Importance Sampling Based Internet Traffic Engineering
Under Self-Similar Network Traffic Model

By

Benjamin Zhong Ming Feng
B. Applied Science, University of Waterloo, 2001

A thesis submitted to
The Faculty of Graduate Studies and Research
in partial fulfillment of
the requirements for the degree of

Master of Applied Science in Electrical Engineering

Ottawa-Carleton Institute for Electrical and Computer Engineering
Department of Systems and Computer Engineering

Carleton University
Ottawa, Ontario,
©Copyright
2004, Benjamin Zhong Ming Feng

Library and
Archives Canada

Bibliothèque et
Archives Canada

Published Heritage
Branch

Direction du
Patrimoine de l'édition

395 Wellington Street
Ottawa ON K1A 0N4
Canada

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence

ISBN: 0-612-97406-5

Our file Notre référence

ISBN: 0-612-97406-5

The author has granted a non-exclusive license allowing the Library and Archives Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

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

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

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

Canada

PAGINATION ERROR.

TEXT COMPLETE.

ERREUR DE PAGINATION.

LE TEXTE EST COMPLET.

Abstract

The Importance Sampling based Traffic Engineering (ISTE) technique proposed in this thesis is a brand new approach. Regardless of how large and how complex the network is, as long as the network behaviors of interest can be modeled as stochastic events, and the input of the network can be modeled as a stochastic process, ISTE can predict how the network is going to behave when the network input changes. ISTE will take much less simulation time in comparison to the traditional ad-hoc heuristic approaches which typically require step-by-step simulations or measurements. Our simulation results show that the ISTE approach is fast, simple, and practical for network traffic engineering. Our ISTE approach is much more powerful than all analytical approaches which typically become intractable when more than one node is involved. When applied to more complex scenarios where congestion may be caused by multiple traffic flows, an Alternating Twisting (AT) technique is developed to overcome the increasing noise when twisting single traffic flow.

Acknowledgements

I like to express my deepest appreciation to my thesis supervisors, Professor Changcheng Huang and Professor Mike Devetsikiotis, for their guidance, support, patience, and encouragement.

I also like to thank Yanick Champoux from Alcatel for all his insightful comments and suggestions.

I would like to express my heartfelt gratitude to my parents who has always been there for me. I would not have been here without their invaluable advice, support, and encouragement. Thank you mom and dad, for all the things you have done to provide the best opportunities for me in life and for keeping my spirit up when my future looked uncertain.

Table of Contents

List of Figures	vi
List of Tables	vii
List of Acronyms	viii
1. Introduction.....	1
1.1. Background.....	1
1.2. Motivation.....	2
1.2.1. Network Hot Spot	6
1.2.2. Service Overlay Network.....	8
1.2.3. Service Layer Agreement (SLA) Schemes	10
1.3. Related Work	12
1.3.1. Traditional Traffic Models (Markovian)	13
1.3.2. Self-Similar Traffic Models.....	13
1.3.3. Importance Sampling.....	14
1.4. Contributions.....	15
1.5. Outline.....	16
2. Importance Sampling.....	18
2.1. Importance Sampling Theory	18
2.1.1. Monte Carlo Estimator and Importance Estimator	22
2.1.2. Likelihood Ratio	24
2.1.3. Normalized Variance	28
2.2. Importance Sampling in Rare Network Events Simulation.....	30
2.3. Importance Sampling Based Traffic Engineering.....	33
2.3.1. ISTE Implementation.....	38
2.3.2. Advantages of Importance Sampling Based Traffic Engineering	39
2.3.3. Normalized Variance in ISTE.....	41
3. Simulation.....	43
3.1. Simulation Model.....	44
3.2. Simulation Setup.....	46
3.3. Simulation Results	50
3.3.1. Probability of Overflow	51
3.3.2. Accuracy of the ISTE Prediction	56
3.3.3. Required Simulation Time.....	60
3.3.4. Normalized Variance Results	62
4. Alternating Twisting	65
4.1. Dominating Source Twisting Limitation	65
4.2. Threshold Based Alternating Twisting	72
4.3. Likelihood Ratio in Alternating Twisting.....	75
4.4. Alternating Twisting Simulation Results.....	79
5. Conclusions.....	87
6. Future Work.....	89
Appendix A. Complete Single Queue Simulation Results	90
Appendix B. Complete Tandem Queue Simulation Results.....	92
Appendix C. Complete Traffic Splitting Tandem Queue Simulation Results.....	100
Reference	102

