FAST SIMULATION OF RARE EVENTS
IN MARKOV LEVEL/PHASE PROCESSES

A Thesis Submitted to the College of

Graduate Studies and Research

in Partial Fulfillment of the Requirements

For the Degree of

Doctor of Philosophy

in the Department of Computer Science

University of Saskatchewan

Saskatoon, Saskatchewan

by

Jingxiang Luo

© Copyright Jingxiang Luo, July 2004. All rights reserved.
PERMISSION TO USE

In presenting this thesis in partial fulfillment of the requirements for a Postgraduate degree from the University of Saskatchewan, I agree that the Libraries of this University may make it freely available for inspection. I further agree that permission for copying of this thesis in any manner, in whole or in part, for scholarly purposes may be granted by the professor or professors who supervised my thesis work or, in their absence, by the Head of the Department or the Dean of the College in which my thesis work was done. It is understood that any copying or publication or use of this thesis or parts thereof for financial gain shall not be allowed without my written permission. It is also understood that due recognition shall be given to me and to the University of Saskatchewan in any scholarly use which may be made of any material in my thesis.

Requests for permission to copy or to make other use of material in this thesis in whole or part should be addressed to:

Head of the Department of Computer Science
University of Saskatchewan
Saskatoon, Saskatchewan, Canada
S7N 5A9
Fast Simulation of Rare Events in Markov Level/Phase Processes

Candidate: Jingxiang Luo
Supervisor: Dr. Winfried K. Grassmann

Doctor of Philosophy Thesis
Presented to the College of Graduate Studies and Research
University of Saskatchewan
July 2004

Abstract

Methods of efficient Monte-Carlo simulation when rare events are involved have been studied for several decades. Rare events are very important in the context of evaluating high quality computer/communication systems. Meanwhile, the efficient simulation of systems involving rare events poses great challenges. A simulation method is said to be efficient if the number of replicas required to get accurate estimates grows slowly, compared to the rate at which the probability of the rare event approaches zero.

Despite the great success of the two mainstream methods, importance sampling (IS) and importance splitting, either of them can become inefficient under certain conditions, as reported in some recent studies. The purpose of this study is to look for possible enhancement of fast simulation methods. I focus on the “level/phase process”, a Markov process in which the level and the phase are two state variables. Furthermore, changes of level and phase are induced by events, which have rates that are independent of the level except at a boundary. For such a system, the event of reaching a high level \( n \) occurs rarely, provided the system typically stays at lower levels. The states at those high levels constitute the rare event set. Though simple, this models a variety of applications involving rare events. In this setting, I have studied two efficient simulation methods, the rate tilting method and the adaptive splitting method, concerning their efficiencies. I have compared the efficiency of rate tilting with several previously used similar methods. The experiments are done by using queues in tandem, an often used test bench for the rare event simulation. The schema of adaptive splitting has not been described in literature. For this method, I have analyzed its efficiency to show its superiority over the (conventional) splitting method.

The way that a system approaches a designated rare event set is called the system’s
large deviation behavior. Toward the end of gaining insight about the relation of system
behavior and the efficiency of IS simulation, I quantify the large deviation behavior and its
complexity. This work indicates that the system’s large deviation behavior has a significant
impact on the efficiency of a simulation method.
ACKNOWLEDGEMENT

In the exciting moment of completing this dissertation, I owe thanks to many kind people. They have always stood beside me during frustrating time of research, and have inspired me to reach this goal.

I am sincerely grateful to my supervisor Prof. Winfried K. Grassmann. Without his patience and guidance, personal caring and support, this work would not have been completed. He also suggested many useful insights into my work, and spent countless hours with me to improve the technical writing. I owe thanks to my thesis committee: Prof. Perwez Shahabuddin, Prof. Raj Srinivasson, Prof. Mark Keil, Prof. Dwight Makaroff, Prof. Eric Neufeld, for their help, encourages and for their thoughtful and constructive criticism. Following Prof. Perwez Shahabuddin’s advice, a careful comparison with previous related methods has been included. Prof. Raj Srinivason helped me to improve some mathematical proofs and directed me to related work in the literature.

This research was funded by the discovery grant from the Natural Sciences and Engineering Research Council (NSERC) of Canada. I received the Graduate Scholarship from University of Saskatchewan. My travel cost to an international conference was covered partially by the President Fund / student award from the University. I acknowledge these supports with gratitude.

I would like to acknowledge the following individuals for opportunities of discussing questions relevant to this study: Prof. William Stewart, Prof. Carey Williamson, and Prof. Attahiru S. Alfa. I also appreciate the effort of some anonymous referees in reading my papers. Their opinions and feedback have impact on the topics and the presentation of my work. Special thanks go to Prof. Carl McCrosky, who guided me with patience on the methodology to do scientific research, prior to my entering Ph. D. program. I would also like to thank the following people for various of help: Prof. Jim Greer: the head of the department, Mrs. Maureen Desjardins, Mr. Greg Oster, Ms. Jan Thompson and other technical and office staff. Thanks also go to Mr. & Mrs. Goerzen (Con & Elsie), Mr. & Mrs. Blocks (Peter & Arlene), Wen’guang (Wayne) Wang & Yanping Zhao, and many other friends, who have made my stay in Saskatoon a memorable experience. Last but not least, I thank my wife Qun (pronounced as Chyeun) more than any words could express. It is due to her that I could dedicate myself to the research during the past several years. I am in debt to my daughter Ruiping, for spending too little time with her.

This dissertation is dedicated to my wife Qun, and to my parents in China.
# Table Of Contents

PERMISSION TO USE ................................................. i

ABSTRACT .......................................................... ii

ACKNOWLEDGEMENT ................................................ iv

Table of Contents .................................................. v

List of Figures ...................................................... viii

List of Tables ....................................................... ix

1 Introduction ...................................................... 1

1.1 Performance Evaluation Via Simulation ......................... 1

1.2 Importance Sampling Methodology and an Analogy .............. 3

1.2.1 The methodology ........................................... 4

1.2.2 An analogy ................................................ 5

1.2.3 Some points of view ....................................... 6

1.3 Redistribution of Penalties .................................... 7

1.4 An Overview of the Dissertation ................................ 8

1.4.1 Assumptions .............................................. 8

1.4.2 The focus ................................................ 9

1.4.3 Relation to other studies ................................ 9

1.4.4 The plan .................................................. 11

2 Fast Simulation: Background ................................... 13

2.1 What Is Efficiency and “Efficient Simulation”? ................ 14

2.2 Stochastic System Models – Discrete Event Systems vs. Markov Event Systems .................................................. 17

2.2.1 Multi-dimensional state space, phase-type distributions .... 19

2.3 Random Walks and the Large Deviation Rate .................. 20

2.3.1 Cramer’s theorem on large deviation rate .................. 21

2.4 Importance Sampling .......................................... 26

2.4.1 The methodology .......................................... 27

2.4.2 Conditions for bounded relative error in IS ................. 28

2.4.3 Exponential twist; the asymptotic decay rate ............... 31

2.4.4 More examples: rare events in a single GI/G/1 queue ....... 36

2.4.5 Exponential twist in multi-dimensional state space ......... 40

2.5 Cases of Inefficiency Reported in IS Simulation ............... 43
2.6 Importance Splitting ............................................ 45
2.7 Inefficient Cases Reported in Importance Splitting .......... 46

3 Efficiency Consideration of Fast Simulation – A Large Deviation Perspective 48
3.1 Exponential Twist Parameters and Large Deviation Rate Function .... 49
3.2 Analysis of Inefficient Cases in Exponential Twist – Literature .... 51
3.3 Behavior Perspective of Inefficient Cases in Exponential Twist ..... 53
  3.3.1 Large deviation behavior of free random walks ................. 54
  3.3.2 Large deviation behavior of restricted random walks .......... 57
3.4 Remarks ............................................................ 59

4 Rate Tilting of Level/Phase Processes 60
4.1 Background ......................................................... 62
  4.1.1 Level/phase model ........................................... 62
  4.1.2 The hitting probability, some metrics of interest ............... 63
  4.1.3 An example; complex boundary behavior ....................... 66
4.2 The Rate-Tilting Transform .................................... 67
  4.2.1 The construct of rate-tilting ................................ 67
  4.2.2 The formula of the likelihood ratio (LR) ....................... 71
4.3 Effectiveness of Rate-Tilting ................................... 73
  4.3.1 The geometric decay of $\gamma_n$ ............................. 74
  4.3.2 The bound for $\bar{L}_n/\gamma_n$ ............................... 78
4.4 Some Experiments on Tandem Queues .......................... 82
  4.4.1 Calculation of the relative error for the estimator ............. 82
  4.4.2 Experimental results ......................................... 83
  4.4.3 Verification of fast simulation ................................ 87
4.5 On Pilot Study Using Approximation and Simulation .......... 88
4.6 Comparison with some previous methods that change event rates .... 90
  4.6.1 Comparison with the Parekh-Walrand estimator ............... 90
  4.6.2 Complete cycles, A-cycles, and estimators for stationary probability involving rare event set .......................... 91
  4.6.3 Comparison with the method of Kroese & Nicola ............... 94
  4.6.4 The result of this comparative study ......................... 96
4.7 Summary ............................................................ 97

5 Further Study on Rate Tilting, Simulation Efficiency and System Behavior 98
5.1 Rate Tilting vs. Exponential Twist: Similarities and Differences .... 99
5.2 Main Results ....................................................... 107
5.3 P-F Right Eigenvector As Asymptotics of the Local Potentials ..... 109
5.4 Some Results From the Matrix Geometric Solution ................ 113
  5.4.1 The matrix geometric solution and the dominant term in phase distribution ........................................ 114
  5.4.2 Factorization of the matrix $\Gamma(x)$ ........................ 116
5.5 The Asymptotic Phase Distribution in Tilted System ............... 117
  5.5.1 An eigenvalue correspondence theorem in rate tilting .......... 118
  5.5.2 Asymptotic phase distribution – typical and large deviation ... 122
5.6  Report on Some Numeric Calculations ........................................ 123
5.7  Outlook ....................................................................................... 129

6  A Study of the Adaptive Splitting Method ...................................... 131
   6.1  The Model and The Splitting Strategy ....................................... 133
   6.2  Adaptive Splitting: Making It Efficient ..................................... 136
       6.2.1  Variance and simulation cost of the adaptive splitting ......... 137
       6.2.2  Efficiency of different $h$-strategies ................................. 139
   6.3  Savings of Simulation Cost and Inconsistency Index in Adaptive Splitting ......................................................... 143
       6.3.1  Inconsistency index ............................................................. 144
       6.3.2  Savings of simulation cost ..................................................... 145
   6.4  A Numerical Experiment ............................................................ 147
   6.5  Summary .................................................................................. 149

7  Summary of Contributions, Reflections and Remarks ..................... 150
   7.1  The Main Issues ......................................................................... 150
   7.2  Summary of Contributions ........................................................ 152
   7.3  Outlook .................................................................................... 154

References ......................................................................................... 155
List of Figures

1.1 An illustration of fast simulation using IS ........................................... 4
1.2 An illustration of fast simulation using penalty redistribution ............. 7

2.1 An interpretation of the Legendre transform ........................................ 24
2.2 A diagram of the M/M/1 model ............................................................. 30
2.3 A diagram of splitting with splitting degree 2 ....................................... 46
2.4 The inconsistency in setting up the thresholds ...................................... 46

3.1 Multiple entry points to the target set in \( \mathbb{R}^2 \) .......................... 55
3.2 An example of random walk in \( \mathbb{R}^2 \), with and without barriers .......... 57

5.1 A diagram of the P-F right eigenvector \( \eta_{(i)} \): models 1(a) and 1(b) .... 126
5.2 A diagram of the P-F right eigenvector \( \eta_{(i)} \): models 2(a) and 2(b) .... 126
5.3 A diagram of the P-F left eigenvector \( \vec{u}_1 \) (in tilted system): models 1(a) and 1(b) ................................................................. 127
5.4 A diagram of the P-F left eigenvector \( \vec{u}_1 \) (in tilted system): models 2(a) and 2(b) ................................................................. 127

6.1 A diagram of splitting with splitting degree 2 ........................................ 132
6.2 A diagram of two-threshold model ....................................................... 133
6.3 The speed-up factors by using adaptive splitting .................................. 148
List of Tables

4.1 Efficiency of rate tilting: tandem queues, model 1a) . . . . . . . . . . . . . . . . . . . . . 84
4.2 Efficiency of rate tilting: tandem queues, model 1b) . . . . . . . . . . . . . . . . . . . . . 84
4.3 Efficiency issue: tandem queues, model 1c) – movable server model . . . . . 85
4.4 Sensitivity analysis of rate tilting, using model 1a) . . . . . . . . . . . . . . . . . . . . . 87
4.5 Comparison of efficiency: rate tilting vs. the method of Parekh & Walrand 91
4.6 Efficiency comparison of rate tilting vs. the method of Kroese & Nicola . . 94
Chapter 1

Introduction

To evaluate the performance of stochastic systems, Monte-Carlo simulations are frequently used. However, when the metrics for evaluation depend heavily on events that occur only rarely, it is a great challenge to simulate efficiently. This dissertation investigates approaches to reduce the computational challenge of the problem.

1.1 Performance Evaluation Via Simulation

This thesis deals with stochastic systems. A system is a collection of entities that interact while they evolve. With respect to the objective of my study, a system at any particular instant is described using a set of variables: the state variables. An event is an occurrence that triggers an instantaneous change of the state, e.g., an arrival or a departure in a queueing scenario. If the generation of events is governed by a stochastic process, or if the effect of an event has a random component, then the system is a stochastic system. Moreover, according to whether the changes of state occur continuously, or at separate points of time, one categorizes systems as discrete systems and continuous systems. A discrete system (also a discrete event system, or DES) is the one in which the state variables change instantaneously at separate points of time. In this thesis, only discrete systems will be discussed.

The quality of a system is measured by some performance metrics, such as the expectation of the proportion of time in a state, the probability of entering a rare event set in a regenerative cycle, etc. Sometimes so-called penalties are used to specify the performance metrics. Penalties are associated with each individual replica of the system according to the replica's history. They are actually functions, which can be either state-based (a
cost is accumulated on a per-time-unit basis or per-visit basis) or event-based (a cost is
induced on the change between states). One can then specify the performance metric as
the expectation of the penalty.

One distinguishes between two types of systems: terminating systems, where there is a
“natural” event that terminates each run, and non-terminating systems. For terminating
systems, performance metrics are usually referred to as “transient performance metrics”. For non-terminating system, one uses metrics based on averages, either averaged over a
fixed time horizon or over regenerative cycles, which are defined as below. For a process,
we may be able to identify some epochs \( \{t_i\}, i = 1, 2, \ldots \) such that for any \( i \), the sample
path after \( t_i \) is completely independent of the path before \( t_i \). Such points \( t_i \)'s are called
regeneration points, and an regenerative cycle is the interval between two consecutive re-
generative points. A system may be non-terminating, yet can be divided into regenerative
cycles, where the marking event is the return to a regenerative state. For instance, in a
stable M/G/1 queue, an idling server with no customer in the system is a regenerative state. Note that for a finite state Markov chain, every state in the chain is a regenerative state. The expected length of a regenerative cycle usually depends on the state chosen.
The so-called cycle averages are on “per regenerative cycle” basis.

For Markov chains and Markov event systems, there are analytic algorithms that can
calculate the metrics sought, but the computational complexity may be prohibitive. We
know that the size of the state space increases exponentially with the number of state
variables. As a consequence, the space of possible outcomes can be very large, so to
analytically calculate the performance metrics may be close to impossible. If this is the
case, one must rely on Monte-Carlo simulation, which is a computer-based statistical
sampling experiment.

We might have some set of outcomes (e.g., the system entering some set of states),
that occur only rarely. When a metric heavily depends on such a set of rarely occurring
outcomes, the system will be referred to as rare event system.

In the literature, the term “event” has two different meanings: 1) changes of system
states (e.g., arrivals to a queue), 2) a set of outcomes, such as entering a specific set of
states. The fact that there are two meanings may unfortunately cause some confusions,
but not much can be done about this, since both usages are common. Readers have to infer the exact interpretation from the context.

Recently, much effort has been put into investigations of efficient simulation methods when rare events are involved [32]. The reason is that rare events are often very important. Examples include system failure, a high error rate in data communication, a very long packet delay in a communication system, and so on. Such events, though rare, can induce extreme penalties to the system performance. Since users expect service of high quality, the technical standards for performance are becoming more and more stringent, e.g., it is required in some standards that the cell loss rate should be less that $10^{-7}$ [63]. There is a great interest to simulate rare event systems, which is a challenging task. If we want to find the probability of a rare event, then it is possible that the absolute error of the estimate is small yet the relative error is large. This large relative error is due to the lack of observations of the rare event of interest. To have sufficient observations to derive a statistically significant estimate, we are forced to simulate for a very long period. Hence, to reduce the simulation run length significantly, we must use techniques which modify the system in some way, instead of simulating the system as is.

1.2 Importance Sampling Methodology and an Analogy

*Monte Carlo simulation* is a computer-based statistical sampling experiment. In simulation, the sampling space (denoted by $\Omega$) is the space of all possible outcomes of a replica’s evolution. We generate a limited number of *replicas* of the system, their evolution forming respective *sample paths* (also called *replica evolution paths*). We denote a sample path by $\omega$, $\omega \in \Omega$. The metrics sought are analyzed from a collection of the sample paths. Hence, for the assessment of simulation, sampling theory is applicable.

What is the *efficiency* of a simulation? To explain this term, assume the computation budget is fixed. Given this fixed budget, if we can obtain a more accurate estimate from approach $B$ than from approach $A$, then we say that approach $B$ is more efficient. The theme for the investigation of rare events simulation is, given that we have specified both a system and a performance metric of the quality of the system, how we can simulate
efficiently. In this thesis, the term efficient simulation techniques refers to the related methodology and approaches in efficient simulation. Efficient simulation is also referred to as fast simulation.

### 1.2.1 The methodology

To answer the challenge of simulating rare event systems, several approaches to enhance the simulation efficiency have been proposed. Here I review the essential idea of importance sampling (or IS, for short), which is a mainstream efficient simulation method. Most of the work in this research will in some way relate to the importance sampling.

![Diagram](attachment:image.png)

**Figure 1.1:** An illustration of fast simulation using IS

In importance sampling, there are two system models, with different system parameters. One is the original model \( Q \) (referred as the “target system”), and we are interested in some performance metrics of \( Q \), the other is a modified system \( \tilde{Q} \) which is to be simulated. \( \tilde{Q} \) will be referred to as the “shadow system”. The key link between the target and the shadow system is that the performance metric \( \gamma \) of the target system \( Q \) must be recoverable from observing the outcomes in the shadow system \( \tilde{Q} \). As shown in Figure 1.1, an importance sampling simulation goes like \( A1 \rightarrow B1 \rightarrow B2 \rightarrow A3 \). The functionality of each stage is as follows:

- **Step 1** (\( A1 \rightarrow B1 \)). Derive a shadow system from the target system.
It is worth noting that deriving a good “shadow system” is not trivial: it has attracted much research effort.

- **Step 2** ($B1 \rightarrow B2$). Simulate the shadow system.

- **Step 3** ($B2 \rightarrow A3$). Find a relation between the performance measure from the outcome of the shadow system, and the performance metric of the original system. This relation has to be derived carefully.

There is another technique called *importance splitting*. I will discuss this topic in section 2.6.

### 1.2.2 An analogy

To explain the intuition behind IS, the following analogy will be used. We all understand that scientific experimentation or observation can be made more efficient with some prior information on what to observe and where we are more likely to observe a phenomenon of interest. With this knowledge, we can create the conditions favorable to the occurrences of the phenomenon. For example, consider using a microscope to observe cells of biological organisms. To ensure a proper observation, there are two relevant questions: a proper magnifying scale, and the place where we should focus. The proper magnifying scale depends on the size of the objects: the cells. The focus of the microscope will become more important if we want to observe particular cells, say, malformed cells. By random focusing, the chance of observing a malformed cell is very small, even if we know it does exist. In simulation for rare events using IS, the situation is very similar. The “magnifying scale” corresponds to the increased likeness to observe events of interest, and “where to focus” should be answered by the sample paths that dominate the contributions to the metrics of interest (such metric may be the probability of entering a certain set of states in the system). Precisely, the corresponding prior information is the following:

1. the *asymptotic decay rate*, which helps to decide “magnifying scales”,

2. the *most likely path*, which helps to decide “where to focus”.

5
Both concepts of the *asymptotic decay rate* and the *most likely path* will be elaborated in Chapter 2.

Compared to applying a microscope, in IS simulation, there are differences. First, the change of system dynamics must be made before we start a simulation. Second, if we find that the change of system dynamics results in a simulation that is not optimally efficient, and we decide to change to other system dynamics, we are already wasting simulation effort. Hence, apply IS properly is difficult.

1.2.3 Some points of view

IS simulation is much more complex than direct simulation. This accounts for why some simulation practitioners have resisted using IS simulation [36]. There, the author made the following comment:

One problem with almost all such (importance sampling) technique is that their careless use may backfire, that is, lead to increase in variance. These techniques have been developed by statisticians for use by statistically sophisticated analysts only. Their use by beginners is not recommended.

However, there is a great potential to reduce simulation effort significantly. In [59], the author made a discreet but an optimistic remark:

Hence, it is important to stress that not all IS techniques are overly complex, and many have an intuitive basis . . . It is always worth trying some basic IS methods since, for simple systems, useful run-time savings may result without any great technical effort from the simulator. On the other hand, substantial improvement of the simulation run time for a complex system is often very difficult, and may constitute a considerable research task.

Readers may consult Townsend et al. [61], where a variety of examples are given in application of importance sampling. In these examples, by employing IS properly, the simulation effort can be reduced by orders of magnitudes.
1.3 Redistribution of Penalties

The redistribution of penalties is based on a different idea. This method has not yet been published, but is currently being investigated [24]. The method applies to the situation where the performance metric sought is expressed by average penalty (either time average or cycle average).

Note that there is just one system: the target system. One does not change the dynamics of the system, therefore distribution of the possible sample paths is not changed. The idea is that the penalties ascribed to the rare event set are redistributed to those states that are close to (i.e. very likely to enter) the states in the rare event set, meanwhile keeping the expectation of estimates for the metrics unchanged.

Through the redistribution of penalties as I just described, it is possible to reduce the variance of the estimator. If the performance metric $\gamma$ can be known beforehand, then we may assign $\gamma$ to every individual replica. In particular, if the metric sought is an expectation which is estimated by a time average, and we let a penalty of $\gamma$ be accumulated per time unit in a simulation run, irrespective of the system state, then we will definitely get $\gamma$ as the estimate for the time-averaged penalty, with zero variance.

Realistically, we do not know the metric beforehand. However, this discussion does
provide a guideline that penalties will be distributed as evenly as possible among all possible states. How can this be done? When the performance metric is a time-averaged penalty, it is possible to make use of the balance equation on the rates or transition probabilities to redistribute the penalties. If a penalty function is specified to induce such a performance metric, then obviously the penalties assigned to the designated rare event set (say, $C$) should be much larger than the penalties outside $C$. To get the penalties more evenly ascribed to all states, one aims to reduce the penalties inside $C$ and to increase the penalties outside. With this strategy, consequently, those frequently visited states get a little increase in the penalty; meanwhile, the penalties for the rare event set drops significantly. In another work [24], it is shown with an example that a more accurate estimate can be expected. Here, I mention the method as a fresh idea of fast simulation. However, there is a major challenge of this method, that is, to find a schema of distributing the penalties that is certain to reduce the variance and yield efficient simulation in many situations. This method is currently not as well understood as IS. I do not pursue the method further in this thesis.

1.4 An Overview of the Dissertation

1.4.1 Assumptions

In application where the events of interest are rare, the simulation efficiency will become a serious concern. Yet, we have to rely on Monte-Carlo simulation in many situations of evaluating performance, since analytically calculating the performance metrics entails too much work, thus is close to impossible. In the development of this dissertation, we always assume some events of interest are rare, and we assume that efficient simulation methods are developed for those applications where simulation must be used.

Note that efficient rare event simulation is a difficult problem, and we cannot expect to have a panacea which works efficiently for all situations. It is important that fast simulation methods be evaluated critically: I will indicate both their merits and limitations.
1.4.2 The focus

Most of this work will be applying the importance sampling principle, and, to a lesser extent, the importance splitting. Importance sampling (IS) is a mainstream method used to enhance simulation efficiency, and great successes have been achieved using the method. However, cases have been reported in the literature [20] [18] where some approaches applying IS fail to provide the simulation efficiency we want, and this has caused some worry and heated discussion in recent years [20] [21].

The goal of this dissertation is to provide some analysis on why importance sampling (IS) or importance splitting fails to be efficient sometimes, and improve the existing fast simulation methods on the efficiency. Most results on the efficiency will be theoretically established. For example, I will show that when the event of interest becomes rarer, my proposed methods keep the relative error bounded, and when the state space increases, the simulation effort using these methods increases slowly. For experimentation, small to medium stochastic models are used, in order to ease the burden on describing and validating simulation models. The purpose of these experiments is to verify the correctness and the efficiency of the estimator, or show that when the state space becomes large, the proposed algorithms scale up well. Hence, we use models of small to medium size as test benches. The emphasis is placed on the exploration of new methods, and I have not yet simulated very large systems. Of course, models with large space are important. Indeed, the rare event methods become more advantageous when dealing with large models.

1.4.3 Relation to other studies

Because of the importance of rare event simulation, there is considerable literature on this topic. The researchers work in different, but closely related directions.

There is one type of rare event simulation which I should describe first, even though it is not in the center of my work. In this type of simulation, either known frequencies, or their distributions can be extracted from the empirical data. Approaches in this direction can be combined with the study of system policy, with the emphasis placed on how different policies perform when taking into account the occurrences of rare events. For example,
suppose in a service node, different clients will bring different workloads to the server, and suppose the service time is proportional to the workload. The workloads conform to a probabilistic distribution, which is known. Moreover, the exact workload of a client is known when it arrives. The goal is to assess the scheduling policy to serve the waiting customers. If we want to minimize the total waiting time experienced by all clients, it is better to process the light workloads first. However, we do not want to penalize the clients with heavy workloads to such an extent that they would never go through the service or experience outrageously long waiting time. Here, a perfect knowledge of the occurrence probability of the rare events of interest is presumed to be known.

Now suppose we have several scheduling policies that we want to assess through simulation. The method to handle this situation is called probing method, or the virtual measure method, also known as the method of asking “what if” questions, as described in [4]. This method has been in the literature for a long time. For the example just described, we can “inject” clients with heavy workloads into the system at a higher frequency. These injected events are called probes. They will provide more opportunities to assess the effect of heavy workloads with respect to the different policies. In this scenario, both the original occurrence probability of the rare events of interest (assumed or known from empirical data) and the sampling frequency of rare events (as by our choice) are known, so it is not difficult to calculate their ratio, and use that ratio to assign weights to our observations. By this we can obtain the performance metrics in the original system when taking into account the rare events.

In the rare event problem that I will address in this thesis, the rare event probability is the goal we are seeking. In comparison, the question discussed above has a different focus, where the investigation of the consequence of the rare events to the system is the goal, and the rare event probability is assumed to be known. These kinds of questions do have an important link to my study: once one has found the probability of entering a certain rare event set, then the “probing” method may be applied to have a quick examination on how different system policies affect the system performance metrics. I call the problem just mentioned the “just-in-case” decision problem, which is certainly of practical interest.

In many cases relevant to this study, the probability of entering a rare event is given by
the tail distribution of a certain random variable of interest. For instance, large deviation in a random walk, or, extreme congestion or delay in a stable service network are given by certain tail distributions. Since there is a link between the occurrence probability of rare events and the tail distribution of a random variable, it is desirable to characterize how fast the tail distribution vanishes. This work dates back to Cramer (1938) [11], who considered a random walk without a boundary, also called a free random walk. Of course, in practical scenarios, random walks can have boundaries, and this is the topic of Chapter 3.

There are a number of reasons why the asymptotic decay rate alone is not sufficient. One reason is that, for the purpose of performance engineering, such approximations using the asymptotic decay rate might not be accurate enough (see e.g., [63]). This provides the motivation for many recent investigations on efficient simulation methods.

1.4.4 The plan

In the past, there has been considerable effort devoted to developing efficient simulation techniques to estimate rare events. However, this problem is not yet completely solved. The goal of this thesis is to explore new simulation techniques that are easy to implement and that are efficient.

The rest of this thesis is arranged as follows.

Chapter 2 will review some background material relating to rare events, simulation efficiency, and the principle and application of the IS methods. Particular attention is paid to the exponential twist, a specific IS method that has found wide applications, and a number of examples are given. Cramer’s theorem on the large deviation rate function and its connection with the exponential twist will also be discussed in this chapter.

Chapter 3 will examine the situations where applying the exponential twist method fails to be efficient. These cases are frequently encountered in applications, so they cannot be dismissed as merely exceptions. Instead, they reveal severe limitations of the exponential twist. I analyze the reasons for the inefficiency reported in these cases. It will be indicated that many existing fast simulation methods fail to be efficient, because they cannot effectively generate the most likely paths leading to the rare events of interest.
This is due to the complicated system behavior when boundaries are imposed to the state space, or due to complications arising in multi-dimensional state spaces. This analysis serves as a guide and a convenient reference point for later development.

Following Chapter 3, efficient simulation methods are developed and methods reported in literature are improved. There are certain circumstances where the proposed methods work efficiently. Under these circumstances, it will be shown that the new methods can effectively generate the paths that are likely to lead to the rare events of interest. Chapters 4 and 5 will discuss a specific importance sampling method, namely, rate tilting applied to level/phase processes. In that chapter, it will be suggested that the use of rate tilting can lead to efficient simulation under certain condition, called “compliant boundary” condition, which ensures the efficiency of rate tilting. Chapter 5 shows several different large deviation behaviors, and their respective consequences in simulation efficiency. Chapter 6 will extend the importance splitting technique in literature to the adaptive splitting methods for the multi-threshold setting, where on crossing a threshold, different splitting degrees are allowed for different replicas.

In the final chapter (Chapter 7), I will summarize the main contribution and conclude this work.
Chapter 2

Fast Simulation: Background

In recent literature, the discussion of efficient simulation (also called “fast simulation”) usually goes together with rare event simulation. The reason is that rare event systems pose particular challenges to simulation efficiencies. Note that the simulation efficiency degrades drastically in the presence of rare events, so a modest enhancement of efficiency does not help much. Consequently, in rare event simulation, we are seeking a pronounced enhancement of efficiency.

This chapter contains various background on fast simulation. Section 2.1 discusses the criteria of simulation efficiency. Following this, I discuss some specifications for the stochastic processes used, i.e. discrete event systems vs. Markovian systems (Section 2.2), and random walks (Section 2.3). The random walk is the model used by Cramer [11] for presenting his large deviation result, which is the basis for many fast simulation approaches. Cramer’s theorem is presented in Section 2.3.1. After that, the importance sampling (IS) method and the importance splitting method are reviewed. In Section 2.5, some inefficient cases of IS reported in the literature are included.

Heidelberger [32] gave a good review of fast simulation approaches and their applications. Here, the focus is different. I emphasize the basic principles that support the fast simulation approaches. My intention is to provide a good basis for understanding the merits and the limitations of IS, and to seek remedy or improvement when the existing (IS based) fast simulation approaches fail to be efficient.
2.1 What Is Efficiency and “Efficient Simulation”?

Because Monte Carlo simulation is a computer-based statistical sampling experiment, we must use statistical techniques to design and analyze simulation experiments. Several statistic terms must be defined before proceeding to details. An estimator is a function obtained from a series of simulation runs to estimate the metric in question, and it can be regarded as a random variable. If the expectation of an estimator is the same as the true value of the metric, then we say the estimator is unbiased. In this situation, the error of the estimator comes only from the variance. The expectation of a random variable is usually estimated using an average. When the variable is sampled from independent identical distributed (i.i.d.) samples, we conclude from the central limit limit theorem that the distribution of the value for the estimator conforms to a normal distribution. Based on this, the confidence interval, that is, an interval which includes the true value of the metric with a prescribed probability, can be calculated. In calculating the confidence interval, the variance plays a central role in assessing the accuracy of an estimate. This method uses the normal distribution approximation for the distribution of the estimator, so it deserves noting that the confidence interval may be not accurate enough for small samples.

The accuracy of an estimator is measured by its statistical error. We make this statement precise as follows. The difference between the result produced by an estimator and the true value of the metric is called absolute error. The ratio of the absolute error to the true value of the metric is called relative error. From the analysis above, provided we produce the estimate from large i.i.d. samples, and there is no bias, the variance and the true value of the metric together settle the distribution of the estimator error. In this situation, one can use variance to measure the accuracy of the estimator.

The accuracy of an estimator is decided by both the estimation method and the number of samples. Typically, the variance decreases when more samples are taken. It is thus unfair to compare the efficiency of two sampling methods without mentioning how many samples are taken in each method. In sophisticated sampling, the cost for taking each samples may be different. To account for this, we introduce a concept work, which is the effort involved in sampling, or, in the context of this study, the simulation effort. A good
measure of the efficiency for employing the method will be,

\[
\text{work-normalized variance} = \text{work} \times \text{variance}. \tag{2.1}
\]

This measure was suggested by Glynn and Whitt [22], and actually dates back to Hammer-sley and Handscomb [31]. Comparing the work-normalized variances of two simulations is equivalent to comparing the variances resulting from a fixed simulation budget. To support the notion of “work-normalized variance”, consider independent sample surveys. If \( n \) samples are taken, and we estimate the expectation by the arithmetic mean, then the variance of the estimator is inversely proportional to \( n \). Thus, the variance of the estimator multiplied by \( n \) is an invariant.

The work (i.e., simulation effort) is expressed as a cost, which means that a cost model is required to measure the effort involved in a particular simulation. Also, note that if we are aiming at an efficient simulation, there will inevitably be some analytic effort involved (unless we adopt a naive simulation, which is usually inefficient for rare event simulation). This analytic effort takes place prior to the simulation, and is typically ignored, but should be incorporated in a more elaborate version to measure efficiency. However, I will include only the simulation effort, and will ignore the analytical effort prior to the simulation, even though there is a trade-off between the analytic effort spent prior to simulation and the simulation efficiency.

Above, we compare the efficiency of one method with another method. This is called \textit{comparative efficiency}. Alternatively, one may study the degradation of efficiency of one particular method as the events become rarer. This is called the \textit{asymptotic efficiency}, which is discussed now.

We consider a series of performance metrics \( \gamma_n \), where \( \gamma_n \to 0 \) as \( n \to \infty \). This is a typical situation in the study of rare events. The discussion below basically follows the discussion in [19]. In rare event simulation, one typically requires a reasonably small relative error, and therefore tries to bound the relative error. In the following, we assume that the budget for the simulation effort is fixed. The criterion for a \textit{bounded relative error}
can be expressed as:

$$\limsup_{\gamma_n \to 0} (\text{Var}[\hat{\gamma}_n] / \gamma_n^2) < \infty. \quad (2.2)$$

Here $\hat{\gamma}_n$ is an estimator for $\gamma_n$. The ratio $\text{Var}[\hat{\gamma}_n] / \gamma_n^2$ is the square of the coefficient of variation.

From (2.2), and taking logarithms, we have $\limsup_{\gamma_n \to 0} (\log \text{Var}[\hat{\gamma}_n] - 2 \log \gamma_n) = D < \infty$ where $D$ is a constant. Dividing both sides of the formula considered by $\log \gamma_n$, and noting that $\log \gamma_n \to -\infty$ as $n \to \infty$ (since $\gamma_n \to 0$ as $n \to \infty$), we have

$$\limsup_{\gamma_n \to 0} (\log \text{Var}[\hat{\gamma}_n] / \log \gamma_n) - 2 = 0. \quad (2.3)$$

Though equation (2.3) follows from equation (2.2), I will show later that the converse is not true, thus (2.3) is weaker. The criterion suggested by (2.3) is called as the asymptotically optimality.

If we drop the assumption that the budget for the simulation is fixed, then the variance in (2.3) must be substituted by the product of variance and effort; the effort is denoted by $\text{Work}(n)$. The asymptotically optimality is now defined by:

$$\limsup_{\gamma_n \to 0} \left( \log \text{Var}[\hat{\gamma}_n] + \log \text{Work}(n) / \log \gamma_n = 2. \right.$$  

A simulation method that achieves asymptotic optimality is considered as a good method.

The following example shows that the bounded relative error and the asymptotic optimality are indeed different.

**Example 2.1** This example shows the difference between bounded relative error and asymptotic optimality.

Let us suppose the metrics sought take the form of $\gamma_n = cx^n$, where $c$ and $x$ are known constants. On the other hand, suppose we have constructed an estimator $\hat{\gamma}_n$ such that

$$\text{Var}[\hat{\gamma}_n] = dnx^{2n},$$

where $d$ is an arbitrary constant. At this point, we assume that the simulation effort is fixed, hence we do not need to account for $\text{Work}[n]$. Now I show that the estimator $\hat{\gamma}_n$ is asymptotically
optimal, but it does not have a bounded relative error. For this purpose, we do some calculation,

$$
(\text{Var}[\hat{\gamma}_n]/\gamma_n^2) = (dn^{2n})/(c^2 n^2) = n(d/c^2).
$$

Obviously, as $n \to \infty$, the formula above approaches $\infty$. Hence the estimator does not have a bounded relative error. On the other hand, by straightforward calculation, when $n \to \infty$, we have

$$
\frac{\log \text{Var}[\hat{\gamma}_n]}{\log \gamma_n} - 2 \to 0.
$$

By definition, this implies that the estimator $\hat{\gamma}_n$ is asymptotically optimal.

In the rest of this work, I will use the criterion of the bounded relative error for the development of efficient estimators. Estimators with bounded relative error are of course asymptotically optimal (when the estimators are required to be only asymptotically optimal, some of the conditions in later development can be relaxed).

