ENERGY CONSUMPTION IN DATA CENTERS WITH DETERMINISTIC SETUP TIMES

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF APPLIED MATHEMATICS
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

AYTAÇ KARA

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
FINANCIAL MATHEMATICS

SEPTEMBER 2017

Approval of the thesis:

ENERGY CONSUMPTION IN DATA CENTERS WITH DETERMINISTIC SETUP TIMES

submitted by **AYTAÇ KARA** in partial fulfillment of the requirements for the degree of **Master of Science in Department of Financial Mathematics, Middle East Technical University** by,

Prof. Dr. Bülent Karasözen
Director, Graduate School of **Applied Mathematics** ________________

Assoc. Prof. Dr. Yeliz Yolcu Okur
Head of Department, **Financial Mathematics** ________________

Assoc. Prof. Dr. Ali Devin Sezer
Supervisor, **Financial Mathematics, METU** ________________

Asst. Prof. Tuan Phung-Duc
Co-supervisor, **Faculty of Engineering, Information and Systems, University of Tsukuba** ________________

Examining Committee Members:

Prof. Dr. Gerhard Wilhelm Weber
Scientific Computing, METU ________________

Assoc. Prof. Dr. Ali Devin Sezer
Finacial Mathematics, METU ________________

Asst. Prof. Özge Sezgin Alp
Accounting and Financial Management, Başkent University ________________

Date: ________________

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: AYTAÇ KARA

Signature :

ABSTRACT

ENERGY CONSUMPTION IN DATA CENTERS WITH DETERMINISTIC SETUP TIMES

Kara, Aytaç
M.S., Department of Financial Mathematics
Supervisor : Assoc. Prof. Dr. Ali Devin Sezer
Co-Supervisor : Asst. Prof. Tuan Phung-Duc

September 2017, 44 pages

Data centers, which are networks consisting of thousands of computers, are central objects in the global computation infrastructure. Typical data centers today may consume as much electricity as a small town. Thus, it is of interest to build models of these centers that allow one to study / optimize their energy usage. One of the models for the energy usage based on queueng theory is the one developed in "Exact Solutions for $M/M/c/Setup$ Queues" by Tuan Phung-Duc. The same work carries out a stationary analysis of the developed model to compute the long term average energy cost per unit time. The model of Phung-Duc assumes that the data center consists of c servers and that the servers are in one of the following three modes: running, stopped or in setup. The setup mode is assumed to last a random exponentially distributed time. We modify this model as follows: we replace exponentially distributed setup times with a fixed deterministic setup time. We call the resulting model $M/M/c/dSetup$. We approximate the long term average cost per unit time via simulation and compare this cost with that of the $M/M/c/Setup$ system. Our main finding are as follows: the average energy cost of these systems provide good approximations of one another. Secondly, the average energy cost of the $M/M/c/Setup$ system provides a lower bound for that of the $M/M/c/dSetup$ system.

Keywords : queueing theory, energy consumption, average cost of energy per unit time, data centers, deterministic setup time, electricity markets

ÖZ

SABİT KURULUM SÜRELİ VERİ MERKEZLERİNDE ENERJİ TÜKETİMİ

Kara, Aytaç
Yüksek Lisans, Finansal Matematik Bölümü
Tez Yöneticisi : Doç. Dr. Ali Devin Sezer
Ortak Tez Yöneticisi : Yardımcı Prof. Tuan Phung-Duc

Ağustos 2017, 44 sayfa

Binlerce bilgisayardan oluşan veri merkezleri, küresel hesaplama alt yapısının temel yapıtaşlarından birisini oluşturmaktadır. Bugünkü tipik veri merkezleri küçük bir kasaba kadar elektrik tüketebilir. Bunlar veri merkezlerinin modellenmesini ve bunların enerji kullanamının analizini önemli araştırma konuları yapmaktadır. Bu yönde geliştirilmiş ve kuyruk teorisi üzerine kurulu modellerden biri Tuan Phung-Duc'un "Exact solutions for M/M/c/Setup" makalesinde verilmiştir. Bu makale önerdiği modelin durağan analizini yapmış, bu analizi kullanarak modelin birim zaman başına ortalama enerji maliyetini hesaplamış ve bu maliyetin model parametrelerine nasıl bağlı olduğunun bir analizini vermiştir. $M/M/c/Setup$ modeli c veri sunucusundan oluşur ve bu sunucuların "açık," "kapalı" ve "açılıyor" (setup) hallerinden birinde olduğunu varsayar.
$M/M/c/Setup$ modelinde, her sunucu için, açılma zamanının üssel dağılıma sahip bir rassal değişken olduğu varsayılır. Bu çalışmamızda $M/M/c/Setup$ modelini şu şekilde değiştiriyoruz: açılma süresini rassal bir değişken almak yerine sabit bir zaman alıyoruz; bu modele $M/M/c/dSetup$ diyelim. Bu varsayım altında sistemin birim zaman başına ortalama enerji maliyetini simülasyon kullaranak yaklaşık olarak hesaplıyoruz ve $M/M/c/Setup$ sisteminin ortalama enerji maliyetiyle karşılaştırıyoruz. Analizimizin temel sonucu şudur: $M/M/c/Setup$ sisteminin ortalama birim zaman enerji maliyeti $M/M/c/dSetup$ sisteminin ortalama birim zaman enerji maliyetinin altında kalmakla beraber bu iki sistemin ortalama birim zaman enerji maliyetleri birbirlerine çok yakın çıkmaktadır.

Anahtar Kelimeler: kuyruk teorisi, enerji tüketimi, veri merkezleri, sabit kurulum süresi, birim zamana düşen ortalama enerji maliyeti, elektrik piyasaları

To My Family

ACKNOWLEDGMENTS

I would like to express my gratitude to my thesis supervisor Assoc. Prof. Dr. Ali Devin Sezer for his guidance, motivation and support throughout this study.

I would also like to thank my co-advisor Asst. Prof. Tuan Phung-Duc for his support and help.

Special thanks to my committee members, Prof. Dr. Gerhard Wilhelm Weber and Asst. Prof. Özge Sezgin Alp for their advices, corrections and comments.

I also would like to thank my family and my friends for their support and motivation.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

Introduction

Cloud computing is a service where a company earns money by allowing its users to run programs on its data centers, which are networks of thousands of computers suitable for processing data traffic. Typical data centers today may consume as much electricity as a small town. These servers spend a huge amount of energy to process data and to keep cool. Minimization of power consumption provides savings of a considerable amount of money on behalf of the company and reduce its environmental impact. Thus, it is of interest to study and optimize energy usage of these centers.

As of now, an idle server still uses about 60 % of its peak energy usage [3]. Therefore, a potential way to save power is by turning off idle servers. Nonetheless, off servers need setup time to be an active server; the setup process also consumes power and servers in setup mode cannot process jobs. Thus, there is a nontrivial question of comparison between the two regimes 1) keeping servers always on and 2) turning off idle servers.

To answer questions of this sort, queueing theory provides a natural framework; Chapter 2 of this thesis gives a very brief review of the queueing theory framework. Maccio and Down [14] model a multiple server system with setup times for optimal control of energy aware queueing systems. They use Markov decision process (MDP) and analyze the model to obtain main properties of optimal and suboptimal policies. Gandhi et al. [10] determine a few closed form approximations for the ON-OFF policy in which in a large number of servers can be in setup mode at once. There is a wide literature on queueing systems with setup time both with applications to data centers and other manufacturing processes, the works see [4, 5, 6, 20] for single system servers and [1, 8, 9, 10, 12, 14, 15] treat multi server systems. See [17] for a review of many of these works.

A well known model of data centers that allow servers to be in setup mode is the $M/M/c/Setup$ model. This model is reviewed in Chapter 3 of this thesis. [8, 9] analyze the $M/M/c/Setup$ model with ON-OFF policy using a recursive renewal reward approach (RRR). Phung-Duc [17] derives exact solutions of the stationary measure of $M/M/c/Setup$ sytem using the generating function approach and matrix analytic method. The generating function approach gives closed form expressions for the joint stationary queue length distribution and the conditional decomposition formula. On the other hand, the matrix analytic approach gives to a recursive algorithm to obtain the joint stationary distribution and performance measures. Chapter 3 gives a summary of how [17] applies these approaches to the $M/M/c/Setup$ system.

From a finance point of view, [17] and other works taking the queueing theory approach to the cost analysis of data centers, use the concept of "average cost post unit time" or equivalently "Long-run average cost" as their financial measure. This is one of the basic quantities that one can use to analyze the costs of any business. Average cost per unit time is reviewed below. We will also be using this quantity to measure the energy costs of data centers.

The processes that a computer goes through as it turns on, i.e., the whole setup process, are usually constant and one expects that, at least approximately, the setup process uses the same time and energy each time it is repeated. Therefore, it makes sense to take the duration of this process as a constant, as opposed to the exponentially distributed random assumption made for this time in the $M/M/c/Setup$ model. Motivated by this observation, the goal of this thesis is to study the $M/M/c/dSetup$ model, which differs from the $M/M/c/Setup$ model only as follows: in $M/M/c/Setup$ the setup time for a server is exponentially distributed with mean $1/\alpha$; in the $M/M/c/dSetup$ model, the setup time is deterministic and it equals $1/\alpha$. The $M/M/c/dSetup$ model is introduced in Chapter 4. The parameters of the $M/M/c/dSetup$ model are as follows: $1/\alpha$: the deterministic setup time, λ: the rate of the arrival of jobs to the data center; the arrival process is assumed to be Poisson, c: the number of servers in the data center, μ: the rate at which a server finishes serving a job, the duration of this service is assumed to be exponentially distributed. The $M/M/c/Setup$ has the same set of parameters; only the interpretation of α is different: in $M/M/c/Setup$, $1/\alpha$ is the mean time that it takes a server to finish its setup, the duration of the setup is assumed exponentially distributed. We will denote the average cost per unit time of the $M/M/c/Setup$ system by E_∞ and that of the $M/M/c/dSetup$ by E^d_∞. These will be functions of the system parameters. Chapter 5 approximate E_∞ and E^d_∞ using simulation and compares these costs as the system parameters vary. Our main finding are as follows: the average energy cost of these systems provide good approximations of one another. Secondly, the average energy cost of the $M/M/c/Setup$ system provides a lower bound for that of the $M/M/c/dSetup$ system. We list some directions for future research in the Conclusion, which is our Chapter 6.

1.1 Long-run average cost / Average cost per unit time

Let $C(t)$ denote the rate of spending of the business at time t, the "long-run average cost", or "average cost per unit time" of the business will then equal

$$AC = \lim_{T\to\infty} \frac{1}{T} \int_0^T C(t)dt.$$

If C is a stable process, it will have a stationary measure μ and by the Ergodic Theorem the above limit will equal

$$AC = \int_0^T x\mu(dx).$$

The greatest cost associated with running a data center is the cost of energy to run it. Running a data center is a complex task involving many considerations other than energy costs. But given the importance of energy costs, this thesis will only focus on the energy costs. Let us now talk about the the average energy cost unit time for the $M/M/c/Setup$ and $/M/M/c/dSetup$ models. In these models we will have the following processes: R_t: the number of servers running at time t, i.e., in ON mode, and S_t: the number of servers in setup mode in time t. We will assume that a server in ON [Setup] mode consumes energy at a constant rate C_a [C_s]. Then the total energy spent at time t will be

$$\int_0^T (R_t C_a + S_t C_s) dt.$$

We will further assume that the cost of unit energy is constant at all times. For this reason it suffices to focus on the total energy consumption which is represented by the above integral. The long time unit energy consumption rate is

$$\lim_{T \to \infty} \frac{1}{T} \int_0^T (R_t C_a + S_t C_s) dt.$$

If the underlying system is stable and has a stationary measure μ, the Ergodic Theorem quoted above implies

$$\lim_{T \to \infty} \frac{1}{T} \int_0^T (R_t C_a + S_t C_s) dt = \mathbb{E}_\mu[R_t] C_a + \mathbb{E}_\mu[S_t] C_s. \tag{1.1}$$

We will denote this last expression by E_∞^{MMC} for the $M/M/c$ model reviewed in Chapter 2, E_∞ for the $M/M/c/Setup$ model reviewed in Chapter 3 and E_∞^d for the $M/M/c/dSetup$ model given in Chapter 4. To compute these quantitites we only have to compute the expectations appearing in (1.1). Very simple explicit formulas exist for them for the $M/M/c$ case (in this case S_t is always zero, so we have to only compute the first expectation), which are reviewed in Chapter 2. The approximation of them given in [17] for the $M/M/c/Setup$ case is reviewed in Chapter 3. Finally, in Chapter 5 we approximate the same expecatation via simulation for the $M/M/c/dSetup$ model.

CHAPTER 2

Basic Queueing Systems

In this chapter, queueing systems (or queueing theory) and models are presented with their many feautures. We begin by an introduction. Afterwards, classification of queueing systems are given and performance parameters are explained. Then, one of the basic models of queueing systems, $M/M/c$ Queue is described.

2.1 Queueing theory / Queueing systems

Queueing systems is the mathematical study of waiting lines [19]. The primary quantities of interest in a queueing model are lengths and waiting times. In current literature queueing systems often appears as a subfield of operations research, which is the general field concerned with businesss / industrial decision making.

Queueing systems dates back to the beginning of the 20th century and the research conducted by A. K. Erlang (1878–1929), who worked for the telecom company in Copenhagen and studied telephone traffic [13]. Telecominications remain an active area of application of queueing theory [13]. The early works of A.K. Erlang already included the main elements of queueing theory [13]: arrival process, service process, departure process and waiting of customers, servers, etc. The next section explains some of the terminology of queueing systems.