List of Figures

Figure 1 Traffic Engineering Optimization Component Implementation	3
Figure 2 Queueing Theory Limitation	4
Figure 3 Network Hot Spot.....	6
Figure 4 Service Overlay Network	8
Figure 5 Service Overlay Network's Black Box View.....	9
Figure 6 PIPE SLA model	11
Figure 7 HOSE SLA model.....	11
Figure 8 RANGE SLA model.....	12
Figure 9 Probability of Event Occurrence in a Simulated System	20
Figure 10 System Behavior Under Different Input Processes.....	21
Figure 11 Event Occurrence Trace	22
Figure 12 Importance Estimator Versus Monte Carlo Estimator	24
Figure 13 Importance Sampling in Rare Event Simulation Summary	32
Figure 14 Importance Sampling based Traffic Engineering Process.....	35
Figure 15 Normalized Variance in ISTE	42
Figure 16 Network Simulation Topologies.....	45
Figure 17 Simulation Runs and Measurements	47
Figure 18 Input Traffic Process Shift/Twist	49
Figure 19 Single Queue Probability of Overflow Result.....	51
Figure 20 Tandem Queue, end-to-end Probability of Overflow Result.....	52
Figure 21 Tandem Queue, Q1 Probability of Overflow Result.....	53
Figure 22 Tandem Queue, Q2 Probability of Overflow Result.....	53
Figure 23 Split Traffic Tandem Queue, end-to-end Probability of Overflow	54
Figure 24 Split Traffic Tandem Queue, Q1 Probability of Overflow.....	55
Figure 25 Split Traffic Tandem Queue, Q2 Probability of Overflow.....	55
Figure 26 Percentage of Difference for Single Queue Scenario.....	56
Figure 27 Percentage of Difference for End-to-end Performance Prediction, Tandem Queue Scenario	57
Figure 28 Percentage of Difference for Q1 Performance Prediction, Tandem Queue Scenario.....	57
Figure 29 Percentage of Difference for Q2 Performance Prediction, Tandem Queue Scenario.....	58
Figure 30 Percentage of Difference for End-to-End Performance Prediction, Split Tandem Queue Scenario	58
Figure 31 Percentage of Difference for Q1 Performance Prediction, Split Tandem Queue Scenario.....	59
Figure 32 Percentage of Difference for Q2 Performance Prediction, Split Tandem Queue Scenario.....	59
Figure 33 Required Simulation Time Comparison, Single Queue Scenario	60
Figure 34 Required Simulation Time Comparison, Tandem Queue Scenario	61
Figure 35 Required Simulation Time Comparison, Split Traffic Tandem Queue Scenario	61
Figure 36 Normalized Variance for Single Queue Scenario	62
Figure 37 Normalized Variance for Tandem Queue Scenario.....	63

Figure 38 Normalized Variance for Tandem Queue Scenario.....	63
Figure 39 ISTE Limitation Scenario 1 Result, twisting the non-dominating source.....	66
Figure 40 ISTE Limitation Scenario 1 Result, twisting the dominating source.	66
Figure 41 ISTE Limitation Scenario 2 Result, twisting the non-dominating source.....	68
Figure 42 ISTE Limitation Scenario 2 Result, twisting the dominating source	68
Figure 43 ISTE Limitation Scenario 1, Normalized variance.	70
Figure 44 ISTE Limitation Scenario 2, Normalized variance	71
Figure 45 Alternating Twisting.....	72
Figure 46 AT-ISTE algorithm pseudo-code	74
Figure 47 Alternating Twisting Simulation Scenario 1	79
Figure 48 Probability of Overflow Trajectory in Alternating Twisting Simulation Scenario 1.....	80
Figure 49 Alternating Twisting Simulation Scenario 2	81
Figure 50 Probability of Overflow Trajectory in Alternating Twisting Simulation Scenario 2.....	82
Figure 51 Alternating Twisting Simulation Scenario 3	83
Figure 52 Probability of Overflow Trajectory in Alternating Twisting Simulation Scenario 3.....	84
Figure 53 Individual Queue Behavior during AT	85

