport}}$.
- Number of busy ambulances at the dispatch notification (NB_1), scene arrival (NB_2), scene departure (NB_3), and hospital arrival (NB_4) moments
- Whether the patient is transported to hospital ($I_T = 1$ for hospital transportation, $I_T = 0$ otherwise)
- Whether the call is evaluated as life threatening with a Delta/Echo priority ($I_U = 1$ for Delta/Echo priority, $I_U = 0$ otherwise)
- Whether light and siren is deployed during the travel and transportation times ($I_S = 1$ for light and siren, 0 otherwise; light and siren is deployed for priority numbers 1 to 4)
- To capture changeovers, we identify whether the responding ambulance received the dispatch notification in the standby mode ($I_C = 1$ for regular service) or it received

Measure	T^{Chute}	T^{Travel}	T^{Scene}	$T^{\text{Transport}}$	T^{Hospital}	T^{SWOT}	T^{SWT}
Mean	0.96	7.13	28.59	16.57	69.38	44.49	117.50
Median	0.73	5.78	23.35	14.57	56.97	34.58	105.9
Standard deviation	3.84	6.87	30.76	11.01	51.04	44.57	55.50
Coefficient of variation	4.00	0.96	1.08	0.66	0.74	1.06	0.47

Table 3.2: Descriptive statistics for service time intervals (minutes)

Measure	$\overline{NS}$	$\overline{NB}$	$\overline{L}$	D (km)	D' (km)
Mean	42.78	18.84	0.44	3.89	13.27
Median	43.00	19.00	0.43	3.11	12.37
Standard deviation	7.28	6.97	0.13	3.07	6.94
Coefficient of variation	0.37	0.17	0.29	0.79	0.52

Table 3.3: Descriptive statistics for explanatory variables

while returning from a previous service ($I_C = 0$ for extended service). We identify a service as an extended service if the time between the start of the service and the finish of a previous call responded by the same crew is less than 10 minutes.

- We compute the EMS load at the dispatch notification (L_1), scene arrival (L_2), scene departure (L_3), and hospital arrival (L_4) of a service as the fraction of busy ambulances by $L_i = NB_i/NS, i = 1, \ldots, 4$. Then, we compute the average load during the whole service time of a call $\overline{L}$ as the average of load at the moments included in the service.

- We quantify extended load in the server level (O) by computing the time length a specific ambulance crew has been continuously serving since their last changeover.

Tables 3.2 and 3.3 provide descriptive statistics of the variables of interest. Hospital time is the longest time interval with an average of 69.38 minutes and chute time is the shortest time interval with an average of almost a minute. The average service time for a without-hospital transportation service is 44.49 minutes, whereas the average service time for services with hospital transportation is 117.50 minutes. Almost 58% of services include hospital transportation. This sets the average service time of all calls at 86.84 minutes.

In the remainder of this section, we provide our regression models and we test our hypotheses. In the regression models $\mathbf{X}$ denotes a vector of control variables including

Coefficient	T^{Chute} (min.) Model (3.1)
Intercept	$-1.60(1.47)$
I_C	$0.23(0.04)^{***}$
L_1	$-0.28(0.13)^{*}$
NS	$0.05(0.03)$
D	$0.06(0.00)^{***}$
I_U	$-0.02(0.03)$
R^2	0.0032
P-val.	$< 2.2e-16$

Table 3.4: Effect of load on chute time ($***$, $**$, $*$ denote statistical significance at the 0.1%, 1%, and 5% significance levels, respectively. Standard errors are shown in parentheses.)

day, time, and day and time interaction variables, ϵ is the error term, and "$\times$" represents interactions. We have dummy variables for days of the week and dummy variables for hours of the day, plus interactions between these two sets of dummy variables.

3.4.1 Testing Chute Time Hypotheses

We test hypotheses H 1 (*chute time increases with changeover*) and H 2 (*chute time decreases with load*) by Model (3.1). A significant positive coefficient for I_C supports hypothesis H 1 and provides evidence for its predecessor mechanism M 1.1 (the *setup* mechanism). A significant negative coefficient for L_1 supports hypothesis H 2 and provides evidence for its predecessor mechanism M 1.2 (the *early task initiation* mechanism). We include D in Model (3.1) as it may take longer for ambulance crew to find the address and the best route if scene location is outside the normal coverage region of a unit.

$$T^{\text{Chute}} = \beta_0 + \beta_{I_C} I_C + \beta_{L_1} L_1 + \beta_{NS} NS + \beta_D D + \beta_{I_U} I_U + \beta_{\mathbf{X}} \mathbf{X} + \epsilon. \tag{3.1}$$

Table 3.4 presents the regression coefficients of Model (3.1). The coefficient for I_C supports H 1, which can be explained by the *setup* mechanism as proposed in M 1.1. Based on the coefficient of L_1, an increase in load from 10% to 90% shortens the chute time by almost 15 second. This provides an evidence for mechanism M 1.2 and the *early task initiation* mechanism.

3.4.2 Testing Travel and Transportation Times Hypotheses

We test hypotheses H 3 (*travel time decreases with changeover*) and H 4 (*travel time increases with load*) by Model (3.2). The predecessor mechanism for H 3 is mechanism M 2.5 (the *network arrangement* mechanism), which we test by Model (3.4). Significant negative coefficients for I_C in Models (3.2) and (3.4) support H 3 and its predecessor M 2.5, respectively.

Hypothesis H 4 has three predecessor mechanisms: M 1.6 (the *social pressure - speedup* mechanism), M 2.2 (the *geographical dispersion* mechanism), and M 2.3 (the *geographical speedup* mechanism). We test M 2.2 by Models (3.3) and (3.4). We expect the magnitude and significance of the coefficient of L_1 attenuate from Model (3.2) to Model (3.3). A significant positive coefficient for L_1 in (3.4) supports the *geographical dispersion* mechanism.

Model (3.5) tests the other two mechanisms for H 4: the *social pressure - speedup* and *geographical speedup* mechanisms. A significant positive coefficient for D supports the *geographical speedup* mechanism and a significant positive coefficient for L_1 provides evidence for the *social pressure - speedup* mechanism.

$$T^{\text{Travel}} = \beta_0 + \beta_{I_C} I_C + \beta_{L_1} L_1 + \beta_{NS} NS + \beta_{I_U} I_U + \beta_{I_S} I_S + \beta_{\mathbf{X}} \mathbf{X} + \epsilon, \tag{3.2}$$

$$T^{\text{Travel}} = \beta_0 + \beta_{I_C} I_C + \beta_{L_1} L_1 + \beta_D D + \beta_{NS} NS + \beta_{I_U} I_U + \beta_{I_S} I_S + \beta_{\mathbf{X}} \mathbf{X} + \epsilon. \tag{3.3}$$

$$D = \beta_0 + \beta_{I_C} I_C + \beta_{L_1} L_1 + \beta_{NS} NS + \beta_{I_U} I_U + \beta_{I_S} I_S + \beta_{\mathbf{X}} \mathbf{X} + \epsilon, \tag{3.4}$$

$$S = \beta_0 + \beta_{I_C} I_C + \beta_{L_1} L_1 + \beta_D D + \beta_{NS} NS + \beta_{I_U} I_U + \beta_{I_S} I_S + \beta_{\mathbf{X}} \mathbf{X} + \epsilon. \tag{3.5}$$

Table 3.5 presents the coefficients of regression Models (3.2)-(3.5). Although the coefficient of I_C in Model (3.4) supports the *network arrangement* mechanism (M 2.5), its coefficient in Model (3.2) supports the opposite of what we hypothesized in H 3: Travel time of a regular service, the first service after a changeover, is longer than travel time of an extended service. One explanation for this result is that regular services require the acceleration phase to travel on residential or arterial roads before reaching to the cruising speed phase corresponding to highway travel (Budge et al. 2010). Whereas, extended services are likely to be initiated while the ambulance is already in the cruising speed phase. Therefore, the

Coefficient	T^{Travel} (min.) Model (3.2)	T^{Travel} (min.) Model (3.3)	D (km) Model (3.4)	S (km/min.) Model (3.5)	S (km/min.) Model (3.5) $D < 5$
Intercept	$7.75(2.36)^{**}$	$3.69(2.29)$	$4.05(1.25)^{**}$	$-7.41(7.83)$	$-4.87(5.16)$
I_C	$0.26(0.07)^{***}$	$0.43(0.07)^{***}$	$-0.10(0.04)^{**}$	$-1.24(0.24)^{***}$	$-0.78(0.15)^{***}$
L_1	$4.20(0.21)^{***}$	$0.04(0.20)$	$4.38(0.10)^{***}$	$-0.50(0.67)$	$0.99(0.43)^{*}$
D		$0.97(0.01)^{***}$		$0.48(0.02)^{***}$	$0.25(0.04)^{***}$
NS	$0.03(0.05)$	$0.03(0.05)$	$0.01(0.03)$	$0.17(0.17)$	$0.14(0.10)$
I_U	$-1.42(0.05)^{***}$	$-1.32(0.05)^{***}$	$-0.13(0.03)^{***}$	$-0.09(0.17)$	$0.01(0.10)$
I_S	$-4.55(0.11)^{***}$	$-2.83(0.10)^{***}$	$-1.54(0.06)^{***}$	$-0.09(0.35)$	$-1.04(0.25)^{***}$
R^2	0.0487	0.2543	0.0431	0.0063	0.0015
P-val.	$< 2.2e-16$	$< 2.2e-16$	$< 2.2e-16$	$< 2.2e-16$	0.0003

Table 3.5: Effect of load on travel distance and speed ($***$, $**$, $*$ denote statistical significance at the 0.1%, 1%, and 5% significance levels, respectively. Standard errors are shown in parentheses.)

advantage of higher travel speed for extended services surpasses the advantage of shorter distance for regular services. This is verified by the coefficient of I_C in Model (3.5), where speed is the dependent variable.

Model (3.2) results support hypothesis H 4. Results of Models (3.3) and (3.4) support mechanism M 2.2 (the *geographical dispersion* mechanism). Based on Model (3.4), travel distance increases by 0.43 kilometers for 10% increase in load. The attenuation of the effect of load in Model (3.3) compared to Model (3.2) suggests that the *geographical dispersion* mechanism is the dominant mechanism that affects the travel time. The coefficients of determination (R^2) of Models (3.2) and (3.3) imply that distance explains almost 20% of the variability in the travel time.

Based on Model (3.5), traveling speed for farther distances is faster. This implies the *geographical speedup* mechanism (M 2.3) as distance itself increases with load. The coefficient of L_1 in (3.5) is not significant. This is partly because the *geographical speedup* mechanism dominates the *social pressure-speedup* mechanism. When we run Model (3.5) for shorter travel distance services (less than 5 kilometers) where the power of the *geographical dispersion* mechanism is limited, the coefficient of L_1 becomes significant with a negative sign. This provides evidence for the *social pressure-speedup* mechanism, at least in short distances.

We run similar regression models as Models (3.2)-(3.5) on the transportation time to test hypothesis H 5 (*transportation time increases with load*) and the three predecessor mechanisms M 1.6 (the *social pressure-speedup* mechanism), M 2.2 (the *geographical dispersion* mechanism), and M 2.3 (the *geographical speedup* mechanism). We did not find enough evidence to support H 5 or any of the mechanisms. Unlike scene locations, ambulances transport patients to a fixed set of hospitals regardless of the EMS load. This explains why the *geographical dispersion* mechanism is not an effective mechanism for the transportation time. Although the transportation speed increases with distance, we do not observe the *geographical speedup* mechanism here since distance is not driven by load.

3.4.3 Testing Scene Time Hypotheses

We test hypotheses H 7 (*scene time increases with load below a critical threshold and decreases with load above the threshold*) and H 8 (*scene time increases with extended load*) by Model (3.6). A significant negative coefficient for L_2^2 supports the concave relation between the scene time and load. It also implies the predecessor mechanisms M 1.3.a (the *task increase* mechanism) and M 1.3.b (the *workload smoothing* mechanism). Also, a significant positive coefficient for T^{Response} supports the *service complication* mechanism (M 3.1) as response time increases with load due to longer travel time. A significant positive coefficient for O supports H 8 and the predecessor mechanism M 1.7 (the *overwork-slowdown* mechanism). We include interaction terms $L_2^2 \times I_U$ and $L \times I_U$ in Model (3.6) as we believe the outlined mechanisms are more pronounced for less urgent patients.

$$\begin{aligned} T^{\text{Scene}} =& \beta_0 + \beta_{L_2^2} L_2^2 + \beta_{L_2} L_2 + \beta_O O + \beta_{T^{\text{Response}}} T^{\text{Response}} + \beta_{NS} NS + \beta_{I_U} I_U + \\ & \beta_{I_T} I_T + \beta_{L_2^2 \times I_U} L_2^2 \times I_U + \beta_{L \times I_U} L \times I_U + \beta_{\mathbf{X}} \mathbf{X} + \epsilon. \end{aligned} \tag{3.6}$$

Table 3.6 presents results of Model (3.6). The results support the concave relation between load and scene time (H 7) and provides evidence for the *task increase*, *workload smoothing*, and *service complication* mechanisms. Figure 3.2 plots scene time versus load for transported urgent and non-urgent services based on Table 3.6 results and average values of other independent variables. Model (3.6) also supports the slowdown mechanism and its effect on increasing scene time due to the extended load.

Coefficient	T^{Scene} (min.) Model (3.6)
Intercept	$-1.95(8.91)$
L_2^2	$-29.47(4.79)^{***}$
L_2	$26.40(4.42)^{***}$
O	$0.01(0.00)^{***}$
T^{Response}	$0.50(0.01)^{***}$
NS	$0.63(0.23)^{**}$
I_U	$3.93(1.83)^{**}$
I_T	$-12.18(0.20)^{***}$
$L_2^2 \times I_U$	$28.37(9.24)^{**}$
$L_2 \times I_U$	$-24.15(8.41)^{**}$
R^2	0.0647
P-val.	$< 2.2e-16$

Table 3.6: Effect of load on scene time ($***$, $**$, $*$ denote statistical significance at the 0.1%, 1%, and 5% significance levels, respectively. Standard errors are shown in parentheses.)

Figure 3.2 shows how scene time changes with load for urgent and non-urgent services based on Model (3.6) and average values for $NS = 42.79$, $T^{\text{Response}} = 8.09$ minutes, and $O_2 = 6.75$ minutes.