### 2.2 Stochastic System Models – Discrete Event Systems vs. Markov Event Systems

Discrete event systems (DES), introduced in Chapter 1, are very general. It is typically difficult to solve such systems analytically or assess the relevant performance metrics. Consequently, additional restrictions are imposed to make the system Markovian (Markov event system, or MES), so as to be more tractable.

Although the state space of a DES can be arbitrary, for the tractability reason, I use a $d$-dimensional vector, with each component taking integer values, to represented the system state. The change of the system state is triggered by events, and each event has a corresponding effect. The effect is expressed by the difference between the state before the event and the state after the event (again a $d$-dimensional vector). When we simulate a DES, the events have to be scheduled. The time between the scheduling of an event and the time the event takes effect in the system will be called incubation time. (Another term is “inter-event” time, which is basically the same as “incubation time”; however, some events may be dormant during a certain time interval, like an idling server has no service completion). Generally, when describing the dynamics for a DES, we specify the distribution function of all incubation times. If all incubation times are independent
exponential random variables, then the time since the event in question was scheduled
does not matter [26]. In this situation, each event has a release rate that depends only on
the current system state. One calls such systems Markovian systems.

One can also view the system evolution history as a sequence of transitions among the
states, and associate each possible transition with a release rate. In this regard, we are
using the continuous time Markov chain (CTMC), with the release rates for all possible
transitions specified in the infinitesimal generator matrix. If the time spent in each state
of the system is not of concern, then one can consider only the state immediately following
the $i$-th transition, $i = 1, 2, \ldots$ In this way, the corresponding process becomes a discrete
time Markov chain (DTMC). This DTMC is embedded in the original CTMC. Of course,
there are also systems where the model is such that the events occur at some fixed points of
time. For example, in communication, the time is often slotted, with packets of messages
arriving and departing at only the start (or end) of each slot. Such models are also
conveniently modeled by a DTMC.

On the other hand, simulation practitioners would prefer to use events, their effects
and release rates to describe a system. This may also be practically convenient since
the number of events is much smaller than the number of all possible transitions. In
continuous time stochastic processes that are event based, the event release rate $r(t_0)$ and
the distribution of the incubation time are related. Let $T$ be the incubation time of a
certain event, and let $F(T) = P(T \leq t)$ be its distribution. Now suppose the time since
the event in question was scheduled is $t_0$, then the event release rate at instant $t_0$ is the
density of the incubation time distribution, conditional on that this event has not been
released in $[0, t_0)$. Note in the formula for $r(t_0)$ below, $F'(t_0)$ is the density function at $t_0$,
and $\hat{F}(t) = 1 - F(t)$.

$$r(t_0) = \frac{F'(t_0)}{\hat{F}(t_0)}.$$ 

Note that when $F(t)$ is exponentially distributed, $r(t_0)$ is independent of time $t_0$. Furth-
more, if all these event release rates depend only on the current state, then the system is
Markovian, and from these rates it is not difficult to find the infinitesimal generator ma-
trices. The infinitesimal generator is a typical starting point for many analytic methods.
Also, details of constructing the infinitesimal generator matrix are available in [7] [8]. For this reason, the infinitesimal generator matrix is often assumed known in later discussion.

2.2.1 Multi-dimensional state space, phase-type distributions

When Markov processes are used to model real world scenarios, the dimension of the state space is usually high. Clearly, the model becomes more powerful as we use more variables. On the negative side, the analysis of high dimensional models can become very involved.

We stated that a state can be represented as a $d$-dimensional vector. For a simple example, consider a network of $d$ servers, with each component corresponding to the length of line in front of the respective server. However, state variables need not be length of queues. Indeed, any variable that somehow relates to the system dynamics can be a candidate for a state variable, as the following examples show.

In the queueing context, one may allow different type of customers. Customers belong to different types, in the sense that they come from different sources, or they are assigned different priorities for entering the service, or they have different service requirements (e.g., the distribution of service times required for different type customers may differ). In many situations, it is necessary to provide the number of customers of each type in the system, hence the dimension of the state space is drastically increased.

Another possibility is to add *supplementary variables*, which allows one to deal with non-exponential incubation times. Typically, incubation times are continuous, but including continuous variables in the state space poses severe challenges to the system tractability. However, since exponential incubation times do not lead to continuous supplementary variables, one might be able to use exponential random variables as building blocks for constructing other continuous distributions. This idea leads to the *phase-type distribution* ([46], Chapter 2). In this case, we model (or approximate) a non-exponential distribution, using a number of exponentially distributed “phases”. This approach is called the *method of phases*. In this method one assumes that there are different phases, and the rate of completing the incubation time depends on the phase one is in. More specifically, following Neuts [46], we consider a Markov process on the states \{1, 2, \ldots, m + 1\} with infinitesimal
generator

\[ Q = \begin{bmatrix} Q^* & q^0 \\ 0 & 0 \end{bmatrix} \] (2.4)

where each element in the \( m \times m \) matrix \( Q^* \) is non-negative except for the diagonal. The diagonal element is chosen such that

\[ Q^* \cdot \mathbf{I} + \mathbf{q}_0 = \mathbf{0}. \]

The initial probability vector is given by \((\mathbf{v}, v_{m+1})\), with \( \mathbf{v} \cdot \mathbf{I} + v_{m+1} = 1 \). Here \( \mathbf{I} \) is a vector with all components being 1. A probability distribution \( F(.) \) on \([0, \infty)\) is a phase-type distribution if and only if it is the time until absorption in a finite Markov process of the form given in equation (2.4). We call the pair \((\mathbf{v}, Q^*)\) a representation of \( F(.) \). For models using phase-type distributions in queueing theory and reliability theory (see [42], [46]).

### 2.3 Random Walks and the Large Deviation Rate

We now present the so-called large deviation results, which are the basis for many fast simulation approaches. These results were introduced by Cramer [11] who used the model of random walks. Random walks are actually much simpler than Markov event systems. However, by exploring this model thoroughly, we can gain valuable insight on occurrences of rare events.

We start this section by describing the free random walks along a straight line. Let \( \{X_i\} \) be independent identically distributed (i.i.d.) random variables taking values in the set of real numbers. Let \( S_T = \sum_{i=1}^{T} X_i \), let \( \bar{X}_T = (1/T) \sum_{i=1}^{T} X_i \). We use \( S_T/T \) or \( \bar{X}_T \) to specify the actual drift rate, which is observed from a replica of the random walks during the first \( T \) steps.

To generalize the random walks on a line to random walks in \( d \)-dimensions, let \( \bar{X}_i \) be a vector with \( d \) components. In analogy to the random walks on the line, we define \( \bar{S}_T = \sum_{i=1}^{T} \bar{X}_i \). The resulting process is called free random walks in \( d \) dimension space \( \mathbb{R}^d \). 

20
In practical scenarios, we may have to consider the existence of absorbing or reflecting barriers, a topic addressed in the next chapter. Random walks with either absorbing or reflecting barriers are referred to as the \textit{generalized random walks}, in contrast to the free random walks.

2.3.1 \textbf{Cramer’s theorem on large deviation rate}

First, we introduce the notation of UTLE, which stands for “up to logarithm equivalence”. This notation will be frequently used in later discussion. Consider two sequences \(\{a_n\}\) and \(\{b_n\}\). Suppose that when \(n \to \infty\), both sequences converge to 0. Further, if \(\{a_n\}\) and \(\{b_n\}\) satisfy \((1/n)(\log a_n - \log b_n) \to 0\) as \(n \to \infty\), then we write \(a_n \simeq b_n\) (UTLE). To understand the meaning of this, suppose there are constants \(-\infty < \underline{d} < \overline{d} < +\infty\), such that

\[ \underline{d} b_n \leq a_n \leq \overline{d} b_n. \]

Taking logarithm for each term, and dividing by \(n\), we get

\[ (1/n)(\log \underline{d} + \log b_n) \leq (1/n) \log a_n \leq (1/n)(\log \overline{d} + \log b_n). \]

Obviously, sequences \(\{a_n\}\) and \(\{b_n\}\) are UTLE, since as \(n \to \infty\), we have \((1/n) \log \underline{d} \to 0\) and \((1/n) \log \overline{d} \to 0\). We may also say that when the ratio \(\frac{a_n}{b_n}\) is bounded, sequences \(\{a_n\}\) and \(\{b_n\}\) are UTLE.

Also, for any two sequences \(\{a_n\}\) and \(\{b_n\}\), we have

\[ a_n + b_n \simeq \max(a_n, b_n) \text{ (UTLE)}, \]

which is casually referred to as the principle of “largest-exponent-wins” [34]. This principle can be iterated to apply to finitely many sequences.

Cramer’s large deviation rate function, which is basic to our discussion on rare events, is now introduced. Here, I restrict the discussion to one dimension random walk, however, the large deviation rate function in the following discussion can be extended to multi-dimensional case. Let \(\{X_t\}\) be a sequence of i.i.d. random variables, and consider the
random walk governed by \( \{X_i\} \). Let

\[ M(\theta) = E[\exp(\theta X_1)], \]

where \( \theta \in \mathbb{R} \). One calls \( M(\theta) \) the moment generating function wherever \( M(\theta) \) is finite. The large deviation rate function \( I(.) \) is very important. On the subset of \( \mathbb{R} \) where \( M(\theta) \) is finite, \( I(.) \) is defined as,

\[ I(z) = \sup_{\theta} \{\theta z - \log M(\theta)\}. \]  \hfill (2.5)

Sometimes \( \log M(\theta) \), i.e., the logarithm of the moment generating function, is referred to as the cumulant generating function. The following theorem is due to Cramer [11] (I present it in the following form so as to be accessible to more readers; this is slightly different from the original form).

**Theorem 2.1** For any fixed value \( x \in \mathbb{R} \), and \( \delta > 0 \), one has,

\[ (1/T) \log P(|S_T/T - x| < \delta) \rightarrow -I(x) \quad (T \rightarrow \infty, \delta \to 0^+). \]  \hfill (2.6)

Note the right-hand side of the formula above is (almost) independent of \( \delta \). This is because for any constant \( \delta \), as \( T \to \infty \), \( (1/T) \log \delta \to 0 \) (note \( \delta \) should be arbitrarily small but never be 0).

This formula is basic to the large deviation theory. When the actual drift \( x \) differs from \( E[X_1] \), then as \( T \) increases, \( |S_T - TE[X_1]| = |Tx - TE[X_1]| \) also increases, which is the reason why we call \( I(x) \) the large deviation rate function. Note that for random walks, we differentiate between the expected drift rate and the actual drift rate \( S_T/T \). When the actual rate moves further away from the expected rate, the probability of observing such an occurrence becomes smaller. It can be said that Cramer’s theory quantifies the rate of the probability \( P(|S_T/T - x| < \delta) \) decreases.

Using the notation of UTLE, formula (2.6) is expressed as:

\[ P(S_T/T \approx x) \simeq \exp\{-TI(x)\} \quad \text{(UTLE)}. \]
To extend the discussion above, we determine the probability that the observed rate function falls into a set $D \subset \mathbb{R}$. We have,

$$P(S_T/T \in D) \simeq \sum_{x \in D} \exp\{-TI(x)\} \simeq \sup_{x \in D} \exp\{-TI(x)\} \quad \text{(UTLE, according to 2.3.1)}$$

$$= \exp\{- T \inf_{x \in D} I(x)\}.$$ 

Hence, by defining

$$I(D) = \inf_{x \in D} I(x) \quad (2.7)$$

we can write that

$$(1/T) \log P(\bar{X}_T \in D) \rightarrow I(D),$$

where $I(D)$ is said to be the large deviation rate function on set $D$.

One of the unusual (and non-intuitive) aspect of this theorem is that the large deviation rate function at a set depends essentially on one point $\bar{x}$, which is the point such that $I(\bar{x}) = \inf_{x \in D} I(x)$, and which we call the dominant point of drift with respect to set $D$. Also, if $E[X_i] \in D$, then $E[X_i]$ is the dominant point of drift, and the related probability is 1, as it should be (by the law of large numbers).

Note that the cumulant generating function $\log M(\theta)$ provides much information about the distribution of the random variable. In particular, the slope of $\log M(\theta)$ at $\theta = 0$ is the expected drift rate of the random walk in question. To show this, we calculate as,

$$\frac{d}{d\theta}[\log M(\theta)]|_{\theta=0} = M'(0)/M(0).$$

Here, $M'(0)$ denotes $\frac{d}{d\theta}[M(\theta)]|_{\theta=0}$, and it takes the value of $E[X_1]$, as is well known. It is obvious that $M(0) = E[\exp\{0X_1\}] = 1$. On the other hand, by the law of large numbers, we have

$$S_T/T \rightarrow E[X_1] \quad (T \rightarrow \infty)$$

23
almost surely. Hence, as $T \to \infty$,

$$S_T/T \rightarrow \frac{d}{d\theta}[\log M(\theta)]|_{\theta=0}, \quad (2.8)$$

almost surely, that is the slope of $\log M(\theta)$ at $\theta = 0$ is the drift rate. Technically, $I(.)$ is the Legendre transform of $M(\theta)$.

A geometric interpretation of the large deviation rate function has been proposed in the literature ([34], page 8). To understand this interpretation, consider Figure 2.1. For any $x \in \mathbb{R}$, we can find the slope $x$ on the curve $\log M(\theta)$ at $\theta = \bar{\theta}$. Then we draw a tangent at the point $(\bar{\theta}, \log M(\bar{\theta}))$, which will intersect the axis $\theta = 0$ at $(0, -I(x))$. We will call such $\bar{\theta}$ as the adjunct generating exponent to the drift rate $x$. (If such drift rate $x$ cannot be found as a slope in curve $\log M(.)$, then we set $I(x) = +\infty$.) Also, the tangent of curve $\log M(\theta)$ at $\theta = \bar{\theta}$ has the slope $x$, which implies the tangent being considered is parallel to the line of $\theta x$. This interpretation of $I(.)$ provides insights which will be used later.

![Figure 2.1: An interpretation of the Legendre transform](image)

It has been shown (e.g. [3]) that the moment generating function is always convex, i.e. $M''(\theta) > 0$ for any $\theta$. From this convex property, and the geometric interpretation given above, two further properties follow, as described in the following two remarks.

**Remark 1:**
Let \( \theta \) be the adjunct generating exponent to the drift rate \( x \). Then \( x \) increases as \( \theta \) increases. This is because the slope on the curve \( \log M(\theta) \) always increases.

**Remark 2:**

Consider the function \( I(x)/x \), where \( I(x) \) is the large deviation rate function. Suppose \( I(x)/x \) achieves the minimum at \( x = x^* \). Let \( \theta^* \) be the adjunct generating exponent to \( x^* \). Then one must have \( M(\theta^*) = 1 \). In addition, such \( x^* \) is unique.

Now I give a proof. Note \( M(\theta^*) = 1 \) implies \( \log M(\theta^*) = 0 \). Hence in the geometric interpretation, such \( \theta^* \) is the intersection of \( \log M(\theta) \) with the \( \theta \) axis (denoted as point \( J \) in Figure 2.1), and \( x^* \) is the slope of \( \log M(\theta) \) at point \( J \).

Now take a point \( C \) (\( C \neq J \)) of the curve \( \log M(\theta) \), and draw a tangent at \( C \). Suppose that the slope of the curve at \( C \) is \( x \) (\( x \neq x^* \)). Let the intersection of the tangent and the \( \theta \) axis be point \( A \). Now we show that \( I(x)/x = |OA| \). This follows from

\[
I(x) = |OB|(\text{according to the geometric interpretation of } I(x))
\]

and

\[
x = |OB|/|OA|.
\]

The last equality holds because line \( BC \) is parallel to line \( OS \). On the other hand, As \( C \) moves towards \( J \) along the curve, the point \( A \), which is the intersection of the \( \theta \) axis and the tangent that passes through \( C \), also moves towards \( J \). In this situation, the points \( C \), \( J \) and \( A \) becomes coincident. Consequently, we have \( I(x)/x = |OJ| \). Note that \( \log M(\theta) \) is a convex function, hence it always holds that \( |OA| \geq |OJ| \), with equality only when \( C \) coincides with \( J \), i.e., \( M(\theta) = 1 \).

Hence, we conclude that \( I(x)/x \) achieves the minimum when the adjunct generating exponent \( \theta = \theta^* \), \( \theta^* > 0 \) is such that \( M(\theta^*) = 1 \). To see that \( x^* \) is unique, note that \( x^* \) is the slope at \( J \). Because that point \( J \) is unique, \( x^* \) is unique, too.

There has been considerable effort to generalize Cramer’s theorem. Most relevant to the study of fast simulation are the studies which dispense of the requirement that the sequence \( \{X_i\} \) in random walks must be independent. This is useful when applying
Cramer’s theorem to more involved stochastic processes, where some dependent structure is inevitable. Gartner & Ellis completed this generalization in their work [17].

In spite of these general results, explicit formulas for the exact tail probability are not generally available. Often, a huge amount of work is required to find the exact probability, and this makes explicit calculation impractical. This provides us the reason to resort to fast simulation.

The most likely path

We introduce the concept of the most likely path. Random walks on a line can be represented by a plot with $T$ on the $x$-axis and $S_T/T$ on the $y$–axis. Consider now many replications of the random walk, and let $\Delta_\delta(y) = \{(T, S_T/T) : y - \delta \leq S_T/T < y + \delta\}$ be the set of all possible paths that lie between two parallel lines; as $\delta \to 0$ and $T \to \infty$, these points $(T, S_T)$ lies almost on the line $S_T/T = y$. Now we locate $y = y_0$ such that the rare events of interest occurs at $\Delta_\delta(y_0)$ dominantly, i.e. $P(\Delta_\delta(y_0))$ is exponentially larger than $P(\Delta_\delta(y'))$ for any $y' \neq y_0$, as $n \to \infty$ and $\delta$ sufficiently small. One calls $(T, S_T)$ such that $S_T/T = y$ the most likely path.

Intuitively, rare events are most “likely” to occur along a path that is close to the so-called “most likely path”. Along the path, drift rate $S_T/T$ is almost $y$ (between $y - \delta$ and $y + \delta$), where $\delta$ can be made arbitrarily small. Note that the existence of the most likely path is theoretically based upon Cramer’s theorem (and the “UTLE” analysis).

2.4 Importance Sampling

There is a large body of literature in IS, and it is hard to trace the idea of IS to its origin. IS has been successfully used by many researchers in evaluating performance metrics that depends heavily on rare events. (See [32] for a survey.) Most relevant to this study are the fast simulation methods to estimate the probability of extremely long queues (excessive backlogs) [50] [14].

A slightly different approach is called importance splitting, which will be addressed in section 2.6.
2.4.1 The methodology

The essence of importance sampling has been introduced in Chapter 1. In this method, the probability distribution that generates the samples is changed to encourage the occurrences of rare events, on which the performance metric of interest heavily depends. We are aiming at sufficient observations of events of interest. This allows us to find rare event metrics relating to the original system more efficiently.

The likelihood ratio (also known as the "Radon-Nikodym derivative" in [3] [52]), is the ratio of the probability that an event will occur under the original distribution to the probability it will occur under the new sampling distribution. This ratio depends on the particular sample path. Let \( \omega \) be a sample path observed, \( P(.) \) and \( \tilde{P}(.) \) be the probability measure of the sample path distribution in the original system and the new system, respectively. Then the likelihood ratio \( L(\omega) \) can be expressed as,

\[
L(\omega) = \frac{P(\omega)}{\tilde{P}(\omega)}.
\]

(2.9)

The likelihood ratio plays an important role in IS. It is readily proved that a reduction of the variance of the estimator will result if \( L(\omega) < 1 \) on the rare events of interest [32]. Of course, not every change of system dynamics will enhance simulation efficiency. Some changes may degrade the efficiency, and an example is given by Heidelberger [32] to show that certain change will even result in an infinite variance for an estimator!

We now describe the probability of entering a designated set of states, and how an estimator for this probability can be constructed using IS. Let \( C_n \) be a specific set of states in the system being considered. Consider a sample path \( \omega \) that starts from a designated state (say state 0), and stops when it either hits \( C_n \) or returns to 0, i.e. a sample path in a regenerative cycle. We suppose the probability of entering \( C_n \) approaches 0 as \( n \) increases. The indicator function \( 1_{C_n}(\omega) \) is defined as,

\[
1_{C_n}(\omega) = \begin{cases} 
1, & \text{if } \omega \text{ enters } C_n \\
0, & \text{otherwise}
\end{cases}
\]
Let

\[ \gamma_n = E_P[1_{C_n}(\omega)], \]

(2.10)

where \( E_P \) is the expectation operator, given \( P \), in which \( P \) is the probability measure over all sample paths generated by system dynamics. In later discussion, we also refer \( \gamma_n \) as the hitting probability (on a per-cycle basis).

If IS is applied, the hitting probability becomes the expectation of \( L(\omega)1_{C_n}(\omega) \), that is

\[ \gamma_n = E_P[1_{C_n}(\omega)] = E_P[L(\omega)1_{C_n}(\omega)]. \]

(2.11)

An estimator suggested by (2.11) is as follows. We generate \( K \) replications in the new system, for which the distribution of sample paths is \( \tilde{P}(\cdot) \), then we estimate \( \gamma_n \) by using

\[ \hat{\gamma}_n = \frac{1}{K} \sum_{\{\omega: 1_{C_n}(\omega) = 1\}} L(\omega). \]

We will call this estimator IS estimator.

2.4.2 Conditions for bounded relative error in IS

I discuss now a general guideline to select an appropriate change of measure, such that the relative error is bounded, in accordance with equation 2.2. As before, let \( P \) and \( \tilde{P} \) be the sampling distribution of the original system and of the new tilted system, respectively. Corresponding to the metric \( \gamma_n = P(C_n) \), let us define a new hitting probability in the tilted system as,

\[ \gamma'_n = \tilde{P}(C_n) = E_{\tilde{P}}[1_{C_n}(\omega)]. \]

(2.12)

The following theorem provides the sufficient conditions such that the relative error of an IS estimator is bounded. Results similar to this appear in \[57] \[33]. I choose to present it in the following form to fit the discussion in this thesis.

**Theorem 2.2** In order that the fast simulation estimator for \( \gamma_n \) has a bounded relative error, the following two conditions are sufficient:

(A1) There is a function \( 0 < \Phi(n) < \infty \), and values \( 0 < d_1 < d_2 < \infty \), such that for any
sample path \( \omega \) and for any \( n \), the following inequalities hold,

\[
0 < d_1 \Phi(n) < \inf_{\omega \in C_n} L(\omega) < \sup_{\omega \in C_n} L(\omega) < d_2 \Phi(n) < \infty. \tag{2.13}
\]

Here \( d_1, d_2 \) must be constants that are independent of \( n \).

(A2) In the tilted system, the corresponding hitting probability \( \gamma_n \) is bounded away from 0.

That is, there is a value \( c > 0 \) such that

\[
\gamma_n' = E_{\tilde{P}}[1_{C_n}(\omega)] > c > 0. \tag{2.14}
\]

Note that (A1) indicates the likelihood ratio is well behaved in the sets of rare events, and
(A2) indicates that the rare event of interest is not rare any more under the new selected sampling distribution \( \tilde{P} \). This intuitively explains the enhancement of efficiency based on the new simulation: we have more observations of the rare event of interest, which lead to more accurate estimation.

**Proof:** Let \( \tilde{\gamma}_n \) denote the IS estimator. Define

\[
\epsilon^2 = \frac{\text{Var}_{\tilde{P}}[\tilde{\gamma}_n]}{\gamma_n^2}. \]

Obviously, \( \epsilon^2 > 0 \). If the IS estimator being considered satisfies both conditions (A1) and (A2), i.e., assume equations (2.13) and (2.14). Then

\[
1 + \epsilon^2 = \frac{\text{Var}_{\tilde{P}}[\tilde{\gamma}_n] + \gamma_n^2}{\gamma_n^2} = \frac{E[\tilde{\gamma}_n^2]}{(E[\tilde{\gamma}_n])^2} \leq \frac{(\sup_{C_n} L(\omega))^2}{(\inf_{C_n} L(\omega))} \frac{E_{\tilde{P}}[1_{C_n}(\omega)]}{(E_{\tilde{P}}[1_{C_n}(\omega)])^2} \leq \frac{d_2^2 \Phi^2(n)}{d_1^2 \Phi^2(n)} (1/ E_{\tilde{P}}[1_{C_n}(\omega)]) = \frac{d_2^2}{d_1^2 \gamma_n} \text{ (formula 2.12)} \]

\[
< \frac{d_2^2}{d_1^2 c} < +\infty \text{ (from equation (2.14))}. \tag{2.15}
\]

Hence \( \epsilon^2 < \infty \). Use the definition (2.2) for bounded relative error to complete the proof.
We now examine an example.

**Example 2.2 Excessive backlog in a stable M/M/1 queue.** In an M/M/1 queue, arrivals are Poisson, and service times have exponential distributions. Arrivals that find the server busy will enter the queue. Currently, we assume no limit on the length of queue. We start with an empty queue, and stop as soon as the length of line in the system either reaches $n$ or return to 0. This time will be referred to as a cycle. Let $\omega$ be an outcome whenever we have a complete cycle. If the length of line reaches $n$ in a cycle, we say this is a successful cycle. We are interested in the metric $\gamma$, which is the probability that $\omega$ is a successful cycle.

![Diagram of the M/M/1 model](image)

In an M/M/1 queueing system, an event is either an arrival or a departure, with the respective probability

$$
\lambda' = \frac{\lambda}{\lambda + \mu}, \quad \mu' = \frac{\mu}{\lambda + \mu}
$$

except that when the line is empty, an event must be an arrival. As suggested by Parekh and Walrand [50], to simulate the excessive backlogs in a stable M/M/1 queue, we simulate a system, with the new inter-arrival time and service time distribution still being exponential, however the rates are exchanged, i.e., now we simulate a tilted system with arrival rate $\mu$ and service rate $\lambda$ ($\mu > \lambda$). This is called the Parekh-Walrand estimator,

$$
\hat{\gamma} = E_P[(1_C(\omega)L(\omega))].
$$

In contrast, we denote the direct estimator as $\hat{\gamma}^{\text{dir}}$, which is given as, $\hat{\gamma}^{\text{dir}} = E_P[1_C(\omega)]$.

The likelihood ratio $L(\omega)$ can be calculated as follows. If there are $l$ departures during a success cycle (the cycle reaches $n$), then there must be $n + l$ arrivals (including the
first arrival when the line is empty), bringing the total number of events to \( n + 2l \). The probability of this event is \( P(\omega) = (\lambda')^{n+l-1}(\mu')^l \) under the original measure, and \( \tilde{P}(\omega) = (\mu')^{n+l-1}(\lambda')^l \) under the tilted measure. Therefore, the likelihood ratio is,

\[
L(\omega) = \frac{P(\omega)}{\tilde{P}(\omega)} = (\lambda'/\mu')^{n-1} = (\lambda/\mu)^{n-1}.
\]  

(2.17)

Now consider the variance of the two respective estimators. In the following discussion, let \( m \) be the number of independent samples (note sampling of \( 1_C(\omega) \) are independent of each other). Then their respective variances are

\[
m \text{ Var}[\hat{\gamma}^{\text{dir}}] = \text{ Var}_P[1_C(\omega)] = E_P[1_C(\omega)^2] - \gamma^2
\]

\[
= \gamma - \gamma^2
\]

and

\[
m \text{ Var}[\hat{\gamma}] = \text{ Var}_P[1_C(\omega)L(\omega)] = E_P[(1_C(\omega)L(\omega))^2] - \gamma^2 = E_P[1_C(\omega)^2L(\omega)] - \gamma^2.
\]  

(2.18)

The last equality holds because that \( E_P[L(\omega)1_C(\omega)] = E_P[1_C(\omega)] \). Since \( \lambda < \mu \) (the original queue is stable), \( L(\omega) < 1 \) as \( n > 0 \). Since by (2.17), \( L(\omega) = (\lambda/\mu)^n \), and obviously \( 1^2_{A_n} = 1_{A_n} \), equation (2.18) becomes (\( m \) is the sampling size)

\[
m \text{ Var}[\hat{\gamma}] = (\lambda/\mu)^{n-1} \gamma - \gamma^2,
\]

This indicates a significant variance reduction when \( n \) is large, as compared to the variance of direct estimator.

2.4.3 Exponential twist; the asymptotic decay rate

The method exponential twist is a specific form of IS or “change of probability measure”. The method has been considered in various setting and by several authors [1] [50] [54] [5].

Consider a Markov random walk. Let the sequence \( \{X_i\} \) be an i.i.d. sequence where \( X_i \)
has the cumulative distribution function \( F \) (this will be denoted as \( X_i \sim F \) in the following discussion). Let \( S_T = \sum_{i=1}^{T} X_i, \ T = 1, 2, \ldots, \infty \). Assume \( EX_i < 0 \). For a specified \( s > 0 \) that is sufficiently large, we are interested in estimating the probability that eventually there is an \( T \) such that \( S_T > s \). This probability is expressed as,

\[
q_s = P\{ \exists (T < \infty) \text{ such that } S_T > s \}. \tag{2.19}
\]

Note that in this equation, \( s \) is the rarity index.

In the exponential twist, a new random sequence of \( \{ \tilde{X}_i \} \) is obtained as follows. For a certain parameter \( \theta \), let

\[
d\tilde{F}_\theta(x) = e^{\theta x} dF(x)/M(\theta). \tag{2.20}
\]

Here \( M(\theta) = E[e^{\theta X}] \) is the normalizing factor such that \( \int d\tilde{F}_\theta(x) = 1 \). Now let \( \{ \tilde{X}_i \} \sim \tilde{F}_\theta \), and \( \tilde{S}_T = \sum_{i=1}^{T} \tilde{X}_i \). We will call \( \tilde{S}_T \) an inclined random walk \(^1\).

The hitting probability in the new system (represented as \( \tilde{q}_s \)) is as follows. With respect to the new system, consider

\[
\tilde{q}_s = P\{ \exists (T < \infty) \text{ such that } \tilde{S}_T > s \}. \tag{2.21}
\]

We now verify whether Theorem 2.2 is satisfied. Note when \( EX_i > 0 \), this probability \( \tilde{q}_s = 1 \), which is certainly bounded away from 0. In other words, the event being considered becomes a certain event. For this situation, condition (A1) of Theorem 2.2 is satisfied.

We check now whether the likelihood ratio is bounded uniformly for all paths. Consider a specific sample path until time \( T \), where we take the stopping time \( T \) as the first passage time such that \( S_T > s \), that is:

\[
\omega : (X_1 = x_1, X_2 = x_2, \ldots, X_T = x_T)
\]

If \( P(\cdot) \) and \( \tilde{P}_\theta(\cdot) \) are the probabilities of a sample path, and \( \{X_i\} \) are i.i.d. random

\(^1\)This is also called as tilted random walk. However, we have reserved the word “tilt” for a particular novel method, to be discussed in Chapter 4.
variables, then
\[
L(\omega) = \frac{P(\omega)}{P_\theta(\omega)} = \prod_{i=1}^T \frac{dF(x_i)}{dF_\theta(x_i)} = \prod_{i=1}^T (M(\theta)e^{-\theta x_i}) = (M(\theta))^T e^{-\theta S_T}.
\]

The value of \(M(\theta)\) is important. If \(M(\theta) = 1\), then \((M(\theta))^T\) is always 1, and \(L(\omega)\) is bounded. Otherwise, as \(T \to \infty\), if \(M(\theta) < 1\), then \(L\) approaches to 0; if \(M(\theta) > 1\), \(L\) approaches \(\infty\). It is difficult to bound the likelihood ratio \(L(.)\) in the latter two situations. Hence we decide to choose \(\theta\) such that
\[
M(\theta) = 1. \quad (2.22)
\]

Such a \(\theta\) exists as will be shown later. In the tilted system, \(S_T > s\) occurs with probability 1. Hence we have,
\[
q_s = E_P[L(\omega)] = E_P[(M(\theta))^T e^{-\theta S_T}] = e^{-\theta s} E_P[(M(\theta))^T e^{-\theta(S_T-s)}]. \quad (2.23)
\]

Under proper conditions, it is always possible that \(E_P[e^{-\theta(S_T-s)}]\) converges to a constant as \(s \to \infty\), which means that \(E_P[e^{-\theta(S_T-s)}]\) is bounded (see Asmussen [1]). This, combining with the condition that \(M(\theta) = 1\), leads the result that the likelihood ratio \(L(.)\) is bounded on all paths. This satisfies condition (A2) of Theorem 2.2.

One still needs to know that there is a \(\theta\) such that the twisted random walks have a positive drift (i.e. \(E[X_i] > 0\)) and that \(M(\theta) = 1\), but this has been rigorously established in [1] and [53].

*Remark:* In page 25 of section 2.3.1, we indicate \(I(x^*)/x\) achieves its minimum value when \(M(\theta^*) = 1\). We will show later in section 3.1 that on the most likely path to hit a large deviation set, \(I(x^*)/x\) achieves its minimum, which provides a (more rigorous)
explanation why one should choose \( M(\theta) = 1 \).

The asymptotic decay rate

For fast simulation using the exponential twist, the concept of the asymptotic decay rate is one of the corner stones.

Let us continue the discussion for the example of M/M/1 queue. From formula (2.23), we obtain,

\[
\frac{\log q_s}{s} = -\theta s + \frac{1}{s} \log E_P[e^{-\theta(S_T-s)}]
\]

\[E_P[e^{-\theta(S_T-s)}]\] converge to a non-zero constant. Consequently, \( \frac{\log q_s}{s} \rightarrow -\theta \). Hence the series \( q_s \) has an asymptotic exponential decay rate of \( \theta \).

Above, the rarity index is a real number \( s \in \mathbb{R} \). Sometimes, the rarity index is an integer, e.g., when we use the queue length as the rarity index. For this case, we write \( \gamma_n \) for the metric, and the formula for showing asymptotic exponential decay is as follows:

\[
\frac{\log \gamma_n}{n} \rightarrow \theta.
\] (2.24)

When the process considered has an asymptotic exponential decay rate, then it may be possible for the exponential twist method to induce an effective simulation. Note that the asymptotic exponential decay is a frequently observed pattern, and indeed, the asymptotic decay is the starting point of fast simulation in literature. For example, for the question concerning the probability that the length of queue exceeds \( n \) in a cycle in a M/M/1 queue, it can be shown that the decay rate is asymptotic exponential [50]. Several results are established in the literature. For instance, [63] establishes the asymptotic exponential decay for queues with multiple class customers or multiple service channels, and [5] establishes this result for the discrete queues that typically arise in ATM network.

Once we know that the metric of interest decays asymptotically exponentially, the exponential twist method should be considered. Now we apply the concept for finding the probability of excessive backlogs in an M/M/1 queue. Consider the related queue length process (say \( N_t \), where \( N_t \) is the queue length at \( t \), \( t = 0, 1, 2, \ldots \) are the epochs where the queue length changes (arrivals or departures). Parameters \( \lambda, \mu \) are the arrival rate and

34
service rate, respectively. An event in a M/M/1 queue is either an arrival or a departure, with respective probability \( \lambda' \) and \( \mu' \), where

\[
\lambda' = \lambda/(\lambda + \mu), \quad \mu' = \mu/(\lambda + \mu).
\]

In applying the exponential twist, we find \( \theta \) by solving from the characteristic equation \( M(\theta) = 1 \). Here \( M(\theta) \), the moment generating function, takes the following form.

\[
M(\theta) = \lambda'e^\theta + \mu'e^{-\theta} = 1.
\]

This is a quadratic equation for \( e^\theta \), which has two solutions. One solution is \( e^\theta = 1 \), which corresponds to \( \theta = 0 \). The other is \( e^\theta = \mu'/\lambda' \), which implies \( \theta = \log(\mu'/\lambda') = \log(\mu/\lambda) \).

This latter solution corresponds to the asymptotic decay rate, and can be used to induce a fast simulation.

**Why not extrapolate with known decay rate?**

In discussions of IS methods, the asymptotic decay rate, as discussed above, is often applied as a basis to derive the fast simulation.

Some readers may ask, why not just use extrapolation, if we are interested in the hitting probability, and we already know the asymptotic decay rate? Let us consider the case that the rarity is indexed by an integer \( n \). Actually, from equation (2.24), we can derive

\[
\frac{\gamma_n}{\gamma_{n_0}} \simeq \exp\{-(n - n_0)\theta\} \quad \text{(UPLE)},
\]

where \( \theta \) is a constant independent of \( n \). It is naturally to think that for \( n \) sufficiently large, we should have,

\[
\gamma_n \approx \gamma_{n_0} \exp\{-(n - n_0)\theta\}.
\]

This seems to suggest that once we have \( \gamma_{n_0} \) and the asymptotic decay exponent \( \theta \), we may as well use extrapolation to find the value of \( \gamma_n \) for \( n > n_0 \). However, there are reason for not adopting this schema. First, we only get \( \theta \) as the asymptotic decay rate, and we do not know how fast the decay rate converges. But without this knowledge, we cannot control
the extrapolation error. Second, and more importantly, if our estimate of this decay rate
is not exact, using extrapolation will magnify the error exponentially. The rapid increase
of the error can reach such an extent that the estimate is off by magnitudes. This can
be checked by some simple calculation. If the estimate is 3% lower than the actual value
(this is a modest situation), $(1 - 0.03)^{75} = 0.102$, which means that if $n - n_0 = 75$, the
probability will be off by a factor of 0.102, that is, by an order of magnitude. On the other
hand, using fast simulation, we can get a bounded relative error under proper conditions.
In conclusion, using fast simulation will be much more accurate, when compared to the
extrapolation with known decay rate.