2.1.1 Mathematical Description of Queueing Systems

Here are some of the main concepts that appear in queueing theory:

- **Arrival process:** The stochastic definition of customer arrivals. It may depend on the current state of the system, including the number of customers in the system. In basic queueing models where the arrival times are independent, the arrival process is described by the interarrival time distribution.

- **Service process:** The stochastic definition of customer service. Customer service may also depend on the current state of the system; the simplest and most commonly used assumption on service times is that of independent and identically distributed (iid) service times.

- **System structure:** The resources of the queueing system, which includes the number of servers and the size of the waiting room, and their interconnection.
- **Service discipline:** A set of rules that assigns the service order and service mode of customers. The most known service orders are FCFS (first come, first served), FIFO (first in, first out), and LIFO (last in, first out). All customers can be served in parallel. This service discipline is known as Processor Sharing (PS). Service order has a significant role when different types of customers reach to the system.
- **Performance parameters:** One should take into consideration and compute some performance parameters to construct a detailed model of a queueing system. The most known performance parameters are system utilization, mean and distribution of waiting time, loss probability, etc.

2.2 Classification of Basic Queueing Systems

In 1953, D. G. Kendall introduced a classification and a standard notation of basic queueing systems. The current version of this set of notations consists of six elements – *A/B/c/d/e-x*, where [13]

- A is the type of arrival process,
- B is the type of service process,
- c is the number of servers,
- d is the maximum number of customers in the system,
- e is the population of of customers,
- x describes the service discipline.

In basic queueing systems, *A* and *B* take one of these options:

1. M – memoryless, attributes to exponentially distributed arrival or service time,
2. D – constant arrival or service time,
3. E_r – order r Erlang distributed arrival or service time,
4. H_r – order r hyperexponentially distributed arrival or service time,
5. G or GI – i.i.d. random arrival or service time with any general distribution;

d, e, and x are omitted if they receive their default values: $d = \infty$ an infinite system capacity, $e = \infty$ an infinite customer population, and $x = FCFS$ (first come first served) service discipline.

2.3 Queueing System Performance Parameters

The optimal operation of queueing systems can be investigated by numerous performance parameters. Some important of them are as follows: [13]. Customer loss probability, Waiting time distribution, Mean waiting time, Distribution of a server's busy period and Queue length distribution. We refer the reader to [13] for definitions and in depth analysis of these measures for a broad range of queueing systems. In the present thesis, only the last one of these will be used, what follows is a review of the queue length distribution.

2.3.1 Queue length distibution and expected queue length

Let $L(t),\ t \geq 0$, be the number of customers in the system which includes customers in the servers taking service and also waiting customers in the buffer at the time t. Let $\bar{L}_k(t)$ be the period of time in $(0, t)$ so that there are k customers in the system:

$$\bar{L}_k(t) = \frac{1}{t}\int_0^t I_{\{L(s)=k\}}\, ds. \tag{2.1}$$

If

$$p_k = \lim_{t\to\infty} \bar{L}_k(t),\ k \geq 0, \tag{2.2}$$

exists, then $p_k,\ k \geq 0$ is the time average queue length distribution. If L has a stationary distribution μ and it is stable, the Ergodic Theorem (see, e.g., [2, Theorem 1.6.4, page 50]) implies

$$p_k = \mathbb{P}_\pi(L_t = k).$$

A very important function of the queue length distribution is the expected queue length:

$$\mathbb{E}_\pi[L_t] = \sum_{k=1}^{\infty} kp_k;$$

the average unit cost per unit time measure that will be used in the current work is a direct function of this quantity.

2.4 $M/M/c$ queue

The most classical queueing systems models are the $M/M/1$, $M/M/c$, $M/G/1$ and $G/M/1$ queues, all of which are thoroughy treated in many references, including [13]. Of these, the most relevant to the data center models studied in this thesis is the $M/M/c$ model; the $M/M/c/Setup$ and $M/M/c/dSetup$ models of the next two chapters can be seen as natural extensions of this model. The present section reviews this model. The $M/M/c$ model corresponds to the following data center: c servers, jobs arriving to the data center following a constant rate λ; the service times are iid and exponentitally

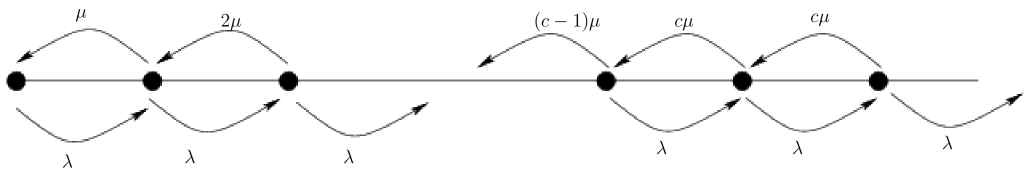

Figure 2.1: The dynamics of the $M/M/c$ queue.

distributed with rate μ. The $M/M/c$ model of data centers assumes that the servers are always kept on, so there is no setup mode.

If none of the servers in the system is available at an arrival, then the new customer rests in the buffer. When i $(1 \leq i \leq c)$ servers are not available, service processes of these servers happens at the same time. Thanks to the memoryless property of the service time distribution, other service times are independent exponentially distributed random variables as well. The minimum of i independent exponentially distributed random variables with $i\mu$.

The stationary distribution and the energy consumption rate for this system can be computed explicitly. The formulas for the stationary distribution of the $M/M/c$ queue system are well known, see, e.g., [13, Chapter 6]. For sake of completeness and ease of reference, we give a derivation of these formulas in this chapter. We will use these results in our comparison in Chapter 5.

The state space for this model is $\mathbb{N} = \{0, 1, 2, 3, ...\}$; the state process represents the number of customers in the system. To compute the stationary distribution it suffices to consider this system at service completions and arrrivals. Let X_n denote this random walk, and let $F_n = \sigma(X_1, X_2, ..., X_n)$; its dynamics are as follows:

$$X_{n+1} = X_n + I_n$$

with

$$P(I_{n+1} = v|F_n) = P(I_{n+1} = v|X_n) = p(X_n, v),$$

where

$$p(x,v) = \begin{cases} \frac{\lambda}{c\mu+\lambda}, & v = 1, \\ \frac{\min(c,x)}{c\mu+\lambda}, & v = -1, \\ \frac{c-\min(x,x)}{c\mu_\lambda}, & v = 0, \end{cases}$$

$x \in \mathbb{N}$.

These dynamics are shown in Figure 2.1

Then, the stationary distribution π_{ON} is the solution of the following equation:

$$\pi_{ON}(x) = \sum_{v \in \{-1,0,1\}} \pi_{ON}(x - v)p(x, v), x \in \mathbb{N}. \tag{2.3}$$

Proposition 2.1. *The stationary distribution π_{ON} of the always-ON system is*

$$\pi_{ON}(x) = \begin{cases} \pi_{ON}(0)\frac{\lambda^x}{x!\mu^x}, & \text{if } x \leq c, \\ \pi_{ON}(0)\frac{\lambda^x}{c!c^{x-c}\mu^x}, & \text{if } x > c\ . \end{cases} \tag{2.4}$$

Proof. For $x = 0$, (2.3) reads

$$\pi_{ON}(0) = \frac{\mu}{c\mu + \lambda}\pi_{ON}(1) + \frac{c\mu}{c\mu + \lambda}\pi_{ON}(0),$$

which implies

$$\pi_{ON}(1) = \frac{\lambda}{\mu}\pi_{ON}(0); \tag{2.5}$$

this gives $\pi_{ON}(1)$ in terms of $\pi_{ON}(0)$. Now summing both sides of (2.3) for $x = 0, 1$ gives

$$\begin{aligned} \pi_{ON}(0) + \pi_{ON}(1) &= \pi_{ON}(0) + \pi_{ON}(1)\frac{c\mu}{c\mu + \lambda} + \frac{2\mu}{c\mu + \lambda}\pi_{ON}(2), \\ \frac{2\mu}{c\mu + \lambda}\pi_{ON}(2) &= \frac{\lambda}{c\mu + \lambda}\pi_{ON}(1), \\ \pi_{ON}(2) &= \frac{\lambda}{2\mu}\pi_{ON}(1). \end{aligned}$$

Repeating the same argument with $x = 0, 1, 2, ..., k$,

$$\pi_{ON}(k) = \frac{\lambda}{k\mu}\pi_{ON}(k-1), \tag{2.6}$$

for $k < c$ and

$$\pi_{ON}(k) = \frac{\lambda}{c\mu} \tag{2.7}$$

for $k \geq c$. This completes the proof. □

The explicit formula (2.4) gives the following formula for $\pi_{ON}(0)$, the probability of an empty system:

Proposition 2.2. *The stationary probability that the $M/M/c$ system is empty equals*

$$\pi_{ON}(0) = \frac{1}{\sum_{x=0}^{c-1}\frac{r^x}{x!} + \frac{r^c}{c!}\frac{1}{1-\rho}}. \tag{2.8}$$

Proof. By definition $\pi_{ON}(\mathbb{N}) = 1$, i.e.,

$$\sum_{x=0}^{\infty}\pi_{ON}(x) = 1.$$

Substituting (2.4) in the above equation gives

$$\pi_{ON}(0)\left(\sum_{x=0}^{c-1}\frac{r^x}{x!}+\frac{r^c}{c!}\sum_{x=c}^{\infty}\rho^{x-c}\right)=1,$$
$$\pi_{ON}(0)\left(\sum_{x=0}^{c-1}\frac{r^x}{x!}+\frac{r^c}{c!}\frac{1}{1-\rho}\right)=1,$$

which yields (2.8). □

Let B_n denote the number of busy servers in the $M/M/c$ system, i.e.

$$B_n = \min c, X_n.$$

For the computation of the energy consumption rate of an $M/M/c$ (see (1.1)) system we need to know the expected number $\mathbb{E}_{\pi_{ON}}[B_0]$ of busy servers under the stationary distribution. Most remarkably, although the computation of $\pi_{ON}(0)$ requires a sum of c terms, $\mathbb{E}_{\pi_{ON}}[B_0]$ has a very simple formula:

Proposition 2.3. *The expected number of busy servers under the stationary distribution equals*

$$\mathbb{E}_{\pi_{ON}}[B_0] = r. \tag{2.9}$$

Proof. By definition

$$\mathbb{E}_{\pi_{ON}}[B_0] = \mathbb{E}_{\pi_{ON}}[\min c, X_0] = \sum_{x=1}^{\infty}\min c, x\pi_{ON}(x). \tag{2.10}$$

For $1 \le x < c$, (2.6) gives

$$\min c, x\pi_{ON}(x) = x\pi_{ON}(x) = x\frac{\lambda}{\mu x}\pi_{ON}(x-1) = r\pi_{ON}(x-1).$$

For $x \ge c$, (2.7) gives

$$\min c, x\pi_{ON}(x) = c\pi_{ON}(x) = c\frac{\lambda}{c\mu}\pi_{ON}(x-1) = r\pi_{ON}(x-1).$$

Substituting these in (2.10) gives

$$\mathbb{E}_{\pi_{ON}}[B_0] = \sum_{x=1}^{\infty}\min(c,x)\pi_{ON}(x). \sum_{x=1}^{\infty} r\pi_{ON}(x-1) = r,$$

which proves (2.9). □

In the $M/M/c$ model, the servers are assumed to be always in ON mode; if a server is ON but not serving, i.e., it is idle, it is assumed to consume energy at a constant rate C_i, Therefore, deviating slightly from (1.1) to take into account the difference between idle and running ON servers, we get the following formula for the average energy consumption rate of the $M/M/c$ system per unit time:

$$E_{\infty}^{MMC} = c\rho C_a + c(1-\rho)C_i.$$

Following [17] we will assume $C_i = 0.6C_a$.

CHAPTER 3

Exact Solutions for M/M/c/Setup Queues

Phung-Duc [17] analyzes the same $M/M/c/Setup$ model with ON-OFF policy and, the main purpose of the paper is computing exact solutions of the joint stationary queue length distribution.

To achieve this goal, the author first presents the model and uses two methods which are generating function approach and matrix analytic method. Furthermore, comparison of methods including renewal reward approach, presentation of some variant models such as $M/M/c/Setup/Sleep$ and $M/M/c/Setup/Delayoff$ and performance measures are also included in the paper. In this chapter, we will summarize his work in detail.

3.1 M/M/c/Setup Model and Notations

- **Arrival:** Jobs arrive at the system based on Poisson process with rate λ.

- **Service:** Service time has the exponential distribution with mean $\frac{1}{\mu}$.

  After the service is completed, if there are still waiting jobs in the queue then a server takes to serve. Otherwise, server is turned off. After the arrival of a job, if there is an OFF server then it is turned on and that job is placed in the buffer. But a server needs time to setup and be active to serve jobs.

- **Setup:** Setup time also has the exponential distribution with mean $\frac{1}{\alpha}$.

  Consider there are two jobs in the system. If the service of one job is finished before the setup of a server, then the waiting job goes to an active server and the server in setup process is turned off.

  Let j stands for the number of customers and i stands for the number of active servers in the system.

  The number of servers in setup process is $\min j - i; c - i$ and a server is in either BUSY or OFF or SETUP situation. The number of active servers is smaller than or equal to the number of jobs in the system ($i \leq j$).