List of Tables

Table 1 Network Behavior Extrapolation Using ISTE	39
Table 2 Notations for Simulation Result Figures	50

List of Acronyms

FGN:	Fractal Gaussian Noise.
IS:	Importance Sampling
ISTE:	Importance Sampling Based Traffic Engineering
ISTE-AT:	Importance Sampling Based Traffic Engineering Alternating Twisting.
LRD:	Long Range Dependent
QoS:	Quality of Service
SLA:	Service Layer Agreement
SON:	Service Overlay Network
TE:	Traffic Engineering
VPN:	Virtual Private Network

1. Introduction

Traffic engineering (TE) is critical for data networks such as today's Internet, or future MPLS networks. It has been widely studied by the research community [1]. Due to the complexity of Internet traffic and network topology, analytical approaches quickly become intractable. Simulation has been considered as one of the major vehicles for traffic engineering. Unfortunately simulation may introduce significant costs and require unduly long time to obtain meaningful results. This thesis investigates an Importance Sampling based traffic engineering approach which is general enough to address most of the traffic engineering issues and fast enough to obtain useful results in a reasonable time.

1.1. Background

Traffic Engineering is the aspect of network engineering dealing with the issue of performance evaluation and performance optimization of networks [1]. There are many different types of traffic engineering (TE) systems: online or offline; time dependent, state dependent or event dependent; long time scale, medium time scale, or short time scale; Centralized or Distributed; etc. [1]. Regardless the type of the system, one of the primary goals of any traffic engineering mechanism is to optimize the performance of a network. The performance of concern could be the end-to-end performance of an entire network, the performance of a network segment, or the performance of a single node in the network. There is always a target or a requirement specified for the network performance; which implies each traffic engineering system must be composed of at least three components: a measurement component, an analysis component, and an optimization component [2].

The measurement component of the TE system could be used to measure any one of the different aspects of the network performance: bandwidth, delay, delay jitter, utilization, packet loss, buffer overflow, etc. [1]. The analysis component first analyzes the information from the measurement component and then carries out performance evaluation to see if the predefined performance requirement is met. If the analysis component reports that the network performance needs improvement, the optimization component will decide the best course of action to optimize the network performance. Some of the optimization actions could involve changing the internal properties of the network: expanding link capacity, set up new physical/logical path(s), adjust packet schedulers, change dropping policies, etc. [2]. There are also other optimization actions involving making adjustments to the inputs of the network by adjusting the traffic shapers or re-configuring the admission controllers.

This thesis assumes the measurement & analysis component of the TE system is available and effective [3] [4] [5] [6]. This thesis proposes this brand new Importance Sampling based Traffic Engineering (ISTE) approach that can be implemented as part of the optimization component of the TE system. ISTE can provide one feasible solution or one feasible course of action that will optimize the network performance to meet the preset requirement. This thesis also assumes the optimization component is capable of changing the network input, such as using the traffic shapers to change the mean rate of the ingress traffic.

1.2. Motivation

The optimization component decides on the course of action to improve the network performance. The optimization component needs to know or needs to predict how the

adjustments carried out will affect the network behavior [2]. In order to achieve a high utilization of network resources, it is necessary to avoid overshooting the requirement. As a result, it is very useful to know, as accurately as possible, how the network behavior will change for a given change in the network input.

Figure 1 Traffic Engineering Optimization Component Implementation

The optimization component can be implemented using analysis based approaches, such as effective bandwidth analysis [7], queueing theory analysis [8], etc; when the network becomes too complex for the analytical approaches, simulation can be used to overcome this problem. After the simulated network is created, the optimization component can be implemented using heuristic simulation based approaches, such as trial & error [9], gold section search [10], steepest descent/ascent [11], etc. As indicated by Figure 1, the optimization component could always be implemented as the hybrid of those two kinds of approaches.