2.4.4 More examples: rare events in a single GI/G/1 queue

Free random walks can be exploited to find some important metrics relating to a GI/G/1
queue. In a GI/G/1 queue, both the inter-arrival time and the service time are independ-
ent, and each has a known probability distribution. Arrivals that find the server busy
will enter the queue. We consider two problems in this section. One is how to find the
probability that the waiting time of some clients exceeds a designated value $s$. The other
is how to find the probability that the length of line exceeds $n$ in a regenerative cycle.

Example 2.3 A GI/G/1 queue and the tail waiting time

Following [13], let $X_1, X_2, \ldots$ be mutually independent random variables with a common
distribution $F$. We define an induced sequence of random variables $W_0, W_1, \ldots$, by Lindley’s
recursion [13] (in the following $x^+ = \max(x, 0)$):

$$
\begin{align*}
W_0 &= 0, \\
W_j &= (W_{j-1} + X_j)^+, \quad j > 0.
\end{align*}
$$

(2.25)

Lindley’s recursion appears in several contexts, in particular in the GI/G/1 queue, with
customers served first in, first out (FIFO). Note we always assume that the 0-th customer
arrives when the system is empty (hence this one has a 0 waiting time). Lindley’s recursion
can describe the waiting time of individual customer by setting $X_j$ in (2.25) to

$$X_j = (B_{j-1} - A_j).$$

Here, $A_j$ is the time elapsed between the arrival of $(j-1)$-th and $j$-th customer, and $B_{j-1}$ is the service time spent on $(j-1)$-th customer. This relation is well known in queueing theory.

We assume that the distributions of inter-arrival time and service time are known. By Lindley’s recursion, the random walks underlying the sequence $\{W_j\}$ is not a free random walk, since it involves the $\text{max()}$ function. Fortunately, the question in discussion can be significantly simplified by using the theory of ladder points. (For free random walks $S_k = \sum_{i=1}^{k} X_i$, we say that $S_k$ is an (ascending) ladder point if for any $0 \leq j \leq k - 1$, it always holds that $S_k > S_j$). It has been shown that the distribution of the waiting time variable $W_j$ is identical to the distribution of the following random variable [13]:

$$U_j = \text{max}\{0, S_1, ..., S_j\}.$$  

Hence the waiting time sequence is connected with $\{U_j\}$, the maxima of a free random walk; although these two sequences are not the same, they have the same distribution. For the purpose of estimating the tail probabilities, it is not necessary to make any distinction between the distribution of $U_j$ and the distribution of the waiting in question.

When the expected service time is less than the expected inter-arrival time, then $E[X_i] = E[A_i] - E[B_i] < 0$. The random walks generalized by $\{X_i\}$ therefore have a drift towards $-\infty$. Hence, in a direct simulation of GI/G/1 queue, we usually observe only a few (ascending) ladder points before $S_j$, $j = 1, 2, ...$ becomes negative. To encourage a long sequence of ladder points leading to large value $S_j$, the exponential twist technique, as discussed before, can be applied.

It is very important to note that Lindley’s recursion holds only if the queueing discipline is FIFO. In other words, the relation holds when passing (a late arrival receives service before an early arrival) is not allowed. Generally, it is difficult for the “no passing”
condition to hold in a network of queues. That accounts for the fact that there are fewer results applying fast simulation to the network of queues. In fact, the situation for network of queues is currently not as well understood as in the case of single queue, and many questions in networks remain open.

Note that different description of sample paths may be used under different circumstances. In particular, we want to facilitate the calculation of the likelihood ratio. This point is illustrated by the following example.

Example 2.4 The queue length process in a GI/G/1 In a GI/G/1 queue, consider the queue length process (say $N_t$, where $t = 0, 1, \ldots$ are the epochs where the queue length changes). Unfortunately, unless the queue considered is M/M/1, i.e., arrivals are Poisson and service times are exponential, $N_t$ is not a Markov chain. Hence, the fast simulation methods on Markov chains are not directly applicable.

To overcome this difficulty, Sadowsky suggested the following approach [53]. He works in continuous time, so the time $t$, which is the incubation time, takes continuous values. For the purpose of estimating the tail probability of excessive backlog, a sample path is specified by the “event sequence” in $[0, T]$, where the system starts empty, and at time $T$ the queue length either reaches a designated length $n$ or returns to 0. In the event sequence, the events in the order of their occurrences, their types and the times when they occur are recorded. From this sequence, the backlogs (i.e., queue length) at each epoch can be read off, and so can the inter-event time.

For a successful path (i.e., the queue length reaches $n$ at time $T$), there must be an integer $k$ such that the arrival time of customer $k$ is earlier than the departure time of the $(k - n)$-th customer. Analytically, this relation can be expressed as follows:

$$
\sum_{j=1}^{k} A_j < \sum_{j=1}^{k-n} B_{j-1},
$$

where $\{A_j\}, \{B_j\}$ are respectively the sequence of inter-arrival time and service time. To explain the inequality above, note the left hand side gives the arrival time of $k$-th customer, and the right hand side is the earliest time that the $(k - n)$-th customer enters the service, as he has to wait for all customers arrived in front to go through the server (there may be
idle times, so the actual time entering the service may be later).

Like the previous example, this example also considers a GI/G/1 queue. However, the events of interest are different. It is noteworthy that even in the same model, there are different ways to specify a sample path. Which one to choose depends on the convenience of calculation.

Remark: Although Sadowsky is only considering the queue length process, his approach incorporates inter-event times. This is actually the approach of supplementary variables. Also, it is noteworthy that there are different ways to specify a sample path, even in the same model. For example, the sample path considered in this example is different from the last example dealing with the tail waiting time, although both involve a GI/G/1 queue. Which sample path to choose depends on several factors: the metrics of interest, and the convenience of modeling and calculating the likelihood ratio. In this example, a sample path is specified by the event sequence (including the incubation times), and the stopping time is decided by the condition that backlog exceeding a certain level.

With Sadowsky’s approach just described, one can change the system dynamics and calculate the likelihood ratio as follows. Let $A, B$ denote the random variables of inter-arrival time and service time. Let $F_a(t), F_b(t)$ be the corresponding cumulative distribution function, i.e. $F_a(t) = P(A \leq t), F_b(t) = P(B \leq t)$. In order to enhance the efficiency of sampling, samples are taken according to new functions $	ilde{F}_a(\cdot), \tilde{F}_b(\cdot)$. With exponential twist, we try the following transform,

\[
\begin{align*}
    d\tilde{F}_a(t) &= \frac{e^{-\theta A} dF_a(t)}{M_a(-\theta)}, \\
    d\tilde{F}_b(t) &= \frac{e^{\theta B} dF_b(t)}{M_b(\theta)}.
\end{align*}
\]

(2.26)

where $M_a(-\theta) = E[e^{-\theta A}]$, and $M_b(\theta) = E[e^{\theta B}]$ are the respective normalization factors. (Of course, suppose $\theta$ is chosen such that both $M_a(-\theta)$ and $M_b(\theta)$ are finite). Now we have,

\[
\frac{dF_b(t)}{d\tilde{F}_b(t)} = M_b(\theta) \exp\{ -\theta t \} \quad \text{(from equation (2.26))}.
\]
Since all inter-event time are independent, the probability of observing a particular event sequence is the product of the probability that each inter-event time is observed under the appropriate distribution. We can calculate the probability of observing each inter-event time (both service time and inter-arrival time) appearing in the sample path, and then find their product. This will yield the likelihood ratio $L(\cdot)$. See Sadowsky [53] for details. However, it is worth mentioning that the asymptotic optimality can be achieved if we choose $\theta = \theta^*$ such that

$$M_a(-\theta^*)M_a(\theta^*) = 1. \quad (2.27)$$

The above formula is an important one. Besides the application presented here, this formula plays an important role on fast simulation of bulk queues, with batch arrivals / departures [5] [63]. A connection of this formula with a method studied in my work (i.e., the rate tilting method) will be indicated later in Section 5.1.

The corresponding estimator is sometimes called the *Parekh-Walrand estimator* in literature, since it dates back to Parekh and Walrand [50]. There, the method was proposed heuristically. The approach above follows Sadowsky [53], who provided a rigorous justification for the *Parekh-Walrand estimator*.

### 2.4.5 Exponential twist in multi-dimensional state space

There are two main sources of literature for the exponential twist. One comes from application area, in particular rare events in networks of queues. The other comes from large deviation theory, which typically uses the random walk model.

However, what “exponential twist” means for multi-dimensional state space is not immediately clear. In many occasions, the term has been loosely defined, and may have different meanings in different models. In using the exponential twist to handle rare events in queueing networks, IS is often used as a synonym for the exponential twist (though there are indeed other IS methods). Also, many method are developed first in one dimensional state space; the extension to higher dimensional state space is through some heuristics. For example, in a queueing network, Parekh-Walrand suggests to change the distribution of inter-event times for each arrival and service, to speed up the simulation.
In recent literature, there are some studies trying to formally extend the exponential twist method to multi-dimension state space. For example, Asmussan and Rubinstein [2] attempt this for Markov additive process, which is a multi-dimensional Markov process with a certain structure. However, the resulting transform of the system dynamics is significantly more complicated than the term “exponential twist” in its usual meanings. (Actually, this method is in line with my rate tilting method, though the focus and the intended application are different). Hence, I do not use the term “exponential twist” to refer to such methods.

Following the literature dealing with large deviation theory, the exponential twist in multi-dimensional state space is defined as follows. On state space $\mathbb{R}^d$, let us consider the free random walks $\tilde{S}_T = \sum_{i=1}^T \tilde{X}_i$. Here $\{\tilde{X}_i\}$ is an i.i.d. random sequence of random walks, all vectors are in $\mathbb{R}^d$. For simplicity, we consider $d = 2$, but the extension to the case $d > 2$ is simple. Let $\tilde{X} = (X(1), X(2))$. The joint distribution function of $\{\tilde{X}\}$ is given by

$$P(X(1) \leq x(1), X(2) \leq x(2)) = F(x(1), x(2)).$$

In exponential twist, we apply $\tilde{\theta} = (\theta_1, \theta_2)$ to induce a new distribution $\tilde{F}$. Function $\tilde{F}$ is given as follows,

$$d\tilde{F}(x(1), x(2)) = \frac{\exp\{\theta_1 x(1) + \theta_2 x(2)\}dF(x(1), x(2))}{M(\theta_1, \theta_2)}.$$

(2.28)

where $M(\theta_1, \theta_2) = E[\exp\{\theta_1 X(1) + \theta_2 X(2)\}]$ is the moment generating function. We may also regard $M(\ldots)$ as a normalizing factor such that $\tilde{F}$ is a probability distribution. The sequence $\tilde{S}_T = \sum_{i=1}^T \tilde{X}_i$, where sequence $\{\tilde{X}_i\}$ conforms to the distribution $\tilde{F}$, is called the inclined random walk. Hence, the exponential twist method is extended to higher dimensions, for the model of free random walks.

Note that there are no bounds imposed in free random walks, (i.e., it is possible for every component of $\tilde{S}_T$ to approach $\pm\infty$ as $T \to \infty$, even though the possibility may be small). This is not the case in many discrete event systems. For instance, in a queueing network, the queue length at each node must be non-negative. Free random walks cannot
model such systems.

Now let us consider a discrete time Markov event system on a multi-dimensional state space with a boundary. To such situations, I attempt to extend formula (2.28), the formula that defines the exponential twist for free random walks.

Suppose we consider a system with two variables, denoted by \((l, j)\) (the reason that we use two different letters for two state variables is to be consistent with later discussion, e.g., the discussion in Chapter 4). The state space is assumed to be discrete and finite. The system is driven by events, each of which has an effect that changes the current state. The effect of an event is expressed by the difference between the state before the event and the state after the event. The effect is completely decided by the event type. The distribution that the next event is of certain type is assumed to be a known distribution, which may depend on the current state (state-dependent). In the system, there are a finite number of events (typically, the number of events is much smaller than the size of the state space).

Typically, \(l, j\) are integers. Since the system is Markov, we can use the transition probability to specify the probability of such transition. When the current state is \((l, j)\), then upon the occurrence of event of type \(e\), the next state becomes \((l', j')\), where \((l' - l, j' - j)\) is completely settled by event \(e\). Let the probability of the above transition be denoted by \(p_{l,j; l', j'}\). Apply the exponential twist with \(\theta = (\theta_1, \theta_2)\) to get

\[
\bar{p}_{l,j; l', j'} = \frac{\exp\{\theta_1(l' - l) + \theta_2(j' - j)\} p_{l,j; l', j'}}{M(\theta_1, \theta_2; l, j)}.
\]  

Again, \(M(\theta_1, \theta_2; l, j)\) is the moment generating function or the normalization factor. This factor may depend on the current state \((l, j)\) (because the transition probability may depend on the current state).

Now consider a CTMC where the probability that the next event is of certain type is proportional to the event release rate. Because of this, in applying the exponential twist to such models, we can directly use the event release rate, i.e., in formula (2.29), we replace transition probability \(p_{i,j}\) with the event release rate \(r_{i,j}\), similarly for the tilted system replace transition probability \(\bar{p}_{i,j}\) with the event release rate \(\bar{r}_{i,j}\). We use the modified
formula to induce the event release rate for the tilted system.

In some applications, the event release rates (in CTMC) or the transition probabilities (in DTMC) are independent of the current state, except at the state boundary. The boundary is a certain set of states. For instance, in an M/M/1 queue, the service rate is a constant unless there is no customer in the line (at that time, the server idles, hence the service rate changes abruptly to 0). To bring this into the context of our discussion, the event release rate becomes independent of the current state, and depends only on the event type e. Correspondingly, the normalization factor $M(\theta_1, \theta_2; l, j)$ becomes independent of $(l, j)$, provided $(l, j)$, the current state, is not at the boundary. This is the case in many applications. States other than the boundary are referred to non-boundary or interior. At non-boundary, both the event release rates and the factor $M(\theta_1, \theta_2; l, i)$ are independent of the current state $(l, i)$. This independence is violated at the boundary.

2.5 Cases of Inefficiency Reported in IS Simulation

In this section, we show some inefficiency cases of IS simulation reported in the literature. Many of these studies relate to rare event simulation in queueing networks.

Example 2.5 An inefficiency case of the first type in IS

This example was reported as an inefficient case, initially by [50]. It involves two queues in tandem. The system in question has two exponential servers in tandem. In front of each server, there is a buffer that can accommodate infinitely many waiting customers. Arrivals are Poisson, and all arrivals join a first line, to be served by the first server. After the service is completed, they join a second line to be served by the second server. After receiving service there, they leave. All rates involved are time homogeneous. The rate of the two servers are usually different, and the slower one is called the “bottleneck server”. Also, a cycle is defined to be the time interval starting from empty queues, and ending at the instant the system is cleared up again. The metric of interest is the so-called “overflow probability”, where overflow refers that the total population waiting for service in front of both servers reaches a level greater than $n$, where $n$ is large. Correspondingly, the rare
event set in this example is

\[ \{ \text{total population waiting for service in front of both servers} \geq n \} \]

For this model, Parekh & Walrand observed that the estimator they designed (the Parekh-Walrand estimator) works efficiently only when the service rates of the two servers are obviously distinct. When their rates become closer, the estimator becomes more and more inefficient.

Later, Glasserman and Kou [20] provided detailed analysis on this phenomenon. They also showed the regions where the mentioned estimator is efficient.

**Example 2.6** *An inefficiency case of the second type in IS*

In the model of tandem queues presented above, consider the following metric of interest:

\[ P\{ N_j > n \text{ is observed in a cycle} \} , \quad (2.30) \]

where \( N_j \) is the length of queue \( j, j = 1, 2 \). Suppose that the \( j \)-th server is the non-bottleneck of the tandem queues (i.e. we are interested in the probability of excessive buildup in a non-bottleneck server). For this situation, the Parakh-Walrand estimator will encounter serious trouble. No matter how the arrival rates and the service rates are changed, provided these rates are constant (state independent and time invariant), there seems no way to find an efficient simulation for the probability given in (2.30).

Above, we have described two examples where applying the exponential twist fails to be efficient. They both appear in queues in tandem. There are wide ranging discussions exploring why the exponential twist becomes inefficiency in these cases. The exploration leads to number of papers pointing to the limitations of IS, particularly the exponential twist. One of my points in this work is that the exponential twist is a simple method, hence intrinsically it cannot "mimic" the behavior when the buildup path to the rare event set is complicated.

The next chapter will be dedicated to the investigation of these inefficiencies. This detailed investigation is worthwhile, since it suggests directions for improvement. These
directions, in turn, motivates the development of fast simulation methods in this thesis.

2.6 Importance Splitting

Though importance splitting has a close connection with the importance sampling, the approach and analysis used in splitting is often distinct from those in importance sampling, and it deserves a separate treatment.

In this section, the setting is as follows. The state space of the system is partitioned, according to the promise to reach the rare event set. In the simplest setting, we use a threshold to partition the state space into two regions: the important region, and the unimportant region. A replica always starts from a state in the unimportant region, and the rare event set of interest is a subset inside the important region. A replica that reaches any state of the important region holds greater promise to enter the rare event set than a replica staying in the unimportant region.

In importance splitting, we do not explicitly change the system dynamics. Instead, to resolve the difficulty of lack of observations of replicas entering the rare event set, we introduce importance splitting as follows. We start some replicas, and apply splitting each time a replica crosses a threshold for the first time, i.e., to split into some new replicas; the new replicas continue evolving independently after the split. In brief, we want to split whenever a replica enters a region holding greater promise (an "important region") to hit the rare event set. When a replica splits, the number of sub-replicas to generate is called the splitting degree. This setting can be extended to multi-thresholds, in which a series of thresholds is set up in an order: the replica must pass through each threshold until it enters some rare event set. Figure 2.3 illustrates the splitting with a stochastic model of two variables. How to set up multiple thresholds will be discussed later.

In order to illuminate the essential idea of splitting, we look at the example: an M/M/1 queue, for which the state space is one dimensional. Here, if letting each queue length be a level, (i.e., when a replica currently has a queue length of \( n \), we say that it is in level \( n \)), then the asymptotic decay rate is \( \rho = \lambda / \mu \). For this model, we choose the splitting degree such that \( \alpha = \mu / \lambda \). With this choice, the number of replicas surviving to every level is
roughly the same. The splitting degree $\alpha > 1$ might not be an integer. In that case, one has to randomize between $\lfloor \alpha \rfloor$ and $\lfloor \alpha \rfloor + 1$ such that the expectation of the splitting degree is exactly $\alpha = (\mu/\lambda)$.

Note that when the state space is one dimensional, the so-called threshold is a separation point on the axis on real numbers ($\mathbb{R}$), thus is a relatively easy case. Complications arise in multi-dimensional state spaces.

### 2.7 Inefficient Cases Reported in Importance Splitting

Like importance sampling, importance splitting in multi-dimensional state spaces can be inefficient, as indicated in [18]. To show this, consider Figure 2.4. There, two replicas

---

Figure 2.4: The inconsistency in setting up the thresholds

crosses threshold B at point $x$ and point $b$, respectively. Let $P(A|B, x)$ be the conditional probability of finally entering threshold A, conditional of the replica already having entered
threshold \( B \), at point \( x \in B \). Such quantities as \( P(A|B, x) \) will be called the local potentials. Similarly we define \( P(A|B, b) \). Let us consider the difference between \( P(A|B, x) \) and \( P(A|B, b) \), where \( x, b \) are two arbitrary points taken from threshold \( B \). If there is a very large difference between them, then we say the threshold is set up inconsistently.

What happens if there is an obvious inconsistency in setting up the thresholds? Consider, e.g., the situation that \( P(A|B, b) \) is much larger than \( P(A|B, x) \), however, conditional on reaching threshold \( B \) only, it is much easier to observe a replica entering through point \( x \) than entering through point \( b \). If this is the situation, then the following phenomenon arises. We will end up observing many replicas that pass through \( x \), however it is not likely that these replicas will finally hit \( A \). Sometimes we call \( x \) trap phase, as it is easy to be entered, but it is difficult to hit the target from it. Hence, much effort is wasted generating replicas that pass through the trap phase, which accounts for the inefficiency of the splitting in this situation.

There are many articles explaining why importance sampling can become inefficient. In comparison, the explanation for the inefficient cases in importance splitting seems fairly simple, with the result that there are not many follow-up articles to investigate this issue further. However, I consider by investigating further on the explanation for the inefficient case of splitting, more insight can be gained.

The inefficiencies for importance sampling will be discussed in the next chapter. Meanwhile, how the importance splitting method can be improved is the topic of Chapter 6.
Chapter 3
Efficiency Consideration of Fast Simulation – A Large Deviation Perspective

The fact that the exponential twist is in some cases inefficient (as reported in literature) indicates that a shadow system (as defined on page 4 of Chapter 1) that can be simulated efficiently cannot be produced by the exponential twist (or generally, the methods based purely on Cramer’s large deviation rate function). It does not mean, in any sense, that such a shadow system does not exist.

In this chapter, readers will see some analysis of the reason why the exponential twist becomes inefficient in certain cases. Several analyses are given in literature. Each of them is interesting, and reflects a particular perspective. My analysis uses mainly a behavioral perspective. The purpose is to show the following: for certain models, the exponential twist cannot produce a system that “mimics” the large deviation behavior in these models. This is because the large deviation behavior is much more complicated than the behavior of the tilted system induced by any possible twist parameters. This point has not been explicitly expressed in previous analyses. Generally, the inefficiency in using the exponential twist could have two causes: (1) the exponential twist could be efficient, however, it is very difficult to find the right twist parameter that makes the simulation efficient; (2) no matter what twist parameters we choose, the exponential twist cannot be efficient. My analysis concentrates on the second possibility, pointing to the following direction. In order to solve these inefficient cases, novel methods that differ significantly from the exponential twist must be introduced.

I start by showing how the twisting parameters $\theta$ used in exponential twist can be derived from Cramer’s large deviation rate function. This is shown for the simple case of
free random walks on a line. This is a problem completely solved, and the exponential twist never fails in this situation. Later I will consider random walks in higher dimensions, or restricted random walks, which can be used to deal with many Markov event system of practical interest, e.g. the queue length process in queuing networks. In this chapter, I will point out some complications arising in higher dimensions which do not exist in the random walks on a line.

3.1 Exponential Twist Parameters and Large Deviation Rate Function

In this section, I show how the twist parameter $\theta$ used in exponential twist can be derived from Cramer's large deviation rate function. This is considered for the simple case of free random walks on a line.

Free random walks on a line - exponential twist and Cramer's large deviation rate function

Let $S_T$ be a random walk in $\mathbb{R}^1$. Let us designate a set $C_s = [s, +\infty)$ where $s$ is sufficiently large. Consider the probability that a replica of the $S_T$ reaches $C_s$, starting from 0. Let $T$ be the first time that the replica reaches $C_s$. Suppose the observed drift rate during the period $[0, T]$ is approximately $x$. Then according to Cramer’s theory, the probability $P$ of following a certain drift rate $x$ to reach $C_s$ is

$$P \approx \exp\{-TI(x)\} \ (UTLE). \quad (3.1)$$

Since $x$ is the observed drift rate during $[0, T]$, we have

$$T \approx S_T / x = s / x + (S_T - s) / x. \quad (3.2)$$

Here $(S_T - s)$ is called an “overshoot”. By substituting the above expression of $T$ into equation (3.1), we find that the probability $P$ of following a certain drift rate $x$ to reach

49
\( C_s \) is

\[
P \simeq \exp\{-(s/x) - (S_T - s)/x\}I(x)\}
\]

\[
\simeq \exp\{-s(I(x)/x)\} \quad (\text{UTLE}). \tag{3.3}
\]

Note the last UTLE equality holds because as \( s \) is sufficiently large, the exponent \((S_T - s)/x \approx (xT - s)/x\) is rather small, as compared with \( s/x \) (actually, as \( s \to \infty, (xT - s)/s \to 0 \) almost surely). Hence, by taking the sum over all possible drift rates \( x > 0 \) (when \( x < 0 \) it is not possible to reach \( C_s \)), we find

\[
\sum_{x > 0} \exp\{-TI(x)\} \simeq \sum_{x > 0} \exp\{-sI(x)/x\} \simeq \exp\{-s \inf_{x > 0} I(x)/x\} \quad (\text{UTLE}) \tag{3.4}
\]

The last UTLE equality holds because of the principle of “large component wins”.

Suppose \( x = x^* > 0 \) is the point such that \( I(x)/x \) achieves the minimum (page 25). Then from the curve of \( \log M(\theta) \), we find the adjunct generating exponent with respect \( \theta^* \) to the drift rate \( x^* \) (see page 24 for a definition of adjunct generating exponent). If we apply the exponential twist with parameter \( \theta^* \) to the original free random walk model, then the inclined random walk will have a drift rate of \( x^* \). This seems to be true by definition of \( I(x) \), but we calculate it anyhow as an additional check. To this end, we calculate the derivative of \( \log M(\theta) \) at a point \( \tilde{\theta} \in \mathbb{R} \), by using the definition for \( M(.) \) and find the derivative,

\[
\frac{d}{d\bar{\theta}}[\log M(\theta)]|_{\theta = \bar{\theta}} = \frac{d}{d\bar{\theta}}[M(\theta)]|_{\theta = \bar{\theta}} \]
\[
= \int x \frac{e^{\delta \bar{\theta} F(x)}}{M(\bar{\theta})} = E_{d\bar{\theta}}[\tilde{X}_1]. \tag{3.5}
\]

The last equality holds, because \( \frac{e^{\delta \bar{\theta} F(x)}}{M(\bar{\theta})} \) is the new probability measure which defines the inclined random walk. (The usage of the expectation operator is consistent with previous use.) In the equation above, by substituting \( \bar{\theta} = 0 \) and \( \bar{\theta} = \theta^* \), one gets,

\[
\frac{d}{d\bar{\theta}}[\log M(\theta)]|_{\theta = 0} = E_{d\bar{\theta}}[\tilde{X}_1] = E[\tilde{X}_1], \quad \text{(no tilting)}, \tag{3.6}
\]

50
\[ \frac{d}{d\theta} [\log M(\theta)]_{\theta = \theta^*} = E_{d\tilde{F}_{\theta^*}} [\tilde{X}_i]. \] (3.7)

That is, the derivative of \( \log M(\theta) \) at the origin provides the drift rate of the original random walk. If \( \tilde{\theta} = \theta^* \), then the inclined random walk will have a drift rate of \( E_{d\tilde{F}_{\theta^*}} [\tilde{X}_i] \). The fact that we are taking \( \theta^* \) as the adjunct exponent with respect to \( x^* \) indicates that the derivative must be \( x^* \); hence the drift rate of the inclined random walk is also \( x^* \).

Now it suffices to show that the dominant drift rate for the hitting probability (i.e., \( x^* \) such that \( I(x)/x \) achieves the minimum at \( x = x^* \)) is unique. (See Remark 2 at page 25. There we show \( x^* \) is unique by a geometric interpretation of \( x^* \).)

The behavior for the free random walks on a line is now completely clear. For the large deviation behavior, a certain drift rate of \( x^* \) can be identified. A replica of the system, on hitting the large deviation set \( C_s \) (where \( s \) is sufficient), is most likely to illustrate the drift rate \( x^* \). The drift rate \( x^* \) can be observed as the typical behavior of another random walk. In fast simulation, we can use the exponential twist with \( \theta^* \) to transform the original system to the tilted system. The parameter \( \theta^* \) is the adjunct generating exponent, as mentioned earlier.

### 3.2 Analysis of Inefficient Cases in Exponential Twist – Literature

Quite often, IS is used as the synonym for exponential twist. Indeed, almost all papers mentioning the inefficient cases in IS simulation actually refer to the inefficiency cases in the exponential twist. Some reasons of failure to be efficient for those reported cases are expounded in literature.

Parekh and Walrand [50] reported one inefficient case (this inefficient case has received much attention since the time being reported; I described this inefficient case in page 43). In brief, Parekh & Walrand designed an estimator (called Parekh-Walrand estimator) to effectively simulate the excessive backlog probability for queues in tandem. Unfortunately, this estimator does not always work efficiently. For instance, they found that even in the simple case of two exponential servers in tandem, the estimator will become inefficient
when the rates of the two servers becomes close. In their paper, the main interest was to propose heuristics to apply exponential twisting for fast simulation in a queuing network, so the cases of inefficiency and their reason are only mentioned briefly as a caution. In their work, they blame the inefficiency to the abrupt change of the event rates at the boundary (e.g. the service rate suddenly becomes 0 as the last customer completes the service, and no new customers arrive). As a possible remedy, it was suggested to create a process with the jump rates modified near the boundaries, i.e., a so-called boundary layer construction. The modification is such that the change of event rates near the boundary becomes smooth. However, as they commented, “this numeric approach is rather formidable because of the need to solve a system of differential equations with mixed initial and terminal conditions”. No effort was made to use the boundary layer construction for a practical solution.

In Glasserman and Kou [20], the goal is to provide the conditions that Parekh-Walrand estimator is efficient. The model involves $d$ ($d \geq 2$) queues in tandem, with Poisson arrivals and exponential servers. For this case, they provide a theoretic upper bound of the estimator mentioned (in many other studies, the variance is assessed only experimentally). They establish this bound primarily by using queuing analysis. Because the upper bound of the variance is a function of the model parameters involved (rates of arrival and each service), they can use this bound to provide the exact regions where the Parekh-Walrand estimator is efficient. How to rescue the inefficient cases is not the focus of their work, however.

The work of Juneja [37] is a strikingly different but an interesting work. He focused on what he called cyclic paths, i.e., at two distinct time points, the random walk is at the same position. This may seem a side issue, however, it should be pointed out that cycles exist widely in random walks. With respect to the hitting probability, that is, the probability of entering a designated set, let us consider two paths that are the same when all the cycles are removed. A (heuristic) argument is, that of the two paths, the one with more cycles should contributes less to the hitting probability, and be of less importance. Consequently, this kind of path should be sampled less frequently. Unfortunately, this is not the case if the likelihood ratio on a particular cycle is greater than 1, and therefore some efficiency is lost. Actually, Parekh-Walrand estimator also suffers this loss of efficiency in certain
parameter regions. In the analysis in Juneja [37], though some insights are contained, it cannot explain all inefficient cases.

In Glasserman & Wang [21], several examples are given in which IS estimators based on large derivation rate function actually performs poorer than the corresponding naive estimators. This should be regarded as a cautionary note.

The work in this thesis is different from the efforts mentioned above. It is from the way that the rare event set is approached, or, as we call it, the behavior aspect. In the next section, I will present an analysis on why the exponential twist becomes inefficient in some cases.

3.3 Behavior Perspective of Inefficient Cases in Exponential Twist

In this section, I characterize the so-called large deviation behavior, by using the distribution of sample paths, conditional on reaching the rare event set. Meanwhile, the unconditional distribution of sample paths can be used to describe the typical behavior.

Following, I will consider multi-dimensional Markov event system, and show why in multi-dimensional system, we cannot always use exponential twist to induce an efficient simulation. Also discussed is the large deviation behavior in free random walks, and in restricted random walks.

Concerning the inefficient cases, there are actually two types of inefficiency. In the inefficient cases of the first type, there are many paths to the rare event set which are equally likely. In the inefficient cases of the second type, the likely buildup path to the rare event set is not linear. Both cases can appear in the model of queues in tandem. Indeed, I described two inefficient cases in Example 2.5 (page 43) and Example 2.6 (page 44), which are of the two respective types. They both use the model of queues in tandem.

It should be noted that the reasoning in the rest of this chapter is often based on heuristic arguments. These arguments will highlight the essential distinction behind efficient and inefficient simulations.
3.3.1 Large deviation behavior of free random walks

In this section, I consider random walks on \( \mathbb{R}^d \), with \( d > 1 \). Because of this, the vector notation is used. Each random walk \( \vec{S}_T \) or its underlying sequence \( \{\vec{X}_i\} \) is assumed to be a vector in \( \mathbb{R}^d \).

In the model of free random walks, the system behavior is simple. The most important property that captures this simplicity is the so-called constant drift (also, the linear buildup). This property is described as follows.

1. (typical case) The model of free random walks, as its typical behavior, has a constant drift with rate \( E[X_i] \). To show this, use the law of large numbers, to get

\[
P(|\vec{S}_T/T - E[X_i]| < \delta) = 1, \forall \delta > 0
\]  

(3.8)

2. (large deviation case) Suppose we know that \( \vec{S}_T = \vec{s} \), where \( |\vec{s} - T E[X_i]| \) is assumed to be large. Then for \( 0 < \beta < 1 \), and sufficiently large \( T \), we should have that \( \vec{S}_\beta \approx \beta \vec{s} \).

Indeed, in the large deviation case, the drift rate must also be constant. Suppose in random walks on a line, each \( X_i \) takes one of the two values, either 1 or -1, with \( P(X_i = 1) = p, P(X_i = -1) = 1-p, 0 < p < 1 \). Note we are interested in large deviation behavior, i.e. assume the random walks deviate largely from the expectation. In this case, the random walk is likely to illustrate a constant drift rate, though this constant is quite different from the expectation. Why this is the case? Let us relate this with a classic sampling problem with red and blue balls.

The rule is, in the \( t \)-th move, if a red ball is sampled, then we take a move \( X_t = +1 \), otherwise a blue ball is sampled, and we take \( X_t = -1 \).

With this rule, a sampling sequence corresponds a sequence of random walks. Now when we condition on the final destination (i.e. at \( T \) the random walks reach \( \vec{s} \)), it is conditional on a certain sampling sequence. To see the intermediate position, we collect all samples in an urn, and re-sample. Note the following two points:
- This will certainly produce the same number of red balls and blue balls, though not necessarily the same sequence.

- At intermediate time, the number of red balls and blue balls sampled is in proportion with the colored balls appearing in the final result.

Say the time is $T/2$, then we get about half red balls and blue balls as in the final result. According to the rule mentioned earlier, this implies the random walks get half as far at time $T/2$ as at time $T$. Obviously, the argument is applicable to more general situations.

The most likely path to the rare event set are sought among those paths of constant drift rate. The reasoning here provide a justification. For a rigorous derivation, see [32] and [50],

Despite the constant drift rate in the large deviation behavior of random walk, complications still arise in $\mathbb{R}^d$. Suppose $C$ is a large deviation set. Because a replica of the random walks can reach $C$ at different points of entry, this means that the drift can be towards different directions. To show this, consider Figure 3.1. The target set $C$ is the area that is above line $AB$. Starting from the origin, random walks must cross the line segment between $A$ and $B$, which we refer to as the threshold. Note the entry point could be any point on the threshold.

![Figure 3.1: Multiple entry points to the target set in $\mathbb{R}^2$](image_url)

For the discussion on the probability of entering set $C$, what matters is the distribu-
tion of sample paths among these entry points. There could be several situations which distinguish different large deviation behaviors. First, there could be one dominant entry point, second, there could be more than one dominant entry points, or third, the probability of entering $C$ is equally likely on any point on the threshold to $C$. The first case is the simplest. As shown in Figure 3.1, suppose point $E$ be the dominant point of entry for replicas crossing the line between A and B. For the situation, the question almost reduces to a 1-dimension random walk, since the random walk proceeds dominantly along the path of $O\tilde{E}$. For other situations, we have multiple (more than one) entry points, and the contribution to the hitting probability through these entry points are equal (up to logarithm equivalence). Each entry point and the origin defines a linear buildup path, and along each path, there is a dominant drift rate. Along different path, the dominant drift rate is of course different. The issue is, no matter what twist exponent $\theta$ we choose, it is impossible for the inclined random walk to “mimic” the behavior of buildup on several paths simultaneously.

An inefficient case of the first type in IS: an analysis

Situations as discussed above exist indeed, as I will show below. Consider the model presented in Example 2.5, where we consider queues in tandem. (Queuing models are never free random walks, as no length of queue can become negative, which is the boundary constraint. The analysis below ignores this issue, and focuses on the aspect of multiple entry points.) Let set $C$ be the event that the total number in both lines exceeds some fixed value $n$. When the service rates are close to each other, then it is almost equally likely to buildup at either of the queues. In other words, we have many different drift rates, following different paths to set $C$. The difficulty of efficient simulation for this situation has been indicated above. On the other hand, when the service rates are apart from each other, the buildup is likely to occur dominantly at the bottleneck server. This is interpreted as one dominant point of entry to set $C$, which does not cause the trouble as indicated in multiple dominant entry.
3.3.2 Large deviation behavior of restricted random walks

There is a difference between the large deviation behavior of free random walk and the random walk with barriers. I now explain why.

Even on a line, the large deviation behavior may be slightly different from that of free random walks on a line. Asmussen [1] considered the case of a GI/G/1 queue (where the length of line must be non-negative, hence a barrier), and established the following. He indicated that between the time of an empty system and the time of a large buildup, the most likely path contains a number of short cycles (when the system empties again), and only in the last cycle, the system buildup has a constant rate. The drift rate is therefore not constant except in the last cycle.

In $\mathbb{R}^d$, $d > 1$, the difference between free random walks and restricted random walks poses some real challenges. In this case, the buildup path is not necessary along a line. Let us consider a random walk in $\mathbb{R}^2$, as shown in Figure 3.2. The distribution of $\{\vec{X}_i\}$ is as follows. $\vec{X}_i$ is random taking value from one of the following three: $\delta_1 = (+1,0)$, $\delta_2 = (0,-1)$, and $\delta_3 = (-1,+1)$. We assume that the walk starts at $(0,0)$. If this is a free random walk, the event sequence will be constituted of any number of vectors $\delta_1, \delta_2, \delta_3$ in any possible order. However, if a barrier is imposed, some event sequences must be excluded, because the corresponding random walk would cross the barrier. For the example shown in Figure 3.2, if a replica starts at $(0,0)$, and a barrier is added to

![Figure 3.2: An example of random walk in $\mathbb{R}^2$, with and without barriers](image)
prevent $S_T$ from having any negative component, then the following condition must be satisfied:

$$\# \{\text{events type-3}\} \leq \# \{\text{events type-1}\} \quad (3.9)$$

where $\# \{\text{events type-} j \} \geq \{\text{events type-} j \}$ denotes the number of events of type- $j$ occurring during the considered interval $[0,j]$. Otherwise, the replica at a certain epoch will end up at a place where the first coordinate becomes negative.