We will denote the continuous time $M/M/c/Setup$ system by $\mathcal{X}_t$; $\mathcal{X}_t(1)$ will denote the total number of customers in the system at time t, and $\mathcal{X}_t(2)$ will denote the number of running servers. $\mathcal{X}$ is a piecewise constant process. Let T_n denote the n^{th} jump time of this system. That all times (interarrival, service and setup times) are exponentially distributed and iid and the PASTA ("Poisson arrivals see time averages") property [11, page 264] of the system implies that the discrete time process embedded random walk $X_n = \mathcal{X}_{T_n}$ is a constrained random walk on $\mathbb{Z}_+^2$ with the following dynamics:

$$X_{n+1} = X_n + Y_n \tag{3.1}$$

where Y_n's distribution depends on X_n as follows; here we set $S = c(\alpha + \mu) + \lambda$. The increments Y_n, $n = 1, 2, 3, ...$, take values in

$$\mathcal{Y} = \{(0,1), (1,0), (-1,0), (-1,-1), (0,0)\}$$

with probabilities:

$$\begin{aligned}
P(Y_n = (0,1)|X_n = x) &= \frac{\min x_1 - x_2, c - x_2\alpha}{S}, \\
P(Y_n = (1,0)|X_n = x) &= \frac{\lambda}{S}, \\
P(Y_n = (-1,0)|X_n = x) &= 1_{(0,\infty)}(x_1 - x_2)\frac{x_2\mu}{S}, \\
P(Y_n = (-1,-1)|X_n = x) &= 1_{\{0\}}(x_1 - x_2)\frac{x_2\mu}{S},
\end{aligned}$$

and

$$r(x) \doteq P(Y_n = (0,0)|X_n = x)$$

is set so that $\sum_{y\in\mathcal{Y}} P(Y_n = y|X_n = x) = 1$. The increment $(0,0)$ corresponds to a "null-event."

Null events are added to the system so that the process that counts the number of events in the system is a Poisson process with constant rate S.

As with $\mathcal{X}$, the first component $X_n(1)$ of X denotes the total number of jobs in the system and the second component $X_n(2)$ of X denotes the number of servers currently handling a job. Thus $X_n(1) - X_n(2)$ is the number of jobs waiting for service and $\min c - X_n(2), X_n(1) - X_n(2)$ is the number of servers in setup mode.

We assume that the initial position $X_0 = x$ satisfies $x_1 \geq x_2$; i.e., initially the total number of jobs int the system is greater than or equal to the total number of servers handling a job. This and the dynamics of X imply

$$X_n \in \mathcal{D} \doteq \{x : x_1 \geq x_2, x_1, x_2 \geq 0\}.$$

These dynamics are depicted in Figure 3.1 (this illustration shows only the nonzero increments and the numerators of the jump probabilities).

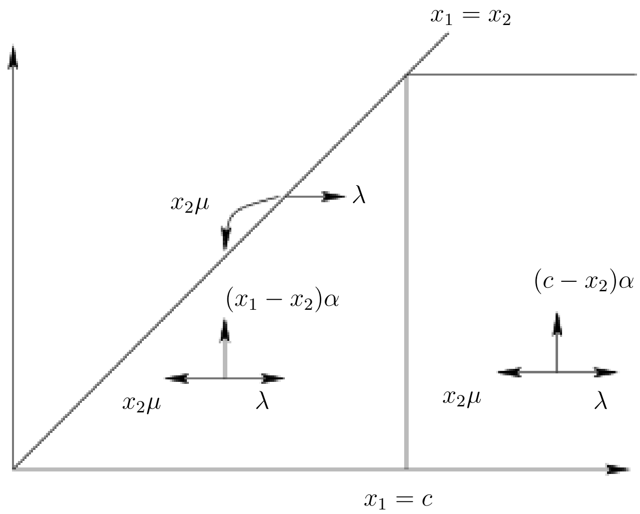

Figure 3.1: The dynamics of X.

3.2 Average power consumption and total energy cost for the $M/M/c/Setup$ system

For the energy consumption of the data center, [17] makes the following assumptions: an active server [a server in setup mode] consumes energy at a constant rate $C_a > 0$ [$C_s > 0$].

Remember that $\mathcal{X}_t(1)$ denotes the number of customers in the system and $\mathcal{X}_t(2)$ denotes the number of active servers in the system at time $t \in [0, \infty)$ (i.e., in continuous time). Then the number of servers S_t in setup mode at time t is

$$S_t = \min \mathcal{X}_t(1) - \mathcal{X}_t(2), \mathcal{X}_t(2)).$$

The average unit time energy consumption rate of the data center is

$$\begin{aligned} E_\infty &\doteq \mathbb{E}_\pi[\mathcal{X}_t(2)C_a + R_t C_s] && (3.2) \\ &\doteq \mathbb{E}_\pi[\mathcal{X}_t(2)C_a + \min \mathcal{X}_t(1) - \mathcal{X}_t(2), cC_s] && (3.3) \\ &= C_a \mathbb{E}_\pi[\mathcal{X}_t(2)] + C_s \mathbb{E}_\pi \min \mathcal{X}_t(1) - \mathcal{X}_t(2), c - \mathcal{X}_t(2)] \end{aligned}$$

As we have already noted, the PASTA ("Poisson arrivals see time averages") property [11, page 264] of the system implies that to compute the stationary measure of the continuous time system $\mathcal{X}_t$ it suffices to compute that of the discrete random walk X defined above. (The Poisson "arrival" process in this case is the process that jumps each time an event (including null ones) occurs in the system). The explicit computation of this invariant distribution is reviewed in the next section.

3.3 Stationary distribution

The stability condition for X is given in [17] is $\lambda < c\mu$. Under this stability condition X has a stationary distribution π which satisfies the following set of equations:

$$\begin{aligned} &S(1-r(x)\pi(x_1,x_2)) \\ &\quad = \lambda\pi(x_1-1,x_2) + x_2\mu\pi(x_1+1,x_2) \\ &\quad\; + \min(c-x_2+1, x_1-x_2+1)\alpha\pi(x_1,x_2-1), \quad x_1 > x_2, \\ &S(1-r(x))\pi(x_2,x_2) \\ &\quad = \pi(x_2+1,x_2+1)\mu(x_2+1) + \pi(x_2+1,x_2)x_2\mu, \end{aligned} \tag{3.4}$$

$0 \leq x_2 \leq c$.
The mathematical goal of [17] is the analytic solution of these equations; [17] uses two different methods for this purpose, the generating function approach and the matrix analytic method. The analysis in [17] based on these methods are reviewed in sections below.

Remember that we will use E_∞ of (3.2) to approximate the energy consumption rate of the data center. To compute E_∞ we only need $E_\pi[\mathcal{X}_t(2)]$ and $\mathbb{E}_\pi \max \mathcal{X}_t(2) - \mathcal{X}_t(1), c]$. The PASTA property of the system implies that these expectations equal

$$\mathbb{E}_\pi[X_1(2)], \mathbb{E}_\pi \max X_1(1) - X_1(2), c]$$

where X is the random walk (3.1) and π is its stationary distribution. Because X is a simple two dimensional constrained random walk, one can easily approximate the above expectations using simulation and the law of large numbers:

$$\mathbb{E}_\pi[X_1(2)] \approx \frac{1}{K}\sum_{k=1}^{K} X_k(2), \tag{3.5}$$

$$\mathbb{E}_\pi[\max X_1(1) - X_1(2), c] \approx \frac{1}{K}\sum_{k=1}^{K} \max X_k(1) - X_k(2), c. \tag{3.6}$$

In Chapter 5 below we will use this approximation in the comparison of the energy consumptions of the $M/M/c/Setup$ and the $M/D/c/Setup$ systems.

Note that the number of waiting jobs is $j - i$ in the state $(i; j)$.

3.4 Generating Function Approach (Section 3 in [17])

Generating function approach introduces exact closed form expressions for the joint stationary queue length distribution and the conditional decomposition formula. Explicit expressions for the joint stationary queue length distribution, generating functions and factorial moments of any order are derived.

3.4.1 Explicit Expressions

In the matter of explicit expressions, balance equations for cases including $i = 0$, $i = 1$, the general case $i = 2, 3, ..., c-1$, and $i = c$ are considered.

Denote $\Pi_i(z)$ for the partial generating functions of the number of waiting jobs, i.e.,

$$\Pi_i(z) = \sum_{j=1}^{\infty} \pi_{i,j} z^{j-i}, \quad i = 0, 1, \ldots, c.$$

- **Case *i = 0***

The balance equations for the case as follows:

$$\lambda \pi_{0,0} = \mu \pi_{1,1}, \qquad j = 0, \tag{3.7}$$

$$(\lambda + j\alpha)\pi_{0,j} = \lambda \pi_{0,j-1}, \qquad j = 1, 2, ..., c-1, \tag{3.8}$$

$$(\lambda + c\alpha)\pi_{0,j} = \lambda \pi_{0,j-1}, \qquad j \geq c. \tag{3.9}$$

Let $\widehat{\Pi}(z) = \sum\limits_{j=c}^{\infty} \pi_{0,j} z^j$. Multiplying (3.9) by z^j and summming over $j \geq c$, then we have

$$\widehat{\Pi}_0(z) = \frac{\lambda \pi_{0,c-1} z^c}{\lambda + c\alpha - \lambda z} = z^c \frac{A_{0,0}}{\hat{z}_0 - z}, \quad \Pi_0(z) = \sum_{j=0}^{c-1} \pi_{0,j} z^j + \widehat{\Pi}_0(z) \tag{3.10}$$

where

$$A_{0,0} = \pi_{0,c-1}, \quad \hat{z}_0 = \frac{\lambda + c\alpha}{\lambda}.$$

Equation 3.8 gives

$$\pi_{0,j} = \pi_{0,0} \prod_{i=0}^{j} \frac{\lambda}{\lambda + j\alpha}, \quad j = 1, 2, \ldots, c-1.$$

First equation of 3.10 gives

$$\pi_{0,j} = \frac{\lambda \pi_{0,c-1}}{\lambda + c\alpha} \left(\frac{\lambda}{\lambda + c\alpha} \right)^{j-c}, \quad j \geq c.$$

For the factorial moments, differentiate (3.10) n times and then

$$\widehat{\Pi}_0^{(n)}(1) = \frac{\lambda}{c\alpha} \widehat{\Pi}_0^{(n-1)}(1) + \frac{\lambda}{c\alpha} \pi_{0,c-1} (c-n)_n,$$

$$\widehat{\Pi}_0^{(n)}(1) = \sum_{j=0}^{c-1} \pi_{0,j}(j-n+1)_n + \widehat{\Pi}_0^{(n)}(1),$$

for $n \in \mathbb{N}$. Note that $(\phi)_n$ is called as the *Pochhammer symbol* and see Appendix for more information.

- **Case *i = 1***

The balance equations for the case are

$$(\lambda+\mu)\pi_{1,1} = \alpha\pi_{0,1} + \mu\pi_{1,2} + 2\mu\pi_{2,2},, \tag{3.11}$$

$$(\lambda+\mu+(j-1)\alpha)\pi_{1,j} = j\alpha\pi_{0,j} + \lambda\pi_{1,j-1} + \mu\pi_{1,j+1}, \qquad 2 \le j \le c-1, \tag{3.12}$$

$$(\lambda+\mu+(c-1)\alpha)\pi_{1,j} = c\alpha\pi_{0,j} + \lambda\pi_{1,j-1} + \mu\pi_{1,j+1}, \qquad j \ge c. \tag{3.13}$$

Let $\widehat{\Pi}_1(z) = \sum_{j=c}^{\infty} \pi_{1,j} z^{j-1}$ and then we have $\Pi_1(z) = \sum_{j=1}^{c-1} \pi_{1,j} z^{j-1} + \widehat{\Pi}_1(z)$. Multiply (3.13) by z^{j-1}, sum over $j \ge c$, and rearrange the equation

$$[(\lambda+\mu+(c-1)\alpha)z - \lambda z^2 - \mu]\widehat{\Pi}_1(z) = c\alpha\widehat{\Pi}_0(z) + \lambda\pi_{1,c-1}z^c - \mu\pi_{1,c}z^{c-1}. \tag{3.14}$$

Define $f_1(z) = (\lambda+\mu+(c-1)\alpha)z - \lambda z^2 - \mu$. Then, $f_1(z)$ has two roots which are

$$z_1 = \frac{\lambda+\mu+(c-1)\alpha - \sqrt{(\lambda+\mu+(c-1)\alpha)^2 - 4\lambda\mu}}{2\lambda},$$

$$\hat{z}_1 = \frac{\lambda+\mu+(c-1)\alpha + \sqrt{(\lambda+\mu+(c-1)\alpha)^2 - 4\lambda\mu}}{2\lambda}.$$

Substitute $z = z_1$ into (3.14), then we find

$$\pi_{1,c} = \frac{c\alpha\widehat{\Pi}_0(z_1) + \lambda\pi_{1,c-1}z_1^c}{\mu z_1^{c-1}}.$$

Use mathematical induction to derive a recursive formula for the case $i = 1$.