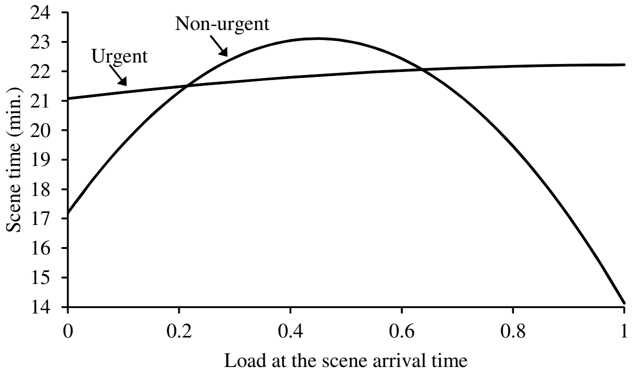


Figure 3.2: Scene time vs. load based on Model (3.6); $NS = 42.79$, $T^{\text{Response}} = 8.09$ minutes, and $O_2 = 6.75$ minutes

3.4.4 Testing Hospital Time Hypotheses

We test hypotheses H 9 (*Hospital time increases with load up to a threshold but decreases with load after the threshold*) and the predecessor mechanisms M 2.1 (the *downstream system congestion* mechanism) and M 1.5 (the *task reduction* mechanism) by Model

Coefficient	T^{Hospital} (min.) Model (3.7)
Intercept	$32.73(19.62)$
L_4^2	$-36.11(8.87)^{***}$
L_4	$68.16(8.32)^{***}$
NS	$0.04(0.51)$
I_U	$5.21(0.49)^{***}$
R^2	0.0501
P-val.	$< 2.2e-16$

Table 3.7: Effect of load on hospital time ($***$, $**$, $*$ denote statistical significance at the 0.1%, 1%, and 5% significance levels, respectively. Standard errors are shown in parentheses.)

(3.7). A significant negative coefficient for L_4^2 supports the concave relation between the hospital time and load. It also provides evidence for the *downstream system congestion* mechanism (M 2.1) when load is below a critical threshold and the *task reduction-early discharge* mechanism (M 1.5) when load is above the critical threshold.

$$T^{\text{Hospital}} = \beta_0 + \beta_{L_4^2} L_4^2 + \beta_{L_4} L_4 + \beta_{NS} NS + \beta_{I_U} I_U + \beta_{\mathbf{X}} \mathbf{X} + \epsilon. \tag{3.7}$$

The results of Model (3.7), as presented in Table 3.7, support the concave relation between load and hospital time (H 9). Plot 3.3 shows the regression line of Model (3.7), estimated by Table 3.7 results, which illustrates the effects of *downstream system congestion* and *task reduction-early discharge* mechanisms.

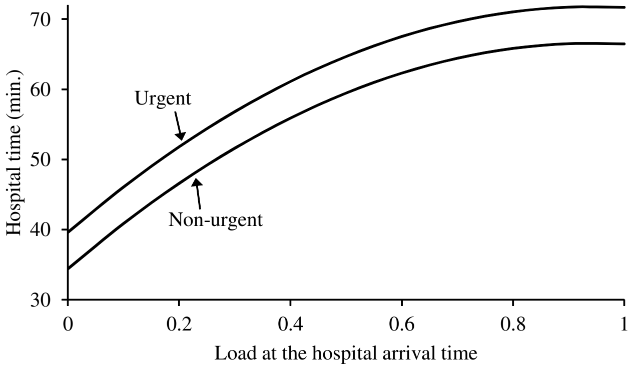


Figure 3.3: Hospital time vs. load based on Model (3.7); $NS = 42.79$

Coefficient	T^{Service}
	Model (3.8)
Intercept	$-5.32(18.45)$
I_C	$6.75(1.20)^{***}$
$\overline{L}^2$	$-33.92(13.03)^{**}$
$\overline{L}$	$21.89(12.13)$
O	$0.14(0.01)^{***}$
I_T	$28.78(3.84)^{***}$
NS	$0.58(0.40)$
I_U	$-0.59(0.41)$
$I_C \times I_T$	$3.26(1.66)^{*}$
$\overline{L}^2 \times I_T$	$-52.23(17.03)^{**}$
$\overline{L} \times I_T$	$125.41(15.65)^{***}$
$\overline{O} \times I_T$	$-0.09(0.02)^{***}$
R^2	0.3923
P-val.	$< 2.2e - 16$

Table 3.8: Effect of load on service time (in minutes) ($***$, $**$, $*$ denote statistical significance at the 0.1%, 1%, and 5% significance levels, respectively. Standard errors are shown in parentheses.)

3.4.5 Testing EMS Total Service Time Hypotheses

We construct Model (3.8) to test hypoteses H 10 (*service time of without and with-hospital transportation services increases with changeover*), H 11 (*service time of without and with-hospital transportation services increases with load below a threshold and decreases with load above the threshold*), and H 12 (*service time of without and with-hospital transportation services increases with extended load*). Significant negative coefficients for I_C and $\overline{L}^2$ when $I_T = 0$ and $I_T = 1$ support H 10 and H 11. Significant positive coefficients for $\overline{O}$ when $I_T = 0$ and $I_T = 1$ support H 12. The results for Model (3.8), presented in Table 3.8, support hypotheses H 10, H 11, and H 12.

$$\begin{aligned} T^{\text{Service}} =& \beta_0 + \beta_{I_C} I_C + \beta_{\overline{L}^2} \overline{L}^2 + \beta_{\overline{L}} \overline{L} + \beta_O O + \beta_{I_T} I_T + \beta_{NS} NS + \beta_{I_U} I_U + \\ & \beta_{I_C \times I_T} I_C \times I_T + \beta_{\overline{L}^2 \times I_T} \overline{L}^2 \times I_T + \beta_{\overline{L} \times I_T} \overline{L} \times I_T + \beta_{O \times I_T} O \times I_T + \beta_{\mathbf{X}} \mathbf{X} + \epsilon. \end{aligned} \tag{3.8}$$

The other hypothesis to test is H 13 (*Service time of a random call decreases with load.*). We cannot test H 13 directly but we can test its predecessors M 1.4 (the *task reduction*

Coefficient	logit$[\Pr(I_T = 1)]$ Model (3.9)
Intercept	$0.67(0.65)$
L_3	$-0.56(0.07)^{***}$
T^{Response}	$0.00(0.00)$
NS	$-0.01(0.02)$
I_U	$-0.35(0.15)^{*}$
$L_3 \times I_U$	$-0.13(0.12)$

Table 3.9: Effect of load on transportation time and probability ($***$, $**$, $*$ denote statistical significance at the 0.1%, 1%, and 5% significance levels, respectively. Standard errors are shown in parentheses.)

mechanism: *the probability of hospital transportation decreases with load*) and M 3.2 (the *service complication* mechanism: *the probability of hospital transportation increases with longer response time when EMS load is high*). We test mechanisms M 1.4 and M 3.2 by logistic Model (3.9). A significant negative coefficient for L_3 supports M 1.4 and the *task increase* mechanism. A significant positive coefficient for T^{Response} supports M 3.2 and the *service complication* mechanism.

$$\begin{aligned}\text{logit}[\Pr(I_T = 1)] =& \beta_0 + \beta_{L_3} L_3 + \beta_{T^{\text{Response}}} T^{\text{Response}} + \beta_{NS} NS + \beta_{I_U} I_U + \\ & \beta_{L_3 \times I_U} L_3 \times I_U + \beta_{\mathbf{X}} \mathbf{X} + \epsilon. \end{aligned} \tag{3.9}$$

Table 3.9 presents the results of Model (3.9). The results support the *task reduction* mechanism to avoid long hospital times in higher EMS load but they do not provide evidence for the *service complication* mechanism's effect on the probability of transporting patients to hospital. According to Table 3.9, as load increases from 10% to 90%, the probability that a patient is transported to hospital decreases from 49% to 38% for non-urgent calls. This result can be interpreted as an indication to support H 13 but we cannot construct a model to test it directly.

CHAPTER 4

Modeling Load and Overwork Effects in Queueing Systems with Adaptive Servers

4.1 Introduction

Most capacity planning and queueing models are based on an assumption that servers work at a constant speed. This assumption is a simplification of reality, and researchers have documented various ways in which the assumption is violated. A typical finding is that servers speed up when the system *load*, usually measured by the system occupancy, increases (Edie 1954, Kc and Terwiesch 2012, Gans et al. 2010, Kuntz et al. 2011, Chan et al. 2012, Tan and Netessine 2012). Some researchers have hypothesized, and in some cases verified, that such speedup cannot be sustained indefinitely, and therefore servers slow down when load remains high over an extended period (Sze 1984, Gans et al. 2010, Dietz 2011), a situation that has been referred to as *overwork* (Kc and Terwiesch 2009).

We believe it is important to study adaptive server behavior from three perspectives: (1) empirically, to establish fundamental knowledge about whether, how, and why servers in real systems adapt, which we focused on in Chapters 2 and 3, (2) analytically, to develop tractable models that incorporate the main aspects of real server behavior, and (3) prescriptively, to investigate the impact on solution quality of accounting for adaptive server behavior in models used to generate solutions. In this chapter, we focus on the second perspective, of developing tractable models. We also touch on the third perspective, by illustrating possible negative impacts on solution quality that might result from ignoring adaptive server behavior.

We extend the commonly used Erlang C capacity-planning model to allow server speed to depend on load and overwork, we derive the performance measures of the extended model, and we investigate the "errors" that can result from using a constant-server-speed model to predict performance or prescribe capacity levels. Extending the Erlang C model to allow server speed to depend on load is not difficult and has been accomplished by researchers such as Jackson (1963). It is more challenging to allow server speed to also depend on overwork while maintaining model tractability, and it appears that no one has undertaken that analysis.

The modeling challenge is to operationalize overwork through an index that does not require detailed memory of the past history of the system. The indices that we investigate are all based on keeping track of "high-load periods" through the concept of a *k-partial busy period*, which is a period during which k or more of the s servers in the system are busy serving users. Typically, we select k to correspond to the average number of busy servers—for example, in a 10-server system with 80% long-term average utilization, we would set k to $0.8 \times 10 = 8$, which means that overwork begins to impact server speed when 8 or more of the 10 servers are simultaneously busy. (Our models do not restrict k to equal the average number of busy servers, however.) During a k-partial busy period, one could track various cumulative overwork measures. We have investigated three cumulative measures: number of service completions, elapsed time, and elapsed time summed over all busy servers. We focus on the first measure, which leads to a tractable two-dimensional Markov chain, with one dimension corresponding to load and the other dimension corresponding to overwork. The Markov chain has a special structure that allows it to be formulated as a quasi-birth-and-death (QBD) process. We exploit this special structure to design efficient algorithms to compute system performance measures.

4.2 Literature Review

We survey two streams of related work: First, empirical studies that document that service rates vary when system conditions change (Section 4.2.1) and second, analytical and simulation studies that investigate the performance of state-dependent systems or develop optimal service rate control policies (Section 4.2.2).

4.2.1 Empirical Studies

Batt and Terwiesch (2012) categorize means through which system load affects service rate as either speedup or slowdown mechanisms. We use the speedup and slowdown categories to organize our review of empirical papers.

Speedup effects have been observed in many contexts. Edie (1954), the earliest empirical study we know of, reports that toll booth holding times (service times) at the Lincoln Tunnel and the George Washington Bridge decrease with traffic volume and the number of open booths because (1) the collectors expedite the operation under the backed-up traffic pressure and (2) the drivers are more likely to have their payment ready before they reach the toll booth. Sze (1984) anecdotally reports that Bell System telephone operators work faster during overloaded periods to work off the queue. Tan and Netessine (2012) and Staats and Gino (2012) document speedup behavior of restaurant waiters and bank loan application processors, respectively.

There are also several reports of speedup effects in healthcare systems. Kc and Terwiesch (2009) find evidence of speedup in two distinctly different operations in a hospital, patient transportation and cardiothoracic surgery, where patient transportation time and patient length of stay (LOS) decrease with the number of busy transporters and the number of occupied beds, respectively. Kc and Terwiesch (2012) and Chan et al. (2012) argue that when a cardiac intensive care unit (ICU) is full and a new patient needs to be admitted, care providers are likely to discharge the most stable patient early. Batt and Terwiesch (2012) observe speedup effects for several emergency department (ED) patient care tasks. Kuntz et al. (2011) report a nonlinear relation between bed occupancy level and hospital LOS and confirm that physicians use their discretion over early discharge when load is very high. However, load does not appear to affect LOS when the utilization is low.

Turning to slowdown mechanisms, Sze (1984) lists the slowdown behavior of telephone operators after long high-load periods without relief, as one of the complexities of workforce management in call centers. Dietz (2011) observes a positive correlation between call volume and average service time when call volume is high and hypothesizes that “shift fatigue" leads to longer service times. Gans et al. (2010) define *run length*, the number of services an agent has performed since the last gap of longer than one hour, as a proxy for how overworked a call center agent is, and they find that higher run length is associated with longer average call times for some agents.

Kc and Terwiesch (2009) show that the load effect for in-hospital patient transportation time and cardiothoracic surgery patient LOS is not permanent and overwork, measured as the excess load over a time period that extends a specified number of time periods into the past, eventually slows down transporters and medical staff. Batt and Terwiesch (2012) also find evidence of slowdown in such ED tasks as lab specimen collection and X-ray imaging. Armony et al. (2010) show that service rate decreases with load when the number of patients within an ED is high and conjecture that ED medical staff slow down when they feel overwhelmed by the system pressure.

4.2.2 Analytical and Simulation Studies

Analytical queueing models with load-dependent service rates have a long history, dating back to Jackson (1963), who obtained the joint probability distribution of the queue lengths in a network with Markovian routing. In this system, arrivals and service completions at each station follow generalized Poisson processes with mean rates that depend on the total number of users and the queue length at each station, respectively. Welch (1964) and Harris (1967) focus on the $M/G/1$ model, extending it to allow the service time distribution to depend on whether the system is empty when service begins (Welch 1964) or on the queue length when service begins (Harris 1967).

Gans et al. (2010) and Powell and Schultz (2004) use simulation to investigate the behavior of systems with state-dependent service rates. Gans et al. (2010) demonstrate that accounting for adaptive server behavior improves capacity planning and Powell and Schultz (2004) show that adaptive server behavior benefits system throughput.

Assuming that service rate is controllable and waiting and service costs associated with a service rate are known, Crabill (1972) studies how to optimally choose state-dependent service rates for a single-server queue to minimize the long-run expected cost. George and Harrison (2001) show that the optimal service rate is increasing in the queue length. Berk and Moinzadeh (1998) present an analytical model where patient discharge time is affected by the occupancy level of the healthcare unit and explore the impact of early discharges on effective capacity. Chan et al. (2012) develop a state-dependent queueing model of an ICU in which physicians discharge patients earlier when the ICU is overloaded in order to accommodate new urgent patients. Speedup can alleviate high congestion in some situations, but may reduce future bed availability due to readmission of prematurely discharged patients. They investigate polices to determine when and how speedup should be used.

We make the following contributions: (1) We extend multi-server queueing models where the service rate depends on load to also incorporate dependence on overwork. (2) We formulate our model as a level-dependent QBD process that can accommodate any functional dependence of service rates on load and overwork and we provide formulas and efficient algorithms to compute steady state probabilities and system performance measures. (3) Our experiments demonstrate the magnitude of the errors that result from using fixed-service-rate models to predict performance. (4) We illustrate several types of unintended consequences that can result from using fixed-service-rate models to prescribe staffing in systems with state-dependent service rates, including oscillatory staffing, unstable staffing, and convergence to a suboptimal staffing level.

4.3 Operationalizing Overwork

In this section, we discuss alternative ways to operationalize the concept of overwork—a situation where the system has been under a heavy load for an extended time period. We review how other researchers have operationalized overwork and related constructs in order to measure overwork empirically, we discuss why it is challenging to incorporate previously proposed overwork measures in a stochastic model, and we outline a family of overwork measures that can be incorporated in a Markov chain through the addition of only one state variable.