With a little bit more effort, we can even analyze how the imposition of a boundary changes the sample path distribution. Again, suppose a replica starts at $(0,0)$. It has a drift to the southwest. The target rare event set is at the north, designated as $C_{s_2}$, with $s_2 > 0$. In the model being considered, entering $C_{s_2}$ is a rare event. If this is a free random walk, then the dominant point of entry might be at $(0,s_2)$. If a replica starts at $(0,0)$, the line between them lies along the $y$-axis. The barrier we impose is along the $y$-axis. That barrier changes the probability distribution. Actually, among $\delta_1, \delta_2, \delta_3$, the only move that has an upward component is $\delta_3$, however this move must be disabled on the barrier. The implication is that the probability of following the $y$-axis, conditioning on reach $C_{s_2}$, will be decreased. Similar situations happen at and close to any barriers, where the probability of following a certain path is changed (reduced or increased).

When a barrier is imposed, additional constraints like the one above have to be taken into account. Such constraints will disqualify some paths from being included in the sample space. The corresponding analysis would usually be more involved than the situation when there is no barrier.

In conclusion, in restricted random walks, in large deviation behavior, the buildup is not necessarily along a linear path, neither does Cramer's large deviation rate function hold. Indeed, the function $I(.)$ is derived under the assumption of free random walk, hence not applicable when there are barriers.

*An inefficient case of the second type in IS: an analysis*

Now consider the inefficient case in IS described in Example 2.6, where we examine the queue length buildup in front of a non-bottleneck server in queues in tandem. Suppose
that there are $d$ queues, and we use $(N_j), j = 1, 2, ..., e$ to represent the queue length vector. Previously (in section 2.4.4), we have established a link of such a queue length process to the random walk. Because the queue lengths must be non-negative integers, there are barriers. With our analysis on the behaviour of restricted random walks, we expect that the queue length buildup in this situation may be not linear, and this is indeed the case [20]. The non linear buildup explains the reason for the inefficient case of second type.

Moreover, this analysis illustrates (again), that an efficient simulation for the problem being studied is difficult. The way out of this difficulty is to create a new system with a somewhat different structure.

3.4 Remarks

As the analysis in this chapter suggests, a method based on the large deviation rate function alone is not sufficient to ensure efficiency. We analyzed from a “behavioral” perspective. At the end of Chapter 2, the analysis of inefficient cases when using splitting suggests we should recognize the potentials local to the state entering a threshold, and enforce replicas that have reached states with high potentials to hit the target. In a way, the analysis given here on the inefficient cases in exponential twist agrees with the analysis given previously concerning the inefficient cases in importance splitting. Both analyses point to the direction that more detailed asymptotic information should be sought to ensure efficiency. Where the traditional exponential twist method fails, incorporating more detailed asymptotics is the only way to produce an efficient simulation. The failure case of the exponential twist and the importance splitting motivates the works in Chapter 4 - 6 of this thesis.
Chapter 4
Rate Tilting of Level/Phase Processes

In this chapter\(^1\) and in the one that follows, we explore “rate tilting”, which is a fast simulation method. We use certain event-based Markovian models, called level/phase processes. The development of methods other than the method of exponential twist is necessary. As indicated by an analysis in Chapter 3, the method of exponential twist has some intrinsic shortcomings. That analysis also showed that the asymptotic decay rate alone is not sufficient to ensure that a fast simulation is efficient. This suggests the direction to seek more detailed characteristics of the original system, and incorporate that information into fast simulation.

The numerical solution of Markov chains has attracted much attention, and has advanced considerably [26]. Nonetheless, exploration for Monte-Carlo simulation is still worthwhile as simulation is advantageous when solving problems with large state space. In exploring fast simulation, it is always a good idea to start from simple but representative and extensible models. We consider level/phase processes as such models. In rate tilting, the rates of events are changed. The method of changing event rates has been investigated previously [5] [41]. However, this study differs from previous studies in several aspects.

Now the level/phase process is described. This process is a Markov process with a two dimensional state space, described by two variables: the \textit{level} and the \textit{phase} (for more background, see Section 4.1). The level takes non-negative integer values and the phase takes discrete values. The changes of states (levels and phases) are induced by events. The rates of events are independent of the level, provided that the current level has a value high enough. However, this “level-independence” is typically violated at lower levels.

---

\(^1\)A major part of this chapter is based on a presentation at N8MC'03 by the author [43].
These levels are called \textit{boundary levels}. Note that the system behavior at the boundary levels turns out to be important in the question that we study. The level/phase process is positive recurrent (also called “stable”) if the expected number of levels going down per time unit is always greater than the expected number of levels going up per time unit. Unless stated otherwise, the given process is positive recurrent. A positive recurrent process must be stable. For a related discussion, see [42].

In a positive recurrent level/phase process, let a regenerative cycle be the period that a system starts from a particular state \( s \), and terminates when either it enters a set \( C \) or returns to \( s \). Let us take \( s \) to be \((0,0)\) (level 0, phase 0). Specifically, let \( C = C_n \) denote the set of states at or above level \( n \), and define \( \gamma_n \) be the probability of reaching \( C_n \). If \( n \) is large, then a direct simulation to find \( \gamma_n \) is inefficient because in a stable process, reaching level \( n \) becomes a rare event. Hence, importance sampling methods become advantageous. In the approach suggested here, we create a new process, called “tilted” process, which is constructed from the original process by a “rate-tilting” transform: in this transform, we increase the rates of going up one or more levels and decrease the rates of going down. In this fashion, one will reach the target level \( n \) more often, and one might expect that this type of rate tilting would always increase the efficiency of the simulation. Unfortunately, this is not always the case. My work will clarify this issue by providing criteria ensuring that rate tilting will be advantageous.

The method of changing rates in queueing systems with the intention of making rare events occur more often has been pioneered by Parekh and Walrand [50]. A similar method has been used later by Chang et al. [5] to investigate a discrete queueing system. Kroese and Nicola [41] modified the tilting method of Chang et al. to solve a problem involving tandem queues. This chapter generalizes these ideas to arbitrary level/phase processes. However, the approach developed here does have some new features, as detailed in the following. The generalized eigenvalue approach is used here, in the sense of [44] (see also [30], [28]) to calculate explicitly the dominant eigenvalue and its associate eigenvector, which are then used to derive a fast simulation. (Approaches using eigenvalues in fast simulation can be found in [41] [5], which are slightly different, as they resort to moment generating function, which I do not). Also, in the approach developed here, instead of
relying on the theory of Markov additive processes [35][2][48][41], we rely on the level/phase process, which can explicitly account for the boundary levels. The handling of boundary is a major difference between this approach and the one developed earlier. Concerning the simulation efficiency, I will show that the relative estimation error of $\gamma_n$ remains bounded as $n$ increases, provided certain conditions to be discussed are met. Note that an efficient simulation will justify the cost we pay for finding the dominant eigenvalue. Furthermore, we apply the fast simulation to a model (i.e. movable server model), which has not been previous studied in the context of rare event simulation. Also, a detailed comparison with previous methods in the methodology and in the efficiency will be conducted.

4.1 Background

In this section, we formally define the level/phase process. Then rate tilting transform, which changes the system dynamics, is introduced. This is a linear transform, which uses the dominant eigenvalue and its associated eigenvector as parameters in the transform (about linear transform and matrix theory, see [56]). We simulate the tilted system, and use the importance sampling (IS) estimator to estimate the hitting probability in the original system.

4.1.1 Level/phase model

In continuous time, the process is given by an infinitesimal generator. (The infinitesimal generator is the standard term used in discussion Markov chains. In an infinitesimal generator, any off-diagonal element represents a rate going from one state to another, and any diagonal element is a rate of leaving a certain state, multiplied by -1.) In the level/phase process, the infinitesimal generator matrix $Q$ has a certain structure. Let $Q = (Q_{ij})$, where $i, j$ are levels. The sub-matrix $Q_{ij}$ contains all transition rates of going from a phase of level $i$ to a phase of level $j$. In other words, matrix $Q$ is block structured, with each block containing the transition rates between the respective levels. $Q$ in a level/phase process has the following features:

- block-banded: we assume that $Q$ is “block-banded”, i.e. $Q_{ij} = 0$ unless $i - g \leq j \leq
\[ i + h, \text{ where } g, h > 0 \text{ are integers;} \]

- Level invariance: \( Q_{ij} = Q_{i-1,j-1} \) holds true for almost all \((i, j)\), which we take to mean that there is a value \( N_0 \) such that \( Q_{ij} = Q_{i-1,j-1} \) for \( i > N_0 \), which in turn implies \( Q_{ij} = Q_{j-i} \) for \( i > N_0 \).

As an infinitesimal generator, the sum of each row of \( Q \) must equal 0. We will call this as the regularity condition. Outside the boundary, the regularity condition can be expressed as follows

\[
( \sum_{d=-g}^{h} Q_d ) \cdot \vec{1} = \vec{0},
\]

(4.1)

Here, \( \vec{1} \) is a vector with all components being 1. At boundary level(s), the regularity condition usually takes a different form.

The level/phase process is related to the QBD (quasi-birth-death) process and to the M/G/1 and the GI/M/1 paradigms of Neuts [46] [47]. If \( g = h = 1 \) (i.e., the level increases or decreases at most one in a single transition), then the process is called a QBD process. If \( g = 1 \) and \( h = \infty \), then the process is covered by the M/G/1 paradigm. If \( g = \infty \) and \( h = 1 \), then the process is covered by the GI/M/1 paradigm. In this discussion, it is required that both \( g \) and \( h \) to be finite because we require the matrix \( Q \) be block-banded.

In many occasions, it is convenient to consider the so-called “jump chain” [10], which is a discrete time Markov chain embedded at the epoch of each transition. The jump chain can be found from a continuous time Markov process. This will be discussed in Section 4.2.2.

### 4.1.2 The hitting probability, some metrics of interest

The hitting probability \( \gamma_n \) has been introduced previously. The following two metrics are also of interest. The first one is the stationary probability of being in a high level \( n \). In continuous time, and for ergodic processes, this measures the portion of time the process spends in level \( n \). In discrete time, this similarly measures the percentage of visits to a state in level \( n \). It is convenient to consider a DTMC, where the said stationary probability is the percentage of visits to states at level \( n \), and is denoted by \( \pi_n \). To see the relation
between $\hat{\pi}_n$ and the hitting probability $\gamma_n$, let $\nu_n$ denote the expectation of the number of visits to level $n$ in a cycle, conditional that the replication hits $n$. We have,

$$\gamma_n \nu_n = \hat{\pi}_n \ E[\text{length of a regenerative cycle}]. \quad (4.2)$$

By definition, both the L.H.S. and the R.H.S. represents the expected number of visits to level $n$ in a cycle, so they must agree. To find $\hat{\pi}_n$, one first finds $\gamma_n$, $\nu_n$, and $E[\text{length of a regenerative cycle}]$, then uses the formula above.

Another metric is the expected time to reach a high level $n$, starting from a given state. This problem can be reduced to the problem of finding the hitting probability $\gamma_n$ by exploiting the following ratio formula. Given a specific starting state, let $t_H$ be the time to hit the target level, and let $t_E$ be the time to return to the starting state. In this case, the following formula applies (see [39] [32]):

$$E[t_H] = \frac{E[\text{min}(t_H, t_E)]}{\gamma_n}. \quad (4.3)$$

Fast simulation is only needed to find $\gamma_n$, but not for finding $E[\text{min}(t_H, t_E)]$, because the event of either hitting the starting state or the target level is not a rare event.

A performance evaluator may be interested in metrics other than the ones listed above. To find such metrics, the key is to incorporate a penalty/reward structure. For each sample path $\omega$, ascribe a penalty $\phi(\omega)$, which is a specified function. Then similar to the hitting probability $\gamma_n$, we define a metric,

$$\gamma = E_P[\phi], \quad (4.4)$$

where $P$ is the probability measure in the space of possible simulation outcome. Different function $\phi(.)$ will induce different metrics.

In the following discussion, we distinguish between rewards for being in a state and rewards for changes between states. When the system is at state $X$, rewards for being in a state are accumulated at a rate $\phi(X)$ per unit time; if a reward is for changing the state, then each time the system transfers from state $X$ to state $X'$, a reward impulse of $\Delta_{X,X'}$
is added to the total reward. Thus we have $\Phi(\omega)$:

$$\Phi(\omega) = (1/T) \left( \int_0^T \phi(X(t))dt + \sum_{X,X' \in S} \Delta_{X,X'} \right), \quad (4.5)$$

where the sum is for all transitions $(X \rightarrow X')$ that occurs during the time horizon $[0,T]$; $S$ is the state space. $\Phi(\omega)$ is one kind of time-averaged cost. To be general, in the above formula, $T$ can be either fixed or random stopping time (e.g. $T$ can be a regeneration point). From this example, we see that we can use the penalty function to specify metrics generally. For more examples, see [6] [25].

To bring this discussion into the setting of rare events, suppose $\Phi_n(\omega)$ takes finite, non-zero values only if $\omega$ includes a visit to levels above $n$, otherwise $\Phi_n(\omega) = 0$. Let $\gamma_n$ denote again the (cycle based) hitting probability. Let $\nu[\Phi_n]$ denote the expected accumulated cost in a cycle, which is decided by function $\Phi_n(\omega)$, and $\pi[\Phi_n]$ denote the expected time-averaged cost. Then we should have (similar to equation (4.2)),

$$\gamma_n \nu[\Phi_n] = \pi[\Phi_n] \ E[\text{length of a regenerative cycle}]. \quad (4.6)$$

Note that the value of $\pi[\Phi_n]$ and $\nu[\Phi_n]$ are considered on a continuous time basis, because we want to capture the reward average for being in a state. In discrete time, the length of a regenerative cycle has a slightly different meaning: that length refers to the number of transitions in a cycle. In continuous time, the length should be interpreted as the length of the time interval of a cycle. In formula (4.6), the time average $\pi[\Phi_n]$ can be converted, to be expressed by the other three metrics. Among these three, only $\gamma_n$ is a rare event metric based on cycles. $E[\text{length of a regenerative cycle}]$ is not a rare event, and is independent of $n$. The metric $\nu[\Phi_n]$ is not a rare event metric for its own sake, but is conditional on hitting $n$, which is a rare event. To find $\nu[\Phi_n]$, we implement a simulation with on/off fast simulation, as follows. First, from the initial state, we switch on the fast simulation to observe replicas entering the rare event set. Once this occurs, we immediately switch off the fast simulation, so that the system can return to the initial state. During the simulation, we record the accumulated cost during the complete cycle.
By running a number of replicas this way, we can find the corresponding accumulated cost. This will provide the estimate for $\nu[\Phi_n]$.

With the analysis above, $\gamma_n$ is indeed the key metric to be estimated in rare event simulation. Hence, in this chapter and chapters following, estimating $\gamma_n$ will be our focus.

### 4.1.3 An example; complex boundary behavior

As an example of the level/phase model, we use the tandem queues. In this queueing system, there are two exponential phase servers in tandem. Arrivals are Poisson, and all arrivals join a first line, to be served by the first server. After the service is completed, they join a second line to be served by the second server. After receiving service there, they leave. In this model, we could use either the length of second line as level, or the length of first line as level. The length of the other line becomes phase. Note the only boundary level in this model is level 0, because only at level 0, no departures are possible. In some models, a server becomes slower if fewer than a certain number of customers are waiting. In this case, the boundary will include several levels.

As indicated in Chapters 2 and 3, the boundary plays a critical role in affecting the system behavior. This is illustrated now using the model of tandem queues just described. In this example, there are two types of boundary behaviors. Suppose the level is the length of the second queue. Incorporating different boundaries leads to the following three models:

- Example 1a) (the build-up at a bottleneck server). The second server is the bottleneck. For this situation, [19] indicates that in the dominant path to reach the target, there is no significant build-up at the first buffer. This implies that there are not many transitions at the boundary.

- Example 1b) (the build-up at a non-bottleneck server). The first server is the bottleneck. For this situation, [19] suggests the dominant way to reach the target is that the first buffer builds up to a certain level at first, and then the second buffer starts to build up. Hence, there are many transitions at the boundary.
- Example 1c) (the build-up at a movable, non-bottleneck server) The fast server (assumed to be the second server), when seeing a small number of clients, helps to clear up the backlogs of the other server. That is, we consider the so called movable server model [30]. A movable server will significantly reduce the transitions at the boundary.

Note that Example 1a) and 1c) have only a small number of transitions at the boundary, whereas Example 1b) has many. This difference turns out to be important later. Note also that the importance of the boundary depends not only on the system, but also on the performance metrics we are seeking. If the metrics relate to such a cycle which always starts and ends at a set of states above the boundary, then obviously, the boundary is not a concern.

### 4.2 The Rate-Tilting Transform

In order to estimate $\gamma_n$, we first construct a tilted system, which implicitly generates a new distribution of the sample paths. Suppose the distribution of the original distribution and tilted distribution of paths are $P(\omega)$ and $\tilde{P}(\omega)$, respectively, and their likelihood ratio is $L(\omega)$. So the importance sampling estimator as defined in (2.11) can be applied. Secondly, in accordance with the right hand side of this formula, we estimate the expectation of $L(\omega)1_{C_n}(\omega)$ based on the observation of the tilted system. For a level/phase process, we give the following condition called canonical condition $^2$, as follows. From any phase $i$ of level $k$, there is a path to every phase $j$ at some level $l > k$. Except for pathological cases, this canonical condition is always met, and we assume it to be valid.

#### 4.2.1 The construct of rate-tilting

The state space can be divided into two regions, the non-boundary and the boundary. The boundary are those levels below or at $N_0$. Note that in non-boundary levels, the transition rates are specified by the matrices $Q_d, -g \leq d \leq h$. In boundary levels, the transition

---

$^2$This condition is sometimes called the regularity condition in some discussion in the literature. We use the term “canonical condition” and reserve the word “regularity condition” for other meaning.
rates are specified by the matrices \( Q_{ij}, i \leq N_0, j - i \leq h \). The so-called “rate-tilting transform” is basically the same in both regions, except some special care is required at the boundary.

To facilitate the discussion, we use \( B \) to denote the set of states of the boundary. Suppose the original level/phase process has the generator matrix \( Q \). We now suggest the following algorithm for the “rate-tilting transform”. Here, \( x \) is a real scalar that satisfies \( 0 < x < 1 \).

*Algorithm 1: rate tilting*

Step 1. For real \( x, 0 < x \leq 1 \), define \( \Gamma(x) = \sum_{d=-g}^{h} x^{h-d} Q_d \). \( \Gamma(x) \) is called a matrix polynomial [23].

Step 2. (characteristic equation) Consider the following equation,

\[
\Gamma(x) \eta(x) = \vec{0}. \tag{4.7}
\]

The equation above is referred to as the *characteristics equation*. Any pair \( (x, \eta(x)) \) that satisfies the characteristics equation is called an *eigen-pair*, with \( x \) being the *generalized eigenvalue* and \( \eta(x) \) being the corresponding *eigenvector*. See [23] for more details.

Of all solutions of \( x \), we are only interested in the \( x \) with the largest norm. Denote this solution by \( x_1 \).

Actually, \( x_1 \) is the Perron-Frobenius eigenvalue\(^3\) [56] of a certain rate matrix. Hence according to the Perron-Frobenius theory, \( x_1 \) is positive and single, and \( \eta(x_1) \), the eigenvector associated with \( x_1 \), can be made positive by multiplying with an appropriate constant.

Step 3. (rate-tilting transform) The rate-tilting transform for non-boundary is as

\(^3\)Later, scalar \( x_1 \) will always be used as the eigenvalue.
follows,
\[
\tilde{Q}_d = (\text{diag}[\tilde{\eta}(x_1)])^{-1} \ (x_1)^{-d}Q_d \ (\text{diag}[\tilde{\eta}(x_1)]), \quad -g \leq d \leq h \quad (4.8)
\]

where \(\text{diag}[\tilde{\eta}(x_1)]\) denotes a diagonal matrix with entries equal to the elements of vector \(\tilde{\eta}(x_1)\). Transform (4.8) is applied for all non-boundary levels. Since \(x_1\) is positive, and \(\text{diag}[\tilde{\eta}(x_1)]\) is positive, the transform properly produces matrices with all entries remaining non-negative, except for the diagonal elements of \(\tilde{Q}_0\).

Step 4. (new boundary construct) We extend “rate-tilting” naturally to the boundary, i.e., we use,
\[
\tilde{Q}_{kl} = (\text{diag}[\tilde{\eta}(x_1)])^{-1} \ ((x_1)^{-(l-k)}Q_{kl}) \ (\text{diag}[\tilde{\eta}(x_1)]) \ ((k,.) \in B). \quad (4.9)
\]

Then we set the diagonal elements of \((\tilde{Q}_{kk})\) such that the sum of row is 0. Other constructs at the boundary are possible, provided the sum of row is 0, i.e., the regularity condition holds.

deckend Algorithm 1.

To save space, let \(\tilde{\eta}_h = \tilde{\eta}(x_1)\). In the following, \(\eta_{i(i)}\) denotes the \(i\)-th component of \(\tilde{\eta}(x_1)\), and we abbreviate it to \(\eta_{ij}\). If a transition is from a non-boundary state to any other state, then we expand formula (4.8), yielding
\[
(\tilde{Q}_d)_{ij} = x_1^{-d}(\eta_{i(i)/\eta_{j(j)})}(Q_d)_{ij}, \quad -g \leq d \leq h. \quad (4.10)
\]

This gives the tilted rates we require.

The regularity condition (4.1) holds for non-boundary levels: those levels above \(N_0\):
\[
\sum_{d=-g}^{h} (\tilde{Q}_d) \cdot \bar{I}
= \sum_d (\text{diag}[\tilde{\eta}_l])^{-1} \ (x_1^{-d}Q_d) \ (\text{diag}[\tilde{\eta}_l]) \cdot \bar{I} \quad \text{[by formula(4.8)]}
= (\text{diag}[\tilde{\eta}_l])^{-1} \left( \frac{\Gamma(x_1)}{x^n} \right) \tilde{\eta}_l \quad \text{[by def. of } \Gamma(.)]\]
\[ \bar{Q} = 0 \quad \text{[by characteristic equation (4.7)].} \]

Besides, by the construction of the algorithm, it is obvious that the level invariance still holds. So \( \bar{Q} \) is truly an infinitesimal generator generator of a level/phase process.

It turns out to be convenient to consider the so-called “jump chain” \([10]\) embedded at the epoch of each transition. This means that if two replications visit the same set of states in the same order, they are considered as the same, irrespective of how long each replication stays in a state. Indeed, \( \gamma_n \) will have the same value in the original chain and in the jump chain. For the hitting probability related to a CTMC, using the jump chain will simplify the calculation of LR.

To find this jump chain, we proceed as follows. Let \( Q \) be the infinitesimal generator. Let the diagonal elements of \( Q \) be denoted by \(-r_{(i,j)}\), that is, \( r_{(i,j)} \) is the sum of rates of all transitions from level \( l \), phase \( j \). Let

\[ \Delta = \text{diag}[r_{(0,0)}, \ldots, r_{(0,m)}, r_{(1,0)}, \ldots, r_{(1,m)}, \ldots, r_{(N,0)}, \ldots, r_{(N,m)}], \quad (4.11) \]

where \( m \) is the number of phases, and \( N \) is the highest level considered in the model. Then,

\[ P = I + \Delta^{-1}Q. \quad (4.12) \]

Here \( I \) denotes the identity matrix. \( P \), the transition matrix of the jump chain, is a stochastic matrix with vanishing diagonal elements.

Using the jump chain, we discuss the characteristic equation in discrete time, as in the following remark.

**Remark:** Given \( Q \), which is the infinitesimal generator of a CTMC, we consider the jump chain embedded at each transition epoch. The jump chain is defined by (4.12). By expanding this equation, we can get relations between \( P_d \) and \( Q_d \). Now define \( \Gamma^{(D)}(x) = \sum_{d=-h}^{h} x^{h-d}P_d \), similar to step 1 of Algorithm 1. Note in continuous time, the characteristic equation is (4.7), so we get

\[ \Gamma^{(D)}(x) \cdot \bar{\eta}(x) = x^h \bar{\eta}(x). \quad (4.13) \]
This is the discrete version of the characteristic equation, which will be used later.

4.2.2 The formula of the likelihood ratio (LR)

We calculate the LR for an arbitrarily designated path $\omega$. Consider the jump chain just mentioned, because how long a replication stays at a state has no effect on $\gamma_n$, the metric of interest.

The main purpose of this subsection is to show the following result:

**Theorem 4.1** Consider a sample path $\omega^*$ that starts from some phase $j_0$ of level 0 and ends at some phase $j_r$ of level $n$. Then the likelihood ratio of such a sample path is,

$$L(\omega^*) = x^n(\eta(j_0)/\eta(j_r))\Pi^*$$

where $\Pi^*$ is a factor that depends only on the transitions at the boundary $B$.

**Proof:**

Consider the DTMC $P$, which is the jump chain embedded in $Q$. When considered in discrete time, the probability of following a path is the product of the probability of the transitions that occur along the path, because of the Markov property. We start from a single transition from state $(l, j)$ to state $(l', j')$, $(l', j') \neq (l, j)$. Let $L_{\text{sg}}$ (the subscript 'sg' stands for a single transition) denote the ratio of the probability that this transition occurs in $P$ to the probability it occurs in $\tilde{P}$, where $\tilde{P}$ relates to the jump chain of the tilted process. Now let $\tilde{t}_{(l,j)}$ be defined like $t_{(l,j)}$, except that it relates to the tilted system. Then,

$$L_{\text{sg}} = \frac{(Q_{l'}w)_{j,j}/t_{(l,j)}}{(Q_{l'}w)_{j,j}/\tilde{t}_{(l,j)}} \quad \text{[by expanding formula (4.12)]}$$

$$= \frac{(Q_{l'}w)_{j,j} \tilde{t}_{(l,j)}}{(Q_{l'}w)_{j,j} t_{(l,j)}}$$

(4.15)

To calculate this ratio, we must consider two cases.

*Case 1:* single transition starting in a non-boundary state, i.e., $(l, \cdot) \notin B$. 

71
According to equation (4.10) with $d = 0$, $i = j$,

$$r(i, j) = (Q_0)_{j,j} = (Q_0)_{j,j} = \tilde{r}(i,j).$$

(4.16)

Making use of formula (4.10) with $d = l' - l$, $L_{sg}$ becomes,

$$L_{sg} = \frac{(Q_{ll'})_{j,j'}}{(Q_{ll'})_{j,j'}} = (x_1)^{l' - l} (\eta(j)/\eta(j')), \text{ where } (l, .) \notin B. \quad (4.17)$$

Case 2: single transition starting in a boundary state, i.e., $(l, .) \in B$.

For the boundary, $r(i, j) = \tilde{r}(i, j)$ does not necessarily hold. Let $\varphi_{l,j} = \tilde{r}(i,j)/r(i,j)$; we call $\varphi_{l,j}$ correction factor. We write $L_{sg}$ as:

$$L_{sg} = \frac{(Q_{ll'})_{j,j'}}{(Q_{ll'})_{j,j'}} = (x_1)^{l' - l} (\eta(j)/\eta(j')) \varphi_{l,j}, \text{ where } (l, .) \in B. \quad (4.18)$$

Now consider a sample path $\omega^*$ that starts from some phase $j_0$ of level 0 and ends at some phase of level $n$, as follows:

$$\omega^* : (l_0, j_0) \rightarrow (l_1, j_1) \rightarrow (l_t, j_t) \rightarrow (l, j_0) \rightarrow (l, j_0) \rightarrow (l, j_0)$$

(4.19)

where $(l_t, j_t)$ is the level and the phase after $t$-th transition. Apply formula (4.17) and (4.18) as appropriate, for the 1st, 2nd, .. until the $\tau$-th transition, and take the product to get:

$$L(\omega^*) = \prod_{l=0}^{\tau}(x^{l+1} - l \eta(j_0)/\eta(j_{l+1})) \prod_{l_1, j_1} \varphi_{l_1, j_1} = x^n (\eta(j_0)/\eta(j_\tau)) \Pi^*(\omega^*). \quad (4.20)$$

Here $\Pi^*(\omega^*)$ denotes the product of all correction factors at the boundary. The exponent of $x$ in the last expression is $n$ because $l_\tau - l_0 = n$.

This proves the Theorem. Equation (4.14) allows us to analyze the efficiency of rate-tilting.
4.3 Effectiveness of Rate-Tilting

Rate-tilting, as just discussed, is in the category of importance sampling (IS). Some authors' proofs for the efficiency of an IS estimator relies on the result that the probability of the event of interest in the tilted system is bounded away from 0, no matter how high the level is (see, eg. [5] [41]). The basis of such proofs is Theorem 2.2 (page 28). However, for any level/phase process, such a result is not always easy to establish. Moreover, as was stated, the conditions in that theorem are sufficient for efficiency, but may be stronger than necessary.

In order to judge the efficiency of a simulation method better, we should look for new conditions, which are convenient to use for the situation under discussion, and which are both sufficient and necessary. We do this in the following.

Let \( C_n \) be the event set of interest, and define \( \tilde{L} \) as follows:

\[
\tilde{L} = E_P[L(\omega)1_{C_n}(\omega)]/\gamma_n = E_P[L(\omega)|1_{C_n}(\omega) = 1].
\]  

\( \tilde{L} \) is the expectation of the likelihood ratio \( L(\omega) \), conditional on that the path \( \omega \) does hit level \( n \). The following Theorem 4.2 opens an avenue to discuss the efficiency of estimating \( \gamma_n \). Note that the underlying method may be generalized to deal with other IS estimators.

**Theorem 4.2** The importance sampling estimator of \( \gamma_n \) that is based on \( E_P[L(\omega)1_{C_n}(\omega)] \) has a bounded relative error if and only if, as \( \gamma_n \to 0 \), \( (\tilde{L}_n/\gamma_n) \) has a uniform upper bound, independent of \( n \).

**Proof:**

We find the variance of the estimator based on (2.11) as follows,

\[
\begin{align*}
\text{Var}_P[L(\omega)1_{C_n}(\omega)] &= E_P[L^2(\omega)1_{C_n}(\omega)] - (E_P[L(\omega)1_{C_n}(\omega)])^2 \\
&= E_P[L(\omega)1_{C_n}(\omega)] - \gamma_n^2 \quad \text{(Note, } L(\omega)\bar{P}(\omega) = P(\omega)) \\
&= \tilde{L}_n\gamma_n - \gamma_n^2. 
\end{align*}
\]  

(4.22)
Since the estimator is unbiased, one can calculate the square of the relative error as follows:

\[ \epsilon^2 = (\text{Var}[L(\omega)1_{C_n}(\omega)]/\gamma_n^2) = (\bar{L}_n/\gamma_n) - 1. \] (4.23)

Theorem 4.2 follows from the formula above. □

We will show that, provided a certain condition is satisfied, \( \bar{L}_n/\gamma_n \) is upper bounded. This condition, we call as the compliant boundary condition, primarily involves the system behavior at the boundary. Essentially, we show that when the compliant boundary condition is satisfied, \( \gamma_n \) decays asymptotically geometrically, and \( L_n \) decays with roughly the same rate.

4.3.1 The geometric decay of \( \gamma_n \)

The main result of this subsection is the asymptotic decay of \( \gamma_n \). We have,

**Theorem 4.3** Given a positive recurrent level/phase process that satisfies the canonical condition, \( \gamma_n \) has an asymptotic geometric decay, i.e., when \( n \) is sufficiently large,

\[ \gamma_n = c(x_1)^n + o((x_1)^n) \] (4.24)

where \( c \) is a scalar.

Note that the value of \( \gamma_n \) is obtained from simulation. We do not need to know the scaler \( c \). We also assume that \( x_1 \) is known, or can be obtained (this topic will be addressed in section 4.5).

**Remark:**

To make the analysis easier, we convert the level/phase process to a QBD process by re-blocking, i.e. rearranging the infinitesimal generator into larger blocks. This can be done for any level/phase process. For a simple example, in the case that \( g = 1 \) and \( h = 2 \), we can arrange the blocks as follows,

\[ \overline{Q}_{-1} = \begin{bmatrix} 0 & -Q_{-1} \\ Q_{-1} & 0 \end{bmatrix}, \quad \overline{Q}_0 = \begin{bmatrix} Q_0 & Q_1 \\ Q_{-1} & Q_0 \end{bmatrix}, \quad \overline{Q}_1 = \begin{bmatrix} Q_2 & 0 \\ Q_1 & Q_2 \end{bmatrix} \]
such that $\overline{Q}_l = 0$ as $l > 1$ or $l < -1$. Now $\overline{Q}$ has a structure that can be identified as a QBD process, except that the number of phases is doubled. This conversion facilitates the theoretical development. Another advantage is that QBD is a special case of both GI/M/1 and M/G/1 paradigm, so many results in that category become applicable for our analysis. Note some authors also use re-blocking to design efficient algorithms [16], so it is also useful from the view of computation, but we will not address this issue.

I now present some supporting arguments for Theorem 4.3. I will relate $\gamma_n$, the hitting probability, to the stationary probability of being in level $n$.

Concerning $\nu_n$ in (4.2), we have the following:

**Lemma 4.1** In a QBD process, let $\nu_n$ be the expectation of number of visits to level $n$ in a cycle, given that level $n$ is hit. Then $\nu_n$ converges to a constant $\nu > 0$ as $n \to \infty$.

The proof of Lemma 4.1 will be given later in this section (as the proof is fairly lengthy).

From (4.2) and Lemma 4.1, as $n \to \infty$, $\gamma_n$ and $\hat{\pi}_n$ decay at the same rate. In other words, the expression of $\gamma_n$ and $\hat{\pi}_n$ are the same, except for a constant factor. The proof of Theorem 4.3 can be accomplished by showing that $\hat{\pi}_n$ has an asymptotic geometric decay, i.e. there is some constant $c' > 0$, such that as $n \to \infty$,

$$\hat{\pi}_n = c'(x_1^n + o(x_1^n)).$$  \hspace{1cm} (4.25)

Note that we define $\vec{\pi}_n$ as a vector of $m$ components, where $m$ is the number of phases, each component being the stationary probability of being in a phase of level $n$; the scalar $\pi_n$ is simply the sum of these components, i.e. $\pi_n = \vec{\pi}_n \cdot \vec{1}$.

Using the approach of eigenvalue solution [28] [44], we have,

$$\vec{\pi}_n = \vec{c}x^n + o(x^n).$$  \hspace{1cm} (4.26)

where $\vec{c} > 0, x > 0$, since $\vec{\pi}_n > 0$. More rigorously, suppose that the process $Q$ is positive recurrent and canonical. Then obviously, the jump chain associated with $Q$ is also positive recurrent and canonical. Applying the matrix geometric solution to the jump chain $P$ (see
Neuts [46], pp. 9-10), the vector $\tilde{\pi}_n$ ($n > N_0$) assumes the follow form,

$$\tilde{\pi}_n = \tilde{\pi}_{N_0} R^{n-N_0}. \quad (4.27)$$

in which $R$ is the so-called rate matrix for $P$. (In [46], the formula is given by $\tilde{\pi}_{n+1} = \tilde{\pi}_n R$.

Also, the rate matrix for $P$ has the same set of eigenvalues as the rate matrix for $Q$.)

Note that the rate matrix $R$ is non-negative, and $\text{sp}(R) < 1$. From the well-known Perron-Frobenius theorem ([60][56]), $\text{sp}(R) > 0$ is a simple eigenvalue of $R$, and all other eigenvalues have a norm strictly less than $\text{sp}(R)$, unless the following exceptional circumstance on $R$ arises:

(E1) all eigenvalues of $R$ are 0, or $R$ is reducible.

The canonical condition ensures that (E1) never holds. It then follows that, $\tilde{\pi}_n = c' (\text{sp}(R))^n (1 + o(1))$. Right multiply both sides by $\overline{1}$, and denote $\tilde{\pi}_n = \tilde{\pi}_n \cdot \overline{1}, c' = c' \cdot \overline{1}$, then we have $\tilde{\pi}_n = c' (\text{sp}(R))^n (1 + o(1))$. Now in order to prove (4.25), it remains to show the following fact.

**Lemma 4.2**

$$\text{sp}(R) = x_1, \quad (4.28)$$

where $x_1$ is the eigenvalue solution of equation (4.7) with the maximal norm, and $R$ is the rate matrix for $P$. This follows from [45].

Note the theory we just developed applies to any banded level/phase process.

The remainder of this section is devoted to the proof of Lemma 4.1. We assume that $n > N_0$ (i.e., level $n$ is above the boundary levels). To facilitate the discussion, we use the term below-$n$ sojourn, which begins immediately after the last epoch when the level is at or above $n$, and ends immediately before it returns to level $n$ or above. We decompose $\nu_n$ into three parts as follows. Let $\nu_n^{(0)}$ denote the expected number of visits to level $n$ from the first hit of $n$ to the instant that the replication exits to a below-$n$ level. After these visits, there are other visits to level $n$, which can be further divided into two parts. Some visits are contributed by those paths that, after hitting level $n$, have never visited level $N_0$
or below (denoted by \( \nu_n^{(1)} \)), and the rest (denoted by \( \nu_n^{(2)} \)), which are contributed by those replications that have hit \( n \), visit the boundary, and then come up to visit level \( n \) again.

Now we write,

\[
\nu_n = \nu_n^{(0)} + \nu_n^{(1)} + \nu_n^{(2)}.
\] (4.29)

The replications contributing to \( \nu_n^{(0)} \) are always above-\( n \), thus above the boundary. We conclude that \( \nu_n^{(0)} \) is a constant, because of the level invariance. Now we show that as \( n \) increases,

(a.) \( \nu_n^{(2)} \to 0 \);

(b.) \( \nu_n^{(1)} \) increases, with a finite upper bound, hence converges to a constant.