Lemma 3.1.

$$\pi_{1,j} = a_j^{(1)} + b_j^{(1)}\pi_{1,j-1}, \quad 2 \le j \le c, \tag{3.15}$$

where

$$a_j^{(1)} = \frac{j\alpha\pi_{0,j}}{\lambda+\mu+(j-1)\alpha - \mu b_{j+1}^{(1)}}, \quad b_j^{(1)} = \frac{\lambda}{\lambda+\mu+(j-1)\alpha - \mu b_{j+1}^{(1)}},$$

for $j = c-1, c-2, \ldots, 2, 1$. *Moreover,*

$$0 < a_j^{(1)}, \quad 0 < b_j^{(1)} < \frac{\lambda}{\mu}, \quad j = 1, 2 \ldots, c.$$

The generating function $\widehat{\Pi}_1(z)$ is shown as below:

$$\widehat{\Pi}_1(z) = z^{c-1}\left(\frac{A_{1,0}}{\hat{z}_0 - z} + \frac{A_{1,1}}{\hat{z}_1 - z}\right), \tag{3.16}$$

where

$$A_{1,0} = \frac{A_{0,0}\hat{z}_0}{f_1(\hat{z}_0)}, \quad A_{1,1} = -\frac{A_{0,0}\hat{z}_0}{f_1(\hat{z}_0)} + \pi_{1,c-1}.$$

To find the partial factorial moments, take the derivative of (3.14) and then put $z = 1$ into that equation. Hence, we have a recursive formula which is

$$\begin{aligned}\widehat{\Pi}_1^{(n)}(1) = \frac{c}{c-1}\widehat{\Pi}_0^{(n)}(1) + \frac{n(\lambda - \mu - (c-1)\alpha)\widehat{\Pi}_1^{(n-1)}(1) + \lambda n(n-1)\widehat{\Pi}_1^{(n-2)}(1)}{(c-1)\alpha} \\ + \frac{\lambda\pi_{1,c-1}(c-n+1)_n - \mu\pi_{1,c}(c-n)_n}{(c-1)\alpha}.\end{aligned} \tag{3.17}$$

- **General Case, where *i* = 2, 3, ..., *c*-1**

The balance equations for the case are

$$(\lambda + i\mu)\pi_{i,i} = \alpha\pi_{i-1,i} + i\mu\pi_{i,i+1} + (i+1)\mu\pi_{i+1,i+1}, \quad j = i, \tag{3.18}$$

$$(\lambda+i\mu+(j-i)\alpha)\pi_{i,j} = \lambda\pi_{i,j-1}+(j-i+1)\alpha\pi_{i-1,j}+i\mu\pi_{i,j+1}, \; i+1 \le j \le c-1, \tag{3.19}$$

$$(\lambda + i\mu + (c-i)\alpha)\pi_{i,j} = \lambda\pi_{i,j-1} + (c-i+1)\alpha\pi_{i-1,j} + i\mu\pi_{i,j+1}, \quad j \ge c. \tag{3.20}$$

Let $\widehat{\Pi}_i(z) = \sum_{j=c}^{\infty} \pi_{i,j}z^{j-i}$ and then we have $\Pi_i(z) = \sum_{j=i}^{c-1} \pi_{i,j}z^{j-i} + \widehat{\Pi}_i(z)$. The rest of the process is similar to the case $i = 1$:

$$\begin{aligned}&[(\lambda + i\mu + (c-i)\alpha)z - \lambda z^2 - i\mu]\,\widehat{\Pi}_i(z) = (c-i+1)\alpha\widehat{\Pi}_{i-1}(z) \\ &+\lambda\pi_{i,c-1}z^{c-i+1} - i\mu\pi_{i,c}z^{c-i}.\end{aligned} \tag{3.21}$$

Define $f_i(z) = (\lambda + i\mu + (c-i)\alpha)z - \lambda z^2 - i\mu$. Its two roots are

$$z_i = \frac{\lambda + i\mu + (c-i)\alpha - \sqrt{(\lambda + i\mu + (c-i)\alpha)^2 - 4i\lambda\mu}}{2\lambda},$$

$$\hat{z}_i = \frac{\lambda + i\mu + (c-i)\alpha + \sqrt{(\lambda + i\mu + (c-i)\alpha)^2 - 4i\lambda\mu}}{2\lambda}.$$

Substitute $z = z_i$ into (3.21), then

$$\pi_{i,c} = \frac{(c-i+1)\alpha\widehat{\Pi}_{i-1}(z_i) + \lambda\pi_{i,c-i}z_i^{c-i+1}}{i\mu z_i^{c-i}}. \tag{3.22}$$

Equations (3.19) and (3.22) together bring $\pi_{i,j}$ $(i+1 \leq j \leq c)$.

Lemma 3.2.

$$\pi_{i,j} = a_j^{(i)} + b_j^{(i)} \pi_{i,j-1}, \quad i+1, i+2, \ldots, c, \tag{3.23}$$

where

$$a_c^{(i)} = \frac{(c-i+1)\alpha \widehat{\Pi}_{i-1}(z_i)}{i\mu z_i^{c-i}}, \quad b_j^{(i)} = \frac{\lambda z_i}{i\mu}$$

and

$$a_j^{(i)} = \frac{(j-i+1)\alpha\pi_{i-1,j} + i\mu a_{j+1}^{(i)}}{\lambda + i\mu + (j-i)\alpha - i\mu b_{j+1}^{(i)}}, \quad b_j^{(i)} = \frac{\lambda}{\lambda + i\mu + (j-i)\alpha - i\mu b_{j+1}^{(i)}}$$

for $j = c-1, \ldots, i+1$. *Moreover,*

$$0 < a_j^{(i)}, \quad 0 < b_j^{(i)} < \frac{\lambda}{i\mu}.$$

Also, the generating function $\widehat{\Pi}_i(z)$ *is shown as below:*

$$\widehat{\Pi}_i(z) = z^{c-i} \left(\sum_{j=0}^{i} \frac{A_{i,j}}{\hat{z}_j - z} \right), \tag{3.24}$$

where

$$A_{i,j} = \frac{A_{i-1,j}\hat{z}_j}{f_i(\hat{z}_j)}, \quad A_{i,i} = -(c-i+1)\alpha \sum_{j=0}^{i-1} \frac{A_{i-1,j}\hat{z}_j}{f_i(\hat{z}_j)} + \pi_{i,c-1}.$$

To find the partial factorial moments, take the derivative of (3.21) *n* times and then put $z = 1$ into that equation. Hence, we have a recursive formula which is

$$\begin{aligned}\widehat{\Pi}_i^{(n)}(1) = \frac{c-i+1}{c-i}\widehat{\Pi}_{i-1}^{(n)}(1) + \frac{n(\lambda - \mu - (c-i)\alpha)\widehat{\Pi}_i^{(n-1)}(1) + \lambda n(n-1)\widehat{\Pi}_i^{(n-2)}(1)}{(c-i)\alpha} \\ + \frac{\lambda\pi_{i,c-1}(c--i+2-n)_n - i\mu\pi_{i,c}(c--i+1-n)_n}{(c-i)\alpha}.\end{aligned} \tag{3.25}$$

- **Case *i* = *c***

The last case is a little bit different than the others. The balance equations for the case are

$$(\lambda + c\mu)\pi_{c,c} = \alpha\pi_{c-1,i} + c\mu\pi_{c,c+1}, \quad j = c, \tag{3.26}$$

$$(\lambda + c\mu)\pi_{c,j} = \lambda\pi_{c,j-1} + \alpha\pi_{c-1,j} + c\mu\pi_{c,j+1}, j \leq c+1, \tag{3.27}$$

Let $\widehat{\Pi}_c(z) = \sum\limits_{j=c}^{\infty} \pi_{c,j} z^{j-c}$ and we have $\Pi_c(z) = \widehat{\Pi}_c(z)$. After the multiplication of 3.27 and summation over $j \leq c$, we find

$$(\lambda + c\mu)\widehat{\Pi}_c(z) = \frac{\alpha}{z}\widehat{\Pi}_{c-1}(z) + \lambda z \widehat{\Pi}_c(z) + \frac{c\mu}{z}(\widehat{\Pi}_c(z) - \pi_{c,c}) \tag{3.28}$$

Then we have

$$\widehat{\Pi}_c(z) = \frac{\alpha\widehat{\Pi}_{c-1}(z) - c\mu\pi_{c,c}}{z-1}\frac{1}{c\mu - \lambda z} = \frac{\alpha\Big(\widehat{\Pi}_{c-1}(z) - \widehat{\Pi}_{c-1}(1)\Big)}{z-1}\frac{1}{c\mu - \lambda z}, \tag{3.29}$$

where $\alpha\widehat{\Pi}_{c-1}(1) = c\mu\pi_{c,c}$ is used in (3.29).

Define $f_c(z) = (\lambda + c\mu)z - \lambda z^2 - c\mu$, substitute $\widehat{\Pi}_{c-1}(z)$ in terms of (3.24) with $i = c-1$ and then

$$\widehat{\Pi}_c(z) = \left(\sum_{j=0}^{c} \frac{A_{c,j}}{\hat{z}_j - z} \right) \tag{3.30}$$

where $\hat{z}_c = \dfrac{c\mu}{\lambda}, \quad A_{c,j} = \dfrac{A_{c-1,j}}{\hat{z}_c - 1}, \quad A_{c,c} = -\sum\limits_{j=0}^{c-1} \dfrac{A_{c-1,j}\hat{z}_j}{f_c(\hat{z}_j)}.$

To find the partial factorial moments, take the derivative of (3.28) n times, rearrange the result with using l'Hopital's Rule and then put $z = 1$ into the equation. Hence, we have a recursive formula which is

$$\widehat{\Pi}_c^{(n)}(1) = \frac{\alpha\widehat{\Pi}_{c-1}^{(n+1)}(1) + \lambda n(n-1)\widehat{\Pi}_c^{(n-2)}(1) + 2\lambda\widehat{\Pi}_c^{(n-1)}(1)}{(n+1)(c\mu - \lambda)}. \tag{3.31}$$

Note that $\pi_{0,0}$ is computed with the normalization condition such that $\Pi_0(1) + \Pi_1(1) + \ldots \Pi_c(1) = 1$.

Now, we can determine explicit result for the factorial moments and joint stationary distribution, because explicit expressions for the generating functions are known.

Furthermore, the generationg function for the number of waiting jobs $\Pi(z)$ is

$$\Pi(z) = \sum_{i=0}^{c} \Pi_i(z).$$

3.4.2 Conditional Stochastic Decomposition

We have known that

$$\Pi_c(z) = \frac{\alpha(\Pi_{c-1}(z) - \pi_{c-1,c-1}) - c\mu\pi_{c,c}}{(z-1)(c\mu - \lambda z)},$$

$$\Pi_c(1) = \frac{\alpha \Pi'_{c-1}(1)}{c\mu - \lambda z}.$$

Denote $Q^{(c)}$ as the conditional queue length with the condition that c servers are busy. Then we have

$$\mathbf{P}(Q^{(c)} = i) = \mathbf{P}(N(t) = i + c \mid C(t) = c).$$

Denote $P_c(z)$ as the generating function of $Q^{(c)}$. It can be derived such that

$$\begin{aligned} P_c(z) = \frac{\Pi_c(z)}{\Pi_c(1)} &= \frac{\alpha(\Pi_{c-1}(z) - \pi_{c-1,c-1}) - c\mu\alpha\Pi'_{c-1}(1)\pi_{c,c}}{(z-1)} \frac{1-\rho}{1-\rho z} \\ &= \frac{\sum_{i=0}^{\infty} \left(\sum_{j=i+1}^{\infty} \pi_{c-1,c-1+j} \right) z^i}{\Pi'_{c-1}(1)} \frac{1-\rho}{1-\rho z}, \end{aligned}$$

where $c\mu\pi_{c,c} = \alpha(\Pi_{c-1}(1) - \pi_{c-1,c-1})$ is used in the second equality.

Note that $(1-\rho)/(1-\rho z)$ is the generating function of the number of waiting jobs in the $M/M/c$ system without setup times (i.e., $Q^{(c)}_{ON-IDLE}$).

Define

$$p_{c-1,i} = \frac{\sum_{j=i+1}^{\infty} \pi_{c-1,c-1+j}}{\Pi'_{c-1}(1)}, \quad i \in \mathbb{Z}_+.$$

Then we have

$$\sum_{j=i+1}^{\infty} \pi_{c-1,c-1+j} = \mathbf{P}(N(t) - C(t) > i \mid C(t) = c-1)\, \mathbf{P}(C(t) = c-1),$$

and

$$\Pi'_{c-1}(1) = \mathbf{E}[(N(t) - C(t) \mid C(t) = c-1)]\, \mathbf{P}(C(t) = c-1)$$

Hence,

$$p_{c-1,i} = \frac{\mathbf{P}(N(t) - C(t) > i \mid C(t) = c-1)\, \mathbf{P}(C(t) = c-1)}{\mathbf{E}[(N(t) - C(t) \mid C(t) = c-1)]}.$$

Note that $N(t) - C(t)$ is the number of waiting jobs for the last server in setup mode.

Thus, $p_{c-1,i}$ $(i = 0, 1, 2, \ldots)$ stands for distribution of the number of waiting customers in front of a random waiting customer provided that $c-1$ servers are active and the last server is in setup mode. Therefore, our decomposition result is

$$Q^{(c)} = Q^{(c)}_{ON-IDLE} + Q_{Res}$$

where Q_{Res} stands for the number of extra jobs because of the setup time.

3.5 Matrix Analytic Method (Section 4 in [17])

The matrix analytic approach is based on quasi-birth-and-death process (QBD) and is used to find a recursive algorithm for the stationary distribution.