Overwork might result in human servers (for example, hospital porters or call center agents) becoming fatigued, resulting in a slowdown in service delivery. In settings where a "server" represents a bundle of human and other resources, overwork could influence service speed through more complex mechanisms. If one views a bed and the resources needed to serve a patient in a cardiothoracic surgery ward as a server, for example, then Kc and Terwiesch (2009) argue that overworked physicians tend to prescribe more testing, which can delay patient discharge.

We know of two empirical studies that operationalize overwork. Kc and Terwiesch (2009) measure overwork as the average excess load over the last T time periods. For example, suppose that the average load is 2 requests per server per hour, but in the previous $T = 5$ hours, each server has handled 3 requests per hour. The resulting overwork is $3 - 2 = 1$ *extra* request per server per hour for a total amount of $1 \times 5 = 5$ units of overwork. In a similar vein, Gans et al. (2010) define "run length" as the number of calls an agent has answered since the last service gap of longer than an hour to measure overwork for an individual agent. This means that if the last service gap of an agent ended one hour ago and during the previous hour the agent handled 50 calls, then the agent overwork is measured as 50. Notice that both operationalizations have one free parameter ("T" for Kc and Terwiesch (2009) and "one hour" for Gans et al. (2010)) for which the most appropriate value is not obvious. Kc and Terwiesch (2009) choose a value for T that maximizes model fit, but Gans et al. (2010) do not vary the "one hour."

Although the overwork variables that Kc and Terwiesch (2009) and Gans et al. (2010) defined can be measured empirically, they require historical information about service completions in previous time periods and from a stochastic modeling perspective, using these definitions requires a high-dimensional state space. If one were to use the Kc and Terwiesch (2009) operationalization in a Markov chain, then one would need to include T state variables—the number of users in the system in each of the last T time periods. If the system capacity is N and one also includes a state variable for the current number of users in the system, then the cardinality of the state space is $(N+1)^{T+1}$. The exponential growth in the cardinality of the state space as the number of state variables increases is known as the *curse of dimensionality* (Bellman 1961). With the Gans et al. (2010) operationalization, the curse of dimensionality is even more pronounced, because one cannot bound how far back the model's "memory" should reach in order to capture the last service gap that was longer than an hour.

We propose a family of tractable overwork measures, all of which are defined using the concept of a k-partial busy period, $k = 1, \cdots, s$, which is a period that commences when an arrival finds $k-1$ users in the system and ends when a departure leaves the system with $k-1$ users (Artalejo and Lopez-Herrero 2001). Every overwork measure that we consider equals zero when a k-partial busy period begins, increases during the k-partial busy period, and is reset to zero when the k-partial busy period ends. Three examples of such overwork measures are (1) the number of users served, $J(t)$, (2) the elapsed time, $E(t)$, and (3) the cumulative service time across all servers, $C(t)$,—all measured up to time t in the current k-partial busy period. In this chapter, we focus on,

$$J(t) = \text{the number of users served up to time } t \text{ in the current } k\text{-partial busy period}$$
$$(0 \text{ if the system is not in a } k\text{-partial busy period}), \tag{4.1}$$

and we incorporate it as the second state variable in a continuous-time Markov chain (CTMC).

If the average system utilization (proportion of busy servers) is ρ and one sets $k = \lceil \rho s \rceil$, then $J(t)$ is analogous to the Kc and Terwiesch (2009) measure in that both measures increase during periods when the load is above average. During periods when the load is below average, the Kc and Terwiesch (2009) measure decreases gradually, but our measure $J(t)$ is reset to zero

4.4 Model Formulation

In our generalization of the Erlang C model, users arrive according to a Poisson process with rate λ and wait in an infinite capacity first-come-first-served queue for the first available of s parallel and identical servers. Servers are never idle when customers are waiting. The rate $\mu_{i,j}$ at which every busy server completes its current service depends on the state variables $I(t)$ (number of users in the system) and $J(t)$ (defined in (4.1)). The overwork measure $J(t)$ increases by one with every service completion when $I(t) \geq k$ and $J(t)$ is reset to zero when $I(t)$ falls below k. The resulting Model M_1 is a CTMC with two infinite-range state variables and state space $\Omega_1 = \{(i,j) : i = 0, \cdots, k-1; j = 0\} \cup \{(i,j) : i =$

$k, k+1, \cdots; j = 0, 1, \cdots\}$. In Section 4.5, we transform Model M_1 into Model M_2, in which $J(t)$ has finite range. We do not specify service time distributions but the assumption that the holding time in each system state is exponential with a state-dependent mean implies that service time distributions are phase-type.

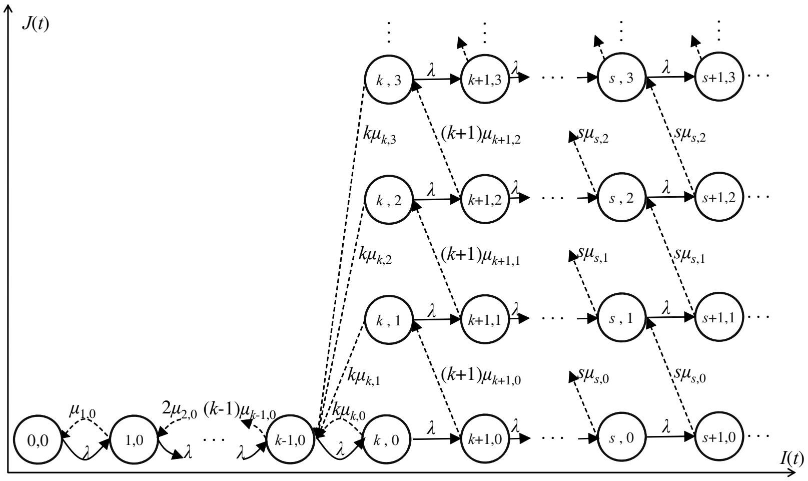


Figure 4.1: State transition diagram for Model M_1

We use $B(t) = \min\{I(t), s\}$ to denote the number of busy servers at time t and we represent particular values of $B(t)$, $I(t)$, and $J(t)$ by b, i, and j. We measure load as the ratio b/s. Figure 4.1 shows a transition rate diagram for Model M_1. The four transitions types are:

- User arrival, with rate λ, resulting in a transition to state $(i+1, j)$.
- User departure from state $(i, 0)$, where $1 \leq i \leq k-1$, with rate $\mu_{i,0}$, resulting in a transition to state $(i-1, 0)$.
- User departure from state (k, j) that terminates a k-partial busy period, with rate $k\mu_{k,j}$, resulting in a transition to state $(k-1, 0)$.
- User departure from state (i, j), where $i \geq k+1$, that does not terminate a k-partial busy period, at rate $b\mu_{i,j}$, resulting in a transition to $(i-1, j+1)$.

If all service rates are equal ($\mu_{ij} = \mu, \forall (i,j) \in \Omega$), then the marginal steady-state distribution of I matches the system size distribution of the Erlang C model. When $k = s = 1$, the marginal distribution for J can be converted to the distribution of the number of users served in a busy period, Y,

$$\Pr(Y = y) = \frac{\pi_{1,y-1}}{\sum_j \pi_{1,j}}, \qquad y \geq 1, \tag{4.2}$$

where $\pi_{i,j}$ is the steady-state probability of state (i, j). We compared the distribution for Y to the closed-form results obtained by Takacs (1955):

$$\Pr(Y = y) = \frac{1}{y}\binom{2y-2}{y-1}\rho^{y-1}(1+\rho)^{-2y+1}, \qquad \rho = \lambda/\mu, \quad y \geq 1. \tag{4.3}$$

In the remainder of the chapter, we make three assumptions about the service rates.

Assumption 4.1. *Service rates are independent of i for $i \geq s$: $\mu_{s+l,j} = \mu_{s,j}$ for $j = 0, 1, \cdots$ and $l = 1, 2, \cdots$.*

This assumption reflects our interpretation of b/s as measuring the system load, which implies that the load effect on service rates saturates when i reaches s. It also implies that the overwork effect on service rates does not depend on the number of users in the system, if all servers are busy. It is possible to relax this assumption to $\mu_{s',j} = \mu_{s'+l,j}$, for $j = 0, 1, \cdots$ and $l = 1, 2, \cdots$, where s' is finite and larger than s.

Assumption 4.2. *Service rates are independent of j for j large enough: There exists an $m \geq 1$ such that $\mu_{i,m+l} = \mu_{i,m}$ for $i = k, k+1, \cdots$ and $l = 1, 2, \cdots$.*

Typically, we expect service rates to decrease with j and if they do, then $\mu_{i,m}$ represents the smallest possible service rate for $I = i$. Even if Assumption 4.2 is judged not to be realistic, if one selects the threshold m to be sufficiently large, then the probability that J is larger than m will be negligible.

Under Assumptions 4.1 and 4.2, M_1 can be transformed into a level-dependent QBD process that we refer to as Model M_2, as we will discuss in Section 4.5.

Assumption 4.3. *Model M_2 is stable: $\mu_{s,j} > 0$, $j = 0, \cdots, m-1$, and $s\mu_{s,m} > \lambda$.*

In Section 4.5.3, we show that the system is stable if and only if Assumption 4.3 holds.

4.5 Model Analysis

We begin by showing, in Theorem 4.4, that under Assumption 4.2, Model M_1 is equivalent to Model M_2, which has a finite range $\{0, 1, \cdots, m\}$ for J. We establish that M_2 is a QBD process, whose special structure, together with general matrix-geometric results, allows tractable calculation of steady-state probabilities and other important system performance measures.

Model M_2 is a CTMC with state variables $(I(t), J(t))$, state space $\Omega_2 = \Omega_1 \cap \{j \leq m\}$, and death rates $b\mu_{i,0}$ for $0 \leq i \leq k-1$ and $b\mu_{i,j}$ for $i \geq k$ and $j \leq m$. Figure 4.2 presents transition rate diagrams for M_1 and M_2.

Theorem 4.4. *M_1 is equivalent to M_2 in the sense that the steady state probabilities $\pi_{i,j}$, for $i \geq 0$ and $0 \leq j \leq m-1$, and the steady state marginal distributions of I are equal in the two models.*

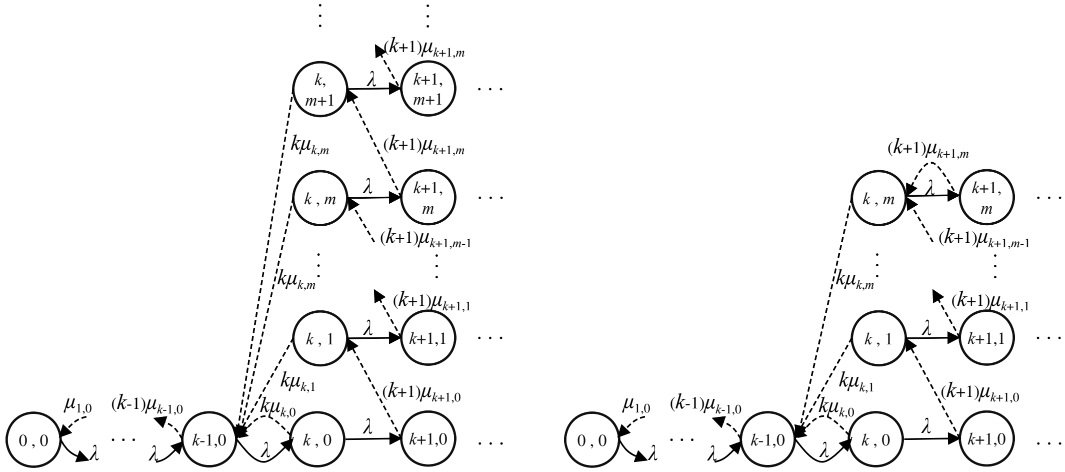


Figure 4.2: Equivalent Markov chains: Model M_1 (left) and Model M_2 (right), for the case $k < s$

Proof. Proof. See Appendix A.1.

In the special case $s = k = m = 1$, M_2 is an $M/M/1$ queue in which the first user in a busy period receives a service with a special service rate. Welch (1964) and Medhi (1996) study this system. We obtained a closed-form solutions for the joint probabilities of this system by solving the balance equations for M_2 and from this solution, we obtained the same distribution for I as Welch (1964) and Medhi (1996). However, obtaining a closed-form solution to the the balance equations for systems with arbitrary s and k and larger m appears to be difficult.

We formulate M_2 as a level-dependent QBD process with I as the level and J as the phase, that is, we order the system states lexicographically as $\{(0,0),\cdots,(k-1,0),(k,0),\cdots,(k,m),\cdots,(s,0),\cdot$ We define the subset $L(i)=\Omega_2\cap\{I=i\}$ of states in level i. Model M_2 is a QBD because it is skip-free to the left and right. The transition matrix is block tridiagonal:

$$\begin{pmatrix}
\mathbf{B}_1 & \mathbf{B}_0 & & & & & & \\
\mathbf{B}_2 & \mathbf{A}_1^{(k)} & \mathbf{A}_0 & & & & & \\
 & \mathbf{A}_2^{(k+1)} & \mathbf{A}_1^{(k+1)} & \mathbf{A}_0 & & & & \\
 & & \ddots & \ddots & \ddots & & & \\
 & & & \mathbf{A}_2^{(s)} & \mathbf{A}_1^{(s)} & \mathbf{A}_0 & & \\
 & & & & \mathbf{A}_2^{(s)} & \mathbf{A}_1^{(s)} & \mathbf{A}_0 & \\
 & & & & & \ddots & \ddots & \ddots
\end{pmatrix},$$

where $\mathbf{A}_0$, $\mathbf{A}_1^{(i)}$ ($\mathbf{A}_1^{(s+l)}=\mathbf{A}_1^{(s)}$, $\forall l\geq 0$), and $\mathbf{A}_2^{(i)}$ ($\mathbf{A}_2^{(s+l)}=\mathbf{A}_2^{(s)}$, $\forall l\geq 0$) are square matrices of order $m+1$, given as

$$\mathbf{A}_0=\begin{pmatrix}\lambda & & \\ & \ddots & \\ & & \lambda\end{pmatrix},\mathbf{A}_1^{(i)}=\begin{pmatrix}-\lambda-b\mu_{i,0} & & \\ & \ddots & \\ & & -\lambda-b\mu_{i,m}\end{pmatrix},\mathbf{A}_2^{(i)}=\begin{pmatrix}0 & b\mu_{i,0} & & \\ & \ddots & \ddots & \\ & & 0 & b\mu_{i,m-1} \\ & & & b\mu_{i,m}\end{pmatrix},$$

and the boundary blocks $\mathbf{B}_1$, $\mathbf{B}_0$, and $\mathbf{B}_2$ are

$$\mathbf{B}_1 = \begin{pmatrix} -\lambda & \lambda & & & \\ \mu_{1,0} & -\lambda - \mu_{1,0} & \lambda & & \\ & \ddots & \ddots & \ddots & \\ & & (k-2)\mu_{k-2,0} & -\lambda - (k-2)\mu_{k-2,0} & \lambda \\ & & & (k-1)\mu_{k-1,0} & -\lambda - (k-1)\mu_{k-1,0} \end{pmatrix}_{k\times k},$$

$$\mathbf{B}_0 = \begin{pmatrix} 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 \\ \lambda & 0 & \cdots & 0 \end{pmatrix}_{k\times(m+1)}, \text{and } \mathbf{B}_2 = \begin{pmatrix} 0 & \cdots & 0 & k\mu_{k,0} \\ \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & 0 & k\mu_{k,m-1} \\ 0 & \cdots & 0 & k\mu_{k,m} \end{pmatrix}_{(m+1)\times k}.$$

Given that $\mathbf{A}_1^{(s+l)} = \mathbf{A}_1^{(s)}$ and $\mathbf{A}_2^{(s+l)} = \mathbf{A}_2^{(s)}$, $l = 1, 2, \cdots$, we can also formulate M_2 as a level-independent QBD with larger boundary blocks by considering all states from $(0,0)$ to $(s-1,m)$ to be boundary states. Although the level-independent QBD representation is more compact, the level-dependent one is more computationally efficient and it is structurally similar to the fixed-service-rate Erlang C model, as we illustrate later in this section.