If both (a.) and (b.) hold, and \( \nu_n^{(0)} \) is independent of \( n \), then \( \nu_n \) converges to a value independent of \( n \); it is obvious that this constant must be positive.

In order to show (b.), we show first that \( \nu_n^{(1)} \) increases. Let \( \nu_n^{(1)}(b) \) be defined analogously to \( \nu_n^{(1)} \), we count however only those paths that at the first below-\( n \) sojourn do not visit any level below \( b \), \( b > N_0 \). Let \( q_n(b) \) denote the probability that in a below-\( n \) sojourn, the lowest level visited is level \( b \). We can calculate \( \nu_n^{(1)} \) as follows,

\[
\nu_n^{(1)} = \sum_{b > N_0} q_n(b) \nu_n^{(1)}(b).
\] (4.30)

Note that

\[
\sum_{b > N_0} q_{n+1}(b) \nu_{n+1}^{(1)}(b)
\]

\[
= q_{n+1}(N_0 + 1) \nu_{n+1}^{(1)}(N_0 + 1) + \sum_{b > N_0} q_n(b - 1) \nu_n^{(1)}(b - 1)
\]

\[
= q_{n+1}(N_0 + 1) \nu_{n+1}^{(1)}(N_0 + 1) + \sum_{b > N_0} q_n(b) \nu_n^{(1)}(b)
\]

\[
> \sum_{b > N_0} q_n(b) \nu_n^{(1)}(b).
\]

Hence,

\[
\nu_{n+1}^{(1)} - \nu_n^{(1)} = q_{n+1}(N_0 + 1) \nu_{n+1}^{(1)}(N_0 + 1) > 0.
\] (4.31)
So as $n$ increases, $\nu_n^{(1)}$ always increases.

Next we show that $\nu_n^{(1)}$ is upper bounded. Take the sum of equation (4.31) for $l = n + 1, n + 2, \ldots$ until $n'$ (where $n' > n$), we get

$$
\nu_{n'}^{(1)} - \nu_n^{(1)} = \sum_{l=n+1}^{n'} q_l(N_0 + 1) \nu_l^{(1)}(N_0 + 1) \leq \sum_{l=n+1}^{n'} \nu_l^{(1)}(N_0 + 1)
$$

$$
= \sum_{l=n+1}^{n'} E[\sharp \hbox{ visits to level } l, \hbox{ after a visit to level } N_0 + 1 \mid 1_{C_l}(\omega) = 1]
$$

$$
\leq E[\sharp \hbox{ visits to level } n + 1, \ldots, n', \hbox{ after a visit to level } N_0 + 1 \mid 1_{C_n}(\omega) = 1].
$$

As $n' \to \infty$, $\nu_{n'}^{(1)} - \nu_n^{(1)}$ is upper bounded, and the bound is independent of $n'$. To show this, note that the system is positive recurrent, so the expected number of visits during a sojourn from level $N_0 + 1$ is finite. Hence, as $n$ increases, $\nu_n^{(1)}$ remains bounded, and it must therefore converge.

Finally, we show that $\nu_n^{(2)} \to 0$. We simply note that for fixed $b < N_0$, after descending down to level $b$, the expected number of visits to level $n$ must converge to 0 as $n$ increases. Otherwise, the sum of all expected number of visits above level $b$ converges to $+\infty$, which contradicts the assumption that the process being considered is positive recurrent. Also, in all paths contributing to $\nu_n^{(2)}$, the lowest level visited is $b < N_0$. Certainly there are only finite levels below $b$, so sum over all $b < N_0$, we get $\lim_{n \to \infty} \nu_n^{(2)} = 0$. □

### 4.3.2 The bound for $\tilde{L}_n/\gamma_n$

In order that the rate-tilting estimator for $\gamma_n$ has a bounded relative error, we require that $\tilde{L}_n/\gamma_n$ has an upper bound (Theorem 4.2). Let

$$
\Omega_{C_n} = \{\omega : 1_{C_n}(\omega) = 1\}. \tag{4.32}
$$

That is to say, $\Omega_{C_n}$ is the set of all sample paths that reach level $n$ in a cycle; using (4.14) (4.21) (4.24), we have,

$$
\frac{\tilde{L}_n}{\gamma_n} \leq \frac{(x_1)^n \left[ \max_{i,j}(\eta(i)/\eta(j)) \right]}{c x_1^n + o(x_1^n)} \frac{\max_{\Omega_{C_n}} \Pi^*(\omega)}{\Pi^*(\omega)}
$$

78
\[
\begin{align*}
&= \left( \frac{1}{c} \right) \left[ \max_{i,j} (\eta(i)/\eta(j)) \right] \left[ \max_{\Omega_C} \Pi^*(\omega) \right] (1 + o(1)) \\
&= \left( \frac{M}{c} \right) \left[ \max_{i,j} (\eta(i)/\eta(j)) \right] (1 + o(1)).
\end{align*}
\]

Here, \( M = \max_{\Omega_C} \Pi^*(\omega) \). If \( M < \infty \), we already have an upper bound for \( \tilde{L}_n/\gamma_n \). For example, when the simulation is designed in such a way that \( \omega \) always starts and ends above the boundary, then \( \Pi^*(\omega) = 1 \) for any sample path \( \omega \). Unfortunately, it is often the case that \( M = \infty \). For this situation, it is more difficult to check whether \( \tilde{L}_n/\gamma_n \) is bounded.

For a sample path \( \omega \), let \( K(\omega) \) denote the number of transitions in the path that go from a boundary state to any other state. Firstly, if for a finite \( K_0 \), \( K(\omega) < K_0 \) for all sample paths, then \( \Pi^*(\omega) \) is uniformly bounded. To show this, we use the largest correction factor, say \( \varphi(i^*, j^*) \), to obtain,

\[
\Pi^*(\omega) < [\varphi(i^*, j^*)]^{K_0} < +\infty. \tag{4.33}
\]

To facilitate further discussion, let

\[
\Omega_{C_n}^{K_0} = \{ \omega : 1_{C_n}(\omega) = 1, K(\omega) < K_0 \}. \tag{4.34}
\]

That is to say, \( \Omega_{C_n}^{K_0} \) is the set of all sample paths that reach level \( n \), and the number of transitions from the boundary state is less than \( K_0 \).

Secondly, even if the likelihood ratio \( L(\omega) \) is extremely large due to a very large \( K(\omega) \), it is often true that such paths are not likely to occur (i.e. they have a very small probability), either in the tilted system, or in the original system. Under such situations, \( \tilde{L}_n/\gamma_n \) may still have a finite upper bound. Elaboration on this issue leads to the condition we call the compliant boundary condition, which is summarized by conditions (B1) and (B2) below:

(B1) If for \( n \) sufficiently large, and \( \epsilon' > 0 \) very small, there is a \( K_0 \), such that

\[
(1/\gamma_n) \sum_{\Omega_{C_n}^{K_0} \Delta \Omega_{C_n}^{K_0}} P(\omega) < \epsilon', \tag{4.35}
\]

79
then we say that the system behavior at the boundary has an negligible effect on the estimate of $\gamma_n$.

(B2) If there is a $K_0$, such that

$$
(1/\gamma_n) \sum_{\Omega_{C_n} \setminus \Omega_{C_n}^{K_0}} \Pi^*(\omega)P(\omega) < S_B < +\infty
$$

is uniformly bounded for all $n$, then we say that the system is well-behaved at the boundary.

There is a particular situation that for fixed $K_0$, $K(\omega) < K_0$ for all sample paths. Then in both (B1) and (B2), the respective sum is 0. So both (B1) and (B2) hold true.

For the situation that (B1) holds but (B2) does not, we developed a modified rate-tilting estimator, given as follows,

$$
E_P[L(\omega)1(\omega \in \Omega_{C_n}^{K_0})]
$$

where $1(.)$ is the indicator function. This means that for any sample path $\omega$ such that $K(\omega) > K_0$, we stop the simulation and exclude the sample path from the statistics collected.

Note, when either (B1) or (B2) is satisfied, we say that the system satisfies the compliant boundary condition. Under the compliant boundary condition, the rate-tilting estimator has a relative estimation error that is bounded (I will show this below).

Though it may be difficult to check by simulation for either assumption (B1) or assumption (B2), it is helpful to use them as heuristics. Of course, an intuition behind the compliant boundary condition is that, for the model being considered, it has only a few transitions at the boundary, along the most likely paths to the rare event set. How to check these conditions theoretically worths to be investigated. The following theorem states the efficiency of rate tilting, provided the compliant boundary condition is satisfied.

**Theorem 4.4** For the rate-tilting estimator of $\gamma_n$,

a. If (B1) is satisfied, then the modified rate-tilting estimator (4.37) has a bounded
relative error; this new estimator has a bias, but the bias can be made arbitrarily small;

b. If (B2) is satisfied, then the rate-tilting estimator has a bounded relative error.

Proof:

Note that by definition,

\[
\bar{L}_n = \mathbb{E}(L(\omega)|1_{C_n}(\omega) = 1) = \sum_{\omega \in \Omega_{C_n}} L(\omega)P(\omega|1_{C_n}(\omega) = 1).
\]

The term \( L(\omega)P(\omega|1_{C_n}(\omega) = 1) \), which appears in equation (4.38), can be summed over in subspace \( (\Omega_{C_n}^{K_0}) \) and in subspace \( (\Omega_{C_n|\Omega_{C_n}^{K_0}}^*) \). Let us denote the result as \( \bar{L}_n^{[K_0]} \), respectively \( \bar{L}_n^* \). (We can do this because the two subspaces are disjoint.) Then,

\[
\bar{L}_n/\gamma_n = \bar{L}_n^{[K_0]}/\gamma_n + \bar{L}_n^*/\gamma_n.
\]

Since \( K_0 \) is finite, according to (4.33), \( \Pi^*(\omega) \) is uniformly bounded on the subspace \( \Omega_{C_n}^{K_0} \). With this, provided we consider at subspace \( \Omega_{C_n}^{K_0} \), the derivation of formula (4.33) goes through, which means that the first term \( \bar{L}_n^{[K_0]}/\gamma_n \) is always bounded.

For the second term \( \bar{L}_n^*/\gamma_n \), we argue as follows. If (B1) holds but not (B2), then the modified rate-tilting estimator results in \( \bar{L}_n^*/\gamma_n = 0 \). In this case, the modified rate-tilting estimator introduces a bias, but if (B1) holds, the bias is negligible, and the relative bias is less than \( \epsilon' \). On the other hand, if (B2) holds, we claim that \( \bar{L}_n^*/\gamma_n \) has a finite upper bound. This is shown as follows:

\[
\bar{L}_n(\omega)/\gamma_n = \frac{1}{\gamma_n^2} \sum_{\Omega_{C_n|\Omega_{C_n}^{K_0}}^*} L(\omega)P(\omega) \quad \text{[by def.]}
\]

\[
= \frac{1}{\gamma_n^2} \sum_{\Omega_{C_n|\Omega_{C_n}^{K_0}}^*} x_1^i \eta_{j_0}^i \Pi^*(\omega)P(\omega) \quad \text{[use (4.14)]}
\]

\[
= [(1/\gamma_n)x_1^i \max_{i,j} \eta_{j_0}^i \eta_{j}^i] [(1/\gamma_n) \sum_{\Omega_{C_n|\Omega_{C_n}^{K_0}}^*} \Pi^*(\omega)P(\omega)]
\]
\[
\frac{x_1^n}{\alpha x_1^n} \max_i \frac{\eta_i}{\eta_j} [S_B] \leq \frac{x_1^n}{\alpha x_1^n + o(x_1^n)} \max_i \frac{\eta_i}{\eta_j} [S_B] \quad \text{[use (4.24)]}
\]
\[< +\infty. \quad (4.40)\]

Note \( S_B < \infty \) follows from (B2).

Hence when either (B1) or (B2) holds true, \( \bar{L}_n/\gamma_n \) has a finite upper bound. Employing Theorem 4.2, we can now conclude that the rate tilting estimator has a bounded relative error. \( \square \)

As a possible application of these conditions, consider to construct a run-time sentinel program, which watches whether the rate tilting applying to a certain model is running efficiently. Usually, we have a rough idea about the probability \( \gamma_n \), and the two amounts \( \bar{L}_n/\gamma_n \) and \( \bar{L}_n^*/\gamma_n \) should be about in the same range. If these amounts exceed a specific value, it would signal that the rate-tilting estimator is not working efficiently and the simulation is aborted.

### 4.4 Some Experiments on Tandem Queues

In this section, I report some experiments on tandem queues. First, we show how the relative error for the IS estimator is calculated. Then I present experimental results. At the end of this section, I explain what has been done to verify that the experiments are carried out properly and the related results are valid.

#### 4.4.1 Calculation of the relative error for the estimator

Even for simple benchmark models, like the tandem queue model, it is very difficult to calculate the exact value of \( \text{Var}[\gamma_n] \) from equation (4.22). However, it is advantageous to have the variance and the relative error in order to see how good the estimator is. For this purpose, we adopt the batch mean method \(^4\), which works under the assumption that different batches are independent (this is not exact, however we usually assume that

\(^4\)To be clear, we point out that in direct simulation, samples of \( 1_{C_n}(\omega) \), which are used to estimate \( \gamma_n \), are independent; however, they are not truly independent when we sample from the tilted distribution, as those paths that reach \( C_n \) are taken with greater probability. Hence, the batch mean method is used to assess the variance.
batches have small correlation if and only the batch size is large enough). Suppose we observe replicas $\omega_i$, $i = 1, 2, \ldots, W$ from the simulation. We group them into $b$ batches, each batch containing $k$ successive samples, i.e. $kb = W$. Now let

$$\hat{\beta}_j = (1/k) \sum_{i=(j-1)k+1}^{jk} [1_{c_n}(\omega_i)L(\omega_i)], \quad (4.41)$$

then $\gamma_n = (1/b) \sum_{j=1}^b \hat{\beta}_j$. Note this is the same IS estimator as before. The variance of $\gamma_n$ can be assessed as,

$$\sigma^2 = Var[\gamma_n] = (1/b) \sum_{j=1}^b (\hat{\beta}_j - \gamma_n)^2. \quad (4.42)$$

Furthermore, when both $k$ and $b$ are sufficiently large, we can apply the central limit theorem to get $\hat{\beta}_j - \gamma_n \approx N(0, \sigma^2)$. Here $N(.,.)$ is the normal distribution. If the confidence level is set at 99%, the desired confidence interval is given as $(\gamma_n - 2.56\sigma/\sqrt{b}, \gamma_n + 2.56\sigma/\sqrt{b})$.

It should be noted that the length of the confidence interval is an approximation when the variance is obtained by simulation.

### 4.4.2 Experimental results

The experiments conducted in the following are designed to address two issues: the efficiency of the rate tilting and the robust of the rate tilting. Both are discussed below.

**The efficiency of rate tilting**

To complement our theoretic work based on bounding the relative error, we study in this section the conditions that rate tilting is efficient through experiment.

In our experiment, we use the model of two queues in tandem, each with a dedicated buffer. We use three sub-models, named 1a) and 1b) and 1c), as they were described previously (see page 66). In all these models, the capacity of the first buffer is fixed at 20. Let $\lambda$ be the arrival rate to the first queue, and let $\mu_1$ and $\mu_2$ be the service rate of the first and second server, respectively. The traffic parameters $\lambda$, $\mu_1$, and $\mu_2$ differ in the sub-models, thus they will be listed before reporting the experiment results. In the movable server model, we assume that the second server, when seeing no clients, joins
the first server immediately. The second server returns when customers are in the second buffer. After the two servers join, the service rate in the first station becomes \( \mu_1 + \mu_2 \). We are interested in the probabilities that the length of second queue exceeds a certain value \( n \). We set the target level to be above 20. Note that in direct simulation, we observe no replication reaching a level above 20. Hence there is no way to estimate \( \gamma_n \) by direct simulation, unless we use many more replications.

**Table 4.1: Efficiency of rate tilting: tandem queues, model 1a)**

Traffic parameters are as follows.
\( \lambda = 1.0, \mu_1 = 4.0, \mu_2 = 2.0 \) (the bottleneck is the second server)
We use 65000 replications.

<table>
<thead>
<tr>
<th>Target level</th>
<th>IS simulation</th>
<th>Exact</th>
</tr>
</thead>
<tbody>
<tr>
<td>65000 replications</td>
<td>1.249 ( 10^{-6} ) ± 2.4%</td>
<td>1.27 ( 10^{-6} )</td>
</tr>
<tr>
<td>( n = 20 )</td>
<td>1.195 ( 10^{-12} ) ± 2.4%</td>
<td>1.21 ( 10^{-12} )</td>
</tr>
<tr>
<td>( n = 40 )</td>
<td>1.31 ( 10^{-18} ) ± 2.4%</td>
<td>1.15 ( 10^{-18} )</td>
</tr>
</tbody>
</table>

To test the efficiency of the rate-tilting method, we run fast simulations using rate tilting, with 65,000 or 130,000 replications, when each cycle is a replication. We simulate on a SUN workstation of Ultra5-10, running an operating system of Sun OS 5.7. The simulation is usually accomplished within a few minutes. We conducted four experiments. The first three experiments are for the purpose of examining the efficiency of rate tilting, and the last one is to test the sensitivity of rate-tilting when \( x^\dagger \) and \( x_1 \) differ, where \( x^\dagger \) is an approximation of \( x_1 \). In these experiments, several typical conditions for the tandem queues are chosen. The choice of the parameters is designed to verify the theoretical results discussed earlier. Other than these considerations, the parameters are chosen ad-hoc.

**Table 4.2: Efficiency of rate tilting: tandem queues, model 1b)**

Traffic parameters are as follows.
\( \lambda = 1.5, \mu_1 = 2.0, \mu_2 = 3.0 \) (the bottleneck is the first server)
We use 130000 replications.

<table>
<thead>
<tr>
<th>Target level</th>
<th>IS simulation</th>
<th>Exact</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>3.172 ( 10^{-6} ) ± 50%</td>
<td>2.87 ( 10^{-6} )</td>
</tr>
<tr>
<td>( n = 20 )</td>
<td>1.074 ( 10^{-13} )</td>
<td>4.67 ( 10^{-13} )</td>
</tr>
<tr>
<td>( n = 60 )</td>
<td>7.067 ( 10^{-21} )</td>
<td>3.53 ( 10^{-20} )</td>
</tr>
</tbody>
</table>
We get the estimates of the hitting probabilities through rate tilting approach and report them in the tables. The results of the first three experiments are summarized in Tables 4.1 to 4.3. The column marked “IS Simulation” is the rate-tilting estimator of $\gamma_n$ by simulation, and the column marked “Exact” is the true value calculated through numeric methods. In reporting the result, I provide an interval estimate where the confidence level is set at 99%. In Table 4.2, some estimates obtained by IS simulation are reported without the confidence interval, because the calculated confidence interval far exceeds 100%, which is meaningless.

Table 4.3: Efficiency issue: tandem queues, model 1c) – movable server model

<table>
<thead>
<tr>
<th>Target level</th>
<th>IS simulation</th>
<th>Exact</th>
</tr>
</thead>
<tbody>
<tr>
<td>$n = 20$</td>
<td>$6.749 \times 10^{-8} \pm 11%$</td>
<td>$6.01 \times 10^{-8}$</td>
</tr>
<tr>
<td>$n = 40$</td>
<td>$4.357 \times 10^{-15} \pm 6.7%$</td>
<td>$4.09 \times 10^{-15}$</td>
</tr>
<tr>
<td>$n = 60$</td>
<td>$3.367 \times 10^{-22} \pm 7.5%$</td>
<td>$3.29 \times 10^{-22}$</td>
</tr>
</tbody>
</table>

By examining these results, we note that the rate tilting approach works fine for Example 1a), but that in Example 1b), there is a degradation of efficiency because of the curse of boundary. Note that Example 1b) has many transitions at the boundary, which degrades the efficiency of rate tilting according to our analysis (see page 66). In Example 1c), rate-tilting works better than in 1b). This is because, in 1c), the second server helps to clear the backlog at the first buffer, thus the build-up of first queue on the boundary drops significantly. This is confirmed by tracing the simulation replications. The analysis shows a strong relationship between the boundary behavior and the efficiency of rate-tilting. We conclude that the rate-tilting estimator works fine provided the compliant boundary condition is met. This is consistent with the theory discussed in this chapter.

To justify the attention to the boundary, we note the following point. In Example 1c), we note that $\gamma_n$, the hitting probability of interest, drops almost two orders of magnitudes when compared with Example 1b), even if the boundary condition is modified only slightly. This shows that $\gamma_n$ may be very sensitive to the changes at the boundary.

In example 1c), the fast server becomes movable, and the probability of excessive
backlog in front of himself drops. For some readers, this may seem counter-intuitive, as the movable server contributes part of his time serving elsewhere, hence has less time tending the queue in front of himself, which seems to lead to higher excessive backlog probability. However, I have verified that the results presented here is valid.

A tentative explanation of this counter-intuitive behavior is as follows. In the movable server model discussed here, the second server (who is faster) helps the other server when he sees no clients, and is back immediately when finishing service on that customer. Hence, those customer the second server helps in the first service can come in earlier to receive the second service. This means that when the second server becomes movable, he can starts the corresponding service earlier, and has more time to clear these “earlier customers” from the system before later arrivals join the queue. It is not a surprise that this will decrease the probability of excessive backlog in front of the second server.

The robustness of rate tilting

The value of tilting parameter $x_1$ may be approximated, so the tilting parameter applied in tilting would deviate from the true value. Let $x^\dagger$ be an approximation of $x_1$. If one extends the rate-tilting transform with approximate tilting parameter $x^\dagger$ to obtain a new matrix $Q^\dagger$, then the regularity condition (as defined in (4.1)) is violated. To make $Q^\dagger$ to be truly an infinitesimal generator, one enforces the regularity condition by changing $Q^\dagger$ on the diagonal elements only. The necessary amounts of change are very small, provided $x^\dagger$ is close enough to $x_1$.

Note that the rate tilting estimator (2.11) is unbiased, irrespective of the tilting parameter. However, the efficiency may suffer. The question arises on how much efficiency will be lost. We will address this using experiments.

The results of the last experiment are summarized in Table 4.4. The row marked “Exact” shows the true value calculated through (non-simulation) numeric computation methods. In each row, we use a different tilting parameter, and examine how the accuracy degrades with a fixed simulation budget (we use 65,000 replications), when the tilting parameter moves away from the optimal $x_1$. We indicate the ratio of the tilting parameter applied in the experiment to the optimal parameter $x_1$. For example, in the row dealing
Table 4.4: Sensitivity analysis of rate tilting, using model 1a)
Traffic parameters are as follows:
$\lambda = 1.0, \mu_1 = 4.0, \mu_2 = 2.0 \ (\rho_1 = 0.25, \rho_2 = 0.50)$
We use 65000 replications.

\[
\begin{array}{|c|c|c|c|}
\hline
\text{Tilted parameter} & n = 20 & n = 40 & n = 60 \\
\hline
\text{Exact} & 1.27 \times 10^{-6} & 1.21 \times 10^{-12} & 1.15 \times 10^{-18} \\
1.2x_1 & 1.207 \times 10^{-6} & 1.268 \times 10^{-12} & 1.498 \times 10^{-18} \\
1.1x_1 & 1.225 \times 10^{-6} & 1.183 \times 10^{-12} & 1.144 \times 10^{-18} \\
1.05x_1 & 1.252 \times 10^{-6} & 1.168 \times 10^{-12} & 1.129 \times 10^{-18} \\
\hline
x_1 \ (\text{opt.}) & 1.249 \times 10^{-6} & 1.195 \times 10^{-12} & 1.131 \times 10^{-18} \\
0.95x_1 & 1.238 \times 10^{-6} & 1.177 \times 10^{-12} & 1.171 \times 10^{-18} \\
0.9x_1 & 1.230 \times 10^{-6} & 1.206 \times 10^{-12} & 1.138 \times 10^{-18} \\
0.8x_1 & 1.216 \times 10^{-6} & 1.220 \times 10^{-12} & 0.779 \times 10^{-18} \\
\hline
\end{array}
\]

with tilting parameter $1.05x_1$, the tilting parameter applied is 5% greater than the true $x_1$.

From these results, we think that the rate-tilting might be reasonably robust. For example, with the relative deviation of the tilting parameter within -5% to +5% of $x_1$, for target level $n = 20$, $n = 40$, and $n = 60$, the additional relative errors induced of estimate of $\gamma_n$, are all within -5% to +5% of the true value of $\gamma_n$. However, as that tilting parameter moves further away, and the target level is high, Table 4.4 shows some obvious degradation of simulation accuracy.

### 4.4.3 Verification of fast simulation

It is necessary to show that the simulation, in particular, the IS estimator, has been implemented properly. Otherwise, the experiment result given may be doubtful. For example, one may suggest that the calculation of likelihood ratio is not handled properly at the boundary level, hence when there are many transitions there, the simulation results goes off the course. For our reason, we need to eliminate this possibility, by verifying the IS estimator. For this purpose, I run both the direct simulation and IS simulation for modest rarity index $n$, to estimate the hitting probability $\gamma_n$, and check that they agree within a precision. When $n < 10$, for the model I investigate (reported earlier), $\gamma_n$ is typically in the range of 1 to $10^{-3}$, so results of sufficient accuracy can be obtained from
either direct simulation or IS simulation, without much difficulty either way. Once both results are available, it is checked whether or not they agree with each other within a prescribed precision. They did agree in all cases.

The agreement of experiment result with some theoretic analysis (shown before) is another indication that our conclusion about the efficiency of rate tilting is valid.

4.5 On Pilot Study Using Approximation and Simulation

Two parameters: the dominant eigenvalue $x_1$ and its associated eigenvector $\tilde{n}_1$ are needed in rate tilting. Prior to this point, we have assumed that $x_1$ and $\tilde{n}_1$ are known. Now, we investigate how this information could be obtained. We call the procedure to find $x_1$ and $\tilde{n}_1$ a pilot study. The pilot study may be performed either by analytic methods, or by pilot simulations.

One may doubt on the efficiency of an approach that involves eigenvalues, since the explicit calculation of eigenvalues and eigenvectors is usually prohibitively expensive for large systems. There is ground for this doubt, yet fortuitously, in quite a number of situations, the dominant eigenvalues $x_1$ can be obtained with less effort than all the other eigenvalues. Once we have found $x_1$, it is usually less difficult to find $\tilde{n}(x_1)$.

Following, a mini-survey is conducted on a number of situations in which the dominant eigenvalue $x_1$ can be found efficiently. In Jackson queueing networks, good approximation of $x_1$ can be obtained by simple calculation. For example, consider the tandem queue example given in the text, with the first buffer of infinite size. Here for the probability that the second queue exceeds a designated $n$, as $n$ increases, the asymptotic decay rate $x_1$ is $\lambda/\mu_2$. When the first buffer has finite size, we have $x_1 < \lambda/\mu_2$, but $\lambda/\mu_2$ can be used as an approximation. In some other cases, we can take advantage of some special structures on $\Gamma(x)$, e.g. $\Gamma(x)$ has a tri-diagonal structure. Better still, there are situations that have a closed-form solution for the dominant eigenvalue (see, e.g., [41], equation (38) in [64] and page 468 in [30]). The model specification could be more involved. For example, in [12], the authors consider a typical multiplexing system, where the eigenvalue problem can be decomposed into small coupled problems. In such a situation, the solution
of the eigenvalues is difficult, except the dominant eigenvalue, which can be obtained more simply than all the others (e.g. Elwalid and Mitra [12], Theorem 7.1, equation (2.20) and Proposition (2.1)).

Another possibility to obtain \( x_1 \) is to scale down the given model. Let us examine the model of two queues in tandem for this purpose. Let the length of first line be the phase, hence \( m \), the number of phases, corresponds to the capacity of the first buffer. If \( x_1 \) changes little as \( m \) changes, we can perhaps use the \( x_1 \) obtained when \( m \) is small to approximate the \( x_1 \) when \( m \) is large. This is true, as found by looking at the exact analytic solution. Suppose, in our example, \( \lambda/\mu_1 = 0.50 \) and \( \lambda/\mu_2 = 0.33 \), where \( \lambda, \mu_1, \mu_2 \) are the rate of arrival, the first server and the second server, respectively. Then the following results are obtained: at \( m = 5 \), \( x_1 = 0.2637 \), at \( m = 10 \), \( x_1 = 0.2921 \), and at \( m = 40 \), \( x_1 = 0.3095 \), at \( m = \infty \), \( x_1 = 0.3333 \). Note that, for small \( m \), finding \( x_1 \) has a low complexity, thus we can resort to an analytic solution whenever we do not have a better approach. This discussion suggests to find \( x_1 \) as follows:

- **Step 0.** For a level/phase process in consideration, identify the basic structure of phases.

- **Step 1.** Scale the problem down to a smaller problem, with a relatively small number of phases. This smaller problem preserves the basic structure of the original one. Solve this smaller problem to find the largest generalized eigenvalue. Let \( x'_1 \) denote the eigenvalue solution for the smaller problem, i.e. with \( m' \) phases, \( m > m' \).

- **Step 2.** Set \( x_1 = x'_1 \) as an approximate value for the larger problem (i.e. with \( m \) phases).

In addition, we might also look at obtaining the dominant eigenvalue and its associated eigenvector by some pilot simulation. In the worst case, if none of the above is applicable, then there is a question of finding the (generalized) dominant eigenvalue \( x_1 \) from equation (4.7). The matrix \( \Gamma(x) \) is of size \( m \times m \), and involves only \( m \) phases. In comparison, the original state space is \( nm \), in which \( n \) is the number of levels in the model. Hence the size of the state space has already been reduced from \( nm \) to \( m \). This is possible only when
we take the eigenvalue approach. To summarize, I am encouraged by all the results above that practical situations exist where the dominant eigenvalue can be found efficiently, yet the complete solution is more complicated.

**On the change of boundary conditions**

It is noteworthy that the parameters $x_1$ and $\tilde{\eta}_1$ are independent of the specification at the boundary levels, because they are the solution of (4.7) which does not involve any boundary level. We claim that this independence is an advantage of the rate tilting method.

Why? Note the changes at the boundary may be of practical importance, because it is often important to compare different policies (e.g. in network control), and such policies may change only the boundaries, as when a server becomes movable (see page 66, Example 1c). Using our method, the pilot analysis for the tilting parameters needs to be found only once, then we can go for efficient simulations, irrespective of the change of boundary (provided the boundary conditions are compliant). However, in a numeric solution method, we have to repeat the numeric solution process for any boundary change.

### 4.6 Comparison with some previous methods that change event rates

As I mentioned earlier, there are previous methods which change event rates with the intention to make rare events occur more often, for instance, the method of Parekh & Walrand and the method of Kroese & Nicola. The comparison of my method with the one of Parekh & Walrand is straightforward. The comparison with the method of Kroese & Nicola is not. In fact, a number of adjustments have to be made to make the two methods comparable.

**4.6.1 Comparison with the Parekh-Walrand estimator**

We find estimates of the hitting probability $\gamma_n$ by using each of the two approaches. The model used is the tandem queue, model 1a). The results are reported in Table 4.5. We can see that the results are almost the same. In conclusion, in this “nice case”, rate tilting
Table 4.5: Comparison of efficiency: rate tilting vs. the method of Parekh & Walrand
Experiment on tandem queues, model 1a)
Traffic parameters are as follows.
\[ \lambda = 1.0, \mu_1 = 4.0, \mu_2 = 2.0 \quad \text{(the bottleneck is the second server)} \]
We use 65000 replications.

<table>
<thead>
<tr>
<th>Target level</th>
<th>Rate tilting</th>
<th>Exact</th>
<th>Parekh-Walrand</th>
</tr>
</thead>
<tbody>
<tr>
<td>n = 20</td>
<td>1.249 \times 10^{-6} ± 2.4%</td>
<td>1.27 \times 10^{-6}</td>
<td>1.251 \times 10^{-6} ± 2.4%</td>
</tr>
<tr>
<td>n = 40</td>
<td>1.195 \times 10^{-12} ± 2.4%</td>
<td>1.21 \times 10^{-12}</td>
<td>1.195 \times 10^{-12} ± 2.4%</td>
</tr>
<tr>
<td>n = 60</td>
<td>1.131 \times 10^{-18} ± 2.4%</td>
<td>1.15 \times 10^{-18}</td>
<td>1.130 \times 10^{-18} ± 2.4%</td>
</tr>
</tbody>
</table>

performs equally well as the Parekh-Walrand estimator. Further explanation on why the two methods lead to results so close will be given in Chapter 5. There, it will be shown that the tilted system resulted from rate tilting “acts” very close to the one used in Parekh & Walrand.

For the “troubled case”: model 1b) or similar, Parekh-Walrand estimator can hardly observe a build-up at the non-bottleneck, but in rate tilting we can. Hence without doubt, the rate tilting method discussed here has an enhanced efficiency.

4.6.2 Complete cycles, A-cycles, and estimators for stationary probability involving rare event set

The method of Kroese & Nicola does not work on complete cycles, as I did in this study. Instead, the so-called A-cycles, also called pseudo regenerative cycles, are used. There are two kinds of metrics that can be estimated using the method of Kroese & Nicola. One kind of metrics are specified on A-cycles, which will be introduced later. The other kind of metrics are the stationary probabilities like \( \pi_n \) (the stationary probability of residing at level \( n \)).

The concept of an A-cycle is important for understanding the method of Kroese & Nicola (to which I compare the efficiency of rate tilting method), so I describe this concept first. Let us choose a set \( A \) of states, in which \( A \) acts as a threshold that divides the state into two parts. A-cycle refers to any sample path between the two successive entries to set \( A \). Fast simulation is applied only to the A-cycles. This concept is introduced to bypass the “boundary” in a level/phase process, such that the process during an “A-cycle” can
be modeled using Markov additive process (note that Markov additive process does not describe the boundary). Suppose in a level/phase process, the boundary consists of levels up to $N$. Then set $A$ can be chosen as any level above $N$, e.g. all states at level $N+1$. On a per $A$-cycle basis, another concept can be introduced, which is the conditional probability of entering level $n$ (with $n > N$) in an $A$-cycle, conditional on starting from $(N + 1, j)$ (level $N+1$, phase $j$). Here, I denote this probability by $(f^A_{N+1, j})$, where the superscript $(A)$ indicates the probability is assessed on a per $A$-cycle basis.

A more meaningful metric is $\pi_n$. For obtaining $\pi_n$, the above probabilities $(f^A_{N+1, j})$, $j = 1, 2, \ldots, m$, are useful. Following [49], this approach proceeds as follows. An estimate of the stationary probability is the ratio of the expected time at level $n$ in an $A$-cycle and the expected $A$-cycle duration. However, for this estimator to work correctly, the starting phase of $A$-cycles must be sampled from the equilibrium distribution, conditioning on set $A$. The equilibrium distribution conditioning on set $A$ is estimated using direct simulation. (Note that in the rate tilting method as discussed earlier, all cycles start from the same designated state. Hence, the equilibrium distribution of the boundary states is not a concern in the rate tilting method.)

In the following, I will refer to the estimator for $(f^A_{N+1, j})$ using the method of Kroese & Nicola as the $K$-$N$ estimator; respectively the estimator for $\pi_n$ using their method as the extended $K$-$N$ estimator. By doing this, a clear distinction is made between the experiment work done on these two kinds of metrics.

Now, I write out the extended $K$-$N$ estimator, with the notation used in this thesis. Again, let $C_n$ denote the target level $n$. For each sample path $\omega$, let $1_{C_n}^{(A)}(\omega)$ be the indicator function, and let $t_{C_n}^{(A)}(\omega)$ be the time remaining in $C_n$ of $\omega$, after reaching $C_n$ in an $A$-cycle, and before leaving $C_n$ again. Let $L^{(A)}(\omega)$ be the likelihood ratio of $\omega$ with respect to the change of probability measure (since fast simulation is applied). Here, the superscript $(A)$ indicates that sample paths are taken on a per $A$-cycle basis. Precisely, let $K$ be the number of replications generated, the estimator for $\pi_n$ are as follows:

$$
\hat{\pi}_n = \frac{(1/K) \sum \omega t_{C_n}^{(A)}(\omega) L^{(A)}(\omega)}{E[\text{duration of an A-cycle}]}.
$$

(4.43)
The denominator is estimated from the direct simulation. In generating the replications, please note the following. To start each replica in an A-cycle, a state of set $A$ must be chosen. This choice is randomized according to the equilibrium distribution, conditional on set $A$. This equilibrium distribution is an estimate obtained by direct simulation. Note that the estimator of $\pi_n$ can also be written as:

$$\hat{\pi}_n = \frac{(1/K) \sum_j \sum_{\omega_j} t_{C_n}^{(4)}(\omega_j) L^{(4)}(\omega_j)}{E[\text{duration of an A-cycle}]}.$$  \hspace{1cm} (4.44)

Here, $\omega_j$ indicates that a replication starts from phase $j$ of set $A$. If $K_j$ is the number of A-cycles starting from phase $j$, then $K_j/K = \hat{\pi}_{A,j}$, where $\hat{\pi}_{A,j}$ is the estimate of stationary probability of being in phase $j$, conditional on set $A$.