3.5.1 QBD Formulation

The infinitesimal of $X(t)$ is shown by

$$Q = \begin{pmatrix} Q_0^{(0)} & Q_1^{(0)} & O & O & \dots \\ Q_{-1}^{(1)} & Q_0^{(1)} & Q_0^{(1)} & O & \dots \\ O & Q_{-1}^{(2)} & Q_0^{(2)} & Q_1^{(2)} & \dots \\ O & O & Q_{-1}^{(3)} & Q_0^{(3)} & \dots \\ \dots & \dots & \dots & \dots & \ddots \end{pmatrix},$$

where O is the zero matrix with an appropriate dimension. A Markov chain with this type of matrix is named as a level dependent quasi-birth-and-death process. Note that $Q_{-1}^{(i)}(i \geq c+1), Q_0^{(i)}(i \geq c)$ and $Q_1^{(i)}(i \geq c)$ are not dependent to i and we have

$$Q_{-1}^{(i)} = Q_{-1} = diag(0, \mu, \dots, c\mu), \quad Q_1^{(i)} = Q_1 = \lambda I,$$

$$Q_0^{(i)} = Q_0 = \begin{pmatrix} -q_0 & c\alpha & 0 & \dots & \dots & 0 \\ 0 & -q_1 & (c-1)\alpha & \ddots & \ddots & \vdots \\ 0 & 0 & -q_2 & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & -q_{c-1} & \alpha \\ 0 & \dots & \dots & 0 & 0 & -q_c \end{pmatrix},$$

where $q_j = \lambda + (c-j)\alpha + j\mu$.

Moreover, $Q_{-1}^{(i)}\ (i \leq c), Q_0^{(i)}\ (i \leq c-1)$ and $Q_1^{(i)}\ (i \leq c)$ are $(i+1)$ x i, $(i+1)$ x $(i+1)$ and $(i+1)$ x $(i+2)$ matrices, respectively. We have

$$Q_1^{(i)} = \begin{pmatrix} \lambda & 0 & \dots & \dots & 0 \\ 0 & \lambda & \ddots & \vdots & \vdots \\ \vdots & \ddots & \ddots & 0 & 0 \\ 0 & \ddots & 0 & \lambda & 0 \end{pmatrix}, \quad Q_{-1}^{(i)} = \begin{pmatrix} 0 & 0 & \dots & \dots & 0 \\ 0 & \mu & \ddots & \ddots & \vdots \\ 0 & 0 & & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & \ddots & (i-1)\mu \\ 0 & \ddots & \ddots & 0 & \imath\mu \end{pmatrix},$$

$$Q_0^{(i)} = \begin{pmatrix} -q_0^{(i)} & i\alpha & 0 & ... & ... & 0 \\ 0 & -q_1^{(i)} & (i-1)\alpha & \ddots & \ddots & \vdots \\ 0 & 0 & -q_2^{(i)} & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & -q_{i-1}^{(i)} & \alpha \\ 0 & ... & ... & 0 & 0 & -q_i^{(i)} \end{pmatrix},$$

where $q_j^{(i)} = (i-j)\alpha + j\mu \ (j = 0, 1, \ldots, i)$.

To find the stationary distribution let

$$\boldsymbol{\pi_i} = (\pi_{0,i}, \pi_{0,i}, \ldots, \boldsymbol{\pi}_{min(i,c),i}), \quad i \in \mathbb{Z}_+, \quad \pi = (\boldsymbol{\pi}_0, \boldsymbol{\pi}_1, \ldots).$$

Then $\boldsymbol{\pi}$ is the unique solution of

$$\boldsymbol{\pi} Q = \mathbf{0}, \quad \boldsymbol{\pi}\mathbf{e} = 1,$$

where **0** and **e** are a row vector of zeros and a column vector of ones with an appropriate size. By the matrix analytic method [16, 18],

$$\boldsymbol{\pi}_1 = \boldsymbol{\pi}_{i-1} R^i, \quad i \in \mathbb{N},$$

and $\boldsymbol{\pi}_0$ is the solution of the boundary equation such that

$$\boldsymbol{\pi}_0 (Q_0^{(0)} + R^{(1)} Q_{-1}^{(1)}) = \mathbf{0}, \quad \boldsymbol{\pi}_0 (I + R^{(1)} + R^1 R^{(2)} + \ldots)\, \mathbf{e} = 1.$$

$R^{(i)}, i \in \mathbb{N}$, is the minimal nonnegative solution of

$$Q_1^{(i-1)} + R^i Q_0(i) + R^i R^{i+1} Q_{-1}(i+1) = \mathbf{O}. \tag{3.32}$$

3.5.2 Rate Matrix

- **Homogeneous Part**

Notice that $Q_1^{(i-1)} = Q_1 \ (i \geq c), Q_0^{(i)} = Q_0 \ (i \geq c)$ and $Q_{-1}^{(i)} = Q_{-1} \ (i \geq c+1)$. Then we have $R^{(i)} = R$ for $i \geq c+1$ and R is the minimal nonnegative solution of

$$Q_1 + RQ_0 + R^2 Q_{-1} = \mathbf{O}.$$

Here, R is an upper diagonal matrix such that $R(i,j) = r_{i,j}$ if $j \geq i$ and $R(i,j) = 0$ when $j < i$ since Q_{-1}, Q_0 and Q_1 are upper diagonal matrix. This kind of QBD is examined in more general way in [21].

Consider the diagonal part of this quadratic equation, then we can determine

$$\lambda - (\lambda + i\mu + (c-i)\alpha) r_{i,i} + i\mu r_{i,i}^2 = 0, \quad i = 0, 1, \ldots, c-1, c.$$

Since R is the minimal nonnegative solution of the quadratic equation, we need to use the smaller root of $r_{i,i}$. Then we have

$$r_{i,i} = \frac{\lambda + i\mu + (c-i)\alpha - \sqrt{(\lambda + i\mu + (c-i)\alpha)^2 - 4i\lambda\mu}}{2i\lambda}, \quad i = 1, 2, \ldots, c-1, \tag{3.33}$$

and

$$r_{0,0} = \frac{\lambda}{\lambda + c\alpha}, \quad \frac{\lambda}{c\mu} < 1.$$

For the nondiagonal part $r_{i,j}$ $(j > i)$, compare the (i,j) element in the quadratic equation. Then we have

$$(c-j+1)\alpha r_{i,j-1} - (\alpha + (c-j)\alpha + j\mu)r_{i,j} + j\mu \sum_{k=i}^{j} = 0.$$

For $j = i+1$, we have

$$r_{i,i+1} = \frac{(c-i)\alpha r_{i,i}}{\lambda + (c-i-1)\alpha + (i+1)\mu - (i+1)\mu(r_{i,i} + r_{i+1,i+1})}, \quad i = 0, 1, \ldots, c-1.$$

For $j = i+2$, we have

$$r_{i,i+2} = \frac{(c-i-1)\alpha r_{i,i+1} + (i+2)\mu r_{i,i+1} r_{i+1,i+2}}{\lambda + (c-i-2)\alpha + (i+2)\mu - (i+2)\mu(r_{i,i} + r_{i+2,i+2})}, \quad i = 0, 1, \ldots, c-2.$$

Hence, for the general case we have

$$r_{i,j} = \frac{(c-j+1)\alpha r_{i,j-1} + j\mu \sum_{k=i+1}^{j-1} r_{i,k} r_{k,j}}{\lambda + (c-j)\alpha + j\mu - j\mu(r_{i,i} + r_{j,j})}, \quad j > i.$$

The rate matrix can be found from the diagonal part and then the upper diagonal parts using these recursive formulae.

- **Nonhomogeneous Part**

To find $R^{(i)} = (i = c, c-1, \ldots, 1)$, use

$$R^{(i)} = -Q_i^{(i-1)} \left(Q_0^{(i)} + R^{(i+1)} Q_{-1}^{(i+1)} \right)^{-1}, \quad i = c, c-1, \ldots, 1. \tag{3.34}$$

which is similar to (3.32). Since the rate matrices are upper diagonal, we need to focus on

$$XA = -Q_0^{(i-1)} \tag{3.35}$$

where $A = Q_0^{(i)} + R^{(i+1)}Q_{-1}^{(i+1)}$ and X are upper diagonal matrices of sizes $(i+1)$ x $(i+1)$ and i x $(i+1)$, respectively. Let $\mathbf{x}_j = (0, 0, \ldots, x_{j,j}, x_{j,j+1}, \ldots, x_{j,i})$ $(j = 0, 1, \ldots, i-1)$ be the j-th row vector of X. Then we have

$$\mathbf{x}_j A = (0, 0, \ldots, -\lambda, 0, \ldots, 0), \quad j = 0, 1, \ldots, i-1,$$

where $-\lambda$ is the $(j+1)$-th entry of the vector. Hence, the solution is

$$x_{i,j} = -\frac{\lambda}{a_{i,j}}, \quad x_{j,l} = -\frac{\sum_{k=j}^{l-1} x_{j,k} a_{k,l}}{a_{l,l}}, \quad l = j+1, j+2, \ldots, i, \tag{3.36}$$

where $a_{i,j}$ is the (i, j) entry of A.

3.5.3 G-matrix

The G-matrix gives the first passage probabilities from one position to the next position in the left hand side.

- **Homogeneous Part**

The G-matrix is also the minimal and nonnegative solution of

$$Q_1 + GQ_0 + G^2 Q_{-1} = \mathbf{O}, \tag{3.37}$$

and it is an upper diagonal matrix. Therefore, the method used for R-matrix can be used to find G-matrix. Let $g_{i,j}$ $(i, j = 0, 1, \ldots, c)$ be the (i, j) element of the matrix.

Compare (0,0) in the both sides (3.37) and we have

$$-(\lambda + c\alpha)g_{0,0} + \lambda g_{0,0}^2 = 0.$$

Then $g_{0,0} = 0$, because $0 \le g_{0,0} \le 1$. In the case of $g_{i,i}$, compare (i, i) elements of the equation. Then we find

$$i\mu - (\lambda + (c-i)\alpha + i\mu)g_{i,i} + \lambda g_{i,i}^2 = 0 \quad (i, j = 0, 1, \ldots, c-1).$$

Remember $0 \le g_{i,i} \le 1$ and

$$g_{i,i} = \frac{\lambda + i\mu + (c-i)\alpha - \sqrt{(\lambda + i\mu + (c-i)\alpha)^2 - 4i\lambda\mu}}{2i\lambda}, \quad i = 1, 2, \ldots, c-1,$$

which is same with z_i we defined before.

Now, compare (c, c) elements and since we need to choose the minimal root, we have $g_{c,c} = \lambda/(c\mu)$ instead of 1.

To find the upper diagonal elements of the matrix, we can still use the same method of R-matrix and find in a recursive way.

Consider $g_{i,i+1}$ $(i = 1, 2, \ldots, c-1)$ and compare $(i, i+1)$ elements, then we have

$$g_{i,i+1} = \frac{(c-i)\alpha g_{i+1,i+1}}{\lambda + (c-i)\alpha + i\mu - \lambda(g_{i,i} + g_{i+1,i+1})}, \quad i = 0, 1, \ldots, c-1.$$

Finally, compare (i, j) elements and then we find

$$g_{i,j} = \frac{(c-i)\alpha + \lambda \sum_{k=i+1}^{j-1} g_{i,k} g_{k,j}}{\lambda + (c-i) + i\mu - \lambda(g_{i,i} + g_{j,j})}, \quad i+1 < j \leq c.$$

- **Nonhomogeneous Part**

The first passage probabilities depend on the level in the nonhomogeneous part. Let $G^{(n)}$ become the probability from level n to level $n-1$. It is the minimal nonnegative solution of

$$Q_{-1} + G^n Q_0 + Q_1 G^n G^{n+1} = \mathbf{O}, \quad n = 1, 2, \ldots, c,$$

where $G^{c+1} = G$. Then we can use the similar nonhomogeneous method in the rate matrix and define $G^{(n)}$ for $n = 1, 2, \ldots, c$.

3.6 Other Parts of the Paper

- **Comparison of the Methods**

When our two methods are compared, we can conclude that the homogeneous part of the QBD formulation corresponds to $\widehat{\Pi}_i(z)$ $(i = 0, 1, \ldots, c)$ of the generating function approach. Also, the nonhomogeneous part of the matrix analytic method corresponds to (i, j); $j = i = 0, 1, \ldots, c,$ $i \leq j \leq c$ of the generating function approach. The matrix analytic method gives a recursive formula for computing the rate matrix.

The matrix analytic method and recursive renewal approach are also equivalent in some way. The quantity $p^L_{i \to d}$ in [8, 9] is identical to $g_{i,d}$ in the matrix analytic method. Notice that matrix R can be found from matrix G. However, recursive renewal approach gives the generating function of the queue length and matrix analityc method is interested of direct computation of the queue length computation.

- **Variant Models**

As variant models, $M/M/c/Setup/Sleep$ and $M/M/c/Setup/Delayoff$, which are also mentioned in [8], are introduced.

In $M/M/c/Setup/Sleep$ model, when a set of $s \leq c$ servers is idle, then it is set to sleep. While $c - s$ servers are idle, then they are turned off. Note that sleep state have a shorter setup time than the off state.

In $M/M/c/Setup/Delayoff$ model, when a server completes its service however does not have a job, then it stays idle for a while.

If we compare our models, it can be affirmed that the boundary part of ”Sleep” model has the same structure with $M/M/c/Setup$ model, but different with ”Delayoff” model. The QBD formulation works to find the rate matrix of the homogeneous part for both models. Moreover, the generating function approach can be used for $M/M/c/Setup/Sleep$ easier than $M/M/c/Setup/Delayoff$ which needs some arrangements.

CHAPTER 4

$M/M/c/Setup$ with Deterministic Setup Times

The main model to be studied in this thesis is the following modification of the $M/M/c/Setup$ model reviewed in the previous section: we replace the setup time distribution with a deterministic time, i.e., each server that is put in setup mode, completes its setup and becomes fully on in a fixed deterministic amount of time. We will refer to this system as $M/M/c/dSetup$. Given that the setup of a computer is a fixed operation at all times and all machines to take the setup time deterministic is a more realistic assumption.