Let $\pi_{i,0}$, $0 \leq i \leq k-1$, denote the steady-state probabilities of the boundary states (one-dimensional section of Model M_2) and the row vector $\boldsymbol{\pi}_i = \{\pi_{i,0}, \pi_{i,1}, \cdots, \pi_{i,m}\}$, $i \geq k-1$, denote the steady state probabilities of the states in $L(i)$. The steady state probabilities of M_2 satisfy

$$\sum_{i=0}^{k-1} \pi_{i,0} + \sum_{i=k}^{\infty} \boldsymbol{\pi}_i \mathbf{1} = 1, \tag{4.4}$$

$$\pi_{i+1,0} = \pi_{i,0} r_i, \qquad 0 \le i \le k-2, \tag{4.5}$$

$$\boldsymbol{\pi}_{k-1} = \pi_{k-1,0} \mathbf{e}_1, \tag{4.6}$$

$$\boldsymbol{\pi}_{i+1} = \boldsymbol{\pi}_i \mathbf{R}^{(i)}, \qquad i \ge k-1, \tag{4.7}$$

where $\mathbf{1}$ is a vector of ones, $\mathbf{e}_1$ is the first unit row vector, and scalars

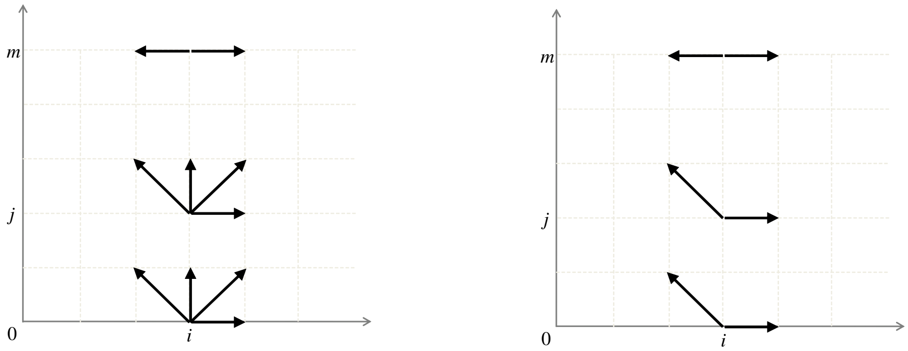
$$r_i = \frac{\lambda}{(i+1)\mu_{i+1,0}}, \qquad 0 \le i \le k-2, \tag{4.8}$$

result directly from the balance equations for the states in the one-dimensional section of the state space. The rate matrix for $L(i)$, $\mathbf{R}^{(i)}$, $i \ge k-1$, records the expected number of visits to $L(i+1)$ between two consecutive visits to $L(i)$ (Latouche and Ramaswami 1999). We discuss the computation of the rate matrices in the next two subsections.

4.5.1 Computing the Rate Matrices

In a level-independent QBD, the rate matrix $\mathbf{R}$ is the minimal nonnegative solution to $\mathbf{R}^2\mathbf{A}_2 + \mathbf{R}\mathbf{A}_1 + \mathbf{A}_0 = 0$ (Neuts 1981) and is computed numerically in most cases. van Leeuwaarden and Winands (2006) introduce a class of level-independent QBD processes with transitions limited to the ones shown in Figure 4.3a, for which $\mathbf{R}$ is explicitly obtainable by counting lattice paths between two particular states multiplied by the constant path probability. The property that facilitates the explicit representation of $\mathbf{R}$ is that if the QBD process leaves a state in level $L(i)$, it returns back to $L(i)$ in a finite number of transitions. The set of transitions in M_2 is a subset of the transitions that van Leeuwaarden and Winands (2006) allow (compare Figures 4.3a and 4.3b). However, in contrast to van Leeuwaarden and Winands (2006), we allow the transition rates to change between and within levels in M_2.

We modify the van Leeuwaarden and Winands (2006) approach to obtain rate matrices



(a) van Leeuwaarden and Winands (2006) model

(b) Model M_2

Figure 4.3: Allowed transitions

$\mathbf{R}^{(i)}, i = k, k+1, \cdots$, for M_2. van Leeuwaarden and Winands (2006) define an excursion as the time interval from the moment that the QBD leaves a particular level i to the moment that it returns to level i for the first time. The rate matrices $\mathbf{R}^{(i)}$ are upper-triangular (because returning to a lower phase is prohibited in M_2, when $i \geq k$) and can be expressed as

$$\mathbf{R}^{(i)} = \begin{pmatrix} R^{(i)}_{0,0} & \cdots & R^{(i)}_{0,m} \\ & \ddots & \vdots \\ & & R^{(i)}_{m,m} \end{pmatrix},$$

where $R^{(i)}_{j,h}$ = expected time spent in state $(i+1, h)$, $h \geq j$, during an excursion with the initial state (i, j), expressed in the expected sojourn time in state (i, j). We extend Property 3.1 in van Leeuwaarden and Winands (2006) to capture the level-dependency of the rate matrices and restate the property as follows:

Property 1. *For an excursion starting from state (i, j), elements $R^{(i)}_{j,h}$ can be decomposed as*

$$R^{(i)}_{j,h} = q^{(i)}_{j,h} E(X^{(i)}_h) \frac{[\mathbf{A}^{(i)}_1]_{j,j}}{[\mathbf{A}^{(i+1)}_1]_{h,h}}, \quad i \geq k, 0 \leq j \leq m, j \leq h \leq m, \tag{4.9}$$

where $q^{(i)}_{j,h}$ is the combined probability of all paths from (i, j) to $(i+1, h)$, $E(X^{(i)}_h)$ is the expected number of visits to state $(i+1, h)$ given that the excursion is in state $(i+1, h)$ for the first time, and $[\mathbf{A}^{(i)}_1]_{j,j}/[\mathbf{A}^{(i+1)}_1]_{h,h} = (\lambda + b\mu_{i,j})/(\lambda + b\mu_{i+1,h})$ is the ratio of the *expected time* $1/|[\mathbf{A}_1^{(i+1)}]_{h,h}|$ *spent in state* $(i+1,h)$ *to the expected time* $1/|[\mathbf{A}_1^{(i)}]_{j,j}|$ *spent in state* (i,j).

Property 1 shows that the task of computing the matrix elements $R_{j,h}^{(i)}$ decomposes into computing the probabilities $q_{j,h}^{(i)}$ and the expected values $E(X_h^{(i)})$. We discuss how to compute these quantities efficiently in the remainder of this subsection.

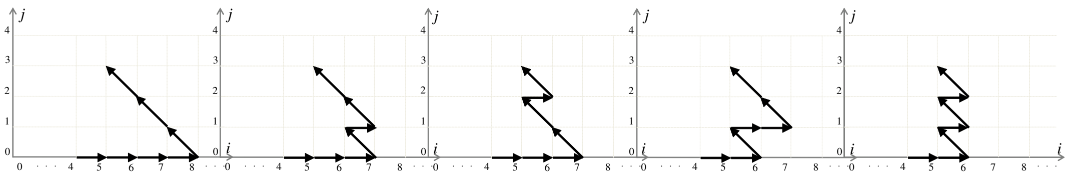
4.5.1.1 Computing $q_{j,h}^{(i)}$:

Let the upper-triangular matrix $\mathbf{q}^{(i)}$ contain elements $q_{j,h}^{(i)}$ for an excursion from level $L(i)$ to $L(i+1)$. We explain how to compute the $q_{j,h}^{(i)}$ probabilities, first for $h < m$ and second, for $h = m$. To calculate $q_{j,h}^{(i)}$ for $h < m$, one needs to add the probabilities of all paths from (i,j) to $(i+1,h)$. In the special case of $s = 1$ and constant service rates where all paths have equal probabilities, van Leeuwaarden and Winands (2006, Theorem 3.1) provide a closed-form solution for $q_{j,h}$ obtained by multiplying the number of paths with the fixed path probability. The state-dependent service rates in M_2, however, result in unequal path probabilities.

As an example, Figure 4.4 shows the five possible paths from state $(4,0)$ to state $(5,3)$. All paths include 4 arrivals and 3 service completions. Denote the probability that the next transition from state (i,j) is an arrival or is a service completion by $\phi_{i,j} = \lambda/(\lambda + b\mu_{i,j})$ and $\psi_{i,j} = b\mu_{i,j}/(\lambda + b\mu_{i,j})$, respectively. When $s = 1$ and the service rates are fixed, the arrival and service completion probabilities are constants $\phi = \lambda/(\lambda+\mu)$ and $\psi = \mu/(\lambda+\mu)$, resulting in $q_{0,3} = 5\phi^4\psi^3$. When service rates are state-dependent, however, each path has a different probability, resulting in

$$\begin{aligned} q_{0,3}^{(4)} = \phi_{4,0}\phi_{5,0}\psi_{6,2}\,(&\phi_{6,0}\phi_{7,0}\psi_{8,0}\psi_{7,1} + \phi_{6,0}\psi_{7,0}\phi_{6,1}\psi_{7,1} + \\ &\phi_{6,0}\psi_{7,0}\psi_{6,1}\phi_{5,2} + \psi_{6,0}\phi_{5,1}\phi_{6,1}\psi_{7,1} + \psi_{6,0}\phi_{5,1}\psi_{6,1}\phi_{5,2})\,. \end{aligned}$$

We interpret $q_{j,h}^{(i)}$ as the probability that an excursion starting from state (i,j) is absorbed in state $(i+1,h)$, by viewing state $(i+1,h)$ as absorbing and all other states that are



(a) (b) (c) (d) (e)

Figure 4.4: Paths from state $(4, 0)$ to state $(5, 3)$

possibly visited in the excursion as transient states. Viewed this way, $q^{(i)}_{j,h}$ is the solution to a set of linear absorption probability equations. Continuing with the Figure 4.4 example, the probability $q^{(4)}_{0,3}$ of reaching state $(5, 3)$ starting from state $(4, 0)$ is the solution to the linear equation set,

$$\begin{aligned}
q^{(4)}_{0,3} &= \phi_{4,0}\delta^{5,3}_{5,0}, \\
\delta^{5,3}_{5,0} &= \phi_{5,0}\delta^{5,3}_{6,0}, \\
\delta^{5,3}_{6,0} &= \phi_{6,0}\delta^{5,3}_{7,0} + \psi_{6,0}\delta^{5,3}_{5,1}, \\
\delta^{5,3}_{7,0} &= \phi_{7,0}\delta^{5,3}_{8,0} + \psi_{7,0}\delta^{5,3}_{6,1}, \\
\delta^{5,3}_{8,0} &= \psi_{8,0}\delta^{5,3}_{7,1},
\end{aligned}$$

$$\begin{aligned}
\delta^{5,3}_{5,1} &= \phi_{5,1}\delta^{5,3}_{6,1}, \\
\delta^{5,3}_{6,1} &= \phi_{6,1}\delta^{5,3}_{7,1} + \psi_{6,1}\delta^{5,3}_{5,2}, \\
\delta^{5,3}_{7,1} &= \psi_{7,1}\delta^{5,3}_{6,2}, \\
\delta^{5,3}_{5,2} &= \phi_{5,2}\delta^{5,3}_{6,2}, \\
\delta^{5,3}_{6,2} &= \psi_{6,2}\delta^{5,3}_{5,3},
\end{aligned}$$

where the variable $\delta^{5,3}_{a,b}$, $a = 5, \cdots, 8$ and $b = 0, \cdots, 8 - i$, is the probability of reaching state $(5, 3)$, starting from any state (a, b) that is on a path between $(4, 0)$ and $(5, 3)$. The above equation set can be solved recursively by using the initial condition $\delta^{5,3}_{5,3} = 1$.

This approach can be used to calculate the elements $q^{(i)}_{j,h}, h < m$ one at a time, but it is more efficient to calculate all entries in a column of $\mathbf{q}^{(i)}$ simultaneously. If we label columns of matrix $\mathbf{q}^{(i)}$ from 0 to m, the elements in column h $(q^{(i)}_{0,h}, \cdots, q^{(i)}_{h,h})$, $h < m$, correspond to the probabilities of absorption in state $(i + 1, h)$, starting from state (i, j), $j = 0, \cdots, h$, as illustrated in Figure 4.5. Note that $q^{(i)}_{h,h} = \phi_{i,h}, \forall h$. The general equations to compute column h, $h = 0, \cdots, m - 1$, are:

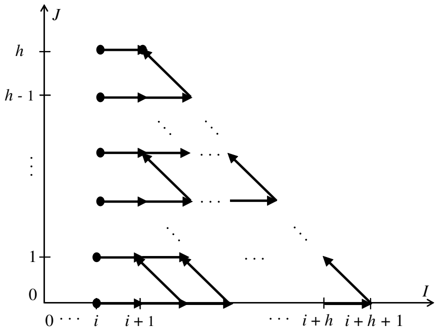


Figure 4.5: Paths from state $(i, j), 0 \leq j \leq h$, to $(i + 1, h)$

$$\begin{cases} q^{(i)}_{h,h} = \phi_{i,h}, & \\ q^{(i)}_{j,h} = \phi_{i,j}\delta^{i+1,h}_{i+1,j}, & j = 0, \cdots, h-1, \\ \delta^{i+1,h}_{i+1,b} = \phi_{i+1,b}\delta^{i+1,h}_{i+2,b}, & b = 0, \cdots, h-1, \\ \delta^{i+1,h}_{a,b} = \psi_{a,b}\delta^{i+1,h}_{a-1,b+1}, & a = i+2, \cdots, i+h+1, \quad b = i+h+1-a, \\ \delta^{i+1,h}_{a,b} = \phi_{a,b}\delta^{i+1,h}_{a+1,b} + \psi_{a,b}\delta^{i+1,h}_{a-1,b+1}, & a = i+2, \cdots, i+h, \qquad b = 0, \cdots, i+h-a. \end{cases} \tag{4.10}$$

These equations can be solved recursively, starting with the initial condition $\delta^{i+1,h}_{i+1,h} = 1$.