In the following, I discuss on how to implement the program such that the distribution of starting phases of A-cycles conforms to the equilibrium distribution, conditional on $A$. To facilitate this discussion, we may think that the extended K-N estimator consists of two stages. First, direct simulation is used to generate the equilibrium distribution on a chosen level outside the boundary. Second, the fast simulation is applied to A-cycles, with the starting phases taken from the equilibrium from step 1. (Note that the generation of equilibrium distribution, conditioning on set $A$, is independently conducted from those fast simulation of A-cycles. Hence regarding them as two stages does not change the approach.) The two stages work together to form a simulation for estimating $\pi_n$. To generate the equilibrium distribution conditional on set $A$, naively, one can run $K$ replications in direct simulation until steady-state is reached. Let this time be denoted by $T_0^*$. Then on each replica, the first state visiting set $A$ after $T_0^*$ is taken, and one uses all such states to constitute an equilibrium distribution, conditioning on set $A$. Another approach is to choose $K = K'K''$, and proceed as follows. Run $K'$ replications until steady-state at time $T_0^*$, and for each replica, take $K''$ successive visits to set $A$ after time $T_0^*$, then use all such states to estimate $\pi_{A,j}$ for each $j$. Notably, the second approach takes less effort, as only $K'$ burn-in period is simulated, but may be less accurate, as correlation exists among successive visits of the same replica. However, whichever of the two approach is chosen, it is essential to note that there is an estimate error for any $\pi_{A,j}$, and this error inevitably
has an impact on the final estimate of $\pi_n$. Unfortunately, how this is handled is out of the scope of the paper of Kroese & Nicola.

4.6.3 Comparison with the method of Kroese & Nicola

For the comparative study with the (extended) K-N estimator, we only look the “troubled” cases where there are many transitions at the boundary. Hence I work on model 1b). The probability being estimated is the stationary probability $\pi_n$.

Note that in the original paper of Kroese & Nicola [41], the method of estimating hitting probability $\pi_n$ is suggested, but no experimental result has been reported. Hence for this comparison, I have to implement the estimators using each of the two methods, and apply them to model 1b). In Table 4.6, the estimates by using my approach, and by using the method of Kroese & Nicola, are reported. From these experiments, we see that the extended K-N estimator yields more accurate results than our method. However, note that significantly more effort is spent in producing the equilibrium distribution. Indeed,

<table>
<thead>
<tr>
<th>Target level</th>
<th>Rate tilting</th>
<th>Exact</th>
<th>extended K-N estimator</th>
</tr>
</thead>
<tbody>
<tr>
<td>$n = 20$</td>
<td>$3.927 \times 10^{-7}$</td>
<td>$3.57 \times 10^{-7}$</td>
<td>$3.784 \times 10^{-7}$</td>
</tr>
<tr>
<td>$n = 40$</td>
<td>$1.183 \times 10^{-14}$</td>
<td>$5.38 \times 10^{-14}$</td>
<td>$5.057 \times 10^{-14}$</td>
</tr>
<tr>
<td>$n = 60$</td>
<td>$9.560 \times 10^{-22}$</td>
<td>$4.78 \times 10^{-21}$</td>
<td>$4.684 \times 10^{-21}$</td>
</tr>
</tbody>
</table>

Table 4.6: Efficiency comparison of rate tilting vs. the method of Kroese & Nicola

The estimate of stationary probabilities for tandem queues, model 1b): Traffic parameters are as follows.

$\lambda = 1.5, \mu_1 = 2.0, \mu_2 = 3.0$ (the bottleneck is the first server)

We use 130000 replications.

Each $A$-cycle must start from some phase, which must be produced using direct simulation. The effort is significant, both because of the necessity of “burn-in” period (such that the steady-state is reached), and because direct simulation is used as part of the solution. In order that the phase distribution on set $A$ converge, it takes a fair amount of time, esp. for those “low-accessible phase” (i.e., $\pi_{A,i}$ is very small). For instance, consider model 1b) as discussed earlier. We choose $A$ as level 1, i.e. any state such that the length of second line is 1. For state $(1, 10)$ (i.e. the length of second line is 1, and the length of first line is 10), let us estimate the conditional stationary probability $\pi_{A,10}$ ($A$ is level 1) by
averaging over all the replications. From the experiment, it is found that as time elapses, the estimate of \( \pi_{A,10} \) converge to the true value very slowly. Suppose \( K' = 2000 \) replicas are used. For each of them, we simulate until time \( T_0 \). At time \( T_0 = 40 \), the estimate \( \hat{\pi}_{A,10} \) is not even close to the true value. Hence, for our purpose, the equilibrium has not yet been reached at this point. Even at time \( T_0 = 1000 \), the estimate \( \hat{\pi}_{A,10} \) still has an error larger than 5\%. Say, we set the burn-in time at 2000. In comparison, for this model with a target level \( n = 60 \), it was observed that in the tilted system, the average duration of a cycle is about 42. By a simple calculation, we find that during the same period for the burn-in of each of the \( K' \) replicas, we can simulate other 47.6 (=2000/42) complete cycles. Hence significantly more effort is spent in generating the equilibrium distribution of the boundary states.

Some readers may suggest to look for those most frequently visited states, and consider that the system is in equilibrium when the averaged probabilities of visiting those states do not change much. Unfortunately, in some models, we risk a severe under-estimate for the stationary probability \( \pi_n \), if from those less frequently visited states, less \( A \)-cycles (than the true equilibrium) are started. This is because that in some models, the less frequently visited states are exactly the states that, conditional on starting from them, an \( A \)-cycle will have a much higher potential to finally reach the target. For example, for model 1b), with the target level \( n = 20 \), we find that as \( j \) increases from 1 to 20, the value of \( \pi_{A,j} \) (where \( A \) is level 1) increases three orders of magnitudes, while the value of \( (f_{20}^{(A)}) \) (i.e. the probability of reaching the target in an \( A \)-cycle, provided starting from phase \( j \) of level 1) decreases by two to three orders of magnitudes. Of course, the tandem queue model 1b) is a case that is particularly hard to handle. In many other situations (e.g. the models considered in [5]), generating the equilibrium distribution is not as hard as this. In those situations, either the number of phases being considered is lower, or the potential to reach the target conditional on different phases does not vary on an extremely large scale.

Also remarkably, using the extended K-N estimator, the error for estimating \( \pi_n \) is much larger than estimating the hitting probability, conditional on starting from a specific state. (In [41], for estimating \( (f_{10}^{(A)}) \) in a model similar to model 1b), a standard relative error of 0.11\% is reported for any level \( n \), which is impressive!) By noting that the likelihood
ratio is much more variable when paths from any state of $A$ are sampled, than the situation when only paths from a specific state are sampled, the increased variance should not be a surprise.

### 4.6.4 The result of this comparative study

The results of this section can be summarized as follows:

1. The rate tilting has a efficiency no worse than the Parekh-Walrand estimator (equally well for the “nice” case, better for the “troubled case”).

2. For the hitting probability, my method can produce both $\gamma_n$, the one defined on complete cycles, or the conditional probability. The method of Kroese & Nicola only reported the conditional probability on a per A-cycle basis. In this regard, they reported different metrics.

3. Both methods can be used to compute $\pi_n$, that is, the probability of residing on a high level. For the tandem queue model 1b), the experiment result favors the method of Kroese & Nicola for accuracy, with a note that significantly more effort is spent in generating the equilibrium distribution among a non-boundary level (level 1, for this example).

4. The conclusion on which estimator is better may be model dependent. In particular, it is shown in my study that for model 1c): the movable server model, rate tilting shows a significant improvement over model 1b).

In my rate tilting method, there is no need to generate the equilibrium distribution $\pi_{A,t}$. This is especially convenient when a model is studied with different boundaries (e.g. in the movable server model, the fast server comes back when helping the other to finish $b$ clients; when $b$ varies, we have different boundary conditions), and in rate tilting it is simpler to assess the variance. Hence when the rate tilting method is efficient, it is a good fast simulation estimator.

The movable server is not studied in the work of Kroese & Nicola. Further work is required to apply (and adapt) the K-N estimator to this model, and compare the efficiency
with that of rate tilting. This may deserve another study.

4.7 Summary

In the rate tilting approach, a proper construct of rate tilting relates to a generalized eigenvalue problem involving the infinitesimal generator matrix of the process being considered. I have shown that the relative error of the hitting probability resulting from the proposed simulation remains bounded as the level increases, provided that the boundary set of the state space satisfies certain conditions.

The method here are developed for any level/phase process. This covers a wide range of models. Particularly, in the queueing context, any Jackson network can be modeled using the level/phase process. Of course, the applications of level/phase processes are not limited to queueing models. The queues in tandem are used merely as an example. However, this example has many features, as many different boundary and different large deviation behaviors can be observed from the model by varying the traffic parameters.

Some comparison with previous methods (the method of Parekh & Walrand, and the method of Kroese & Nicola) has been conducted experimentally. Based on these results, conclusions have been given.

In the next chapter, this model will be studied further. However, the focus will shift to exploring the behavior in the tilted system, and how that behavior relates to the simulation efficiency. This leads to more insights into the method.
Chapter 5
Further Study on Rate Tilting, Simulation Efficiency and System Behavior

In Chapter 4, the rate tilting method has been developed. This chapter expands that material by discussing the system behavior issues relating to rate tilting and its efficiency.

Fast simulation often relies on asymptotics. In the current setting, the level \( n \) is a rarity index. For any probability metric of the system, if it converges as the rarity index \( n \to \infty \), then the limit is called an asymptotic. The asymptotics often assume much simpler forms than the exact metrics, and can be regarded as an approximate knowledge of the system. A simple example in QBD is the asymptotic decay rate, which is the probability that a replica in level \( n \) survives to a higher level \( n + 1 \), provided \( n \) is sufficiently large.

Consider a stable level/phase process. Let us choose an intermediate level \( l \), and a target level \( n \). Let \( \omega \) be the sample path of a replica. Consider a cycle starting at some level, say 0, and ending when it reaches level \( n \) or returns level 0. We define two conditional probabilities, as follows:

\[
(p^0_l)_j = P(\omega \text{ enters phase } j \text{ as it hits level } l \text{ for 1st time } | \omega \text{ starts from level } 0),
\]

\[
(f^n_l)_j = P(\omega \text{ reaches level } n \mid \omega \text{ starts from } (l, j)).
\]

In the setting of the QBD process, a replica must pass through each level one at a time. Hence all replicas reaching \( n \) will pass one phase of level \( l \). As before, let \( \gamma_n \) denote the probability of hitting \( n \). Apply conditioning to phase \( j \) to obtain,

\[
\gamma_n = \sum_j (p^0_l)_j (f^n_l)_j. \tag{5.1}
\]
The discussion in this chapter will be based on equation (5.1). In this chapter, by considering \( (p^n)_j \) when \( l \to \infty \) and \( (f^n)_j \) when \( n \to \infty \), two asymptotics are introduced. The discussion of the two asymptotics helps us to gain a better understanding of the system behavior, and provide probabilistic interpretation for the parameters \( \eta \) used in rate tilting. I also obtain some theoretical results in providing indicators as to when rate tilting runs into problems.

The rest of this chapter is organized as follows. I first compare the rate tilting method and the exponential twist method in Section 5.1. The distinction between the two methods are important. The main results of this chapter are the establishment of the two asymptotics just mentioned. These results are presented in Section 5.2. The rest of this chapter is devoted to the proof and explanation of these results.

### 5.1 Rate Tilting vs. Exponential Twist: Similarities and Differences

Because the rate tilting method and the exponential twist method are discussed using different specifications, I first look at the connections between these models, and then illustrate their similarities and differences.

*The random walk, the DTMC, and the level/phase process: a specific perspective*

The random walk is a DTMC. We distinguish one-dimensional and multidimensional random walks. One-dimensional random walks lead to transition matrices with repeating columns except the situation that the random walks are restricted (there is a barrier). If there are many dimensions, we have to pick one dimension, call it level, and the other dimensions have to be somehow expressed by phases.

If the random walk is bounded, except for the first axis, then the transition matrix looks like the one of a level/phase process, except that the level/phase process is a CTMC. However, CTMCs can always be converted to DTMCs.

Suppose a random walk is governed by an underlying sequence \( \{X_t\} \), let us assume each \( X_t \) takes integer values between \(-g\) and \( h\). The probability of taking a value \( j\),

99
\(-g \leq j \leq h, \text{ is } p_j, \text{ with } \sum_j p_j = 1. \text{ If a particle starts at 0, with the random walk above, then the state space contains all the integer points on the real axis. If the random walk is restricted, e.g., it cannot pass to the left of } x = 0, \text{ then the state space is } \{0, 1, 2, \ldots\}. \text{ Note the recursive relation,}

\[ S_{k+1} = S_k + X_{k+1}, k > 0. \]  

(5.2)

It can be derived from the equation above that the transition matrix describing the transition from \( S_k \) to \( S_{k+1} \) has the following structure:

\[
\begin{pmatrix}
p_{00} & p_1 & p_2 & \cdots & p_h \\
p_{10} & p_0 & p_1 & \cdots & p_h \\
p_{20} & p_{-1} & p_0 & p_1 & \cdots & p_h \\
& \ddots & \ddots & \ddots & \ddots & \ddots \\
p_{-g} & p_{-2} & p_{-1} & p_0 & p_1 & \cdots & p_h \\
& & \ddots & \ddots & \ddots & \ddots & \ddots
\end{pmatrix}.
\]  

(5.3)

Remark:

There are a number of applications whose transition probability matrices have repeating rows, just like \( P \) has. Examples in queuing scenario include discrete GI/G/1 queue, M/D/c, here \( c \) is the number of service channels, and M/G*/1. Hence the transition probability matrices with repeating rows is an important problem [29].

Obviously, \( P \) is a level/phase process, except that the size of each block is 1. Because the size of each block is 1, all vectors and matrices involved in the characteristic equation become scalars. Let us see now the discrete version of characteristic equation (this is equation (4.13), which we repeat below):

\[ x^h \eta = \Gamma^{(D)}(x) \eta. \]

Here, \( \Gamma^{(D)} = \sum_{j=-g}^h x^{h-j} p_j \). For non-trivial solution, we have \( x > 0 \) and \( \eta \neq 0 \). Dividing
both sides by $x^{h\eta}$, we obtain,
\[ 1 = \sum_j x^{-j} p_j. \]  \hspace{1cm} (5.4)

On the other hand, in choosing the optimal twist exponent $\theta^*$, one uses equation (2.22) which is $M(\theta) = 1$, where $M(\theta) = E[exp(\theta X)]$. Equation $M(\theta) = 1$ expands to
\[ 1 = \sum_{j=-h}^{h} e^{\theta p_j}. \]  \hspace{1cm} (5.5)

By introducing the transform $x = e^{-\theta}$, the equation above immediately reduces to equation (5.4). From this analysis, we say that $x$ here plays a similar role as $\theta$ does in exponential twist.

In conclusion, **rate tilting transform is a natural extension of the exponential twist.** In the example discussed here, the two methods are the same.

*Rate tilting differs from the exponential twist in particular type of higher dimensional processes, e.g. higher dimensional random walk model like the tandem queue example.*

Though rate tilting is regarded as an extension of the exponential twist, it is worth noting the extension is really non-trivial. In general, the level/phase process cannot be reduced to a process with a level only, and we have to consider the level/phase process in multi-dimensions (i.e. dimension $d > 1$).

Let $d = 2$ for simplicity. Consider a CTMC. Let $e$ be an event in the system, which cause a change in at least one of the state variables. (Think of the tandem queue mode, where the state variables are the lengths of the queues. Arrivals, moves to the next service node, and departures from the system are the events, which causes the change of queue lengths.) Suppose the event release rate of a certain type $e$ is independent of the current state (except at the boundary). In the following, we only consider the case when the current state, say, $(l, j)$, is not at the boundary, and show that exponential twist and rate tilting are usually different. In the following, for the rate of an event $e$, we simply write $r(e)$, as in Section 2.4.5. An event $e$ always has an effect, say, to transfer the system from $(l, j)$ to $(l', j')$. Note that if the current state is at the boundary, then we should not use...
the notation of $r(e)$ (as discussed earlier, the rate of an event at the boundary depends on the state).

For the system induced by exponential twist, let $\tilde{r}(e)$ be the corresponding event release rate of the tilted system. Similar to formula (2.29), the relation between the event release rate of the new system and the original system is as follows:

$$\tilde{r}(e) = r(e) \frac{\exp\{\theta_1(l' - l) + \theta_2(j' - j)\}}{M(\theta_1, \theta_2)}. \quad (5.6)$$

Because $(l' - l, j' - j)$ is the effect of event $e$, it is independent of the current state $(l, j)$. Also, in (2.29), we use $M(\theta_1, \theta_2; l, j)$; here $l, j$ does not appear in function $M()$ anymore. The reason is that since each event release rate is independent of $(l, j)$ (note we consider only non-boundary states, as indicated earlier), $M()$ is also independent of $(l, j)$.

Let us call $\tilde{r}(e)/r(e)$ the rate adjustment coefficient. The rate adjustment coefficient takes a state independent schema in the formula above. State independent schema means that the rate adjustment coefficient depends on the current event type only, and is independent of the current state.

Rate tilting works differently. If the event $e$ causes a transfer from $(l, j)$ to $(l', j')$, then the event rate in the tilted system is as follows (see equation (4.10)):

$$\tilde{r}(e) = r(e)x_1^{-\eta(j')} \frac{\eta(j')}{\eta(j)}. \quad (5.7)$$

For this situation, the rate adjustment coefficient depends on the current phase $j$, unless $\eta(.)$ takes a particular form. Generally, we say that the rate adjustment coefficient expresses a phase dependent schema.

If the rate adjustment coefficient $\tilde{r}(e)/r(e)$ takes a state independent schema, then we have the following: the fact that $r(e)$ is independent of the current state implies that $\tilde{r}(e)$ is also independent of the current state. However, if the rate adjustment coefficient $\tilde{r}(e)/r(e)$ takes a phase dependent schema, then $\tilde{r}(e)$ may depend on the current phase even if $r(e)$ does not. This turns later to be an importance difference.
Rate tilting has close connections with previous methods. In particular, rate tilting applied to discrete time model of MAP/D/c is equivalent to the method of Chang et al.

Now let us look at a different model: the discrete time MAP/D/c (described below), which was discussed in Chang et al. [5]. The purpose of the following discussion is to investigate the connection of rate tilting with the method of Chang et al., which also changes event rates in a system to speed up simulation. I will show that (1) the level/phase model can be used to model the discrete time model of MAP/D/c, and (2) rate tilting method becomes equivalent to the method of Chang et al. for this particular model.

First, I describe the discrete time model of MAP/D/c, which stands for a queue with Markov arrival process (widely used in the literature, see e.g. [48] [5]) and deterministic service. (Chang et al. considers a feed-forward network of such queues. Here we consider only a single node.) In the model, time is slotted. Arrival packets come in bulk at the start of each time slot, following a Markov arrival process. During each time slot, the server can serve up to c packets, which depart the system at the end of a time slot. Immediately after this instant, the next bulk arrival comes at the start of next time slot. For this model, of interest is the probability that at the end of a time slot, the queue length (backlog) in the system exceeds a large designated number, n. For convenience, we call the arrival and the change of number of jobs in the backlog the input process and the output process, respectively. When comparing the backlog at the end of time slot t to the one at the time slot t − 1, we have:

\[ \text{Input-output relation: the change of number of jobs in the backlog during time slot } t \text{ is the number of new arrivals in slot } t \text{ minus the number of service completions in this time slot.} \]

A deterministic service implies that, in any time slot, the number of service completion is a constant (say, c) unless the backlog has been cleared up during the slot (this happens when the buildup level is less than c, which can be regarded as the system boundary). This will be referred to as a simple input-output relation. Let the backlog (number in the system) be the level. For a Markov arrival process (MAP), neither the distribution for the number of new arrivals nor the change of phase depend on the current backlog. This is
the level invariance in *the input process*, which carries over to the output process, except at the boundary. If there is an upper bound for the number of new arrivals in a time slot (say the number of new arrivals < h), then the output process satisfies the definition of level/phase process. As this model can fit into the framework of a level/phase process, the rate tilting method becomes applicable. Otherwise, when there is no upper bound for the number of arrivals, the output process is slightly more general then the level/phase process (however, the basic argument that follows still goes through).

The boundary (i.e. levels below c) in this model is of less importance, because in this model, there are usually not many transitions at the boundary. In the period prior to a large build-up, the average arrival size must be larger than c, so it takes a limited amount of time to bring an empty system to a system with backlog > c, which is out of the boundary. Typically, on the most likely path to a build-up level > n, the system will not return to the boundary as the backlog keeps increasing. This scenario is different from the continuous time tandem queue model that has been discussed before, where it is possible that there are many transitions at the boundary, i.e., one server is idling while there is a large buildup at the other server; this status occurs on the most likely path (see my remark on the boundary behavior, page 66). With this note, it is reasonable that for the MAP/D/c model, we focus at the non-boundary.

More technical details of the method of Chang et al. is described below.

In a Markov arrival process, let a(t) be the arrival size at time slot t, let j(t) be the phase at the end of time slot t. Now define the conditional accumulated arrival size distribution as,

\[ (F^{[Ar]}_{t})_{ij} = Pr(a(t) \leq a, j(t) = j | j(t - 1) = i). \] (5.8)

Obviously, here it is assumed that the phase space is enumerable, hence the state can be labeled as 0,1,2,...,etc. The superscript [Ar] stands to indicate an arrival process (to distinguish this from the queue length build-up process, which is the usual model discussed previously in this thesis). Meanwhile, a Markov process underlies the transition between phases.

Usually, a(t) only takes integer values. For this situation, the definition can take a
simplified form,
\[ (P_a^{[Ar]})_{ij} = Pr(a(t) = a, j(t) = j|j(t - 1) = i). \] (5.9)

Now let \( P_a \) denote a transition matrix which includes all probabilities for those transitions where the change of level is \( d \). Precisely, using this new notation, the *simple input-output relation* can be expressed as,
\[ (P_a^{[Ar]})_{ij} = (P_{a-c})_{ij}. \] (5.10)

That is, we can simply shift the transition matrix of the input process down \( c \) levels to obtain the transition matrix of the output, i.e., the process of queue length build-up; this works for any level except at the boundary.

In the method of Chang et al., a matrix is defined as follows,
\[ [G(\theta)]_{ij} = \int_{\alpha \geq 0} e^{\theta \alpha} (dF_a^{[Ar]})_{ij} = \sum_{\alpha \geq 0} [e^{\theta \alpha} (P_a^{[Ar]})_{ij}]. \] (5.11)

In their paper, they give the condition that the optimal choice of parameter \( \theta \) is the following:

\[ (C^*) \quad \text{The effective bandwidth of the arrival process is } c. \]

(For the definition of the effective bandwidth, see Chang et al.’s original paper.) According to their analysis, for the MAP/D/c model, condition \((C^*)\) is given by the following equation,
\[ \log sp(G(\theta)) = c\theta. \] (5.12)

Here, for a non-negative matrix, \( sp(.) \) denotes its maximal eigenvalue, also known as the spectral radius.

We are now ready to show the equivalence between the two methods. A simple way to look at this is as follows. Since both the rate tilting method and the method of Chang et al. is applicable to the discrete MAP/D/c model in discussion, and in both methods, the optimal change of sampling is unique, hence the two methods should be similar. To show they are exactly equivalent, we proceed as follows. Substitute the simple input-output relation
into the definition formula of $G(\theta)$ to get

$$
\begin{align*}
[G(\theta)]_{ij} &= \sum_{a \geq 0} [e^{\theta a}(P_{a-c})_{ij}] \\
&= e^{\theta c} \sum_{d \geq -c} [e^{\theta d}(P_d)_{ij}],
\end{align*}
$$

(5.13)

At the last step of the derivation above, we have set $d = a - c$. Now multiply this equation by $e^{-\theta c}$, and let $d = a - c$, $x = e^{-\theta}$, to get

$$
e^{-\theta c}(G(\theta))_{ij} = \sum_{a \geq 0} e^{\theta(a-c)}(P_{a-c})_{ij} = \sum_{d \geq -c} x^{-d}(P_d)_{ij}.
$$

(5.14)

On the other hand, recall that in rate tilting, in choosing the right change of measure, one of the major concerns is the choice of $x$, such that equation (4.13) holds. (Here, the discrete form of the characteristic equation is used, since we are dealing with a discrete time model.) Multiplying equation (4.13) by $x^{-h}$, we get

$$
x^{-h}(\Gamma^{(D)}(x))_{ij} = \sum_{d = c}^{h-c} x^{-d}(P_d)_{ij}.
$$

(5.15)

As the R.H.S. of equation (5.14) and (5.15) are the same, the L.H.S. of the two equations must equal. This brings us to (below, the subscript $(i, j)$ is dropped, as two matrices must equal if all the entries are the same),

$$
e^{-\theta c} (G(\theta)) = x^{-h} (\Gamma^{(D)}(x)),
$$

(5.16)

and consequently

$$
e^{-\theta c} sp(G(\theta)) = x^{-h} sp(\Gamma^{(D)}(x)).
$$

(5.17)

Take logarithms at both sides, to get

$$
-c\theta + \log sp(G(\theta)) = -h \log x + \log sp(\Gamma^{(D)}(x)).
$$

(5.18)

In the method of Chang et al., $\theta$ is such chosen that (5.12) holds, hence in the equation
above, the L.H.S. is 0. On the other hand, from equation (4.13), we see that in rate tilting, the choice of \( x \) is such that \( sp(\Gamma^{(D)}(\theta)) = x^h \), which implied the R.H.S. of the equation above is 0. This means that the choice of parameters are actually the same, subject to the transform \( x = e^{-\theta} \). Further, they construct a conjugate process on the non-boundary, which is exactly the same as what I did in the rate tilting transform (this conclusion can be drawn by comparing the definition of rate-tilting transform in equation (4.8) of this thesis and equation (30) in Chang et al. [5]). For this model, the boundary is of less importance, as indicated earlier. Now it can be concluded that the two methods applied to the particular model as discussed above, become equivalent outside the boundary.

But, note that the equivalence of the two methods holds for this specific model only. Generally, they are intended for different situations. On one hand, if the service time distribution is not deterministic, then the simple input-output relation does not hold. Consequently, an equation corresponding to (5.18) cannot be derived in more general scenarios. On the other hand, for the solution of build-up probability of a feed-forward network of MAP/D/c, the method suggested in [5] is very competitive, as their method works with the effective bandwidth of each individual node, which effectively avoids large state space.

## 5.2 Main Results

In the last section, rate tilting, exponential twist, and the method of Chang et al. are compared in several different models. This helps to clarify the method, and place my development in proper context. Now I return to the main focus, which is the rate tilting, and the related issues on system behavior and efficiency.

The following discussion refers to formula (5.1), which involves two series \((p_0^1)_j, (f^n)_j\). Let \( p_0^1 = ((p_0^1)_1, ..., (p_0^1)_m) \). Similarly, let \( f_l^n = ((f_l^n)_1, ..., (f_l^n)_m) \). As stated, two asymptotics can be obtained, as shown in the following results (R1) and (R2). In the following, note that scalar \( x_1 \) is a constant independent of level and phase; meanwhile vectors \( \vec{u}_1 \) and \( \vec{n}_1 \) does not depend on level \( l \).

**R1 (asymptotic phase distribution of the original system)** When level \( l \) is high enough,
we have,
\[ \tilde{p}_0 = D(x_1)^n \tilde{u}_1 + o(x_1^n) \]  
(5.19)

where \( D \) is an appropriate constant, and \((x_1, \tilde{u}_1)\) is the eigen-pair satisfying \( \tilde{u}_1 \Gamma(x_1) = \tilde{0} \). Here, \( x_1 \) is the largest eigenvalue inside the unit circle.

Hence \( \tilde{p}_0 \), with proper scaling, converge to the vector \( \tilde{u}_1 \). We call the asymptotics as the asymptotic phase distribution for the original system.

R2 (asymptotic local potentials of the original system) When level \( n \) is high enough, for level \( l \) far from the boundary, we have,
\[ \tilde{f}_l^n = D'(x_1)^{n-l} \tilde{\eta}_1 + o(x_1^{n-l}) \]  
(5.20)

where \( D' \) is an appropriate constant, and \((x_1, \tilde{\eta}_1)\) is the eigen-pair satisfying \( \Gamma(x_1) \tilde{\eta}_1 = \tilde{0} \). Again, \( x_1 \) is the largest eigenvalue inside the unit circle.

Hence \( \tilde{f}_l^n \), with proper scaling, converge to the vector \( \tilde{\eta}_1 \). We call the asymptotics as the asymptotic local potentials for the original system.

As discussed previously (Chapters 3 and 4), the efficiency of a simulation method depends on the behavior of the tilted system. This motivates the study of behaviors in the tilted system. By rate tilting, we obtain a tilted system that has an asymptotic phase distribution, meanwhile, starting from any state, a replica of the system will hit any level \( n \) almost surely. The system behavior in the tilted system and its significance will also be shown in this chapter.

5.3 P-F Right Eigenvector As Asymptotics of the Local Potentials

In the rate tilting transform, the eigenvector \( \tilde{\eta}_1 \) is important. A probabilistic interpretation of this eigenvector is given as result (R2) in Section 5.2. Providing such an interpretation also sheds light on why \( \tilde{\eta}_1 \) is important for the simulation efficiency.
In our study, it is convenient to introduce the concept of local potentials as follows. Let \( \tilde{f}_i^{[n]} \) be the vector containing the probabilities \( (\tilde{f}_i^{[n]})_i \) that a process starting in level \( l \), phase \( i \), will finally hit one of the phases of level \( n \). We call each such probability the local potential associated with state \( (l, i) \). The corresponding probability vector \( \tilde{f}_i^{[n]} \) is called the vector of local potentials. Consider \( \tilde{f}_i^{[n]} \), with fixed \( d \) and varying \( n \). As \( n \to \infty \), \( \tilde{f}_i^{[n]} \) may or may not converge. The following shows that, if the process \( Q \) is positive recurrent, then the sequence converges.

**Lemma 5.1** If \( Q \) is positive recurrent, then the sequence \( \tilde{f}_i^{[n]} \) must converge component-wise, as \( n \to \infty \). In other words, there is a vector \( (\tilde{f}_d)_i \), such that, for any component \( i \),

\[
(\tilde{f}_i^{[n]})_i = (\tilde{f}_d)_i (1 + o(1)).
\]

**Proof.**

Let \( \tilde{f}_i^{[n]}(b) \) be defined similarly to \( \tilde{f}_i^{[n]} \), except that we count only those paths where the lowest level visited is level \( b \).

First, when \( b - 1 > N_0 \),

\[
\tilde{f}_{n+1-d}^{[n]}(b) = \tilde{f}_{n-d}^{[n]}(b - 1).
\]

The reason is that \( \tilde{f}_{n+1-d}^{[n]}(b) \) and \( \tilde{f}_{n-d}^{[n]}(b - 1) \) have identical underlying transition matrices. Specifically, if we take any path that contributes to the L.H.S. of the above, and shift down the path one level below, then we have a path that contributes to the R.H.S., and vice versa.

Consider all replicas that starts from level \( n - d \) and finally reaches \( n \). In these replicas, the lowest visited level \( b \) is either above \( N_0 \), or below \( N_0 \). Starting from this relation, we have,

\[
\tilde{f}_{n-d}^{[n]} = \sum_{b \leq N_0} \tilde{f}_{n-d}^{[n]}(b) = \sum_{b > N_0} \tilde{f}_{n-d}^{[n]}(b).
\]

Now we show that the R.H.S. increases as \( n \) increases. Making use of (5.22), we have

\[
\sum_{b > N_0} \tilde{f}_{n+1-d}^{[n+1]}(b) - \sum_{b > N_0} \tilde{f}_{n-d}^{[n]}(b) = \tilde{f}_{n+1-d}^{[n]}(N_0 + 1) + \sum_{b > N_0+1} \tilde{f}_{n+1-d}^{[n+1]}(b) - \sum_{b > N_0} \tilde{f}_{n-d}^{[n]}(b) > 0.
\]
Note in the second line of the above formula, the difference of the two terms in the square brackets $[]$ is 0 by equation (5.22), and the other term must be positive, so we have strict positiveness. Obviously, as $n$ increases, the R.H.S. of (5.23) has an upper bound (1, 1, ..., 1), since each local potential is a probability. Increasing sequences with an upper bound converge, so the R.H.S. converges. The second term on the L.H.S. of (5.23) converges to 0, as each summand of that term converges to 0, because for a stable system, as $n \to \infty$, $f_{N_0}^{[n]} \to 0$ for any fixed $N_0$.

With this, one concludes that the first term on the L.H.S. must also converge. □

Now define $\tilde{f}_d = \lim_{n \to \infty} \tilde{f}_d^{n}$. Because of Lemma 5.1, $\tilde{f}_d$ is well-defined. We call $\tilde{f}_d$ the asymptotics (limit) of the local potentials, and will show it can be calculated from the given infinitesimal generator matrix $Q$. The following development is based on a recurrence relation of $\tilde{f}_d$. To simplify, I will restrict the discussion to the QBD process only. As indicated earlier, any level/phase processes can be transformed to a QBD process by “re-blocking”.

**Theorem 5.1** Consider a QBD process. Let $Q$ and $\tilde{f}_d$ be as defined before.

(a) We have,

$$\bar{0} = Q_{-1} \tilde{f}_d^{n-1} + Q_0 \tilde{f}_d^{n-1} + Q_1 \tilde{f}_d^{n-1-1},$$

(5.24)

where $\bar{0}$ is a vector with all components being 0.

(b) As $d \to \infty$, there is a constant vector $c_1 \bar{v}_1$ and $0 < x_1 < 1$, such that for any $i$,

$$\bar{f}_d(i) = x_1^i (c_1(i) \bar{v}_1) + o(x_1^i),$$

(5.25)

which characterize the asymptotic exponential decay of $\tilde{f}_d$. Here, $x_1 > 0$, and $\bar{v}_1$ is a vector with all components non-negative.

**Remark:**

The theorem above provides a new probabilistic interpretation for $\bar{v}_1$, as the limit of local potentials. Theorem 5.1 (b) is a restatement of results (R2) in Section 5.2. This
result may also give hints on how to find $\eta_1$ via simulation. For example, one may simulate to estimate $(\tilde{f}_d)_i$, and from that, find the components of $\eta$, except for a constant factor. Practically, a number of difficulties need to be solved. This question remains open.

**Proof.**

First, I show (a). Consider the vector of local potentials $(\tilde{f}^{[n]}_i)$. Let $P$ be the transition probability matrix for the jump chain embedded at each transition of $Q$ (matrix $P$ is defined in equation (4.12), page 70). By conditioning on the first transition after a process reaches level $l$, we obtain

$$
\tilde{f}^{[n]}_i = P_{l,i-1}\tilde{f}^{[n]}_{i-1} + P_{l,i}\tilde{f}^{[n]}_i + P_{l,i+1}\tilde{f}^{[n]}_{i+1}
$$

$$
= P_{l-1,i}\tilde{f}^{[n]}_{i-1} + P_{0,i}\tilde{f}^{[n]}_i + P_{1,i+1}\tilde{f}^{[n]}_{i+1}.
$$

By using equation (4.12), and simplifying, the equation above can be reduced to the following,

$$
\tilde{g} = Q_{l-1}\tilde{f}^{[n]}_{i-1} + Q_0\tilde{f}^{[n]}_i + Q_{1}\tilde{f}^{[n]}_{i+1}.
$$

(5.26)

Now let $n \to \infty$. According to Lemma 5.1, we have: $\tilde{f}^{[n]}_i \to \tilde{f}_{n-i}$ when $n \to \infty$. Taking the limit of each term, the equation above reduces to equation (5.24) in the theorem.

Second I show (b). There is a theory of difference equations on vectors. According to that theory, the basic solution of equation (5.24) takes the following form,

$$
\tilde{f}_i = x\tilde{\eta}(x).
$$

(5.27)

Substituting equation (5.27) into (5.24), we obtain,

$$
\tilde{g} = x^{n-i-1}(x^2Q_{-1} + xQ_0 + Q_1)\tilde{\eta}(x).
$$

(5.28)

Let $\Gamma(x) = (x^2Q_{-1} + xQ_0 + Q_1)$, the equation above reduces to

$$
\tilde{g} = \Gamma(x) \cdot \tilde{\eta}(x).
$$

(5.29)
The equation above will be referred to as the characteristic equation for the stochastic process $Q$. We seek solution $x$ in the complex number field.

The general solution of (5.24) must be a linear combination of the basic solutions, as follows,

$$
\tilde{f}(x) = \sum c_j(x)^d \tilde{\eta}(x_j).
$$

Among all $x_j$'s, only those within the unit circle are of interest (otherwise the sum at the right hand of formula above will diverge). Also, from Seneta [56], it is known that for equation (4.7), the solution $x = x_1$ with the maximum norm within the unit circle must be real, and it must be a single root. When $d$ is sufficiently large, all terms almost vanish when compared to the term $c_1(x_1)^d \tilde{\eta}(x_1)$ (because $x_1$ has the maximum norm). When taking the $i$-th component for each vector, we get equation (5.25) stated in this theorem.

\[\square\]

With this discussion, a new insight is gained into why rate tilting is advantageous. Through the use of local potentials in rate tilting, the dynamics of the tilted system is adjusted according to local potentials, so as to encourage the moves to the phases with higher local potentials to hit the target. Hence, the most likely path to the target set can be enforced better, and such paths are intuitively more informative.

The concept of local potentials will also be employed to develop another method called adaptive splitting, which will be addressed in Chapter 7.

*When the phase dependent schema degenerates*

This continues from the discussion on page 101, Section 5.1. The phase dependent schema may degenerate to the state independent schema if $\eta(j)$ takes a certain form. If this happens, then the rate tilting is identical to the exponential twist. For this situation, the rate adjustment coefficients in formulas (5.6) and (5.7) must agree. We have,

$$
\frac{x_1 \eta(x')}{\eta(j)} = e^{-\theta_1 (\theta_2^j - j)}.
$$

(5.31)
By inspection, we have a simple solution to the above equation,

\[ x_1 = e^{-\theta_1} \]

\[ \eta(j) = c(e^{\theta_2}j), \quad (5.32) \]

where \( c \) is a constant. Here, the first formula provides the relation of \( x_1 \) and \( \theta_1 \), and the second formula shows that the asymptotic local potential takes an exponential growth, with respect to the phase variable \( j \). Conversely, if the asymptotic local potential takes an exponential growth as in equation (5.32), then (5.6) concurs with (5.7) when setting \( x_1 = e^{-\theta_1} \).