Let us begin with specifying the dynamics of this model: as in the previous section, let c denote the number of server. Let $1/\alpha$, $\alpha > 0$, denote the constant setup time. We will denote by R_t the number of servers working on a job at time t, and W_t will represent the number of customers in the system waiting for service. As opposed to the $M/M/c/Setup$, we explicitly represent these two quantities with separate processes (in the $M/M/c/Setup$, R_t and W_t was represented together as $X_t(1)$, the total number of customers in the system). A second difference from the $M/M/c/Setup$ is that to have a Markovian model for the $M/M/c/dSetup$ system, we have to keep track of the remaining setup times for the servers in Setup mode; the process $\boldsymbol{S}_t \in \mathbb{R}_+^c$ servers this purpose. For fixed t, $\boldsymbol{S}_t \in \mathbb{R}_+^c$ is assumed to have decreasing components, i.e., $\boldsymbol{S}_t(i) > \boldsymbol{S}_t(j)$ if $i > j$; this means that, by convention, the newer a server is put in setup mode, its index in the vector $\boldsymbol{S}_t$ is smaller. Furthermore, $\boldsymbol{S}_t(i) = 0$ for $i > C_t$; i.e., only the nonzero components of $\boldsymbol{S}_t$ represent the remaining setup time of a server in setup mode. Then the process representing the $M/M/c/dSetup$ model is $\mathcal{X}_t^d = (R_t, W_t, \boldsymbol{S}_t)$. One can represent the total number of servers in setup mode by

$$S_t = \sum_{j=1}^{c} 1_{\{\boldsymbol{S}_t(j)>0\}};$$

the dynamics explained below will always imply $W_t \geq S_t$, i.e., the number of customers waiting for service always exceeds the number of servers being setup.

Here, $\mathcal{X}^d$ is a piecewise deterministic Markov process: R and W components are constant between the jumps of $\mathcal{X}^d$. The S component satisfies the ordinary differential equation (ODE)

$$\frac{d\boldsymbol{S}(j)}{dt} = -1_{\{\boldsymbol{S}_t(j)>0\}}, j = 1, 2, 3, ..., c, \tag{4.1}$$

in between the jumps of $\mathcal{X}^d$. The dynamics (4.1) mean that the remaining setup times diminish uniformly in time until they hit 0. There are three sources of jumps of $\mathcal{X}^d$: arrivals to the system, service completions, and setup completions. At any given time t, we know the time of the next setup time completion (let us denote it by τ_t^s, this is given by the last nonzero element of S_t):

$$\tau_t^s = \inf\{\boldsymbol{S}_t(j) : j = 1, 2, 3, .., S_t\} = \boldsymbol{S}_t(S_t),$$

if $\boldsymbol{S}_t > 0$; if $C_t = 0$, there are no servers in setup mode and the dynamics described below can easily modified to handle this case. In what follows we describe the dynamics only for the case when $S_t > 0$; in particular we are assuming $\tau_t^s < \infty$. In the time interval (t, τ_t^s), the jumps can only arise from arrivals and service completions. The arrivals occur at a constant Poisson rate λ, the service completions occur at a rate μR_t. Therefore, the total jump rate of $\mathcal{X}^d$ in the time interval (t, τ_t^s) is $\lambda + \mu R_t < \lambda c + R_t$; in particular, with probability 1, $\mathcal{X}^d$ will jump at most finitely many times. Let us denote τ_t be the first jump of $\mathcal{X}^d$, after t; as we have just discussed, if $\tau_t < \tau_t^s$, τ_t is either an arrival or a service completion. With probability

$$\frac{\lambda}{\lambda + \mu R_{\tau_t -}}$$

the jump is an arrival, and with probability

$$1 - \frac{\lambda}{\lambda + \mu R_{\tau_t -}}$$

it is a service completion. In the first case, the position $\mathcal{X}^d_{\tau_t}$ of the process right after the jump will be

$$\begin{aligned}
&W_{\tau_t} = W_{\tau_t -} + 1_{\{c = C_{\tau_t -} + R_{\tau_t -}\}},\\
&\boldsymbol{S}_{\tau_t} = \boldsymbol{S}_{\tau_t -} 1_{\{S_{\tau_t -} + R_{\tau_t -} = c\}} + (\boldsymbol{\alpha} + R(\boldsymbol{S}_{\tau_t -})) 1_{\{S_{\tau_t -} + R_{\tau_t -} < c\}},\\
&R_{\tau_t} = R_{\tau_t -},
\end{aligned} \tag{4.2}$$

where $\boldsymbol{\alpha} \in \mathbb{R}_+^c$ is the vector

$$\boldsymbol{\alpha}(j) = \begin{cases} 1/\alpha, & j = 1 \\ 0, & \text{otherwise,} \end{cases}$$

and $L, R : \mathbb{R}^c \to \mathbb{R}^c$ are left and right shift operators; e.g., for $c = 3$, $L([1\ 2\ 3]) = [2\ 3\ 0]$. The first and the second lines of (4.2) mean that if all servers are busy (i.e., either in setup mode or in service mode) the arriving customer is added to the waiting customers queue; if there is a server available, an off server is put in setup mode with new setup time c. In the second case (i.e., the jump is a service completion) the position $\mathcal{X}^d_{\tau_t}$ will be

$$\begin{aligned}
&W_{\tau_t} = W_{\tau_t -} - 1_{\{W_{\tau_t -} > 0\}},\\
&\boldsymbol{S}_{\tau_t} = L(\boldsymbol{S}_{\tau_t -}) 1_{\{S_{\tau_t -} \geq W_{\tau_t -}\}} + \boldsymbol{S}_{\tau_t -} 1_{\{S_{\tau_t -} < W_{\tau_t -}\}},\\
&R_{\tau_t} = R_{\tau_t -} - 1,
\end{aligned}$$

where we are using the fact that $S_t > 0$. The first line means: if there are waiting customers when one of the servers completes service, the server starts serving one of the waiting customers. The second line means: if after this operation there are more servers being setup than customers waiting for service, then turn off the server whose setup time is the longest. If there are waiting customers for each server in setup mode, then continue setting up these servers. Here we see another difference from the $M/M/c/Setup$ model: in the $M/M/c/Setup$ model an arbitrary server in setup mode is turned off as soon as the number of servers in setup mode exceeds the number of waiting customers. In the $M/M/c/dSetup$, we choose the least setup server when we need to turn of a server in setup mode. Note that a service completion may lead to $C_{\tau_t} = 0$, i.e., no servers in setup mode. In that case, the dynamics restart with $C = 0$; as we noted above, the above specification can easily be modified to handle that case.

It remains to specify at τ_t^d, i.e., when one of the servers in setup mode completes setup:

$$
\begin{aligned}
W_{\tau_t^d} &= W_{\tau_t-} - 1, \\
\boldsymbol{S}_{\tau_t^d} &= \boldsymbol{S}_{\tau_t-}, \\
R_{\tau_t^d} &= R_{\tau_t-} + 1,
\end{aligned}
$$

where the first line means the number of waiting customers is reduced by 1 and the third line means that the number of running servers increase by 1 (because, the server is setup and starts one of the waiting customers). The second line simply means, that the remaining servers in setup mode continue with being setup. With this we have completely specified the dynamics of $\mathcal{X}^d$. An alternative way of specifying these dynamics is to use the piecewise Markov process notation of [7], which also can be used to give an explicit construction of the Markov process $\mathcal{X}^d$.

4.1 Comments on the balance equation

Because of deterministic setup times (represented by the S component of $\mathcal{X}^d$), the process $\mathcal{X}^d$ doesn't have a simple embedded Markov chain whose stationary distribution can be used to compute that of $\mathcal{X}^d$. Therefore, for the $M/M/c/dSetup$ model, one must directly study the stationary measure of the continuous time process $\mathcal{X}^d$. The stationary measure μ of $\mathcal{X}^d$ will be a measure on the state space $S = \mathbb{N} \times \mathbb{N} \times [0, 1/\alpha]^c$. The measure μ can be represented by densities $f(i, j, s_1, s_2, ..., s_k, k)$ and the probabilities $f(i, j, 0)$ for $i, j \in \mathbb{N} \times \mathbb{N}$ and $k = 1, 2, .., c$; $f(i, j, s_1, s_2, .., s_k)$ denotes the probability that in steady state at any time, there are i servers running and k customers waiting and k servers in setup with the setup times $s_1, s_2, ..., s_k$; $f(i, j, 0)$ represents the probability, in steady state, that there are i servers running and j customers waiting. By the definition of the process, $f(i, j, 0)$ can be nonzero only when $j = 0$. One can now proceed to write down a setup of coupled infinite system of linear ordinary differential equations for f and expects to find a unique solution to this system when the system is stable, i.e., when $\rho = \lambda/c\mu < 1$; i.e, the stability condition remains unchanged when setup times are assumed deterministic. We are not aware of an explicit solution or an approximate solution of the balance equation for this process; these can be subject of study in future works. For the purposes estimating average energy cost, we do not need the whole stationary measure but only the expectations

$$\mathbb{E}_\infty[R_t], \mathbb{E}_\infty[S_t]$$

under it. An alternative way to approximate these is through Monte-Carlo simulation. This is the approach used in this thesis. The next chapter explains the simulation approach, gives the results of our simulations and our interpretation of them.

CHAPTER 5

Simulation of $M/M/c/dSetup$ and Comparison to $M/M/c/Setup$

Recall that the power cost per unit time for the $M/M/c/Setup$ model is:

$$E_\infty = C_a \mathbb{E}_\pi[R_t] + C_s \mathbb{E}_\pi[S_t],$$

where $\mathbb{E}_\pi$ denotes expectation under the stationary measure. To compute E_∞, we only have to compute the expectations $\mathbb{E}_\pi[R_t]$ and $\mathbb{E}_\pi[S_t]$. For the $M/M/c/Setup$ model, the corresponding expecations are computed by developing explicit formulas for the stationary measure π. In this section, we will use simulation to approximate these probabilities. The simulation approach is based on Ergodic Thereom which says

$$\lim_{T\to\infty} \frac{1}{T}\int_0^T R_s ds = \mathbb{E}_\pi[R_t], \tag{5.1}$$
$$\lim_{T\to\infty} \frac{1}{T}\int_0^T S_s ds = \mathbb{E}_\pi[S_t],$$

almost surely. The simulation idea consist of generating random sample paths of R and S on a computer. Given that R and S are piecewise constant functions of t with finitely many jumps, the above integrals become finite sums for any simulated sample path of R and S; therefore, they are simple to compute. If we take T large enough, the almost sure convergence implies that the computed integrals must be close to the expecations that we intend to compute.

The simulation of $\mathcal{X}^d = (R, C, \boldsymbol{S})$ is straightforward, because these processes are piecewise deterministic with simple dynamics between jumps. Therefore, the description of the process given in the previous chapter serves also as a simulation algorithm. We have conducted the simulations in the Octave computing environment. Figure 5.1 shows the trajectory of the last component of the S process, i.e., this is the graph of shortest remaining setup time; Figure 5.2 shows the joint sample path of the (R, S) process.

An important issue in the approximation suggested by (5.1) is how to choose T so that the prelimit quantity on the left provides a good approximation of the expectation on the right. In all of the simulations below we have chosen T by gradually increasing it until we observed convergence.

The average unit cost of power E_∞ is a function of the system parameters, μ, λ, c and α. The goal of this chapter is a comparison of E^d_∞ and E_∞ as these parameters vary.

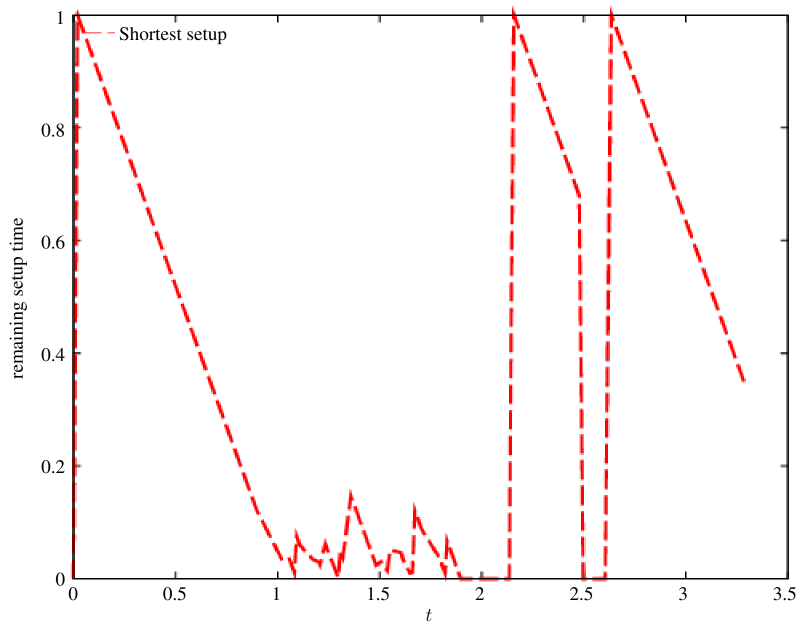

Figure 5.1: The evolution of the shortest setup time in a simulation.