We cannot use equation sets (A.11) and (4.10) for column m of $\mathbf{q}^{(i)}$ $(q^{(i)}_{0,m}, \cdots, q^{(i)}_{m,m})$, because there are infinitely many paths from state (i, j), $j \leq m$, to state $(i + 1, m)$ due to the possible left transitions from states in phase m. To calculate $q^{(i)}_{j,m}$, $j < m$, one can subtract the probability that a path enters $L(i + 1)$ but returns to $L(i)$ before visiting state $(i + 1, m)$ from the probability of a move from $L(i)$ to level $L(i + 1)$, that is:

$$\begin{cases} q^{(i)}_{j,m} = \phi_{i,j} - \sum_{l=j}^{m-1} \psi_{i+1,l} q^{(i)}_{j,l}, \quad j < m, \\ q^{(i)}_{m,m} = \phi_{m,m}. \end{cases} \tag{4.11}$$

4.5.1.2 Computing $E(X^{(i)}_h)$:

If $h < m$, an excursion from (i, j) to $(i+1, h)$ visits $(i+1, h)$ only once. On the boundary, however, given that $(i + 1, m)$ is visited once, the total number of visits to that state is geometrically distributed, with parameter $\psi_{i+1,m}$ (the probability that the next transition is an arrival). Therefore,

$$E(X_h^{(i)}) = \begin{cases} 1, & h < m, \\ 1/\psi_{i+1,m}, & h = m. \end{cases} \tag{4.12}$$

Substituting (4.12) in (4.9) results in the following expression for the non-zero entries in the rate matrix $\mathbf{R}^{(i)}$,

$$R_{j,h}^{(i)} = \begin{cases} q_{j,h}^{(i)} \dfrac{\phi_{i+1,h}}{\phi_{i,j}}, & j = 0, \cdots, m, h = j, \cdots, m-1, \\ q_{j,m}^{(i)} \dfrac{\phi_{i+1,m}}{\phi_{i,j}\psi_{i+1,m}}, & j = 0, \cdots, m, h = m, \end{cases} \tag{4.13}$$

where $q_{j,h}^{(i)}$, $h < m$, and $q_{j,m}^{(i)}$ are obtained from (4.10) and (4.11), respectively.

Even though the transitions between levels $k-1$ and k have a different structure than the transitions between $i-1$ and i where $i > k$ (Figure 4.3), one can verify that the method for computing $\mathbf{R}^{(i)}, i = k, k+1, \cdots$, applies for $\mathbf{R}^{(k-1)}$ as well, as we show in Appendix A.2.

4.5.2 Computational Complexity of Computing $\mathbf{R}^{(i)}$

We first derive the computational complexity of computing matrix $\mathbf{q}^{(i)}$ by solving equation set (A.11) or (4.10). Each equation in (4.10) corresponds to a state in a right triangle, with h states on the vertical leg, h states on the hypotenuse, and a total of $h(h+3)/2$ states. Solving the equation for each of the $2h$ states on the vertical leg or on the hypotenuse requires one arithmetic operation and solving the equation for each of the other $h(h-1)/2$ states requires three operations, for a total of $(3h^2 + 3h + 2)/2$ operations to compute the entries in the h-th column of $\mathbf{q}^{(i)}$, for $h < m$. Computing the m-th column using (4.11) requires $m(m+1)$ operations. In total, the number of operations needed to compute all non-zero entries in the matrix $\mathbf{q}^{(i)}$ is

$$m(m+1) + \sum_{h=1}^{m-1} \frac{3h^2 + 3h + 2}{2} = \frac{m^3 + 2m^2 + 3m - 2}{2}.$$

Once we have computed $\mathbf{q}^{(i)}$, we perform 2 operations if $h < m$ and 3 operations if $h = m$ in (4.13) to compute each nonzero entry of the rate matrix $\mathbf{R}^{(i)}$. The total number of operations needed to compute $\mathbf{R}^{(i)}$ is $(m^3 + 4m^2 + 11m + 4)/2$.

For comparison, Van Houdt and van Leeuwaarden (2011) developed an algorithm with computational complexity $2m^3$ to compute matrix $\mathbf{G}$, which, in turn, is used to compute matrix $\mathbf{R} = \mathbf{A}_0 \left(\mathbf{I} - \mathbf{A}_1 - \mathbf{A}_0\mathbf{G}\right)^{-1}$, for $M/G/1$-type Markov chains with triangular $\mathbf{A}_0$, $\mathbf{A}_1$, and $\mathbf{A}_2$ matrices. Finding the main diagonal of $\mathbf{G}$ requires an iterative algorithm that converges quadratically. The Van Houdt and van Leeuwaarden (2011) algorithm appears to be the most efficient published algorithm that could be used (with some modifications to accommodate state-dependent rates) for our model, but our algorithm is more efficient and easier to implement.

4.5.3 Stability Condition

In Theorem 4.5, we prove that Assumption 4.3 expresses the stability condition of our model.

Theorem 4.5. *Model M_2 is stable if and only if (1) $\mu_{s,j} > 0$, $j = 0, \cdots, m-1$, and (2) $s\mu_{s,m} > \lambda$.*

Proof. Proof. As mentioned in Section 4.5, one can view Model M_2 as a level-independent QBD with rate matrix $\mathbf{R}^{(s)}$ by considering all states from $(0, 0)$ to $(s-1, m)$ to be boundary states. Therefore, Model M_2 is stable if and only if the following ergodicity condition for level-independent QBDs is satisfied (Latouche and Ramaswami 1999, Theorem 7.2.4):

$$\boldsymbol{\nu}\mathbf{A}_0^{(s)}\mathbf{1} < \boldsymbol{\nu}\mathbf{A}_2^{(s)}\mathbf{1}, \tag{4.14}$$

where the row vector $\boldsymbol{\nu} = (\nu_0, \cdots, \nu_m)$ contains the steady state probabilities of the Markov chain that corresponds to the generator matrix $\mathbf{A}^{(s)} = \mathbf{A}_0^{(s)} + \mathbf{A}_1^{(s)} + \mathbf{A}_2^{(s)}$ and is the unique solution to

$$\boldsymbol{\nu}\mathbf{A}^{(s)} = \mathbf{0}, \qquad \sum_{i=0}^{m} \nu_i = 1. \tag{4.15}$$

The Markov chain corresponding to $\mathbf{A}^{(s)}$ is a pure birth process with birth rates $s\mu_{s,0}, \ldots, s\mu_{s,m-1}$, which provides the unique solution $\boldsymbol{\nu} = (0, \cdots, 0, 1)$ in (4.15) if $\mu_{s,j} > 0, j = 0, \cdots, m-1$. If $\mu_{s,j} = 0$, for some $j = 0, \cdots, m-1$, then (4.15) does not have a unique solution. When we substitute $\nu = (0, \cdots, 0, 1)$ in (4.14), we obtain $\lambda < s\mu_{s,m}$.□ □

Note that if the service rates are decreasing in j, then $\lambda < s\mu_{s,m}$ is sufficient for stability.

4.5.4 Performance Measures

Using equations (4.5) and (4.7), we can express the steady state probabilities of the following three state-space regions in terms of the rate matrices and $\pi_{0,0}$:

$$\text{Only load effect: } \Pr(i \leq k-1) = \pi_{0,0}\left(1 + \sum_{i=1}^{k-1}\prod_{j=0}^{i-1} r_j\right), \tag{4.16}$$

$$\text{Both load and overwork effects: } \Pr(k \leq i < s) = \boldsymbol{\pi}_{k-1}\sum_{i=1}^{s-k}\prod_{j=k-1}^{k+i-2}\mathbf{R}^{(j)}\mathbf{1}, \tag{4.17}$$

$$\text{Only overwork effect: } \Pr(i \geq s) = \boldsymbol{\pi}_s \sum_{i=0}^{\infty} \mathbf{R}^{(s)^i} = \boldsymbol{\pi}_s\left(\mathbf{I} - \mathbf{R}^{(s)}\right)^{-1}\mathbf{1}, \tag{4.18}$$

where

$$\boldsymbol{\pi}_{k-1} = \pi_{0,0}\prod_{i=0}^{k-2} r_i \mathbf{e}_1, \text{ and } \boldsymbol{\pi}_s = \boldsymbol{\pi}_{k-1}\prod_{i=k-1}^{s-1}\mathbf{R}^{(i)}. \tag{4.19}$$

To fully characterize the steady state probabilities, we need $\pi_{0,0}$, which we obtain as follows, knowing that (4.16), (4.17), and (4.18) add up to one.

$$\pi_{0,0} = \left[1 + \sum_{i=1}^{k-1}\prod_{j=0}^{i-1} r_j + \prod_{i=0}^{k-2} r_i \mathbf{e}_1 \left(\sum_{i=1}^{s-k}\prod_{j=k-1}^{k+i-2}\mathbf{R}^{(j)} + \prod_{i=k-1}^{s-1}\mathbf{R}^{(i)}\left(\mathbf{I} - \mathbf{R}^{(s)}\right)^{-1}\right)\mathbf{1}\right]^{-1}. \tag{4.20}$$

The Erlang C model is structurally similar to Model M_2, with the rate matrix $\mathbf{R}^{(s)}$ in M_2 playing a similar role to the utilization ρ in the Erlang C model. Table 4.1 illustrates some

Table 4.1: Structural similarities between the Erlang C and M_2 models.

Erlang C

$\pi_{i+1} = \pi_i r_i,\ i = 0, \cdots, k-1$

$\pi_{i+1} = \pi_i r_i,\ i = k, \cdots, s-1$

$\pi_{i+1} = \pi_i \rho,\ i \geq s$

$\pi_0 = \left(1 + \sum_{i=1}^{s-1} \dfrac{r_0^i}{i!} + \dfrac{r_0^s/s!}{1-\rho}\right)^{-1}$

$\Pr(I \geq s) = \Pr(\text{Delay}) = 1 - \pi_0 \sum_{i=0}^{s-2} \dfrac{r_0^i}{i!}$

$W_q = \frac{1}{\lambda}\left(\pi_0 \dfrac{r_0^s}{s!} \dfrac{\rho}{(1-\rho)^2}\right)$

Model M_2

$\pi_{i+1,0} = \pi_{i,0} r_{i,0},\ i = 0, \cdots, k-1$

$\boldsymbol{\pi}_{i+1} = \boldsymbol{\pi}_i \mathbf{R}^{(i)},\ i = k, \cdots, s-1$

$\boldsymbol{\pi}_{i+1} = \boldsymbol{\pi}_i \mathbf{R}^{(s)},\ i \geq s$

$\pi_{0,0} = \left[1 + \sum_{i=1}^{k-1} \prod_{j=0}^{i-1} r_j + \prod_{i=0}^{k-2} r_i \mathbf{e}_1 \left(\sum_{i=1}^{s-k} \prod_{j=k-1}^{k+i-2} \mathbf{R}^{(j)} + \prod_{i=k-1}^{s-1} \mathbf{R}^{(i)} \left(\mathbf{I} - \mathbf{R}^{(s)}\right)^{-1}\right) \mathbf{1}\right]^{-1}$

$\Pr(I \geq s) = \Pr(\text{Delay}) = 1 - \pi_{0,0} \left[1 + \sum_{i=1}^{k-1} \prod_{j=0}^{i-1} r_j + \prod_{i=0}^{k-2} r_i \mathbf{e}_1 \left(\sum_{i=1}^{s-k} \prod_{j=k-1}^{k+i-2} \mathbf{R}^{(j)}\right) \mathbf{1}\right]$

$W_q = \frac{1}{\lambda} \pi_{0,0} \prod_{i=0}^{k-1} r_i \mathbf{e}_1 \prod_{i=k}^{s} \mathbf{R}^{(i)} \left(\mathbf{R}^{(s)} \left(\mathbf{I} - \mathbf{R}^{(\mathbf{s})}\right)^{-2}\right) \mathbf{1}$

of these similarities. In Model M_2, the average queue length L_q is obtained as follows,

$$L_q = \sum_{i=s}^{\infty} (i-s) \boldsymbol{\pi}_i \mathbf{1} = \boldsymbol{\pi}_s \mathbf{R}^{(\mathbf{s})} \left(\mathbf{I} - \mathbf{R}^{(s)}\right)^{-2} \mathbf{1}. \tag{4.21}$$

Using Little's Law and (4.21) results in the expression for the expected queue delay W_q in Table 4.1.

The virtual waiting time for Model M_2 corresponds to the time to absorption in a modified version of the model, where all arrival transitions are removed, all states with one or more free servers are aggregated into a single absorbing state, and the probability distribution for the initial state is the steady-state distribution for M_2, conditional on all servers being busy. The details of this standard approach are discussed, for example, in Ramaswami and Lucantoni (1985). We provide pseudo code for computing the virtual waiting time distribution for M_2 in Appendix A.3.

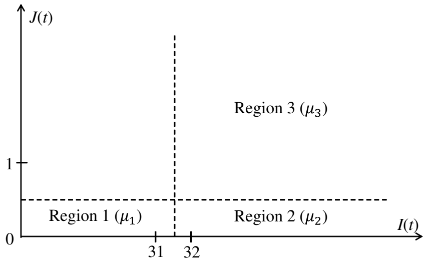


Figure 4.6: Three-service rate system

4.6 Effects of Ignoring Load and Overwork

In this section, we illustrate the magnitude of the errors that can result from using a fixed-service-rate Erlang C model, instead of our state-dependent Model M_2, to predict the performance of a system with load and overwork effects. Model M_2 is general in that it can accommodate any functional dependence of service rates on load and overwork as long as Assumptions 1-3 are satisfied. In this and the next section, however, we focus on a simplified situation with only three different service rates, in order to develop insights and for ease of exposition. Our base case is a 35-server system in which load and overwork begin to impact the service rates when 90% or more of the servers are busy ($k = \lceil 0.9s \rceil = 32$). We specify the service rates per hour as follows:

$$\mu_{i,j} = \begin{cases} \mu_1 = 0.9 & \text{Region 1: } i < k \text{ (load} < 0.9\text{, overwork} = 0\text{),} \\ \mu_2 = 1 & \text{Region 2: } i \geq k, j = 0 \text{ (load} > 0.9\text{, overwork} = 0\text{),} \\ \mu_3 = 0.75 & \text{Region 3: } i \geq k, j > 0 \text{ (load} > 0.9\text{, overwork} > 0\text{).} \end{cases} \tag{4.22}$$

In words, if we use "full speed" to refer to the service rate when at least 32 servers are busy but no service completions have occurred since the 32^{nd} server became busy, the servers slow down to 90% of full speed when fewer than 32 servers are busy and they slow down to 75% of full speed in the overwork region, which begins with the first service completion after the 32^{nd} server became busy and ends when a service completion leaves 31 busy servers. The system is stable if $\lambda < 35 \times \mu_3 = 26.25$.