### 5.4 Some Results From the Matrix Geometric Solution

In results (R1) in Section 5.2, we provide asymptotics for \((p^l_0)\) when \( l \to \infty \) (with proper scaling). The value of \((p^l_0)_j\) gives the probability that a replica of the system enters level \( l \) at phase \( j \), starting from \((0,0)\) (level 0, phase 0). However, in the classic analytic solution, most results are for \( \pi_l \), which is the equilibrium distribution of a system being in a certain phase of level \( l \). Indeed, \( \tilde{\pi}_l \) and \( \pi_l \) are closely related, similar to the relation of the probability of hitting a certain level and the equilibrium probability of being in that level. For this reason, we will look at \( \pi_l \). Following the common use in literature of matrix geometric solution, we use \( \tilde{\pi}_n \), where \( n \) is an arbitrary level (not necessary the target level).

This section considers the equilibrium distribution in the original system, and Section 5.5 considers the equilibrium distribution in the tilted system. The result in this section are not new. However, it is essential for understanding the behavior of the original system, and for understanding the equilibrium in the tilted system, which, in turn, characterize the system behavior in the tilted system. Hence, this section is included for the convenience of readers. Some results from the matrix geometric solution are needed to establish the asymptotics in the tilted system, and its relation with the asymptotics in the original process. Hence, those results are also included in this section.
The method of matrix geometric solution was pioneered by Neuts [46]. In this method, many matrices in question have a probabilistic meaning (i.e., each entry of the rate matrix $R$ has a clear probabilistic interpretation), which is an advantage. On the other hand, in the eigenvalue approach, many amounts have no probabilistic interpretation. However, the computational complexity is reduced by just looking for the dominant eigenvalue.

5.4.1 The matrix geometric solution and the dominant term in phase distribution

Note that the classic matrix analytic method deals with discrete time Markov chains. We may consider the DTMC embedded at each transition epoch for this purpose (as in Chapter 4). However, since CTMC is used later, it is natural to conduct the discussion in terms of the CTMC. Hence, the CTMC is used. Many asymptotics, like the equilibrium distribution and the rate matrix, have different meanings in CTMC and DTMC. However, these differences are minor.

Suppose we are considering a recurrent QBD process in continuous time that has an infinitesimal generator matrix $Q$. If $Q$ is a QBD process, then the matrix geometric solution is applicable, which is as follows:

$$\tilde{\pi}_n = \pi_0 R^n$$  \hspace{1cm} (5.33)

where $\tilde{\pi}_n$ is the equilibrium distribution vector at level $n$. This vector contains the probability of being in any of the phases in this level at any instant. In continuous time, this is interpreted as the expected proportion of time spent in level $n$. The matrix $R$ is called the rate matrix, and it can be shown that all its entries are positive or zero (i.e. non-negative). Its probabilistic will be given later.

For simplicity, assume $R$ has distinct eigenvalues $x_i, i = 1, 2, \ldots, m$. (This restriction can be lifted, as to be shown later). Then according to the well known Jordan standard form, we can always write:

$$R = J^{-1} \text{diag}[x_1, x_2, \ldots, x_m] J.$$  \hspace{1cm} (5.34)
In the following, \( \vec{u}_i \) and \( \vec{v}_j \) denote row vectors and column vectors, respectively. The matrices \( J \) and \( J^{-1} \) can be partitioned as follows:

\[
J = \begin{pmatrix}
\vec{u}_1 \\
\vec{u}_2 \\
\vdots \\
\vec{u}_m
\end{pmatrix}
\]  

(5.35)

and

\[
J^{-1} = (\vec{v}_1 \ \vec{v}_2 \ldots \vec{v}_m).
\]  

(5.36)

By expanding the matrix-form solution (5.33), we have,

\[
\vec{\pi}_n = \sum_j (\vec{\pi}_0 \cdot \vec{v}_j)(x_j)^n \ \vec{u}_j. 
\]  

(5.37)

Here, \( \vec{\pi}_n \) is a row vector. Let \( c_j = (\vec{\pi}_0 \cdot \vec{v}_j) \). Among all terms such that \( c_j \neq 0 \), the term with the largest \( x_j \) will dominate, as \( j \) increases. If we rearrange \( x_j \) such that \( x_1 \) has the largest norm, then \( \vec{u}_1 \ (x_1)^n c_1 \) will dominate the contribution to \( \vec{\pi}_n \). At sufficiently high levels, when \( n \) increases by 1, the probability decays roughly by a factor of \( x_1 \). This also implies that the equilibrium distribution among phases of a level \( n \) (provided \( n \) is sufficiently large) is roughly proportional to \( \vec{u}_1 \), the left eigenvalue associated with \( x_1 \). This vector therefore characterizes the asymptotic phase distribution.

For a non-negative matrix, the eigenvalue with the largest norm must be real and positive, and the associated eigenvector can be scaled so that its component is real and positive. We call this eigenvalue as the P-F eigenvalue. This is according to the Perron-Frobenius theorem in the theory of non-negative matrices, see [56]. As \( R \) is a non-negative matrix, hence \( x_1 > 0 \), and \( \vec{u}_1 \geq 0 \). Note that even in the presence of some multiple eigenvalues, the conclusion that the asymptotic phase distribution equals \( \vec{u}_1 \) is still valid. This is because that the P-F eigenvalue \( x_1 \) of \( R \) must always be single.

Remark: For \( \vec{\pi}_n \), we have established equation 4.2. Similar equations can be established for \( \vec{\rho}_0 \). In order to do this, we essentially use the same method as the one to convert the
equilibrium distribution of being in a level to the hitting probability of that level (how to
do it is shown in Chapter 4). Hence the discussion here is also valid for \( (p_0^j) \), that is, \( (p_0^j) \) is roughly proportional to \( (\bar{n}_1)_j \), abbreviated as \( u(j) \). Precisely, we have,

\[
(p_0^j) = cu(j)(1 + o(1))
\]  

(5.38)

where \( c \) is a constant that depends on level \( l \) only, and is independent of \( j \). Essentially,
this restates result (R1) in Section 5.2.

5.4.2 Factorization of the matrix \( \Gamma(x) \)

Naoumov [45] provides a factorization of \( \Gamma(x) \) using several related matrices. The factorization is of interest, as these matrices have clear physical meanings. From this, we can better understand the eigenvalues and eigenvectors of \( \Gamma(x) \), which relate closely to system behavior.

For a QBD process, we have already used \( R \), and we now provide a probabilistic interpretation for \( R \). We will also need certain matrices \( Y \) and \( G \). Their respective interpretations are given in [27], as follows:

\( R \): the matrix \( R \) records the expected time in level \( n + 1 \), before returning to level \( n \), for each time unit spent in state \( i \) of level \( n \).

\( G \): the matrix \( G \) records the probability of the Markov chain going to level \( n \) for the first time, given that it started in phase \( i \) of level \( n + 1 \).

\( Y \): the matrix \( Y \) is a sub-generator matrix which records the rate of going from level \( n \) to the same level, possibly visiting levels above \( n \), but avoiding any levels below \( n \).

The following result is quoted from Naoumov [45]:

\[
\Gamma(x) = (R -Ix)Y(Gx - I),
\]  

(5.39)
where $\Gamma(x)$ is defined as before. We will refer later to the relation above as Naoumov’s factorization theorem of $\Gamma(x)$. From here, we see immediately the following points: any eigenvalue of $R$ is also an eigenvector of $\Gamma(x)$; and the reciprocal of any eigenvector of $G$ is also the eigenvalue of $\Gamma(x)$. It is also shown in the literature that all eigenvalues of $G$ are within or on the unit circle (because $G$ is a stochastic matrix), and if the process being considered is stable, then all eigenvalues of $R$ are within the unit circle.

Note $x = 1$ is an eigenvalue of $\Gamma(x)$ (with multiplicity 1). Hence it must be either an eigenvalue of $R$, or an eigenvalue of $G$, but not both. With the assumption that $Q$ is a stable (recurrent) process, $R$ cannot have the eigenvalue 1, hence $x = 1$ must be an eigenvalue of $G$. In the next section, we perform an analysis of the P-F eigenvalue and the associated eigenvector for the tilted system.

5.5 The Asymptotic Phase Distribution in Tilted System

For the purpose of studying the large deviation behavior, the two systems (the original and the tilted) are combined, since their behaviors are closely related. The original and the tilted systems are referred to as dual systems. Now the asymptotic phase distribution will be studied. The following result is important to the study here:

R3 For a level/phase process, the asymptotic phase distribution in the tilted system induced by rate tilting, is the same as the asymptotic phase distribution in the original system, conditional on reaching a certain high level.

In the original system (assumably stable), the P-F eigenvalue of the rate matrix is $x_1 < 1$. The associated left eigenvector is $\overline{u}_1$. Unfortunately, it would be quite difficult to observe $\overline{u}_1$ from direct simulation. The reason is that there are few observations for high levels, and for a low level, the boundary plays an important role. Hence the phase distribution typically does not equal $\overline{u}_1$. On the other hand, result (R3) suggests a way to find the asymptotic phase distribution for the tilted system. From that, we learn the asymptotic phase distribution in the original system, conditional on reaching the rare event set. Hence, we have an alternative way to study the large deviation phase distribution, which is important information for the system being studied.
The discussion in this section and subsequent sections has two purposes:

D1 To reveal the working mechanism of rate tilting;

D2 To test the idea of using the asymptotic phase distribution as an indicator for the large deviation behavior, which further differentiate systems that can be efficiently simulated using the exponential twist from those that cannot.

5.5.1 An eigenvalue correspondence theorem in rate tilting

Since we use eigenvalues to understand the system behavior, an eigenvalue correspondence theorem, which converts the eigenvalues in the original system to the eigenvalues in the tilted system, is required. This theorem is used to find the eigenvalues in the tilted system. This is discussed below.

Referring to equation (5.39), it is obvious that any eigenvalue of \( R \) is also a generalized eigenvalue solution such that \( \det(\Gamma(x)) = 0 \). Conversely, if \( x \) is a solution of \( \det(\Gamma(x)) = 0 \), then it must be either a eigenvalue of \( R \), or its reciprocal must be an eigenvalue of \( G \).

Now for the tilted system, one defines, similarly to \( \Gamma(x) \),

\[
\Gamma(y) = y^2 \tilde{Q}_1 + y \tilde{Q}_0 + \tilde{Q}.
\]  

(5.40)

The relation of \( \tilde{Q}_d, d = -1, 0, 1 \) can be found from equation (4.8). Into the definition formula above, we substitute equation (4.8) (in which \( \Delta = \text{diag}[\eta(x_1)] \)), to obtain

\[
\Gamma(y) = y^2 \Delta^{-1}(x_1 Q_1) - 1 + y \Delta^{-1} Q_0 - 1 + \Delta^{-1}((1/x_1) Q_1) - 1
\]

\[
= (1/x_1) \Delta^{-1} \Gamma(x_1 y) \Delta.
\]  

(5.41)

To find generalized eigenvalue solutions for the tilted system, we solve \( y \) from the equation \( \det(\Gamma(y)) = 0 \). The equation is equivalent to,

\[
0 = \det(\Delta^{-1} \Gamma(x_1 y) \Delta) = \det(\Gamma(x_1 y)).
\]  

(5.42)

Hence \( x_1 y \) is a generalized eigenvalue of \( \det(\Gamma(x)) = 0 \), where \( x_1 \) is the tilting factor,
and $y$ is the generalized eigenvalue solution for the tilted system. Immediately we get the following:

**Lemma 5.2** Each generalized eigenvalue of the tilted system is equal to the related generalized eigenvalue of the original system, divided by $x_1$.

The following theorem states the correspondence between the tilted system and the original system in eigenvalues and the associated eigenvectors.

Let $Q$ be a recurrent (stable) QBD process, and $R$ be its associated rate matrix. Suppose the eigenvalues of $R$ are $x_j, j = 1, 2, ..., m$. Let $x_1$ be the P-F eigenvalue of $R$, and let its associated eigenvector be $\vec{\eta}_1 = (\eta_{(1)}, \eta_{(2)}, ..., \eta_{(m)})$. The tilted system is defined by the infinitesimal generator matrix $\tilde{Q}$, which is constructed from rate tilting, with tilting factor $x_1$ and $\vec{\eta}_1$.

**Theorem 5.2** Now consider the tilted process $\tilde{Q}$, and the new rate matrix $\tilde{R}$ associated with $\tilde{Q}$.

(a) The new rate matrix $\tilde{R}$ associated with the process $\tilde{Q}$ has the following eigenvalues: \{1, $x_2/x_1$, ..., $x_m/x_1$\}. Among these eigenvalues of $\tilde{R}$, the P-F eigenvalue of $\tilde{R}$ is 1.

(b) For $\tilde{R}$, the P-F eigenvectors are as follows. The right eigenvector is $\tilde{\mathbf{1}} = (1, 1, ..., 1)$, and the left eigenvector is

$$\tilde{u}_1 = \tilde{u}_1 \cdot \text{diag}[\eta_{(1)}, \eta_{(2)}, ..., \eta_{(m)}].$$

Later, we will refer to this theorem as the *eigenvalue correspondence theorem*.

There are different ways to prove this theorem. Our approach exploits the relation of the rate matrix $R$ with $\Gamma(x)$, and uses the fact that all eigenvalues of $G$ should be within the unit circle (as we consider $G$ a stochastic matrix). We note that the generalized eigenvalues include both those belonging to $R$ and those belonging to $G$, hence it is necessary to identify which belong to which. This “identification” constitutes the essence of the proof below.

Proof.
From previous discussion, we know that $\bar{Q}$ is also the transition matrix of a QBD process. Hence the rate matrix $\bar{R}$ and $\bar{G}$ can be similarly defined.

First we prove part (b) of the theorem. In the original system, $\bar{u}_1$ is the left eigenvector associated with the P-F eigenvalue $x_1$. Then it must also be a generalized eigenvalue solution for $\det(\Gamma(x)) = 0$. Hence, the left eigenvector $\bar{u}_1$ satisfies:

$$\bar{u}_1 \Gamma(x_1) = 0. \quad (5.44)$$

According to (5.41), one has

$$\bar{\Gamma}(y) = (1/x_1) \Delta^{-1} \Gamma(x_1y) \Delta. \quad (5.45)$$

By setting $y$ in equation (5.44) to 1, we get

$$0 = (\bar{u}_1 \Delta) \cdot \Delta^{-1} \Gamma(x_1) \Delta. \quad (5.46)$$

It is obvious now that $\bar{u}_1 \Delta^{-1}$ is the left eigenvector of $\bar{R}$ associated with the eigenvalue 1. According to Lemma 5.2, the eigenvalues in the tilted system are obtained those in the original system divided by $x_1$. Hence the P-F eigenvalue $x_1$ of the original system $x_1$ corresponds to the P-F eigenvalue 1 of the tilted system. Consequently $x = 1$ has just one associated left eigenvector, which is $\bar{u}_1$, as in the following expression: The expression

$$\bar{u}_1 = \bar{u}_1 \Delta = \bar{u}_1 \cdot \text{diag}[\eta(1), \eta(2), \ldots, \eta(m)]. \quad (5.47)$$

Similarly, we can prove that the right eigenvector associated with eigenvalue 1 is $\Delta^{-1} \eta_1 = (1, 1, \ldots, 1) = \bar{1}$. This finishes the proof of part (b).

Now we prove (a). The rationale in identifying which eigenvalues belong to $\bar{R}$ and which belong to $\bar{G}$ is by way of counting the numbers inside and outside the unit circle. The analysis below basically follows [27].

We note that the there are $m$ eigenvalues (counting the multiplicity) for $\bar{R}$, and there are $m$ eigenvalues for $\bar{G}$. This holds similarly for the tilted system, i.e. for $\bar{R}$ and $\bar{G}$. 

120
Because the original process is stable, there are \(m\) eigenvalues inside the unit circle, the eigenvalue 1 on the unit circle, and \(m-1\) outside the unit circle. Dividing by \(x_1, 0 < x_1 < 1\), we obtain the generalized eigenvalues for the tilted system. First, the number outside the unit circle increases by 1, as the generalized eigenvalue 1 for the original system now becomes \(1/x_1 > 1\). There is still one eigenvalue on the unit circle, because the original generalized eigenvalue \(x_1\) becomes the eigenvalue 1 for the tilted system. Within the unit circle, the number decreases by 1, as any eigenvalue \(x_j, j \neq 1\) within the unit circle in the original system, after dividing by \(x_1\), is still within the unit circle, except \(x_1\) itself. For the tilted system, all eigenvalues \(\{x_j/x_1\}\), with \(|x_j| < x_1\), must be a value of \(\tilde{R}\) (any eigenvalues of \(\hat{G}\) cannot be inside the unit circle). All these yield exactly \(m-1\) eigenvalues within the unit circle. There is one more eigenvalue for \(\tilde{R}\), which we claim must be the eigenvalue 1. Otherwise, if \(x = 1\) is an eigenvalue of \(\hat{G}\), then one eigenvalue of \(\tilde{R}\) must take a value outside the unit circle. In this case, if we use \(\pi_n = \pi_0 \tilde{R}^n\) to calculate the expected number of visits to each level in a period, we will find that \(\pi_n\) increases as \(n\) increases. This implies that the system being considered is unstable. On the other hand, that \(\hat{G}\) has an eigenvalue of 1 indicates that \(\hat{G}\) is a stochastic matrix, such that all replicas starting at level \(n\) will come to level \(n-1\), almost surely. Hence the process must be stable. Now we see a contradiction. Hence, the assumption that \(x = 1\) is an eigenvalue of \(\tilde{G}\) cannot be true.

In summary, eigenvalue 1 belongs to \(\tilde{R}\). Because \(x = 1\) has the largest norm (any other eigenvalue of \(\tilde{R}\) has a norm less than 1), hence it must be the P-F eigenvalue of \(\tilde{R}\). This finishes the proof of part (a). \(\square\)

5.5.2 Asymptotic phase distribution – typical and large deviation

With the eigenvalue correspondence theorem just developed, we know that 1 is the P-F eigenvalue in the tilted system, and \(\tilde{u}_1\) (as defined in equation (5.47)) is the associated eigenvector. Note that any other eigenvalue of \(\tilde{R}\) has a norm less than 1. If we write \(\tilde{\pi}_n\) using eigenvalues (similar to formula (5.37) except all eigenvalues and eigenvectors are changed to the ones in the tilted system), then all terms vanish except the one with eigenvalue 1. Consequently, \(c_1\tilde{u}_1\), where \(c_1 = (\tilde{\pi}_0 \cdot \tilde{1})\), dominates the contribution to \(\tilde{\pi}_n\) in
the tilted system.

To facilitate later discussion, let \( u(j) \) be the \( j \)-th component of \( \bar{\alpha}_1 \), and \( \bar{u}(j) \) be the \( j \)-th component of \( \bar{\alpha}_1 \). (This is similar to the use of \( \eta(j) \).) Let us re-examine formula (5.1). The formula was introduced at the start of this chapter, when we consider the hitting probability of level \( n \), conditional on the phase when first entering a certain level \( l \). According to the previous analysis, in a stable level/phase process, the term \( (p'_0)_{j} \) is approximately the typical asymptotic phase distribution, when level \( l \) is sufficiently high. Meanwhile, \( (f_l^n)_{j} \) converges to \( \eta(j) \) (with proper scaling), which is the asymptotic local potential. Following previous results, as \( n \to \infty \), we have,

\[
(p'_0)_{j}(f_l^n)_{j} \to cu(j)\eta(j),
\]

in which the constant \( c \) depends on level \( n \) only, and is independent of \( j \).

Consider starting a large number of replicas. The number of replicas that pass phase \( j \) is proportional to \( (p'_0)_{j}(f_l^n)_{j} \), which is the contribution of those replicas to the hitting probability. On the other hand, expanding equation (5.47) gives

\[
\bar{u}(j) = u(j)\eta(j).
\]

Note that in the original system, \( u(j) \) is the asymptotics of the relative portion of time spent in phase \( j \). However, not all cycles successfully reach level \( n \). If one counts only the portion of time spent in successful cycles (this is like conditioning on reaching level \( n \)), then one gets \( u(j)\eta(j) \). To summarize, equation (5.49) states that, \( u(j)\eta(j) \), the large deviation asymptotic phase distribution in the original system, equals \( \bar{u}(j) \), the typical asymptotic phase distribution in the tilted system. It may be said that the large deviation phase asymptotic distribution becomes the typical asymptotic phase distribution in the tilted system. This is the result (R3), which we have proved now.

Hence, the vector \( \bar{\alpha}_1 \) is an indicator of large deviation behavior of the original system.
5.6 Report on Some Numeric Calculations

From equation (5.49), we realize that $\tilde{\eta}_1$ and $\tilde{u}_1$ can be used as indicators for large deviation behavior of the system.

To characterize the (asymptotic) phase distribution, we can use $\bar{u}_1$ or $\tilde{u}_1$. We are more interested in the tilted system, as that is the system being simulated (in the previous analysis, we have stressed that the simulation efficiency depends on the system behavior). Hence we use $\tilde{u}_1$.

We now need to distinguish between two situations:

(1) type-1 asymptotic phase distribution: (one dominant phase) In this case, there is a unique phase that dominates, i.e., provided the process reaches very high levels, the probability of being in any other phase is fairly small, when compared with the probability of being in the dominant one. When this is the case, $\tilde{u}_1$ has a dominant component, all other components are fairly small, when compared with the dominant one.

(2) type-2 asymptotic phase distribution: (multiple dominant phases) In this case, there are a number of phases such that the probabilities of being in any of these phases are roughly in the same range. Hence there is no single phase that dominates. When this is the case, $\tilde{u}_1$ has many components that are roughly at the same range.

About the asymptotic local potentials, we need to distinguish between the case that $\eta_{(j)}$ grows exponentially as $j$ increases, and the case it does not. In the case that $\eta_{(j)}$ grows exponentially, the rate adjustment takes a state independent schema. In this situation, the rate tilting method and the exponential twist method are almost the same. Otherwise, the rate adjustment takes a phase dependent schema (see the discussion on page 102). For the purpose of making this distinction, we look at $\tilde{\eta}_1$.

In brief, whether or not the buildup path is linear is settled by the asymptotic local potentials, which is $\tilde{\eta}_1$; meanwhile, the vector $\tilde{u}_1$ settles whether or not there is a dominant phase in the buildup path to the rare event set.
In this section, we study the vectors $\tilde{n}_1$ and $\tilde{u}_1$ by conducting some numeric computation. In our experiment, we use the model of two queues in tandem, each with a dedicated buffer. The traffic parameters differ in the sub-models, thus they will be listed before reporting the experimental results. The capacity of the first buffer is fixed at 20. There is no limitation on the capacity of the second buffer, hence the length of queue can be arbitrarily large.

We choose several typical conditions to study the two issues mentioned above. The first two sub-models, named 1a) and 1b), are the same as described previously (see page 66). In these models, we are interested in the probability of reaching a high buildup level in the second queue. Also in these models, the bottleneck server is different.

The respective model parameters follow:

*Model 1a)*: $\lambda = 1.0; \mu_1 = 4.0; \mu_2 = 2.0$.

*Model 1b)*: $\lambda = 1.5; \mu_1 = 2.0; \mu_2 = 3.0$.

Please note that the traffic parameters in models 1(a) and 1(b) are the same as those in Tables 4.1 and 4.2, where we experimentally study the efficiency of rate tilting. This choice is intentional, and helps relate the system behaviors in these models to the efficiencies of rate tilting later.

In many studies involving tandem queues, instead of the queue length at one individual node, the metric of interest is the probability that the total backlog (summing over all nodes) exceeds a certain value. Reflecting this interest, we study the following two models, namely 2a) and 2b). In these two models, we use the sum of the queue length as the level and we use the length of the first line as the phase.

The model parameters are:

*Model 2a)*: $\lambda = 1.0; \mu_1 = 5.0; \mu_2 = 4.0$.

*Model 2b)*: $\lambda = 1.0; \mu_1 = 4.5; \mu_2 = 4.5$.

For the purpose of verifying our theoretical analysis, these experiments were required. Otherwise, the parameters are chosen ad-hoc. For each situation, we calculate $\tilde{n}_1$, the
vector of local potential asymptotics, and \( \tilde{u}_1 \), the vector of phase distributions of visit frequencies. The points are produced in the following way. In figures reporting results on local potentials \( \eta_{(j)} \), the horizontal coordinate is the phase \( j \), and the vertical coordinate is \( \eta_{(j)} \) (scaled by some convenient constant). Note that only those discrete points have real meaning, but we interpolate them with curves to better see the trend of these amounts.

For the P-F right eigenvector \( \tilde{\eta}_{(j)} \) (representing the asymptotic local potentials), the results for model 1a) and 1b) are shown in Figure 5.1, and the results for model 2a) and 2b) are shown in Figure 5.2. On the other hand, for the P-F left eigenvector of the tilted system (representing the large deviation asymptotic phase distribution in the original system), the results for model 1a) and 1b) are reported in Figure 5.3, and the results for model 2a) and 2b) are reported in Figure 5.4.

On the local potentials (reflected in the P-F right eigenvector), we see from Figures 5.1 and 5.2, that in model 1a), the asymptotic local potentials grow (almost) exponentially as the phase index \( j \) increases (because the vertical axis on which we plot the local potentials is set to a logarithmic scale, a straight line implies exponential growth). Meanwhile, in model 2a) the local potential remains almost constant as \( j \) varies. For both models 1a) and 1b), the rate adjustment coefficients follow a state independent schema. (Please refer to equation (5.32), where we stated the condition that a phase dependent schema degenerates to a state independent schema.) Hence, the rate in this model is state independent, and the rate adjustment coefficient takes the state independent schema. Consequently, we expect a linear buildup path to the rare event set. The results conform with the following. According to the time reversal analysis [15] [40] [19], in these two cases, while the buildup of queue occurs dominantly at the bottleneck server at a (almost) constant rate, and the length of queue at the other server remains small. Further, under these conditions, the tilted system develops very closely to the twisted system employed in the exponential twist. Hence the exponential twist also leads to an efficient simulation.

On the other hand, in model 1b), the asymptotic local potentials are not constants, nor do they grow exponentially as phase index \( j \) increases. Consequently, the coefficient of rate adjustment follows a phase dependent schema. This corresponds to another case where the buildup path is not linear, and the tilted system behaves differently from the
Figure 5.1: A diagram of the P-F right eigenvector $\eta_{(j)}$: models 1(a) and 1(b)

Figure 5.2: A diagram of the P-F right eigenvector $\eta_{(j)}$: models 2(a) and 2(b)
Figure 5.3: A diagram of the P-F left eigenvector $\bar{u}_1$ (in tilted system): models 1(a) and 1(b)

Figure 5.4: A diagram of the P-F left eigenvector $\bar{u}_1$ (in tilted system): models 2(a) and 2(b)
twisted system employed in the exponential twist. In this situation, the exponential twist no longer induces a system that can be simulated efficiently, and the proposed rate tilting method has an advantage. More attention should be paid to model 2b). In it, \( \eta(j) \) obviously does not show an exponential growth or decay. However, if we break the curve in question at phase \( j = 10 \), then the first segment shows an exponential decay, and the second segment is almost a constant. However, at this point, the slant of the curve changes abruptly, and consequently, the rate adjustment coefficient changes there abruptly. What is the implication of this? If rate tilting is applied to build the tilted system, then we find an (almost) constant buildup rate in two different regions, and at the location close to the joining place of the two regions, the buildup rate changes sharply. This actually corresponds to multiple buildup paths. At each instant, it depends on the current phase to find which path is most likely. It is also possible to follow one path for a period of time, then cross over to another path.

Now we examine Figures 5.3 and 5.4, which reports \( \tilde{u}_1 \). Models 1a) and 2a) correspond to the type-1 large deviation asymptotic phase distribution, and models 1b) and 2b) correspond to the type-2 large deviation asymptotic phase distribution. In the latter two models, there are slight differences between models 1b) and model 2b). In model 1b), accessing any phase is almost equally likely. In model 2b), the most likely visited phase is either at the low side of the phase \(< 5 \) or high side of the phase \((15 - 20)\), but not in between. There is a branch point at about \( j = 10 \). If a replica is currently in the low side of the phase, it becomes more likely to visit other low phases; the same is true for the high side of the phase. This somewhat reflects the multiple build-up path. In both situations, the exponential twist cannot work efficiently, because it can reinforce only one unique linear buildup path.

As already stated, the traffic parameters in model 1(a) is the same as in Table 4.1. From Table 4.1, one sees that rate tilting is efficient. Meanwhile, Figures 5.1 and 5.3 show that the tilted system for model 1(a) has a relatively simple behavior. Essentially, in the tilted system corresponding to this model, a unique linear buildup path is enforced. In conclusion, for this model, we use a relatively simple tilted system for fast simulation, and we successfully achieve an efficient simulation. This is because the large deviation
behavior in this model is simple, hence a simple tilted system is sufficient to “mimic” the large deviation behavior of this model.

Similarly, the traffic parameters in model 1(b) are the same as in Table 4.2. Table 4.2 shows rate tilting is inefficient when the compliant boundary condition is not satisfied. However, Table 4.3 shows the rate tilting is efficient when the compliant boundary condition is satisfied. The models used in the two tables mentioned only differ in the boundary, which therefore plays a critical role in simulation efficiency, in addition to the asymptotics used to quantify the large deviation behavior. Meanwhile, Figures 5.1 and 5.3 show that the tilted systems corresponding to model 1(b) have much more complicated behaviors than the behaviors shown for model 1(a). This is because the large deviation behavior in this model is very complicated. Hence, it is difficult to induce a tilted system that “mimics” the large deviation behavior of the model. This analysis also suggests that it may be more important to check the compliant boundary condition (discussed in Chapter 4) when the system’s behavior is complicated. Indeed, if the buildup path to the rare event set is not linear, then the process will likely have many transitions at the boundary, thus violating the compliant boundary condition.

5.7 Outlook

I have shown that \( \eta_1 \) represents the asymptotics of the local potentials, and \( \tilde{v}_1 \) represents the asymptotics phase distributions of the tiled system. However, in a large space, obtaining these asymptotics is difficult, and finding them analytically is also complicated.

One possible way out of this difficulty is to down-size the problem. In particular, we solve the asymptotics in a smaller problem (with a smaller \( m, m \) is the number of phases), and then “map” the solution to a larger problem. Previously, I have shown for the particular example of tandem queues, that the asymptotic decay rate \( x_1 \) changes slowly as \( m \) increases. Similar things can be done for \( \eta_1 \) and \( \tilde{u}_1 \). On the other hand, concerning the asymptotics discussed as indicators of system behavior, I have the following result. For the asymptotic local potentials, we only want to check whether \( \eta_{(j)} \) exhibits exponential growth as \( j \) increases. For the asymptotic phase distributions, we only want to check
whether $u_{(j)}$ exhibits the exponential growth as $j$ increases, or whether $\tilde{u}_{(j)}$, $1 \leq j \leq m$ concentrates on one phase or a few phases, or dissipates on almost all phases. The present work shows these properties are mainly settled by the traffic parameters, and are hardly affected by $m$. Hence, analysis of small models can help us gain insight on larger models. Once we know the system’s large deviation behavior, it is easier to assess whether a fast simulation method can work well for a particular model, given its model parameters.

I have discussed the example to down-size the buffer size. This is of practical interest, such as evaluating the buffer overflow probability with respect to different buffer sizes. Another topic to consider is whether down-sizing can be applied in other settings, e.g. number of service channels in a queueing scenario, or even more generally. In this respect, more experimental or theoretical work is required to establish it rigorously.
Chapter 6
A Study of the Adaptive Splitting Method

The previous two chapters were devoted to the development of rate tilting and the analysis of its efficiency. As indicated in Section 2.7, in circumstances involving “trap phases” (i.e. the threshold is not set us consistently), much effort is wasted generating replicas that will not reach the rare event set. In this chapter, I attempt to improve importance splitting method. To overcome the inefficiency caused by the inconsistency of setting up threshold, I introduce the adaptive splitting schema. The adaptive splitting relies on the same asymptotics as that are used for rate tilting. This is an interesting result, which supports my claim that importance sampling method and importance splitting method are two closely related methods.

The origin of splitting techniques can be traced back to as early as 1950’s [38]. Recently, there was a renewed interest in splitting. Splitting has been employed in a series of papers focused on telecommunication [62] [55] [58], and the method looks attractive. Compared to importance sampling, the splitting schema seems much simpler. However, Glasserman and Heidelberger [18] indicated that the seeming simplicity of splitting may be misleading. They also gave counter examples that splitting, as it is now, can fail to be efficient. In fact, no efficiency is guaranteed unless we have a detailed understanding of a process’s rare event asymptotics.

As discussed earlier, in importance splitting, there are a series of thresholds between where a replica starts and the rare event set designated. We apply splitting each time a replica crosses a threshold for the first time, i.e., to split into some new replicas; the new replicas continue evolving independently from the point of the split. The splitting degree is chosen according to the knowledge of some asymptotics in the system. In the existing
splitting schema, the splitting degree neither varies from threshold to threshold, not does it vary with the phase a replica enters as it crosses a threshold. This scenario of splitting is illustrated in Figure 6.1.

![Figure 6.1: A diagram of splitting with splitting degree 2](image)

How the simulation effort is calculated must be stated, as this effort plays a critical role when measuring the simulation efficiency. Measuring the cost is more complicated in the “splitting” scenario than in the IS simulation scenario. Two models have been suggested, namely, the *constant cost model* and the *exact cost model*. The *constant cost model* uses the total number of replicas simulated (including the replicas generated from splitting) as a rough measure. For the simulation cost, the *exact cost model* takes as the cost measure, the total transitions (from one state to another) in the simulation, summing over all replicas. We will adopt the *constant cost model* for simplicity. To justify this, note that for the model to be presented in this chapter, the simulation effort measured by the constant cost and the exact cost differ only by a factor; this factor increases linearly with $n$, the number of levels. A linear factor is considered negligible in our analysis of efficiency, since different splitting degree typically leads to simulation cost that differs exponentially.

Importance splitting may be inefficient under certain circumstances. The reason for the inefficiency is the *inconsistency* in setting up the thresholds, i.e. the existence of “trap phases”. This trouble was indicated in Section 2.6. The problem of inefficiency has challenged the traditional tactics to apply a constant splitting degree, irrespective of the phase. In our study, the splitting degree is allowed to vary with the phase a replica visits when entering the threshold, according to the respective local potential (see Chapter 3
and Chapter 5). We call the corresponding method *adaptive splitting*. Previously used splitting method will be called *non-adaptive splitting*.

The rest of the chapter is organized as follows.

First, I explore and develop the schema of adaptive splitting (Section 6.2). To simplify the investigation, I introduce a two-threshold, multi-phase model, and assume some key characteristics of the process (i.e., the “local potentials”) are known. The effect of the choice of splitting degrees is then studied. I will also show that by adopting the adaptive splitting, one can achieve an enhancement of simulation efficiency; and this enhancement becomes more pronounced as there is an obvious inconsistency in setting up the thresholds. This is supported by a numerical experiment in Section 6.4. Finally, Section 6.5 presents some conclusions.

### 6.1 The Model and The Splitting Strategy

I propose in this section a model to facilitate the study for the efficiency of adaptive splitting. Note that for a full model applying splitting, the analysis of variance and simulation cost of the estimator can become very involved. Hence, a model is introduced that is fairly simple, but still capture the important issues. This is a two-threshold, multi-phase model as in Figure 6.2. We mark the base as level 0 and the target as level $n$. Between the base level and the target level, there is an intermediate level $l$ (chosen as the threshold). Any replica hitting the target must first pass through a state of the

![Figure 6.2: A diagram of two-threshold model](image-url)
intermediate level $l$. Level $l$ includes $m$ phases, labeled from 1 to $m$. We refer to phase 
$i$ of level $l$ as state $(l, i)$, $i = 1, 2, \ldots, m$. The probability of going from the starting state
 to $(l, i)$ is $p_i$. Provided that $(l, i)$ is entered, the conditional probability of reaching the
target in a cycle is $f_i$. This model reminds us of a formulation at the start of Chapter
5. There, the hitting probability is considered, conditional on the enter-$l$ phase. We also
used notation $(p^l_0)_i$ and $(f^n_i)_i$. When both the target $n$ and the threshold $l$ are fixed, all
superscripts and subscripts can be dropped, so I write

$$
\gamma = \sum_{i=1}^{m} p_i f_i. \tag{6.1}
$$

Here, $p_i$’s describes the phase distribution, and $f_i$’s are the local potentials. Clearly,
$0 \leq f_i \leq 1$. By definition, $p_i > 0$, and $0 < \sum_{i=1}^{m} p_i < 1$.

In adaptive splitting, the splitting degrees are allowed to vary according to the phase
visited when entering a threshold. First, $K$ replicas are run from the base level, each
carrying a normal weight of $(1/K)$. During the evolution of each replica, splitting is
applied once the replica reaches the intermediate level $l$, and for any of the replicas, I
assign a splitting degree of $\alpha_i$, when level $l$ is entered at $(l, i)$. In other words, on entering
level $l$ for the first time at state $(l, i)$, a replica is split into $\alpha_i$ replicas, with each split
replica carrying a weight $1/\alpha_i$ of the weight previous to the splitting. Also note that $\alpha_j$
given in the formula above may not be an integer. However, the splitting degree must
be an integer by definition. To avoid this difficulty, we may apply a randomized splitting
degree so that the expected splitting degree of the state considered (averaged over possible
sample paths) is $\alpha_j$. According to [19], this randomization has a negligible contribution
to the variance of the estimator. For simplicity, I will say that the splitting degree $\alpha_j$ is
applied, without mentioning the randomization every time.