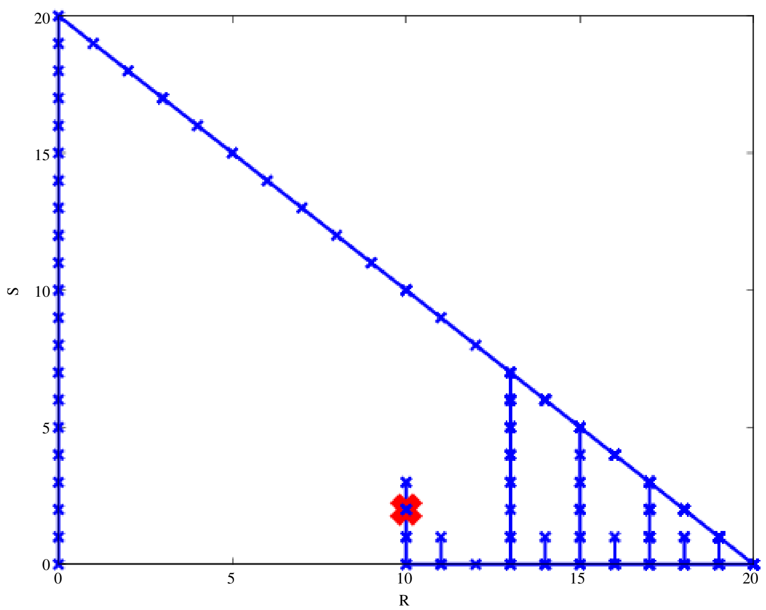

Figure 5.2: A sample path of the (R, S) process in a simulation; the bold point shows the latest position of the process.

To do the comparison we also have to compute E_∞; in [17] this is done by developing formulas for the stationary measure of the underlying process. For the purposes of this thesis we have approximated E_∞ with simulation using the approximations given of (3.5) in Chapter 3. For the simulation of the $M/M/c/Setup$ system it suffices to simulate the two dimensional constrained random walk X given in (3.1) of Chapter 3.

Similar to the $M/M/c/Setup$ system, if the system receives a burst of heavy traffic pushing it away from the empty state, all servers will be ON in the $M/M/c/dSetup$ system; when that is the case the total service rate of the system will be $c\mu$, therefore, when $\mathcal{X}^d$ is sufficiently away from the empty state, $M/M/c/dSetup$ will behave like the $M/M/c$ system. Therefore, a natural stability condition for $M/M/c/dSetup$ is $\lambda < c\mu$, under which one expects that a unique stationary measure exists. We will assume $\lambda < c\mu$ throughout the simulation study. The ratio

$$\rho \doteq \frac{\lambda}{c\mu}$$

is the utilization rate of the system.

In [17] the $M/M/c/Setup$ system is compared to the $M/M/c$ system, i.e., c servers that always remain on. A review of the stationary analysis of this system is given in Section 2.4 in Chapter 2 above. The average unit cost of energy for that system is

$$E_\infty^{MMC} = c\rho C_a + c(1-\rho)C_i$$

, where C_i is the energy consumption rate of an idle server; following [17] we take it to be $C_i = 0.6$. In our comparison study below we will also give E_∞^{MMC}; to ease relating our results to those of [17]. In all of the simulations below we take $\mu = 1$ and $C_a = C_s = 1$, again following [17].

The first three sections below compare E_∞ and E_∞^d as α and c and ρ varies. The last section looks at the structural properties of E_∞^d as a function of the system paramaters.

Our main conclusion from these sections is the following: E_∞ provides a very good approximation of E_∞^d for the range of parameters under consideration.

5.1 Varying α

In this section we present four graphs of E_∞ as a function of α when ρ and c variables are fixed; the ρ and c parameter values for the graphs are: $c = 15$, $c = 25$, and $\rho = 0.5$ and $\rho = 0.7$, respectively.

The relative error made in approximating in E_∞^d by E_∞ is shown by

$$R_e = \frac{|E_\infty^d - E_\infty|}{E_\infty^d};$$

the relative errors for the E values represented in Figures 5.3, 5.4, 5.5 and 5.6 are given in Figure 5.7

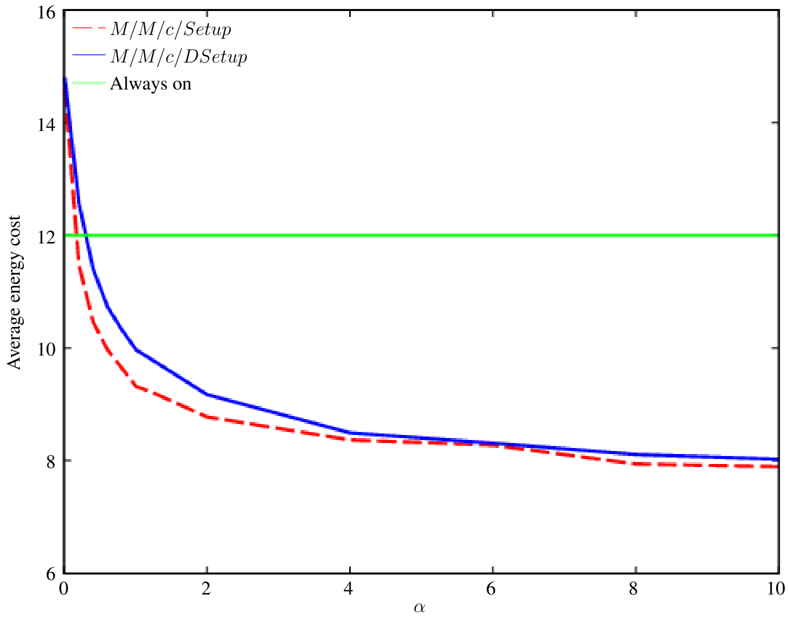

Figure 5.3: E_∞ as a function of α, $\rho = 0.5$, $c = 15$.

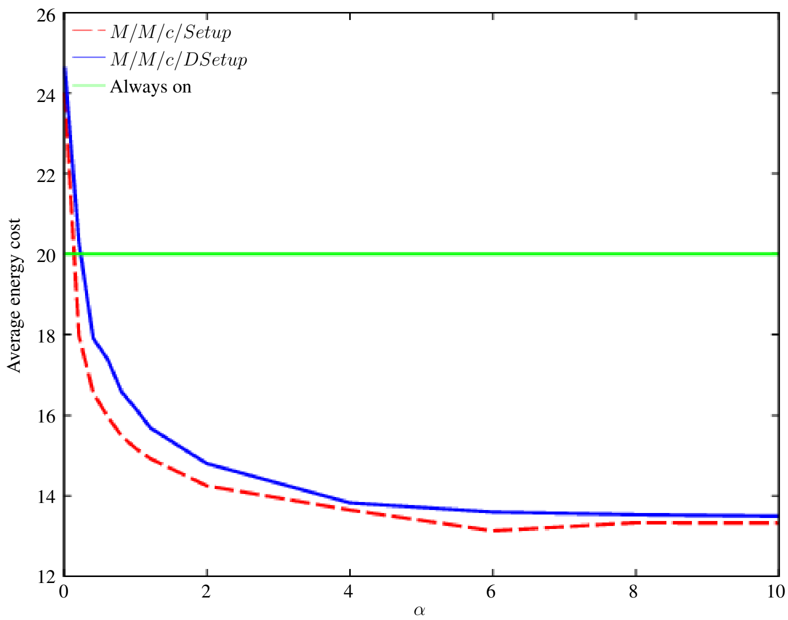

Figure 5.4: E_∞ as a function of α, $\rho = 0.5$, $c = 25$.

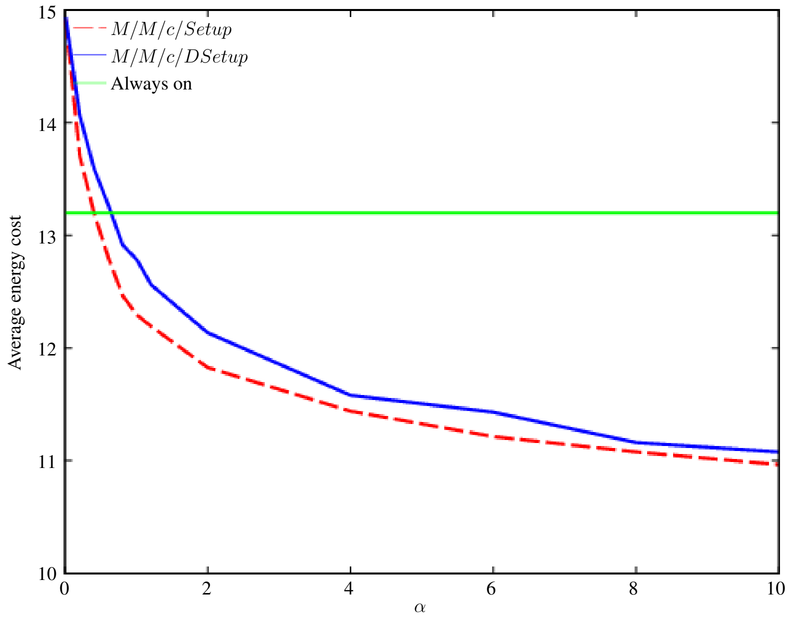

Figure 5.5: E_∞ as a function of α, $\rho = 0.7$, $c = 15$.

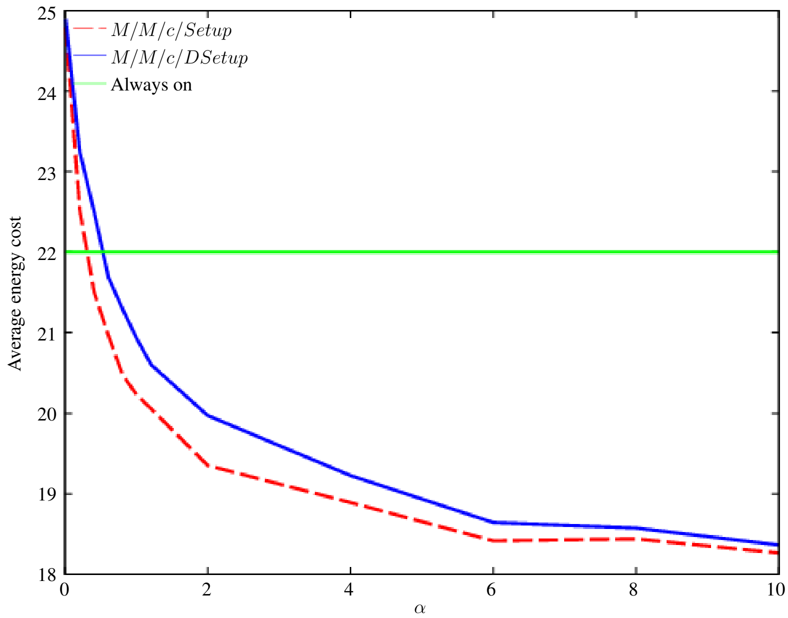

Figure 5.6: E_∞ as a function of α, $\rho = 0.7$, $c = 25$.

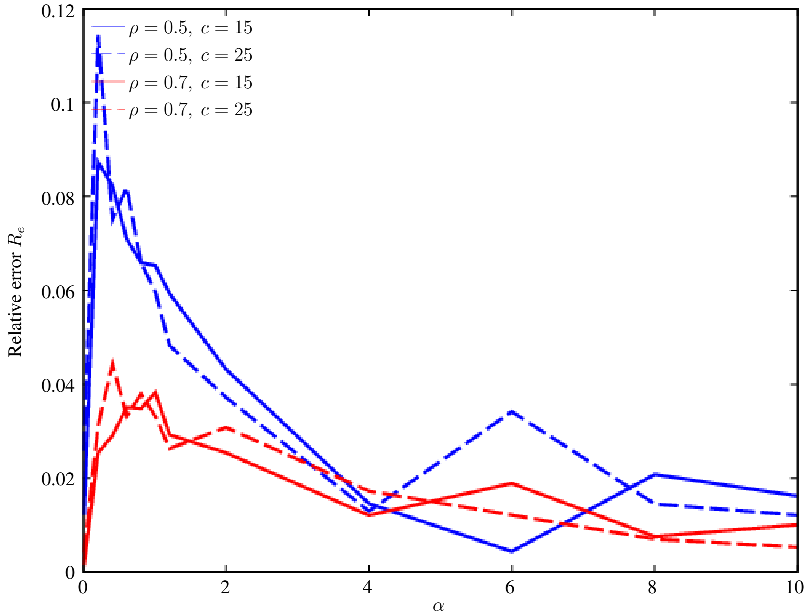

Figure 5.7: R_e as a function of α when ρ and c is fixed as in Figures 5.3, 5.4, 5.5 and 5.6.

Our main observations about these results are as follows:

1. In all of the graphs E^d_∞ and E_∞ are very close to each other or all values of α; the average relative errors (the average of $|E^d_\infty - E_\infty|/E^d_\infty$ over the parameter values used in the respective simulations) are: 0.045 for Figure 5.3, 0.048 for Figure 5.4, 0.022 for Figure 5.5 and $0.0.24$ for Figure 5.6. This suggests that one can use E_∞ as a first approximation for E^d_∞.

2. In all of the graphs E^d_∞ lies above E_∞; it would be interesting to try to prove this rigorously; if correct, therefore, one can use E_∞ always as a lower bound for E^d_∞.

3. The average relative error changes slowly with c for both values of ρ;

4. The average relative error decreases quickly with ρ; this is also clear from Figure 5.7.

5. Because E_∞ and E^d_∞ are very similar, the relation of E^d_∞ to E^{MMC}_∞ is similar to that of E_∞ to the same, as studied in [17].

The next section further studies the function E_∞ as we vary the c parameter.