One alternative to our state-dependent model for the above system is to use the standard

Erlang C model with a well-chosen "representative" service rate. The simplest representative service rate might be a service rate that corresponds to one of the three regions in the state space illustrated in Figure 4.6. Figures 4.7 and 4.8 compare the average delay (logarithmic scale) and the delay probability for Model M_2 with average delays and delay probabilities of three Erlang C models with rates fixed at μ_1, μ_2, and μ_3. The fixed-rate models with μ_1 and μ_2 underestimate the average delay and delay probability, while the fixed-rate model with μ_3 overestimates the two measures. All three fixed-rate models perform poorly for low and moderate arrival rates. The μ_3 fixed-rate model (the one corresponding to the "overwork region") is accurate for arrival rates close to the stability limit. When the arrival rate approaches the stability limit, the Model M_2 probability for the overwork region approaches 1 and the Model M_2 average delay diverges, as it does in the Erlang C model. None of the three fixed-rate models provide a good approximation over the entire arrival rate range. Even if none of three fixed-rate models that correspond to service rates μ_1, μ_2, and μ_3 are accurate, conceivably one can obtain accurate performance measure estimates by either using a weighted average $\bar{\mu}$ of the three service rates as input to a fixed-rate model or by using a weighted average of the outputs from the three fixed-rate models. Let $C(\lambda/\mu, s)$ and $D(\lambda/\mu, s)$ be the Erlang C probability of delay and average delay, respectively. Averaging the input means using $\bar{\mu} = \sum_{i,j} w_{i,j}\mu_{i,j}$, for some set of weights $w_{i,j}$, as the input to the Erlang C model. Averaging the output means estimating the probability of delay and the average delay as $\sum_{i,j} w_{i,j}C(\lambda/\mu_{i,j}, s)$ and $\sum_{i,j} w_{i,j}D(\lambda/\mu_{i,j}, s)$, respectively.

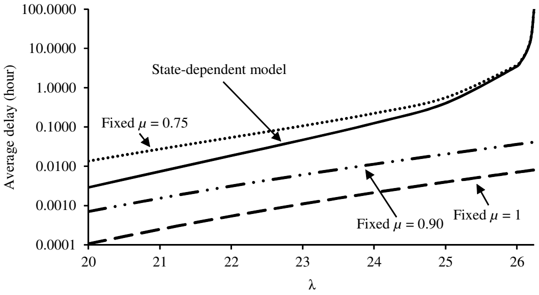


Figure 4.7: State-dependent vs. fixed models: average delay

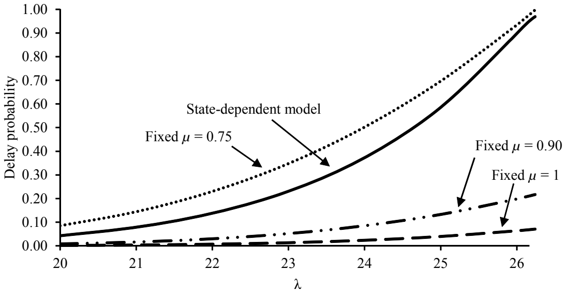


Figure 4.8: State-dependent vs. fixed models: delay probability

It is not clear how one should choose the weights for averaging the input or the output of the Erlang C model. We expect, however, that using the probabilities of the three state space regions shown in Figure 4.6 as weights (or, more generally, setting $w_{i,j} = \pi_{i,j}$) should result in greater accuracy (compared to the state-dependent model) than any set of weights that are determined without solving the state-dependent model. We follow this conservative approach in assessing the accuracy of both input averaging and output averaging.

Figure 4.9 shows that the probability mass shifts from the low-load Region 1 to the high-overwork Region 3 as the arrival rate increases from 20/hour to 26.25/hour (the stability limit), which results in the weighted average service rate $\bar{\mu}$ shifting from $\mu_1 = 0.9$ to $\mu_3 = 0.75$, as shown in Figure 4.10.

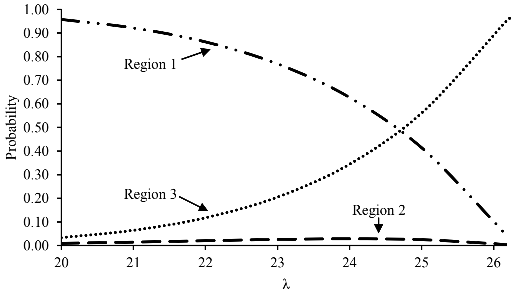


Figure 4.9: State probabilities in service rate regions

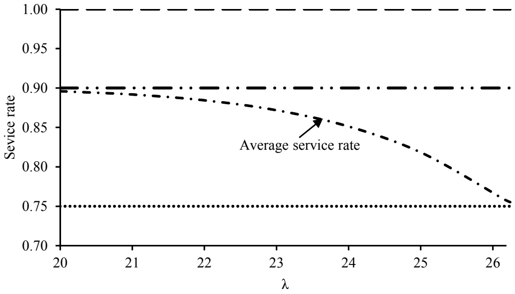


Figure 4.10: Weighted average service rate

Using a weighted average of the three fixed-rate models improves accuracy but considerable and systematic error remains, as we illustrate in Figures 4.11 and 4.12. The output-averaging approximation results in higher delay probability and higher average delay than the input-averaging approximation. This is not a coincidence, but a consequence of Jensen's inequality and convexity properties of $C(.)$ and $D(.)$ with respect to μ (Appendix A.4).

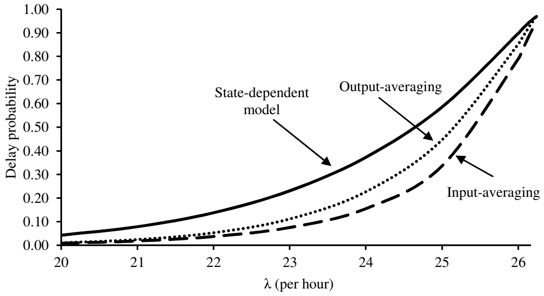


Figure 4.11: State-dependent vs. fixed models: delay probability

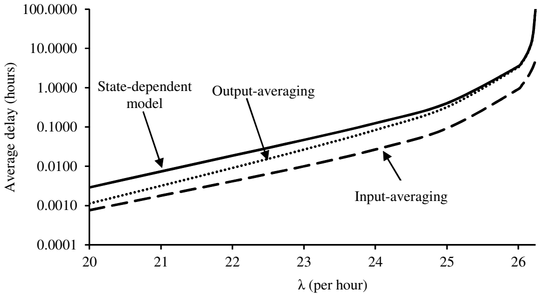


Figure 4.12: State-dependent vs. fixed models: average delay

4.7 Consequences of Using the "Wrong" Model

In Section 4.6, we studied the errors that can occur when one uses the Erlang C model to evaluate performance for a system where the service rates vary with load and overwork. In this section, we investigate the possible consequences of using the Erlang C model (the "wrong" model, if we view the state-dependent Model M_2 as representing reality) on an ongoing basis to set staffing levels in a service system with state-dependent rates. Perhaps the best that one could hope for is that ongoing monitoring of system performance leads to self-correcting behavior, even if one uses an incorrect model, and we find that this does indeed occur in some circumstances. We find, however, that using an incorrect model can also result in a variety of other less desirable system trajectories.

In order to be able to focus on the consequences of using the wrong model, we consider a system whose arrival rate and state-dependent service rates are time invariant. We use the following procedure to imitate staffing decisions in such a system:

1. Set arrival rate (λ), state-dependent service rates ($\mu_{i,j}$), initial staffing ($s = s_0$), initial period counter ($n = 1$), target service level (φ), and monitoring period length (T).

2. Simulate the system for period n and estimate the arrival rate ($\hat{\lambda}_n$) and the service rate ($\hat{\mu}_n$),

$$\hat{\lambda}_n = \frac{A_n}{T}, \qquad 1/\hat{\mu}_n = \frac{\sum_{i=1}^{S_n} X_i}{S_n}, \tag{4.23}$$

   where A_n and S_n are the number of arrivals and service completions in period n, and X_i is the service time of user i whose service finished in period n.

3. Use $\hat{\lambda}_n$ and $\hat{\mu}_n$ in the fixed-rate Erlang C model to find the minimum required staffing for the next period, s_{n+1}, to satisfy φ.

4. Set $n \to n + 1$ and return to Step 2.

The simulation model simulates the state-dependent system, which we assume represents reality. When a server begins serving a user, while in state (i, j), we simulate the service time as exponentially distributed with rate $\mu_{i,j}$. Whenever the system transitions from state (i, j) to (i', j'), we update the remaining service times of the users in service by generating new exponentially-distributed random variates with rate $\mu_{i',j'}$. If the number of servers increases from one period to the next one, then the service of the users in the queue at the end of the previous period, if any, is immediately initiated by the newly added servers. If the number of servers decreases, then the departure of a server, if busy, is postponed until the current service is completed.

In the experiments of this section, we vary the low-load service rate μ_1 in the base-case service-rate function (4.22) to illustrate a range of behaviors that result from the above staffing procedure. We set $\lambda = 20$ per hour and $\varphi = 0.90$ and define the service level as the probability that the virtual wait is less than or equal to 20 minutes. We begin by illustrating the impact of noisy estimation of λ and μ. Figure 4.13 shows simulated staffing for 8 periods, when $\mu_1 = 0.75$ per hour, under three monitoring periods: short ($T = 50$ hours with 1,000 expected arrivals), medium ($T = 250$ hours with 5,000 expected arrivals), and long ($T = 1000$ hours with 20,000 expected arrivals). The optimal staffing for this system, obtained from the state-dependent model, is $s = 31$. The staffing levels obtained from the Erlang C model converge to the optimal value after 4 periods when T is long. For the shorter monitoring periods, the staffing levels take longer to reach the optimal value and the staffing is not guaranteed to remain at the optimal value, because of the estimation noise.

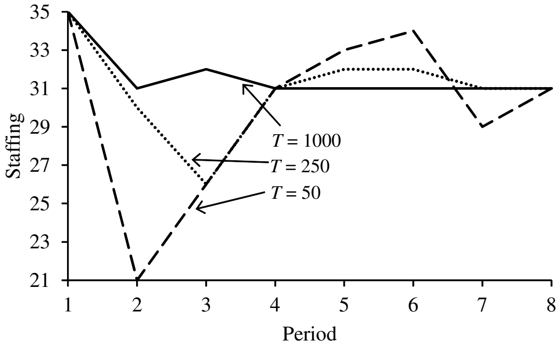


Figure 4.13: The impact of noisy parameter estimation on staffing.

In the remainder of this section, we focus on factors other than parameter estimation noise. Therefore, we assume that the monitoring period is long enough that parameter

Table 4.2: Experiment parameters

Low-load service rate (per hour)	Behavior	Figure
$\mu_1 = 0.50$	Convergence to overstaffing	4.14
$\mu_1 = 0.75$	Convergence to optimal	4.15
$\mu_1 = 0.85$	Convergence to understaffing	4.16
$\mu_1 = 0.90$	Oscillation	4.17
$\mu_1 = 0.95$	Instability	4.18

estimation errors are negligible and that the system reaches steady state, if the system is stable, which means that we can replace the simulation model with the numerical solution of M_2. In what follows, we vary the low-load service rate μ_1 as shown in Table 4.2, to illustrate three main staffing patterns that we have observed: (1) convergence (to a value that is too high, optimal, or too low), (2) oscillation, and (3) instability.

Convergence: As the low-load service rate varies from 0.5 to 0.75 to 0.85 per hour, the staffing converges in all cases, but to a value that changes from too high, to optimal, to too low, as illustrated in Figures 4.14-4.16. The low-load service rate of 0.5 per hour is so much lower than the full-speed service rate of 1 per hour that the Erlang C model cannot capture the speedup, and converges to a staffing level (40 servers) that is far above the optimal value of 32 (Figure 4.14). When the low-load service rate increases to 0.75 per hour—the same value as the overwork service rate—then the Erlang C model approximates the system performance sufficiently well that the staffing converges to the optimal value (Figure 4.15). When the low-load service rate increases above the overwork service rate, to 0.85 per hour, then the Erlang C model's failure to account for the slowdown that occurs due to overwork results in staffing that converges to a value that is too low (29 vs. 31, as shown in Figure 4.16).

Interestingly, we see from Figure 4.14 that the procedure that we have described results in an increase in staffing even when the service level (as computed using M_2) is above target. Similarly, in Figure 4.16, we see one period of a decrease in staffing even though the service level is below the target. These counter-intuitive decisions occur because in our procedure, changes in staffing are determined using the Erlang

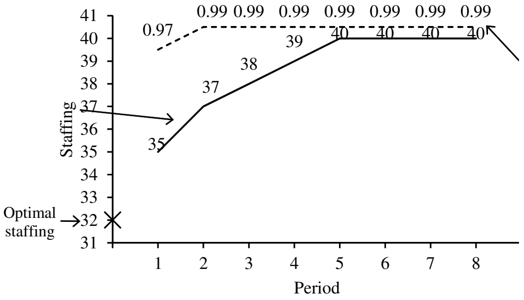


Figure 4.14: Convergence to overstaffing

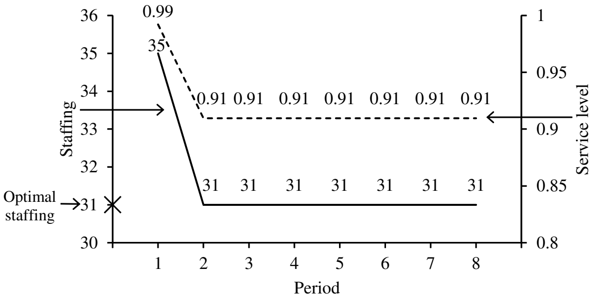


Figure 4.15: Convergence to Optimal staffing

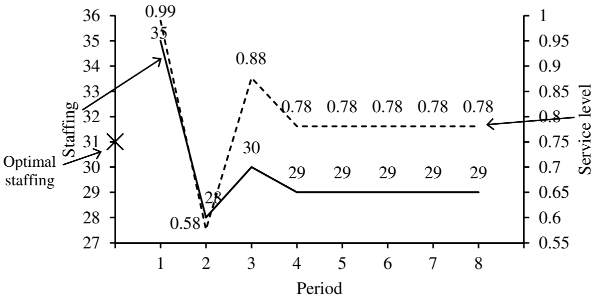


Figure 4.16: Convergence to understaffing

C model. One can envision an alternative “model-free” procedure that determines changes in staffing based only on an empirically estimated service level, $\hat{SL}$. The simplest such procedure might be to increase the staffing by one if $\hat{SL} > \varphi$ and decrease the staffing by one if $\hat{SL} < \varphi$. This procedure would change staffing at the beginning of every monitoring period, assuming that it is unlikely that the empirically estimated service level is exactly equal to its target.

Oscillation: When we increase the low-load service rate to $\mu_1 = 0.90$, then the staffing levels no longer converge, but oscillate (Figure 4.17), between 27 (with $SL = 0.21$) and 31 (with $SL = 0.95$), while the optimal staffing level is 30. The long-term average service level in this oscillating system is 0.58, which is below the target. In addition, in a real system, one expects that the constant staffing changes would be costly and the constant changes in service level might impact customer retention.