Let $\kappa_{ij}$ denote the number of replicas coming from replica $j$ that hits the target, and
on their evolution, level $l$ is entered at phase $i$. With the splitting strategy just described,
we have an estimator of the hitting probability $\hat{\gamma}$ as follows,

$$
\hat{\gamma} = (1/K)[\sum_{j=1}^{K} \sum_{i=1}^{m} (\kappa_{ij}/\alpha_i)]. \tag{6.2}
$$

134
We call $\hat{\gamma}$ as the estimator of adaptive splitting. With $\alpha_i$ varying, we have a family of adaptive-splitting estimators.

Now the question is how the splitting degrees $\alpha_i, 1 \leq i \leq m$ should be chosen for the best efficiency.

The $h$-strategy of choosing the splitting degrees

In our development, the splitting degrees are related to the local potentials, because a larger splitting degree is applied at those phases where the local potentials are greater.

Let us use the following relation as a trial solution:

$$\frac{\alpha_1}{f_1^h} = \frac{\alpha_2}{f_2^h} = \ldots = \frac{\alpha_m}{f_m^h} \quad (6.3)$$

in which $h$ is an exponent parameter, with $-1 < h \leq 1$. From this trial solution, it can be derived that,

$$\alpha_i f_i = \frac{\alpha_i}{f_i^h} f_i^{1+h} = \frac{\alpha_j}{f_j^h} f_i^{1+h}, \quad \forall i, j. \quad (6.4)$$

On the other hand, I enforce the so-called *condition of proper splitting*. In the context of two-threshold model, this means that the number of replicas starting from level 0 should be roughly the same as the number of replicas starting from the level $l$ (when counting the number, all sub-replicas generated after splitting are included). The reason of this is derived from the importance splitting over the model of multi-thresholds. In that scenario, if the number of replicas starting from a level is always less than the replicas starting from one level below, then as the number of levels increases, the number of replicas reaching high levels will become rarer. In the opposite direction, the number of replicas starting from a certain level will grow explosively as the starting level increases, if the number of replicas starting from a level is always more than that number starting from one level below. Both cases will not be efficient. The *condition of proper splitting* has been rigorously established in literature, e.g. see [19].

Now the *condition of proper splitting* in the two-threshold setting can be expressed as
follows,

\[ \sum_i Kp_i \alpha_i f_i = \sum_i Kp_i. \]  

(6.5)

This holds because of the following. First, the number of replicas surviving to state \((l, i)\) is \(Kp_i\). Since the splitting degree is \(\alpha_i\), the number of split replicas continuing from state \((l, i)\) is \(Kp_i \alpha_i\), and of these \((Kp_i \alpha_i f_i)\) replicas survive to the target level. Finally, summing over all states \(i\) of the threshold yields relation (6.5).

Substitute \(\alpha_i f_i\) in equation (6.5) by equation (6.4) to get

\[ \sum_i Kp_i \frac{\alpha_j}{f_j} f_i^{1+h} = \sum_i Kp_i. \]  

(6.6)

Solving for \(\alpha_j\) yields,

\[ \alpha_j = \frac{(\sum_{i=1}^m p_i) f_j^h}{\sum_{i=1}^m p_i f_i^{1+h}}, \quad 1 \leq j \leq m. \]  

(6.7)

To facilitate the discussion, I will use the term \(h\)-strategy. Note two special cases: if \(h = 0\), the splitting degree does not depend on the local potential; this corresponds to the usual (non-adaptive) splitting; if \(h = 1\), the splitting degree is allocated proportionally to the local potentials.

We notice that in stratified sampling [9], the sampling effort allocated is proportional to the variance of a strata. Analogously in adaptive splitting, one may choose the splitting degree proportional to the local potentials. Though the two cases are different, I conjecture that the proportional allocation of splitting degrees in adaptive splitting can be efficient, just as proportional sampling is efficient. To make it rigorous, I will prove this under some reasonable constraints.

### 6.2 Adaptive Splitting: Making It Efficient

Let us fix \(p_i\) and vary \(f_i\), \(1 \leq i \leq m\). Now two issues will be investigated, in which the local potentials \(f_i, 1 \leq i \leq m\) change in different ways. Without loss of generality, we assume throughout that \(f_1 \leq f_2 \leq \ldots \leq f_m\). Firstly, I look at the situation that for every
$i$, $f_i \rightarrow 0$ as the target level $n$ increases, (consider the case that the intermediate level $l$ is fixed and the target $n$ increases, so there are more levels to go from $l$ to $n$). Secondly, I consider the situation that the local potentials vary in a large scale, i.e., $f_m/f_1 \rightarrow \infty$ (note $f_i = \min_i f_i, f_m = \max_i f_i$, according to our earlier assumption). There are many investigations related to the first problem, but here, I attempt to give a solution for the second problem. A relation is set up between the splitting degrees and the local potentials. Using the two-threshold, multi-phase model presented here, I will show that the adaptive splitting is significantly more efficient than non-adaptive splitting.

With the $h$-strategy given, the choice of splitting degrees becomes the choice of the parameter $h$. The main result of this chapter is that $h = 1$ is the best choice, since it is the choice that keeps a uniform bound of relative error (the bound is independent of the distribution of local potentials, as I will show), with much reduced simulation cost when compared with non-adaptive splitting.

### 6.2.1 Variance and simulation cost of the adaptive splitting

To find the efficiency of the “splitting” estimator, we look at both the simulation cost and the variance.

First, let us look at the simulation cost. We will use “work” to refer to the simulation effort, as measured by the total number of replicas simulated. With $K$ initial replicas starting from 0, it is easy to see that

- Total expected work between level 0 and level $l$ is $K(1 - \sum_i p_i)$;
- Total expected work between level $l$ and level $n$ is $\sum_i Kp_i \alpha_i$ (including the replicas generated from splitting)

Hence

$$\text{Expected work} = K(1 - \sum_i p_i) + \sum_i Kp_i \alpha_i. \quad (6.8)$$

Second, we look at the variance of the estimator $\hat{\gamma}$, as defined in (6.2). We use $\hat{\gamma} = (1/K) \sum_j \hat{\gamma}(j)$, where

$$\hat{\gamma}(j) = \kappa_{ij}/\alpha_i, \quad j = 1, 2, \ldots, K \quad (6.9)$$

137
corresponds to an estimate based on the \( j \)-th initial replica starting from the base level 0 (if there is any splitting, we count every hit from the split replicas). We have the following:

**Lemma 6.1** An upper bound for the variance \( V = \text{Var}[\hat{\gamma}] \) is,

\[
V < V^* = \left(1/K\right) \sum_i (p_i f_i / \alpha_i) + \sum_i (p_i f_i^2).
\]

(6.10)

**Proof of Lemma 6.1:**

Let \( \hat{\gamma}' \) be a short notation for \( \hat{\gamma}(\mathcal{U}) \) (since these variables have identical distributions). Let \( M = \text{Var}[\hat{\gamma}]. \) Since the \( K \) initial replicas are identical and independent, from (6.9), we have \( V = \text{Var}[\hat{\gamma}] = (1/K) \text{Var}[\hat{\gamma}'] = (1/K) M. \)

Following, I find an upper bound \( M^* \) for \( M. \) We decompose the variance by conditioning. One has, (see e.g. [51]),

\[
\text{Var}[Z] = E_W[\text{Var}[Z|W]] + \text{Var}_W[E[Z|W]],
\]

(6.11)

in which \( Z \) and \( W \) are any random variables. Add \( E^2[Z] = (E_W[E[Z|W]])^2 > 0 \) to both sides of the formula above, to obtain

\[
E[Z^2] = E_W[\text{Var}[Z|W]] + \{\text{Var}_W[E[Z|W]] + (E_W[E[Z|W]])^2\}
\]

\[
= E_W[\text{Var}[Z|W]] + E_W[E^2[Z|W]].
\]

(6.12)

Now let \( \omega \) be a replica, and \( J_l(\omega) \) be the phase that replica \( \omega \) enters level \( l \) for the first time, and let \( J_l \) be the corresponding random variable. \( J_l \) takes values randomly from \( \{1, 2, \ldots, m\} \), where \( m \) is the number of phases. Replacing \( Z \) by \( \hat{\gamma}' \), and \( W \) with \( J_l \), we calculate,

\[
\text{Var}[\hat{\gamma}'] < E[\hat{\gamma}'^2] = E_{J_l}[\text{Var}[\hat{\gamma}'|J_l]] + E_{A}[E^2[\hat{\gamma}'|J_l]] = M_1 + M_2.
\]

(6.13)

This holds because the split replicas evolve independently, hence simulation of hitting the target from the point of split can be regarded as a Bernoulli trial with the respective
probabilities $f_i$. This immediately leads to,

$$ M_1 = E_{J_i} [\text{Var} [\hat{\gamma} | J_i]] = \left[ \sum_i p_i (f_i (1 - f_i)/\alpha_i) \right] $$

and

$$ M_2 = E_{J_i} [E^2 [\hat{\gamma} | J_i]] = \left[ \sum_i p_i f_i^2 \right]. $$

After dropping the negative term, we get

$$ M = (M_1 + M_2) < \sum_i (p_i f_i/\alpha_i) + \sum_i (p_i f_i^2) = M^*. $$

Note that $V = (1/K)M$, so

$$ V = (1/K) M < (1/K)[\sum_i (p_i f_i/\alpha_i) + \sum_i (p_i f_i^2)] < (1/K)M^* = V^*. $$

This is equation (6.10) in the lemma. □

### 6.2.2 Efficiency of different h-strategies

Note that $V^*/\gamma^2$ is the ratio of variance to the square of the value. The value of $V^*$ depends on how many replicas we have used. To find a measurement of the simulation efficiency, such that the measurement is independent of the number of replicas used, we multiply by $K$. ($KV^*/\gamma^2$) actually measures the square of upper bound of the relative error for the estimator, if running only one replica. Note an upper bound of $KV$ is $KV^* = M^*$, which is given by (6.16), and the metric $\gamma$ is given by (6.1), so the ratio of $KV/\gamma^2$ has the following upper bound,

$$ KV/\gamma^2 < KV^*/\gamma^2 $$

$$ = \frac{\sum_i (p_i f_i/\alpha_i) + \sum_i (p_i f_i^2)}{(\sum_i p_i f_i)^2} = \frac{\sum_i (p_i f_i^2)}{(\sum_i p_i f_i)^2} + \sum_i (p_i f_i/\alpha_i) + 1 $$

$$ = \zeta (p_1, \ldots, p_m, f_1, \ldots, f_m) \left[ \frac{\sum_i (p_i f_i/\alpha_i)}{\sum_i (p_i f_i^2)} + 1 \right], $$

where

\( \zeta \) is a function of $p_i$'s and $f_i$'s, given by

$$ \zeta (p_1, \ldots, p_m, f_1, \ldots, f_m) = \frac{\sum_i (p_i f_i/\alpha_i)}{\sum_i (p_i f_i^2)} + 1. $$

(6.18)
where
\[ \zeta(p_1, \ldots, p_m, f_1, \ldots, f_m) = \frac{\sum_i p_i f_i^2}{(\sum_i p_i)^2}. \] (6.19)

Note the parameter \( h \) does not appear in function \( \zeta(.). \), so \( \zeta \) is the same whatever \( h \)-strategy we adopt.

An important concept here is a uniform bound (w.r.t. \( f_i \)'s) of the relative error, i.e., there is a bound that depends only on \( p_i \)'s, but not \( f_i \)'s. A uniform bound is important, since without it, the estimate becomes very inaccurate when the local potential varies on a large scale. Below, I establish a uniform bound for \( (KV^*/\gamma^2) \). The following lemma establishes a uniform bound for the factor \( \zeta \) of (6.18).

**Lemma 6.2**
\[ \zeta(p_1, \ldots, p_m, f_1, \ldots, f_m) < (1/p_m). \] (6.20)

**Proof:**
\[
\zeta(p_1, \ldots, p_m, f_1, f_2, \ldots, f_m) = \frac{\sum_{i=1}^m p_i (f_i/f_m)^2}{(\sum_{i=1}^m p_i (f_i/f_m))^2} = \frac{\sum_i p_i \xi_i^2}{(\sum_i p_i \xi_i)^2} \quad \text{[use \( \xi_i = f_i/f_m \)}
\]
\[ < \frac{\sum_i p_i \xi_i}{(\sum_i p_i \xi_i)^2} < 1/p_m \quad \text{[if \( i < m \) then \( \xi_i < 1; \xi_m = 1 \)}(6.21)
\]

Hence, \( \zeta(p_1, \ldots, p_m, f_1, \ldots, f_m) \) is bounded uniformly, independently of the amount of \( f_i, 1 \leq i \leq m. \) □

Following, I will prove three basic theorems. Theorem 6.1 shows that in order to keep a uniform bound of the relative error disregarding \( f_i \)'s, one must choose \(-1 < h \leq 1.\)

Theorem 6.2 indicates that the simulation cost, as measured by the number of replicas, decreases as \( h \) increases. Theorem 6.3 shows the savings of the simulation cost when adopting a \( h = 1 \) strategy instead of a \( h = 0 \) strategy.

**Theorem 6.1** Concerning the ratio of \( KV/\gamma^2 \) under different \( h \)-strategies, we have:

1) When \(-1 < h \leq 1\), the ratio \( KV/\gamma^2 \) is bounded uniformly w.r.t. \( f_i \)'s, i.e., there is a bound that depends on \( p_i, 1 \leq i \leq m \) only (independent of any \( f_i \)).
2) When $h < -1$ or $h > 1$, the ratio $KV/\gamma^2$ cannot be bounded uniformly w.r.t. $f_i$’s.

**Proof:**

Since $\zeta(.)$ is uniformly bounded (according to Lemma 6.2), the only factor in (6.18) we still need to bound is $\sum_i \frac{p_if_i}{\alpha_i}$. In this expression, we substitute $\alpha_i$ with its solution (6.7) to get

$$
\frac{\sum_i p_i f_i / \alpha_i}{\sum_i (p_i f_i^2)} = \frac{1}{\sum_i p_i} \frac{(\sum_i p_i f_i^{1-h})(\sum_i p_i f_i^{1+h})}{\sum_i p_i f_i^2}.
$$

(6.22)

Now we consider the two situations of Theorem 6.1.

1) When $-1 < h \leq 1$, we will show

$$
\frac{(\sum_i p_i f_i^{1-h})(\sum_i p_i f_i^{1+h})}{\sum_i p_i f_i^2} < 1.
$$

(6.23)

With this substituted into equation (6.22), we can conclude that

$$
\frac{\sum_i p_i f_i / \alpha_i}{\sum_i (p_i f_i^2)} < \frac{1}{\sum_i p_i}.
$$

(6.24)

The right hand side provides a uniform bound.

I show that (6.23) holds. Consider $(1/z)^h + z^h$, $(z > 0)$ as a function of $h$. It is decreasing at $h < 0$ and is increasing at $h > 0$, so the maximum is reached at either $h = 1$ or $h = -1$. If $z = f_i / f_j$, one has,

$$
(f_i / f_j)^h + (f_j / f_i)^h \leq (f_i / f_j) + (f_j / f_i)
$$

(6.25)

where $-1 < h \leq +1$, $f_1, f_2 > 0$. Multiply the inequality above by the factor $(p_i p_j f_i f_j)$, to obtain

$$
p_i p_j (f_i^{1+h} f_j^{1-h} + f_j^{1+h} f_i^{1-h}) < p_i p_j (f_i^2 + f_j^2).
$$

(6.26)

Taking the sum over all $i < j$ yields

$$
\sum_{i<j} p_i p_j (f_i^{1+h} f_j^{1-h} + f_j^{1+h} f_i^{1-h})
$$
\[
< \sum_{i<j} (p_i p_j f_i^2 + p_j p_i f_j^2) = \sum_i p_i [\sum_{j \neq i} p_j] f_i^2 \\
< \sum_i p_i (1 - p_i) f_i^2.
\] (6.27)

If we add an amount \( \sum_i p_i^2 f_i^2 \) to both sides of the formula above, then we get,

\[
(\sum_{i=1}^m p_i f_i^{1-h}) (\sum_{i=1}^m p_i f_i^{1+h}) < (\sum_{i=1}^m p_i f_i^2).
\] (6.28)

Now dividing by R.H.S. yields equation (6.23).

2) When either \( h < -1 \) or \( h > 1 \), the term \( \frac{(\sum_i p_i f_i^{1-h}) (\sum_i p_i f_i^{1+h})}{(\sum_i p_i f_i^2)^2} \) cannot be bounded uniformly for any given \( f_i, 1 \leq i \leq m \). For simplicity, let \( m = 2 \). We have

\[
\frac{(p_1 f_1^{1-h} + p_2 f_2^{1-h})(p_1 f_1^{1+h} + p_2 f_2^{1+h})}{p_1 f_1^2 + p_2 f_2^2} = \left[ \frac{p_1 f_1^{1-h} p_2 f_2^{1+h}}{p_2 f_2^2} \right] \cdot \left[ (1 + (p_2/p_1)(f_1/f_2)^{h-1})(p_1/p_2)(f_1/f_2)^{h+1} + 1 \right] \\
= \phi_1 \phi_2,
\] (6.29)
in which \( \phi_1, \phi_2 \) denote, respectively, the first and the second fraction enclosed in brackets.

To find examples where \( \phi_1, \phi_2 \) cannot be uniformly bounded, let \( h > 1 \), and suppose \( f_2/f_1 \to \infty \). In the expression above, \( \phi_1 = p_1 f_1^{1-h} f_2^{h-1} = p_1 (f_2/f_1)^{h-1} \); obviously \( \phi_1 \to \infty \) as \( h > 1 \). Meanwhile, the second factor \( \phi_2 \to 1 \). Consequently when \( h > 1 \), \( KV^*/\gamma^2 \to \infty \).

Thus \( KV^*/\gamma^2 \) is not bounded anymore. □

Now the simulation cost is given by (6.8). When adopting the \( h \) strategy, the corresponding simulation cost is denoted as \( \text{Work}[h] \). Now we have,

\textbf{Theorem 6.2} For the two-threshold setting, \( \text{Work}[h] \) decreases as \( h \) increases.

\textbf{Proof:}

Substitute (6.7) into (6.8), and consider \( m = 2 \), we have

\[
\text{Work}[h] = K (1 - p_1 - p_2) + K \left[ p_1 \frac{(p_1 + p_2) f_1^h}{p_1 f_1^{1+h} + p_2 f_2^{1+h}} + p_2 \frac{(p_1 + p_2) f_2^h}{p_1 f_1^{1+h} + p_2 f_2^{1+h}} \right] \\
= K + K (p_1 + p_2) \left[ \frac{p_1 f_1^h + p_2 f_2^h}{p_1 f_1^{1+h} + p_2 f_2^{1+h}} - 1 \right].
\] (6.30)

142
Now for fixed $p_1, p_2, f_1, f_2$, let

\[
\psi[h] = \frac{p_1 f_1^h + p_2 f_2^h}{p_1 f_1^{1+h} + p_2 f_2^{1+h}}.
\] (6.31)

I will show that the derivative $\psi'[h]$ is negative. The calculation of $\psi'[h]$ is straightforward, as follows,

\[
\psi'[h] = \frac{(-1)}{(p_1 f_1^{1+h} + p_2 f_2^{1+h})^2} \sum_{(i,j):i<j} [p_i p_j f_i^h f_j^h (f_i - f_j) (\log f_i - \log f_j)] < 0.
\] (6.32)

The equality holds only when $f_1 = f_2$. Otherwise $\psi[h]$ decreases, and so does Work[$h$].

For $m > 2$, the inequality above can be generalized to,

\[
\psi'[h] = \frac{(-1)}{(\sum_i p_i f_i^{1+h})^2} \sum_{(i,j):i<j} [p_i p_j f_i^h f_j^h (f_i - f_j) (\log f_i - \log f_j)] < 0.
\] (6.33)

With this, the theorem holds generally for $m$ phases, where $m \geq 2$. □

From Theorems 1 and 2, if one wants to preserve the property that the variance ratio of the estimator is bounded uniformly regardless of local potentials, then $h = 1$ (splitting degrees are in proportional to the local potentials) asymptotically corresponds to the minimum effort.

### 6.3 Savings of Simulation Cost and Inconsistency Index in Adaptive Splitting

In this section, I quantify the savings in simulation effort when one adopts $h = 1$ rather than $h = 0$ (i.e. the non-adaptive splitting). Note if the $f_i, 1 \leq i \leq m$, are almost equal, then there may be no saving. Significant savings arise in the situation where there is an obvious inconsistency; i.e. some phases are easily accessed (frequently visited), but once there, it is relatively difficult to hit the target (lower potentials). This relates to the trap phases mentioned earlier (page 47). A choice seems to be $f_m/f_1$, however, even if $f_m/f_1$ is large, if it happens that the phase with lower local potentials is also less visited, the inconsistency needs not be large. For this reason, a better inconsistency index will be
introduced in the next section. Later discussions on the savings of simulation cost are based on this inconsistency index.

6.3.1 Inconsistency index

The inconsistency index that I suggest is:

\[
\varphi = \frac{\left(\sum_i p_i\right)\left(\sum_i f_i\right)}{m \sum_i p_i f_i}.
\]  

(6.34)

To see why \(\varphi\) as defined above is appropriate to reflect the inconsistency of a threshold, note the following obvious facts:

- If the \(p_i\)’s are identical for all \(i\), and the \(f_i\)’s are identical for all \(i\), then \(\varphi = 1/m\), which is a small number. If the \(p_i\) are in proportion to \(f_i\), i.e. \(\frac{p_i}{f_i} = c\) for any \(i\), then \(\varphi < 1\), which is a small number;

- When there is some \(i\) such that \(p_i/f_i, i = 1, 2, \ldots, m\), varies on a large scale, then \(\varphi\) could be quite large. As an extreme case, consider that \(p_1 > 0, p_2 = \ldots = p_m = 0\), and \(f_1 = f_2 = \ldots = f_{m-1} = 0, f_m > 0\), then \(\varphi = +\infty\).

- Consider scaling, e.g. all \(p_i\) (or all \(f_i\)) are multiplied by a constant factor. Then \(\varphi\) remains unchanged.

These are some essential conditions to insure a good inconsistency index, which are satisfied by our proposed \(\varphi\).

The fact that \(\varphi\) involves many variables may cause some difficulties in derivation with formulas. For this purpose, we use \(f_m/f_i\), the ratio of the largest local potential to the smallest local potential. However, \(\varphi\) is preferable for experimental work, due to some of its properties stated above. The following lemma states the relation of \(f_m/f_i\) to the inconsistency index.

**Lemma 6.3** Suppose \(f_i = \min_i f_i, f_m = \max_i f_i\). If the inconsistency index \(\varphi \rightarrow \infty\), then \(f_m/f_i \rightarrow \infty\).
Proof.

Using the definition of $\varphi$, and setting $p'_i = (p_i / \sum_i p_i)$, we have,

$$\varphi = \frac{(\sum_i p_i)(\sum_i f_i)}{m \sum_i p_if_i} = \frac{\sum_i f_i}{m \sum_i p'_i f'_i}.$$  \hfill (6.35)

Note that $f_1 = \min_i f_i, f_m = \max_i f_i$, and that $\sum_i p'_i = \sum_i (p_i / \sum_i p_i) = 1$, we obtain

$$mf_1 \leq \sum_i f_i \leq mf_m$$

$$f_1 \leq \sum_i p'_i f_i \leq f_m.$$  \hfill (6.36)

Using the the two inequalities above and equation (6.34), we get

$$\frac{mf_1}{mf_m} \leq \varphi \leq \frac{mf_m}{mf_1}. \hfill (6.37)$$

From the second half of this inequality, it follows that when $\varphi \to \infty$, then $f_m/f_1 \to \infty$.

In brief, the local potentials must be quite different where there is a large inconsistency. The converse, however, is not necessarily true.

### 6.3.2 Savings of simulation cost

At this point, consider a family of two-threshold models indexed by $\varphi \to \infty$. Within this framework, the savings of simulation cost is studied.

**Theorem 6.3** Consider the two-threshold setting. If the inconsistency index $\varphi \to \infty$, then

$$\lim_{\varphi \to \infty} \frac{\text{Work}[h = 1]}{\text{Work}[h = 0]} = \frac{p_m}{\sum_i p_i} < 1.$$  \hfill (6.38)

The formula above implies a saving in the simulation cost. For simplicity, we prove for $m = 2$ only. The generalization to $m > 2$ is straight-forward.

**Proof**
According to equation (6.30), which defines Work[$h$], and equation (6.31), we have

$$\frac{\text{Work}[h = 1]}{\text{Work}[h = 0]} = \frac{K + K(p_1 + p_2)(\psi[1] - 1)}{K + K(p_1 + p_2)(\psi[0] - 1)}.$$  \hspace{1cm} (6.39)

In the context of a small hitting probability, with many levels between the level $l$ and $n$, we can assume that $f_1 \ll 1, f_2 \ll 1$. In this situation, $\psi[h]$ of equation (6.31) will be sufficiently large so that $\psi[h] \gg 1$. Hence, we should have,

$$\frac{\text{Work}[h = 1]}{\text{Work}[h = 0]} = \frac{\psi[1]}{\psi[0]}(1 + o(1)).$$ \hspace{1cm} (6.40)

Note that $\varphi \to \infty$ implies $f_2/f_1 \to \infty$ (Lemma 6.3) hence $f_1/f_2 \to 0$. Using formula (6.31), with $h = 0$ and $h = 1$, we substitute into the formula above to obtain

$$\lim_{\varphi \to \infty} \frac{\psi[1]}{\psi[0]} = \lim_{\varphi \to \infty} \frac{p_1f_1 + p_2f_2}{p_1f_1^2 + p_2f_2^2} \frac{p_1f_1 + p_2f_2}{p_1 + p_2} = \lim_{\varphi \to \infty} \frac{(p_1f_1 + p_2f_2)^2}{(p_1 + p_2)(p_1f_1^2 + p_2f_2^2)} = \lim_{f_1/f_2 \to 0} \frac{(p_1f_1/f_2 + p_2)^2}{(p_1 + p_2)(p_1f_1^2/f_2^2 + p_2)}$$

$$= \frac{p_2^2}{p_2(p_1 + p_2)} < 1.$$ \hspace{1cm} (6.41)

We now strengthen Theorem 1.

**Theorem 6.4** For the two-threshold model, suppose $f_m = \max_i f_i$. Then for any $-1 < h \leq 1$, and $\varphi \to \infty$ ($\varphi$ index the inconsistency of the local potentials), we have,

$$1 \leq \lim_{\varphi \to \infty} \frac{V[h]}{\gamma^2} \leq \frac{1}{1/(Kp_m)} < 1 + \frac{1}{\sum_i p_i}.$$ \hspace{1cm} (6.42)

where $V[h]$ is the variance of the adaptive splitting estimator, when adopting the $h$-strategy.

This theorem illustrates that, when there is a significant inconsistency of the local potentials, the variance of the estimator does not vary sharply with $h$. From this we conclude that the savings in simulation effort roughly measures the decrease of the work-normalized variance ratio.

**Proof:**
From (6.18), (6.20) and (6.24), we obtain

\[
KV/\gamma^2 \leq \zeta(p_1, \ldots, p_m, f_1, \ldots, f_m)\left(\frac{1}{\sum_i p_i} + 1\right) < (1/p_m)\left(\frac{1}{\sum_i p_i} + 1\right). \quad (6.43)
\]

So,

\[
\frac{KV/\gamma^2}{(1/p_m)} < \frac{1}{\sum_i p_i} + 1. \quad (6.44)
\]

On the other hand, from (6.18) we can get,

\[
KV/\gamma^2 > \zeta(p_1, \ldots, p_m, f_1, \ldots, f_m). \quad (6.45)
\]

Let \( \varphi \to \infty \) (i.e. \( f_1/f_m \to 0 \)) at both sides of the formula above, and note that,

\[
\lim_{\varphi \to \infty} \zeta(p_1, \ldots, p_m, f_1, \ldots, f_m) = 1/p_m. \quad (6.46)
\]

Consequently, \( \lim_{\varphi \to \infty} KV/\gamma^2 \geq 1/(p_m) \), which implies that

\[
\lim_{\varphi \to \infty} \frac{KV/\gamma^2}{(1/p_m)} \geq 1. \quad (6.47)
\]

By combining (6.44) and (6.47), we prove the theorem.

Note that formula (6.47) holds only when \( \varphi \to \infty \), otherwise the lower bound of \( \frac{KV/\gamma^2}{(p_m)} \) can be much smaller. \( \Box \)

### 6.4 A Numerical Experiment

In order to seek evidence of efficiency (other than theoretical results), we want to quantify the enhancement of efficiency when using adaptive splitting as opposed to non-adaptive splitting. To quantify this, we use the speed-up factor, defined as follows,

\[
\text{speed-up factor} = \frac{\text{work-normalized variance using non-adaptive splitting}}{\text{work-normalized variance using adaptive splitting}}. \quad (6.48)
\]
The work-normalized variance for non-adaptive splitting and adaptive splitting has been given previously.

My conjecture is that the speed-up factor tends to increase when the inconsistency index increases. In this section, the conjecture will be verified for the two-threshold model.

![Figure 6.3: The speed-up factors by using adaptive splitting](image)

I constructed a number of examples; in each one I randomly generated some \( p_i, f_i \), and calculated the inconsistency index and the speed-up factor. The results are summarized in Figure 6.3. Let \( x \) be the inconsistency index given by 6.34, and let \( y \) be the speed-up factor. Each point \((x, y)\) in the diagram compares adaptive splitting with non-adaptive splitting. Three references lines \( y(x) = x \), \( y_1(x) = 4x \), and \( y_2(x) = 0.25x \) are also shown in the diagram. One can see that most points are between \( y_1(x) \) and \( y_2(x) \). This result shows that the speed-up factor is roughly proportional to the inconsistency index. This result is very convincing! Note that the inconsistency index spans a wide scale.
6.5 Summary

In this chapter, a novel fast simulation schema called adaptive splitting is proposed. The discussion assumes only two thresholds.

In adaptive splitting, the splitting degree is allowed to vary and is chosen in proportion to the local potentials of the state a replica enters when entering a threshold. In a way, this new schema of splitting is analogous to the proportional sampling in sampling theory. But, the situations in simulation and in sampling are not exactly the same. Hence, a proof of the efficiency for adaptive splitting, like the one given in this chapter, is necessary. The development of an adaptive splitting schema and the proof of its efficiency are the main contributions. I concur with Glasserman & Heidelberger [18], when they state that the efficiency of (non-adaptive) splitting ultimately relies on a detailed understanding of a process’s rare event asymptotics, much as importance sampling does. Only under the right condition can we expect that a (non-adaptive) splitting is asymptotically as effective as the best importance sampling estimator. By the adaptive splitting method, an obvious efficiency gain can be achieved by incorporating a detailed information of rare event asymptotics (i.e. local potentials). Further, adaptive splitting can be extended in principle to multiple threshold adaptive splitting as follows. We can split on each threshold, with splitting degrees in proportion to the local potentials of the state that a replica enters. We consider the vector of local potentials in all phases of a threshold. This vector must converge as the number of levels increases for the level/phase process, we have shown this convergence in Chapter 5.
Chapter 7
Summary of Contributions, Reflections and Remarks

In evaluating system performance, some rare events may be of critical importance. Though being rare, events like system failure, a high error rate in data communication or a very long packet delay in a communication system, can impose severe system penalties. In the recent decade, there were many studies on the rare event simulation methods. This dissertation is a contribution towards this goal. The major contributions are now summarized and possible extensions are discussed.

7.1 The Main Issues

In a large stochastic system, the dynamics leading to the occurrence of rare events are usually intrinsically involved. Analytic algorithms on such problems are often too complicated to be feasible, and simulation suffers from the degradation of efficiency due to the rarity of the events of interest. In order to simulate rare events efficiently, some preliminary information of the system being studied is essential. By incorporating this information, we may have efficient simulation if we find a way to enforce the “most likely path”, when such a path exists.

However, it is not easy to suggest which information we should look for in an efficient simulation. In a way, this decision is determined by two competing factors. If we are looking for too much detailed information, then the effort to obtain this information becomes formidable; on the other hand, if the information lacks sufficient details, then it is difficult to implement an algorithm that ensures efficiency. Hence, the answer to
which information we should look for is not simple at all. From what has been reported in literature, I perceive that this answer is also model dependent, i.e., a given amount of information might be sufficient to ensure efficiency in some models, but not in others. One of my initial motivations was to address this question. Since the existing IS estimators do not always yield satisfactory results, my ultimate purpose is to design better methods to estimate rare event metrics.

My attempt to design good estimators of the rare event metrics has been focused on level/phase processes. In a way, this model arises “naturally”. In the setting I addressed, the rare event happens because of a combination of a large number of normal events (for instance, it is quite normal for a queue length to increase by one, however, it is fairly rare that the queue builds up to a very high level). In this setting, a series of thresholds can be identified in state space, such that the rarity increases when a sample path crosses these thresholds. Also, the sample path must pass through the thresholds one by one. Each threshold, in a sense, can be regarded as a “level”. In multi-dimensional state space, if we are interested in the probability that one of the variables exceeds a certain value, we may regard that variable as the primary variable, which I take as the “level”. I consider this as an advantage, since it is possible to compare the experiment results with the analytic results, which are already available. In particular, we are interested on the asymptotics, which provide the rough information that we are seeking. It deserves attention that the level/phase process is flexible enough to model many features of stochastic processes of practical interest, e.g., multi-dimensional state space (where the phase represents the configuration of all non-primary variables), and phase-type distribution (hence the inter-event time distribution can be more complicated than the exponential distribution). The model consisting of two queues in tandem with Poisson arrivals and exponential servers is presented as an example, but we are looking for fast simulation method applicable in a much broader domain, say any level/phase process. Some of the results should even be applicable to more general discrete event systems.

Another factor that comes to play a role is the “boundary”, i.e., some specified constraints imposed at the boundary of the state space. We emphasize that the boundary has a significant impact on the simulation efficiency, and on the fast simulation method
we choose to ensure efficiency. This is one of the major results in this thesis.

Two kinds of “boundaries” can be identified in level/phase processes. One associates with the level, the other one with the phase. Of course, if either the level or the phase is a queue, then there is a boundary at 0. (In many previous places, I use “boundary” to refer to level boundary.) The level boundary has different behaviors than the phase boundary. Because the phase boundary applies to every level, no matter how high the level is, it will produce a lasting effect to the system large deviation behavior, and it will have a profound influence on rare event metrics. On the other hand, concerning the large deviation behavior, the effect produced by the boundary level might vanish once a replica is far from the boundary. However, we have to be cautious, because in some cases I found that the large deviation behavior at the boundary level is also very important. The condition that I call compliant boundary condition reflects some of my thoughts in this direction. Philosophically, in developing any methods, enough attention should be given to the boundary, since many practical stochastic models allow complicated behaviors at the boundary.

The methods developed in this work have some new features. One prominent feature is that the parameters that I rely on have a clear meaning as being the asymptotics for the process under study. The main features of the proposed methods and other major contributions are summarized in the following section.

7.2 Summary of Contributions

In this dissertation, the main contributions are as follows:

1. I attempted to have an in-depth understanding of the methods reported in literature.
   In particular, I studied extensively the large deviation rate function, which forms the basis of the exponential twist method. This motivated my research and placed my development of efficient simulation approaches in proper context.

2. I studied the rate tilting, which is a specific method of changing event rates in order to speed up simulation of level/phase processes.

152
The conditions for the efficiency were both theoretically justified, and experimentally validated. The verification for the fast simulation method was carefully conducted. This verification was important, because we use the results to see when the simulation is efficient, and when it is not.

The comparison of efficiency with previous methods that change event rates (e.g., the method of Parekh-Walrand, and the method of Kroese-Nicola) was conducted, and the results were reported. The merits and disadvantages associated with each method were indicated.

Under the condition I call compliant boundary, the rate tilting can achieve asymptotic efficiency, in the sense of a bounded relative error. The boundary plays a critical role in deciding whether rate tilting is efficient or not.

3. I introduced the local potentials ($\eta_{(j)}$) and their asymptotics in the level/phase process. This newly developed concept sheds light in understanding the rate tilting. By indicating that $\eta_{(j)}$ is actually asymptotically proportional to the probability of reaching a certain target set, starting from phase $j$ of a certain level, we now understand why the rate tilting works more efficiently than using the exponential twist.

Following the development of this concept, I was able to work out adaptive splitting, which improves the efficiency of (non-adaptive) importance splitting.

The method of adaptive splitting is another promising method. The method is discussed for the two-threshold setting. Details in a more general setting have not been fully developed. However, the result indicated that adaptive splitting leads to a significant improvement on the simulation efficiency compared to non-adaptive splitting.

4. I made use of some asymptotics to quantify the system’s behavior, including both the typical and the large deviation behavior. On one hand, these asymptotics were either used as parameters in rate tilting, or were used in deciding the splitting degrees in adaptive splitting. On the other hand, I did successfully relate these asymptotics
to some eigenvalues and associated eigenvectors of certain matrices. A method to effectively find these eigenvalues and the associated eigenvectors was outlined in a preliminary form.

5. Since there are many importance sampling methods, ranging from the very simple to the very complicated, the one chosen should have the right degree of complexity necessary to enforce the large deviation behavior, i.e., the system behavior, conditional on occurrences of a rare event. The results in this respect are still tentative, and more work is required.

7.3 Outlook

As I discussed novel approaches for fast simulation, several issues that deserve more attention have been already mentioned. Two of them are immediately related to my study. One is to extend the adaptive splitting from two-threshold setting to multi-threshold setting, and the other one is to find effective methods to obtain the asymptotics required for fast simulation.

The asymptotics of the system are important in order to induce an efficient simulation. However, how to obtain these asymptotics is a difficult problem. Note that it is usually not necessary to obtain the exact value of the asymptotics. A rough approximations might also do. (In my thesis, I stated that the efficiency of rate tilting is robust, or insensitive, when the parameter deviates slightly from the exact value.) If a breakthrough can be made in this direction, simulation practitioners would find it easier to apply the methods developed in this thesis.
References