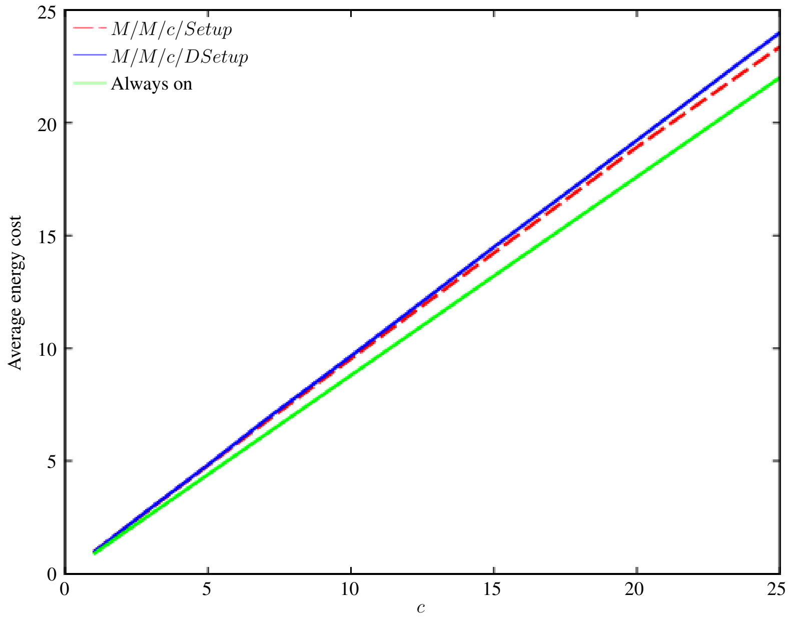

Figure 5.8: E_∞ as a function of c, $\rho = 0.7$, $\alpha = 0.1$.

5.2 Varying number of servers c

The next Figures give graph of E_∞ as a function of c for the following parameter values $\alpha = 0.1$, $\alpha = 1$ and $\rho = 0.7$.

Figure 5.10 shows the E^d_∞ as c varies for both $\alpha = 1$ and $\alpha = 0.1$; as observed in the previous section, as α increases, the setup time $1/\alpha$ decreases and the E^d_∞ also decreases with it:

Our observations on these results are as follows:

1. As we have already observed above, E_∞ and E^d_∞ are close to each other and provide good approximations of one another,

2. E_∞ lies below E^d_∞,

3. The average relative error decreases with α.

The next section looks further into the effect of varying the ρ parameter.

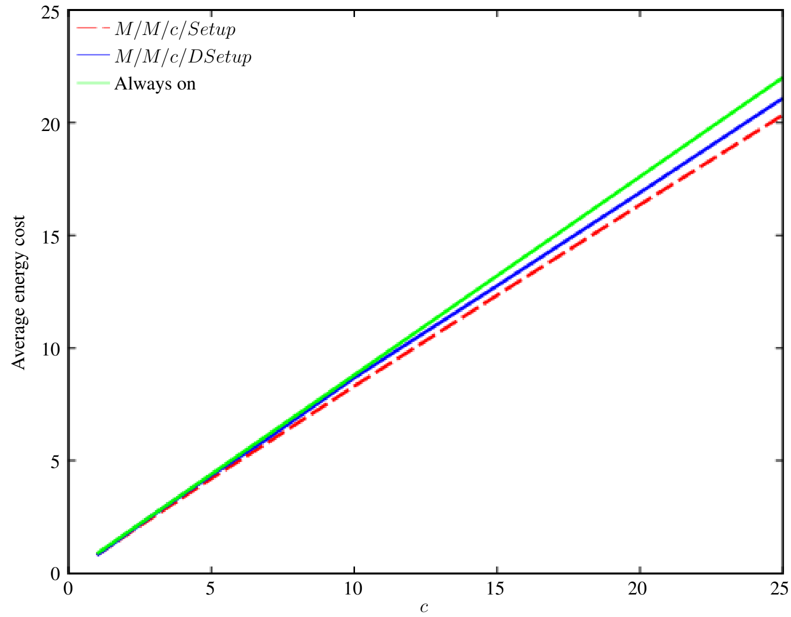

Figure 5.9: E_∞ as a function of c, $\rho = 0.7$, $\alpha = 1$.

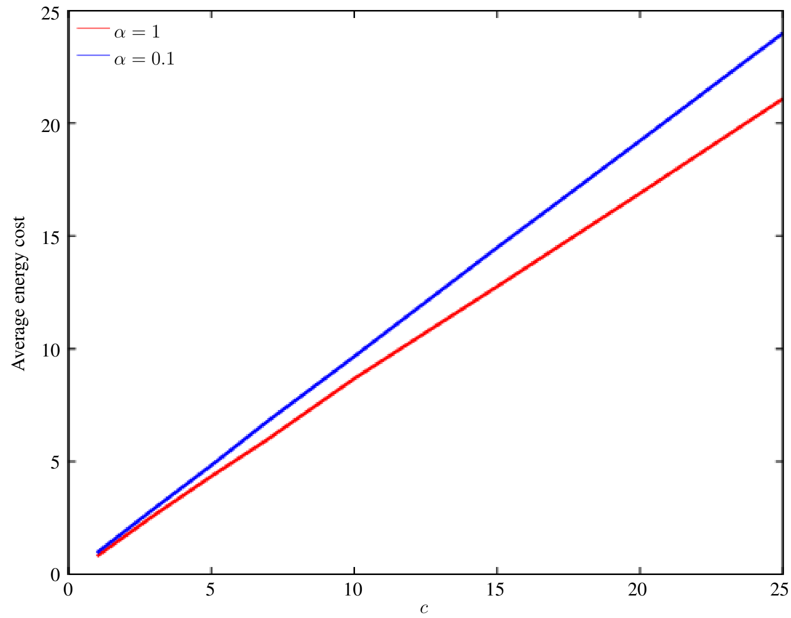

Figure 5.10: E_∞ as a function of c, $\alpha = 0.1$ and $c = 1$; for both plots $\rho = 0.7$.

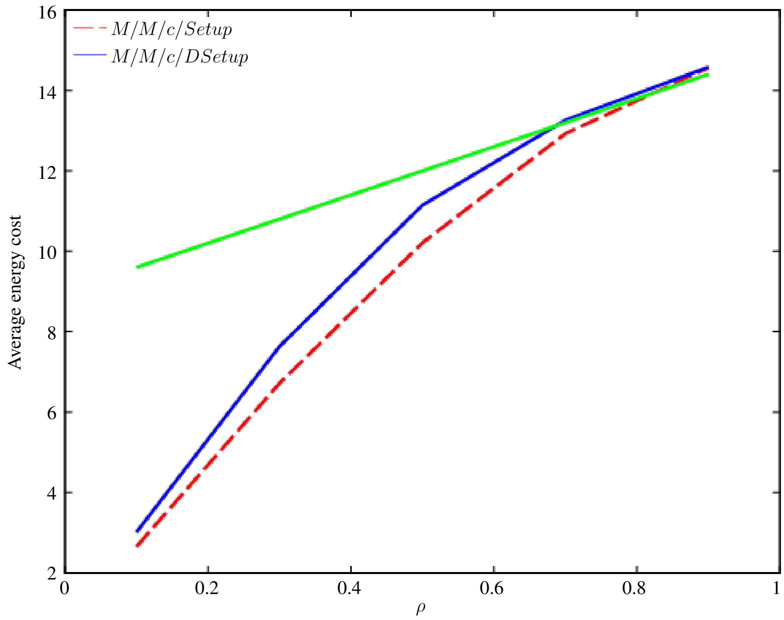

Figure 5.11: E_∞ as a function of ρ, $\alpha = 0.5$, $c = 15$.

5.3 Varying ρ

Figure 5.11 shows the graphs of E_∞, E^d_∞ and E^{MMC}_∞ as a function of ρ for $\alpha = 0.5$ and $c = 15$.

Our observations are as follows:

1. Once again, E_∞ and E^d_∞ provide close approximations of one another, with $E_\infty < E^d_\infty$,
2. Similar to [17] we see that E^d_∞ is increasing in ρ and converges to E^{MMC}_∞ as $\rho \nearrow 1$.

CHAPTER 6

Conclusion and Outlook

Energy management has now became an important problem for servers and data centers, focusing on the reduction of all energy-related costs, including capital, operating expenses, and environmental impacts.

Essentially, even an energy efficient server still consumes about half its full power while doing no work. Servers designed with less attention to energy efficiency usually idle at even higher power levels. It is of interest to study and optimize energy usage of these centers. One of the main theoretical tools in the study of this problem is queueing theory. Within queueing theory two models that are of direct interest to the problem are $M/M/c/Setup$ queues and $M/M/c$ queues. The main financial measure used in this approach is the average energy cost per unit time. Ergodic theorem tells us that this measure can be computed via the stationary measure of the process. The paper [17] develops efficient computational methods based on the generating function approach and matrix analytic approach for the computation of the stationary measure and uses these methods for a study of the average energy cost per unit time for $M/M/c/Setup$ systems. We have reviewed the approach of this paper in Chapter 3.

Setup of a server is usually a constant process, hence a more reasonable assumption for it is that it is deterministic, rather than random, as assumed in $M/M/c/Setup$. The main goal of the present thesis was to understand what happens to the average energy cost per unit time of the system when one modifies random setup times to deterministic setup times (we denoted the modified system by $M/M/c/dSetup$). This problem has been considered in Chapters 4 and 5. Our main tool was simulation which is made possible by the piecewise deterministic dynamics of the underlying processes. Our main finding is that the average energy cost per unit time measures of the $M/M/c/dSetup$ and $M/M/c/Setup$ systems are very close (around 2% to 4% relative error for the parameter values studied in Chapter 5), at least for the parameter values studied in this thesis.

Let us now note three problems for future research:

1. We think that it would be interesting to try to generalize the computations given in [17] to the $M/M/c/dSetup$ framework.

2. Our main financial measure of cost was average unit cost per unit time:

$$P.\lim_{T\to\infty}\frac{1}{T}\int_0^T c_t d_t$$

, where c_t is the energy consumption rate at time t and P is the constant price of energy per unit time. In all of the works using the queueing theory framework P is taken a constant. But as is well known, P is usually a stochastic process. It may be of interest to develop models that take this into consideration.

3. Another possibility is to consider discounted costs of the form

$$\mathbb{E}\left[\int_0^\infty e^{-\int_0^t r_s ds} P_t c_t d_t\right],$$

where r_s is the instantaneous interest rate at time s; this model would also take into account the stochasticity of interest rates.

REFERENCES

[1] J. R. Artalejo, A. Economou, and M. J. Lopez-Herrero, Analysis of a multiserver queue with setup times, Queueing Systems, 51(1), pp. 53–76, 2005.

[2] F. Baccelli and P. Brémaud, *Palm probabilities and stationary queues*, volume 41, Springer Science & Business Media, 2012.

[3] L. A. Barroso and U. Hölzle, The case for energy-proportional computing, Computer, 40(12), 2007.

[4] W. Bischof, Analysis of M/G/1-queues with setup times and vacations under six different service disciplines, Queueing systems, 39(4), pp. 265–301, 2001.

[5] G. Choudhury, On a batch arrival Poisson queue with a random setup time and vacation period, Computers & operations research, 25(12), pp. 1013–1026, 1998.

[6] G. Choudhury, An Mx/G/1 queueing system with a setup period and a vacation period, Queueing Systems, 36(1), pp. 23–38, 2000.

[7] M. H. Davis, *Markov Models & Optimization*, volume 49, CRC Press, 1993.

[8] A. Gandhi, S. Doroudi, M. Harchol-Balter, and A. Scheller-Wolf, Exact analysis of the M/M/k/setup class of Markov chains via recursive renewal reward, in *ACM SIGMETRICS Performance Evaluation Review*, volume 41, pp. 153–166, ACM, 2013.

[9] A. Gandhi, S. Doroudi, M. Harchol-Balter, and A. Scheller-Wolf, Exact analysis of the M/M/k/setup class of Markov chains via recursive renewal reward, Queueing Systems, 77(2), p. 177, 2014.

[10] A. Gandhi, M. Harchol-Balter, and I. Adan, Server farms with setup costs, Performance Evaluation, 67(11), pp. 1123–1138, 2010.

[11] G. Grimmett and D. Stirzaker, *Probability and random processes*, Oxford university press, 2001.

[12] E. Hyytiä, R. Righter, and S. Aalto, Task assignment in a heterogeneous server farm with switching delays and general energy-aware cost structure, Performance Evaluation, 75, pp. 17–35, 2014.

[13] L. Lakatos, L. Szeidl, and M. Telek, *Introduction to queueing systems with telecommunication applications*, Springer Science & Business Media, 2012.

[14] V. J. Maccio and D. G. Down, On optimal control for energy-aware queueing systems, in *Teletraffic Congress (ITC 27), 2015 27th International*, pp. 98–106, IEEE, 2015.

[15] V. J. Maccio and D. G. Down, Exact analysis of energy-aware multiserver queueing systems with setup times, in *Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS), 2016 IEEE 24th International Symposium on*, pp. 11–20, IEEE, 2016.

[16] M. F. Neuts, *Matrix-geometric solutions in stochastic models: an algorithmic approach*, Courier Corporation, 1981.

[17] T. Phung-Duc, Exact solutions for m/m/c/setup queues, Telecommunication Systems, 64(2), pp. 309–324, 2017.

[18] V. Ramaswami and P. G. Taylor, Some properties of the rate perators in level dependent uasi-birth-and-death processes with countable number of phases, Stochastic Models, 12(1), pp. 143–164, 1996.

[19] V. Sundarapandian, *Probability, statistics and queuing theory*, Phi Learning New Delhi, 2009.

[20] H. Takagi, Priority queues with setup times, Operations Research, 38(4), pp. 667–677, 1990.

[21] B. Van Houdt and J. S. van Leeuwaarden, Triangular M/G/1-type and tree-like quasi-birth-death Markov chains, INFORMS Journal on Computing, 23(1), pp. 165–171, 2011.