The oscillation occurs because with 27 servers, the system is very likely to remain in the overwork region, and with 31 servers, the system is very likely to be in the low-load region. In this situation, the Erlang C model is unreliable in extrapolating service

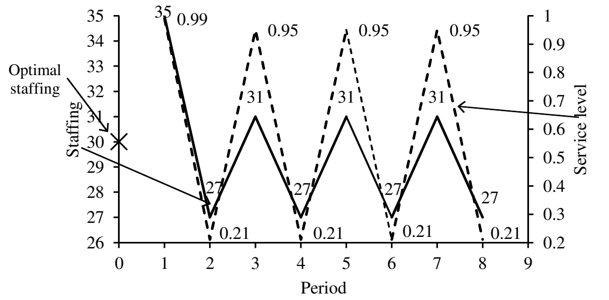
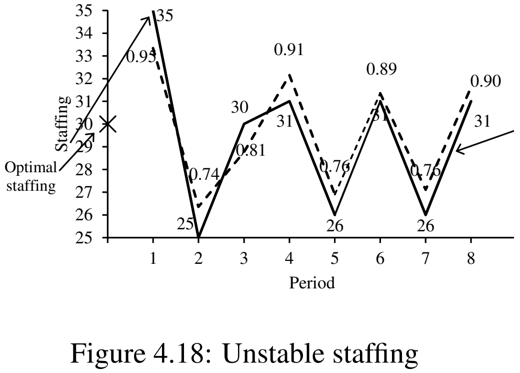


Figure 4.17: Staffing oscillation

levels from the low-load region to the overwork region and vice versa. Specifically, with 31 servers, the average service rate is $\bar{\mu} = 0.89$–quite close to the low-load service rate of $\mu_1 = 0.9$. Similarly, with 27 servers, the average service rate of $\bar{\mu} = 0.77$ is close to the overwork service rate of $\mu_3 = 0.75$. As a consequence, the Erlang C model "overshoots," both when predicting how much to increase staffing and when predicting how much to decrease staffing.

Instability: When we increase the low-load service rate to $\mu_1 = 0.95$, the Erlang C model overestimates the appropriate decrease in staffing so drastically that the system becomes unstable. Starting with 35 servers, the Erlang C model recommends a decrease to 25 servers, which is unstable because $\lambda > 25\mu_3$. For this system, we used simulation (with $T = 1000$ hours) to obtain the service levels shown in Figure 4.18 and the system size sample path in Figure 4.19.



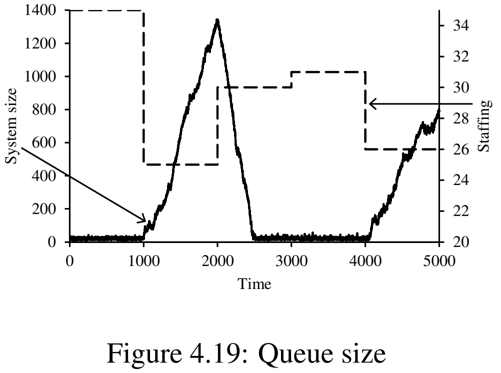


Figure 4.18: Unstable staffing

Figure 4.19: Queue size

CHAPTER 5

Summary of Findings

Empirical researchers have recently challenged the exogenous service times assumption in queueing models by providing evidence for dependence of service times on load in various systems, for example call centers, emergency rooms, and banks. Most of these studies focus on the most obvious manifestation of load in queueing systems, that is, the number of servers currently busy, and its effect on servers. A few studies have also paid attention to the load history in addition to the instantaneous system load. Tracking the load history has unveiled empirical evidence for behaviours like slowdown in response to overwork.

Our aim in Chapters 2 and 3 of this dissertation was to propose a general framework that can be employed by both empirical and analytical researchers to investigate and model service time dependencies in any system. We strove to design a comprehensive framework that can be applied to any system. The proposed framework, which we called it LEST, has three dimensions: (1) load characteristics, (2) system components, and (3) service time determinants.

In the first dimension of the LEST framework, we identified three load characteristics: *changeover*, *load*, and *extended load*, each involving different mechanisms. *Changeover* refers to switching from idle to busy periods or switching from one task to another, which induces mechanisms like setup. *Load* is the congestion level of the system and *extended load* is the past history of load.

In the second dimension of the LEST framework, we identified three system components: *server*, *customer*, and *network*. We recognized that servers are not the only system components that may react to load characteristics. Customers also respond to system load. One obvious example that is well studied is abandonment from a queue. We also came across papers that study queues as nodes in a network and document how busyness of a node affects service times at other nodes. For this reason, we included the *network* as the third system component in our framework.

The third dimension of the framework includes two service time determinants. Sometimes it is the *work content* that is influenced by load characteristics and sometimes it is the *service speed*. The service time is determined by the amount of work required for a service and the rate at which the service is performed.

The proposed framework is beneficial for both empirical and analytical researchers. For empirical researchers, the framework provides a systematic tool to think about the effect of system load on service times from different aspects. The main power of the framework is in provoking questions that lead the researcher to list and explain mechanisms and interactions that cause service times to vary with load. The framework also helps analytical researchers to understand factors that influence service times and how these factors can be translated into state variables in state-dependent models. The framework also emphasizes the importance of characteristics that are often disregarded in models including, single-queue systems vs. queue networks, human vs. inanimate servers or customers, dedicated vs. shared servers, and single vs. multiple customer types.

In Chapter 2, we showed that the findings of published studies can be explained by the framework. The classification of the published studies according to the framework highlights research gaps in the empirical OM literature. In Chapter 3, we applied the framework to an EMS system to investigate possible mechanisms that cause EMS service times to change with load. The framework helped us to hypothesize and identify new mechanisms not previously documented.

Our aim in Chapter 4 was to formulate a tractable and flexible stochastic model that captures two types of adaptive server behaviors that lead to state-dependent service rates: speeding up in response to the system load and slowing down in response to overwork. Markov chains with state-dependent rates can model changes in server speed in response to changes in system load, as measured by the system occupancy, but tracking overwork requires additional state variables.

The model that we formulate can be seen as an extension to the Erlang C model. In addition to the system occupancy state variable that is included in most queueing models, we added one other state variable to capture overwork: The cumulative number of service completions in the current high-load period. This method of operationalizing overwork leads to a model that can be represented as a QBD process with special transition structure that makes it possible to compute steady state probabilities and standard queueing system performance measures efficiently.

We demonstrated through numerical experiments that when service rates depend on load and overwork, use of the Erlang C model provides a poor approximation of the system performance, even if one uses input averaging or output averaging. Using a stylized model of staffing practice, we illustrated how ongoing use of the Erlang C model for staffing with periodically updated arrival and service rate estimates can lead to convergence to a staffing level that is either too high or too low, staffing oscillation, and even to staffing levels that result in an unstable system.

BIBLIOGRAPHY

BIBLIOGRAPHY

Aehlert, B., R. Vroman. 2011. *Paramedic practice today: above and beyond*. Vol. 2. Jones & Bartlett Publishers.

Aksin, O. Z., P. T. Harker (2003). Capacity sizing in the presence of a common shared resource: Dimensioning an inbound call center. *European Journal of Operational Research.* **147**(3), 464–483.

Alanis, R., A. Ingolfsson, B. Kolfal. 2013. A Markov chain model for an EMS system with repositioning. *Production and Operations Management.* **22**(1) 216–231.

Alberta Health Services. 2010. Emergency department surge capacity protocols. Available from `http://www.albertahealthservices.ca/3167.asp`. Last access 10 March 2014.

Asaro, P. V., L. M. Lewis, S. B. Boxerman. 2007. The impact of input and output factors on emergency department throughput. *Academic Emergency Medicine.* **14**(3) 235–242.

Armony M, Israelit S, Mandelbaum A, Marmor YN, Tseytlin Y, Yom-Tov GB (2010) Patient flow in hospitals: A data-based queueing-science perspective. Working paper.

Artalejo JR, Lopez-Herrero MJ (2001) Analysis of the busy period for the $M/M/c$ queue: An algorithmic approach. *Journal of Applied Probability* 38(1):209–222.

Batt JR, Terwiesch C (2012) Doctors under load: An empirical study of state-dependent service times in emergency care. Working paper.

Bellman R (1961) *Adaptive Control Processes: A Guided Tour* (Princeton University Press, Princeton, NJ).

Berk E, Moinzadeh K (1998) The impact of discharge decisions on health care quality. *Management Science* 44(3):400–415.

Bertsekas DP, Tsitsiklis JN (2008) *Introduction to probability* (Athena Scientific, Belmont, Massachusetts).

Bailey, C.D. 1989. Forgetting and the learning curve: a laboratory study. *Management Science*. **35**(3) 340-352.

Bandiera, O., I. Barankay, I. Rasul. 2012. Team incentives: Evidence from a firm level experiment. *Working paper*.

Bitran, G., M. Lojo. 1993. A framework for analyzing the quality of the customer interface. *European Management Journal.* **11**(4) 385–396.

Blackwell, T. H., J. S. Kaufman. 2002. Response time effectiveness: comparison of response time and survival in an urban emergency medical services system. *Academic Emergency Medicine*. **9**(4) 288–295.

Brockmeyer, E., H. L. Halstrm, A. Jensen. 1948. *The life and works of A. K. Erlang*. Transactions of the Dansih Academy of Technical Sciences, Copenhagen.

Brown, L. H. 2006. Multitasking: Function or fallacy? *Business Coaching WorldWide* **2**(3).

Brown, L., N. Gans, A. Mandelbaum, A. Sakov, H. Shen, S. Zeltyn, L. Zhao. 2005. Statistical analysis of a telephone call center: A queueing-science perspective. *Journal of the American Statistical Association.* **100**(469) 36–50.

Budge, S., A. Ingolfsson, D. Zerom. 2010. Empirical analysis of ambulance travel times: the case of Calgary emergency medical services. *Management Science.* **56**(4) 716–723.

Cakir, A., D. J. Hart, T. F. Stewart. 1980. Visual Display Terminals: A Manual Covering Ergonomics, Workplace Design, Health and Safety, Task Organization. John Wiley & Sons, New York.

Carlson, J.G., A.J. Rowe. 1976. How much does forgetting cost? *Industrial Engineering.* **8**(9) 40-47.

Cellier, J.M., H. Eyrolle. 1992. Interference between switched tasks. *Ergonomics*. **35**(1) 25-36.

Chan CW, Yom-Tov G, Escobar G (2012) When to use speedup: An examination of intensive care units with readmissions. Working paper.

Chisholm, C. D., E. K. Collison, D. R. Nelson, W. H. Cordell. 2000. Emergency department workplace interruptions: Are emergency physicians "interrupt-driven" and "multitasking"? *Academic Emergency Medicine.* **7**(11) 1239–1243.

Crabill T (1972) Optimal control of a service facility with variable exponential service times and constant arrival rate. *Management Science* 18(9):560–566.

Delasay, M., A. Ingolfsson, B. Kolfal. 2013. Modeling load and overwork Effects in queueing systems with adaptive servers. *Working paper*.

Dietz, D.C. 2011. Practical scheduling for call center operations. *Omega*. **39**(5) 550–557.

Dshalalow, J. H. 1997. Queueing systems with state dependent parameters. *Frontiers in Queueing: Models and Applications in Science and Engineering*. 61–116.

Edie LC (1954) Traffic delays at toll booths. *Operations Research* 2(2):107–138.

Feero, S., J. R. Hedges, E. Simmons, L. Irwin. 1995. Does out-of-hospital EMS time affect trauma survival?. *The American Journal of Emergency Medicine*. **13**(2) 133–135.

Fisher, M. 2007. Strengthening the empirical base of operations management. *Manufacturing & Service Operations Management*. **9**(4) 368–382.

Forster, A. J., I. Stiell, G. Wells, A. J. Lee, C. Van Walraven. 2003. The effect of hospital occupancy on emergency department length of stay and patient disposition. *Academic Emergency Medicine.* **10**(2) 127–133.

Gans N, Liu N, Mandelbaum A, Shen H, Ye H (2010) Service times in call centers: Agent heterogeneity and learning with some operational consequences. *A Festschrift for Lawrence D. Brown, IMS Collections* 6:99–123.

George JM, Harrison JM (2001) Dynamic control of a queue with adjustable service rate. *Operations Research* 49(5):720–731.

Gladstones, W.H., M.A. Regan, R.B. Lee. 1989. Division of attention: The single-channel hypothesis revisited. *Quarterly Journal of Experimental Psychology: Human Experimental Psychology*. **41**(1) 1-17.

Grassmann W (1977) Transient solutions in Markovian queueing systems. *Computers and Operations Research* 4(1):47–54.

Grassmann W (1983) The convexity of the mean queue size of the $M/M/c$ queue with respect to the traffic intensity. *Journal of Applied Probability* 20(4):916–919.

Green, L. V., S. Savin, N. Savva. 2013. “Nursevendor Problem”: Personnel Staffing in the Presence of Endogenous Absenteeism. *Management Science*.

Gross, D., J. F. Shortle, J. M. Thompson, C. M. Harris. 2008. *Fundamentals of queueing theory*. John Wliey & Sons, New York.

Gupta, S., R. Verma, L. Victorino. 2006. Empirical research published in production and operations management (19922005): trends and future research directions. *Production and Operations Management*. **15**(3) 432–448.

Harel A (2010) Sharp and simple bounds for the Erlang delay and loss formulae. *Queueing Systems* 64(2):119–143.

Harris CM (1967) Queues with state-dependent stochastic service rates. *Operations Research* 15(1):117–130.

Hasija, S., E. Pinker, R. A. Shumsky. 2010. OM Practice-Work Expands to Fill the Time Available: Capacity Estimation and Staffing Under Parkinson's Law. *Manufacturing & Service Operations Management*. **12**(1) 1–18.

Hillier, D. F., G. J. Parry, M. W. Shannon, A. M. Stack. 2009. The effect of hospital bed occupancy on throughput in the pediatric emergency department. *Annals of Emergency Medicine.* **53**(6) 767–776.

Hopp, W. J., S. M. Iravani, G. Y. Yuen. 2007. Operations systems with discretionary task completion. *Management Science*. **53**(1) 61–77.

Inman, R. R. 1999. Empirical evaluation of exponential and independence assumptions in queueing models of manufacturing systems. *Production and Operations Management*. **8**(4) 409–432.

Jackson JR (1963) Jobshop-like queueing systems. *Management Science* 10(1):131–142.

Jaeker, J. B., A. L. Tucker. 2012. Hurry up and wait: Differential impacts of congestion, bottleneck pressure, and predictability on patient length of stay. *Harvard Business School Working Paper*, No. 13052.

Karau, S. J., K. D. Williams. 1993. Social loafing: A meta-analytic review and theoretical integration. *Journal of Personality and Social Psychology*. **65**(4) 681–706.

Kc, D. 2011. Does multi-tasking improve productivity and quality? Evidence from the emergency department. *Working paper.*

Kc D, Terwiesch C (2009) Impact of workload on service time and patient safety: An econometric analysis of hospital operations. *Management Science* 55(9):1486–1498.

Kc D, Terwiesch C (2012) An econometric analysis of patient flows in the cardiac intensive care unit. *Manufacturing and Service Operations Management* 14(1):50–65.

Khudyakov P, Gorfine M, Mandelbaum A (2010) Phase-type models of service times. In preparation.

Krumm, D. 2000. *Psychology at work: An introduction to industrial/organizational psychology*. Worth Publishers, New York.

Kuntz L, Mennicken R, Scholtes S (2011) Stress on the ward–An empirical study of the nonlinear relationship between organizational workload and service quality. *Ruhr Economic Papers* 277.

Latouche G, Ramaswami V (1999) *Introduction to Matrix Analytic Methods in Stochastic Modeling* (ASA SIAM Series on Statistics and Applied Probability, SIAM, Philadelphia).

Latané, B., W. Kipling, S. Harkins. 1979. Many hands make light the work: the causes and consequences of social loafing. *Journal of Personality and Social Psychology*. **37**(6) 822-832.

Levy, Y., U. Yechiali. 1975. Utilization of idle time in an $M/G/1$ queueing system. *Management Science*. **22**(2) 202–211.

Lu, Y. 2013. Data-driven system design in service operations. PhD thesis.

Mas, A., E. Moretti. 2009. Peers at Work. *American Econimc Review*. **99**(1) 112–145.

Mæstad, O., G. Torsvik, A. Aakvik. 2010. Overworked? On the relationship between workload and health worker performance. *Journal of health economics*. **29**(5) 686–698.

Mayer, R.E., R. Moreno. 2003. Nine ways to reduce cognitive load in multimedia learning. *Educational psychologist* **38**(1) 43–52.

Medhi D (1996) Single server queueing system with Poisson input: A review of some recent develepements. In *Advances in combinatorial methods and applications to probability and statistics*. (Birkhauser, Boston, N. Balakrishnan (Ed.)).

Neuts, M.F. 1981. *Matrix-Geometric Solutions in Stochastic Models: An Algorithmic Approach*. The John Hopkins Press, Baltimore.

Pashler, H. 1994. Dual-task interference in simple tasks: Data and theory. *Psychological Bulletin*. **116**(2) 220-244.

Pennebaker, R. 2009. The mediocre multitasker. *The New York Times*. Availabe from `http://www.nytimes.com/2009/08/30/weekinreview/30pennebaker.html`.

Pisano, P., R.M.J. Bohmer, A.C. Edmondson. 2001. Organizational diferences in rates of learning: evidence from the adoption of minimally invasive cardiac surgery. *Management Science*. **47**(6) 752-768.

Powell SG, Schultz KL (2004) Throughput in serial lines with state-dependent behavior. *Management Science* 50(8):1095–1105.

Ramaswami V, Lucantoni DM (1985) Stationary waiting time distribution in queues with phase type service and in quasi-birth-and-death processes. *Stochastic Models* 1(2):125–136.

Robbins, T. R., D. J. Medeiros, T. P. Harrison. 2010. Does the Erlang C model fit in real call centers? *Proceedings of the 2010 Winter Simulation Conference*.

Roth, A. V. 2007. Applications of empirical science in manufacturing and service operations. *Manufacturing & Service Operations Management*. **9**(4) 353–367.

Rubinstein, J.S., D.E. Meyer, J.E. Evans. 2001. Executive control of cognitive processes in task switching. *Journal of Experimental Psychology: Human Perception and Performance* **27**(4) 763–797.

Schultz, K. L., D. C. Juran, J. W. Boudreau, J. O. McClain, L. J. Thomas. 1998. Modeling and worker motivation in JIT production systems. *Management Science*. **44**(12) 1595–1607.

Schultz, K. L., J. O. McClain, L. J. Thomas. 2003. Overcoming the dark side of worker flexibility. *Journal of Operations Management*. **21**(1) 81–92.

Scudder, G. D., C. A. Hill. 1998. A review and classification of empirical research in operations management. *Journal of Operations Management*. **16**(1) 91–101.

Setyawati, L. (1995). Relation between feelings of fatigue, reaction time and work productivity. *Journal of Human Ergology*. **24**(1) 129–35.

Staats, B. R., F. Gino. 2012. Specialization and variety in repetitive tasks: Evidence from a Japanese bank. *Management Science*. **58**(6) 1141–1159.

Steedman, I. 1970. Some improvement curve theory. *International Journal of Production Research*. **8**(3) 189-205.

Stidham, S., R. R. Weber. 1989. Monotonic and insensitive optimal policies for control of queues with undiscounted costs. *Operations Research*. **87**(4) 611-625.

Sze, D. 1984. A queueing model for telephone operator staffing. *Operations Research*. **32**(2) 229–249.

Takacs L (1955) Investigation of waiting time problems by reduction to Markov processes. *Acta Mathematica Hungarica* 6(1):101–129.

Tan, T. F., S. Netessine. 2014. When does the devil make work? An empirical study of the impact of workload on worker productivity. *Management Science*. **60**(6) 1574-1593.

Tanabe, S., N. Nishihara. 2004. Productivity and fatigue. *Indoor Air*. **14**(s7) 126–133.

Taylor, P., P. Bain. 1999. An assembly line in the head: work and employee relations in the call centre. *Industrial Relations Journal*. **30**(2) 101–117.

Van Den Broek, D. (2002). Monitoring and surveillance in call centres: some responses from Australian workers. *Labour & Industry: a journal of the social and economic relations of work*. **12**(3) 43–58.

Van Houdt B, van Leeuwaarden JSH (2011) Triangular $M/G/1$-type and tree-like quasi-birth-death Markov chains. *INFORMS Journal on Computing* 23(1):165–171.

van Leeuwaarden JSH, Winands EMM (2006) Quasi-birth-and-death processes with an explicit rate matrix. *Stochastic Models* 22(1):77–98.

Welch PD (1964) On a generalized $M/G/1$ queuing process in which the first customer of each busy period receives exceptional service. *Operations Research* 12(5):736–752.

Wright, T.P. 1936. Factors affecting the costs of airplanes. *Journal of Aeronautical Science*. **1**(3) 122-128.

APPENDICES

APPENDIX A

Chapter 4: Modeling Load and Overwork Effects in Queueing Systems with Adaptive Servers

A.1 Proof of Theorem 4.4

Proof. Proof. Let $\pi_{i,j}$ and $\pi'_{i,j}$ denote the steady state probabilities for Models M_1 and M_2, respectively. The proof follows by showing that balance equations of Model M_1 provide balance equations of Model M_2 if

$$\pi'_{i,j} = \pi_{i,j}, \qquad 0 \leq j \leq m-1, \tag{A.1}$$

$$\pi'_{i,m} = \sum_{j=m}^{\infty} \pi_{i,j}. \tag{A.2}$$

State $(i,j) \in \Omega_2 - \{(k-1,0), j=m\}$: Model M_1 equations follow,

$$\lambda\pi_{i,0} = \mu_{i+1,0}\pi_{i+1,0}, \qquad 0 \leq i \leq k-2,$$

$$(\lambda + b_i\mu_{i,0})\pi_{i,0} = \lambda\pi_{i-1,0}, \qquad i \geq k, j=0,$$

$$(\lambda + k\mu_{k,j})\pi_{k,j} = (k+1)\mu_{k+1,j-1}\pi_{k+1,j-1}, \qquad 1 \leq j \leq m-1,$$

$$(\lambda + b_i\mu_{i,j})\pi_{i,j} = \lambda\pi_{i-1,j}, + b_{i+1}\mu_{i+1,j-1}\pi_{i+1,j-1}, \qquad i > k, 1 \leq j \leq m-1,$$

which also provide balance equations of Model M_2 if (A.1) holds.

State $(i, j) \in \Omega_1 \cap \{i \geq k, j \geq m\}$: In Model M_1, $\mu_{i,j} = \mu_{i,m}, j > m$, and balance equations follow:

$$(\lambda + k\mu_{k,m})\pi_{k,m} = (k+1)\mu_{k+1,m-1}\pi_{k+1,m-1}, \tag{A.3}$$

$$(\lambda + k\mu_{k,m})\pi_{k,j} = (k+1)\mu_{k+1,m}\pi_{k+1,m-1}, \qquad j > m, \tag{A.4}$$

$$(\lambda + b_i\mu_{i,m})\pi_{i,m} = \lambda\pi_{i-1,m} + b_{i+1}\mu_{i+1,m-1}\pi_{i+1,m-1}, \qquad i > k, \tag{A.5}$$

$$(\lambda + b_i\mu_{i,m})\pi_{i,j} = \lambda\pi_{i-1,j} + b_{i+1}\mu_{i+1,m}\pi_{i+1,j-1}, \qquad i > k, j > m. \tag{A.6}$$

If (A.1) and (A.2) hold, the summation of equations (A.3)-(A.6) over j provides equation set for Model M_2, when $j = m$, as

$$(\lambda + k\mu_{k,m})\pi'_{k,m} = (k+1)\mu_{k+1,m-1}\pi'_{k+1,m-1} + (k+1)\mu_{k+1,m}\pi'_{k+1,m}, \tag{A.7}$$

$$(\lambda + b_i\mu_{i,m})\pi'_{i,m} = \lambda\pi'_{i-1,m} + b_{i+1}\mu_{i+1,m-1}\pi'_{i+1,m-1} + b_{i+1}\mu_{i+1,m}\pi'_{i+1,m}, \qquad i > k. \tag{A.8}$$

State $(k-1, 0)$: Balance equations of state $(k-1, 0)$ in Models M_1 and M_2 follow equation sets (A.9) and (A.10), respectively, that are equivalent if (A.1) and (A.2) hold.

$$\lambda\pi_{k-1,0} = k\sum_{j=0}^{m-1}\mu_{k,j}\pi_{k,j} + k\mu_{k,m}\sum_{j=m}^{\infty}\pi_{k,j}, \tag{A.9}$$

$$\lambda\pi'_{k-1,0} = k\sum_{j=0}^{m-1}\mu_{k,j}\pi'_{k,j} + k\mu_{k,m}\pi'_{k,m}. \tag{A.10}$$

□

A.2 Property 1 for Level $k-1$

In this appendix, we show that the recursion $\boldsymbol{\pi}_{i+1} = \mathbf{R}^{(i)}\boldsymbol{\pi}_i$, Property 1, and the associated algorithm for computing $\mathbf{R}^{(i)}$ is valid for level $i = k-1$, with minor modifications, even though the topology of the transitions between levels $k-1$ and k is different from that for higher levels. This allows us to express the equations for the steady state probabilities more compactly. The only state in level $k-1$ is $(k-1, 0)$. One can form a vector of state probabilities for level $k-1$ by forcing probabilities $\boldsymbol{\pi}_{k-1,j}=0$ for $j\geq 1$, which results in $\boldsymbol{\pi}_{k-1}=(\pi_{k-1,0},0,\cdots,0)$. The structure of the rate matrix $\mathbf{R}^{(k-1)}$ is as follows:

$$\mathbf{R}^{(k-1)}=\begin{pmatrix} R_{0,0}^{(k-1)} & \cdots & R_{0,m}^{(k-1)} \\ 0 & \cdots & 0 \\ & \ddots & \vdots \\ & & 0 \end{pmatrix},$$

To calculate $q_{0,h}^{(k-1)}$, the probability of an excursion from state $(k-1,0)$ to state (k,h), we use the approach from Section 4.5.1.1, modified as shown in equation set (A.11), as like the upper levels, an arrival moves the excursion from state $(k-1,0)$ to level k and a departure from each state of level k terminates the excursion. The other transition of the excursion, in levels $i>k$, follow the structure of Figure 4.3b.

$$\begin{cases} q_{h,h}^{(i)}=\phi_{i,h}, & \text{if } h=0, \\ q_{0,h}^{(i)}=\phi_{i,0}\delta_{i+1,0}^{i+1,h} & \\ \delta_{i+1,b}^{i+1,h}=\phi_{i+1,b}\delta_{i+2,b}^{i+1,h}, & b=0,\cdots,h-1, \\ \delta_{a,b}^{i+1,h}=\psi_{a,b}\delta_{a-1,b+1}^{i+1,h}, & a=i+2,\cdots,i+h+1, \quad b=i+h+1-a, \\ \delta_{a,b}^{i+1,h}=\phi_{a,b}\delta_{a+1,b}^{i+1,h}+\psi_{a,b}\delta_{a-1,b+1}^{i+1,h}, & a=i+2,\cdots,i+h, \qquad b=0,\cdots,i+h-a. \end{cases} \tag{A.11}$$

A.3 Wait Time Distribution

The stationary probability $\overline{W}(x)$ that a user waits more than x time units before entering service in a queue that is modelled as a QBD process can be expressed as (Ramaswami and Lucantoni 1985, Theorem 4)

$$\overline{W}(x)=\sum_{n=0}^{\infty}d_n e^{-\theta x}\frac{(\theta x)^n}{n!}, \tag{A.12}$$

where d_n is the probability that the user waits at least $n+1$ epochs of a uniformizing Poisson process with rate θ in the stochastically equivalent construction of the QBD process. In our level-dependent QBD,

$$\theta = \max_{1 \le j \le m+1} -\{\mathbf{A}_0 + \mathbf{A}_1^{(s)}\}_{jj} = \max_j \{\lambda + s\mu_{s,j}\},$$

$$d_n = \boldsymbol{\pi}_{s-1} \left(\mathbf{I} - \mathbf{R}^{(s)}\right)^{-1} \mathbf{R}^{(s)} \mathbf{H}_n \mathbf{e}, \qquad n \ge 0,$$

$$\mathbf{H}_0 = \mathbf{I}, \qquad \mathbf{H}_{n+1} = \mathbf{H}_n \mathbf{P}_1 + \mathbf{R}^{(s)} \mathbf{H}_n \mathbf{P}_2, \qquad n \ge 0,$$

and

$$\mathbf{P}_1 = \frac{1}{\theta}(\mathbf{A}_0 + \mathbf{A}_1^{(s)}) + \mathbf{I}, \qquad \mathbf{P}_2 = \frac{1}{\theta}\mathbf{A}_2^{(s)}.$$

One can use two approaches to find the stopping criterion for n in the infinite series (A.12). The simpler approach is to set the upper limit for the series equal to (Grassmann 1977, eq. (10))

$$UB = \theta x + 4\sqrt{\theta x} + 5, \tag{A.13}$$

which guarantees that $1 - \sum_{n=0}^{UB} e^{-\theta x}(\theta x)^n/n!$ is less than 10^{-4}.

The second approach requires more effort. Based on (A.12), Ramaswami and Lucantoni (1985) derive the expected wait time in queue as,

$$W_q = \theta^{-1} \sum_{n=0}^{\infty} d_n. \tag{A.14}$$

One can gradually increase the upper limit of n so that the result of (A.14) falls in the acceptable tolerance from the exact expected wait time obtained from the closed form solution of W_q presented in Table 4.1.

A.4 A Note on Figures 4.11 and 4.12

Suppose we view the service rate μ in the Erlang C model as a random variable, with a distribution obtained from the steady-state probabilities for M_2, that is $\Pr\{\mu = \mu_{i,j}\} = \pi_{i,j}$ with $\overline{\mu} = E(\mu)$. Then Jensen's inequality implies that $\mathrm{E}[f(\mu)] > f(\bar{\mu})$ for convex functions f. We show that $C(.)$ and $D(.)$ are convex functions of μ.

The delay probability $C(.)$ is an increasing and convex function of $\rho = \lambda/(s\mu)$ when s is held constant (Harel 2010, Proposition 4). The utilization ρ is is convex in μ. Therefore, $C(.)$ as a function of μ is a composition of a convex increasing function and a convex function, which implies that $C(.)$ is convex in μ.

Grassmann (1983) shows that the average queue length in an Erlang C model is a convex function of μ. Combined with Little's Law, this result implies that the average delay $D(.)$ is convex with respect to μ.