The analytical approaches can calculate how the network will react to changes in the network input. Thus the optimization component can just make one adjustment to achieve the performance optimization target. Effective Bandwidth analysis [7] is based on Large Deviation Theory [12]. Effective Bandwidth is the service rate/bandwidth that should be allocated to achieve a target probability of packet loss, given the network node's buffer size and the properties of the incoming traffic. There are different formulas

which can be used to calculate the effective bandwidth: Direct Effective Bandwidth [13], Courcoubetis Effective Bandwidth [14], Many Sources Asymptotic Effective Bandwidth [15], On-Off Effective Bandwidth [16], Norros Effective Bandwidth [17], or Gaussian Approximation [18]. Because effective bandwidth calculations can determine how much bandwidth is required to achieve the target performance, it can show how much traffic should be allowed into the network to meet the performance requirement. However, the effective bandwidth calculation result is usually conservative [7] and only applicable to single network node analysis. In order to apply the effective bandwidth approach to a network, people typically assume that one node in the network is congested while all other nodes are not. This assumption is typically not true given the dynamics of network traffic and the complexity of network topology in today's Internet.

“Queueing theory is the study of the waiting times, lengths, and other properties of queues.”[19] Queueing theory models every node in the network as a queue. Depending on the queue's incoming traffic model, service time distribution, number of servers in the queue, etc, a different model is used for each node: M/M/1/, M/G/K, etc. [8]. Each model has its own set of equations for the calculation of the queue length, the waiting time, etc. However, when the network is extremely large and complex, its traffic is long range dependent (LRD), and the queueing theory models and equations become intractable [20] [21]. Long range dependent traffic flows are very bursty traffic flows.

Figure 2 Queueing Theory Limitation

Figure 2 is a simple example to show how queueing theory becomes very complicated once there is more than one queue in the network under a self-similar traffic model. Assume the two queues are connected in series. The traffic arriving in the first queue is a self-similar fractal Gaussian random process. There are buffer overflow occurrences in the first queue. As a result, the traffic streams reaching the second queue will have a different type of distribution. It's still possible to predict the queue behavior, but the whole process becomes very cumbersome.

Simulation techniques are often used to overcome the complexity of a network. The heuristic simulation based approach is usually simpler and more accurate than the analysis based approaches: In the simulated network, the performance of the network is optimized in a step-by-step fashion by the heuristic simulation based approach. In each step an adjustment in the network input is made; then, a simulation run is executed; and a measurement of the new network performance is taken. The analysis component will inform the optimization component if the performance requirement has been met or not. If the performance requirement has not been achieved, more adjustments are made by the optimization component. There exist more intelligent heuristic search based algorithms such as gold section search [10] which makes use of the golden ratio; steepest descent/ascent search which makes use of the gradient concept [11].

The heuristic based approach is simple and accurate. However, it may be a long time before the network performance is finally optimized; the performance requirement is always measured in rare events because rare events in a network are usually faulty events such as buffer overflow, packet loss, etc. When the network performance gets close to the target, the events will become rarer since we usually reduce the occurrence of faulty

events. Consequently, the measurement component will need more and more time to collect enough event occurrences [8] so the analysis component can make an accurate decision on whether the performance target is achieved. This implies a very long waiting period after each adjustment is made by the optimization component.

Due to time and equipment constraint, this thesis has only used the ISTE approach in simulated networks instead of real networks. For the rest of the thesis, all networks are simulated networks unless stated otherwise.

The following scenarios show the practical limitations of the analysis based optimization approach and the heuristic based optimization approach:

1.2.1. Network Hot Spot

Figure 3 Network Hot Spot

There is a hot spot where congestion occurs frequently in a network under self-similar traffic. Research has shown real Internet traffic exhibits self-similar characteristics [22] [23] [24]. The packet loss probability on that network node is higher than desired as shown in Figure 3. The optimization component is capable of temporarily reducing the mean rate of the traffic stream entering the network using the network edge traffic shaper. The question is how much the traffic's mean rate should be reduced to lower the packet loss rate down to the desired level. Meanwhile, to achieve a high utilization of network resources, the optimization process should avoid exceeding the target so that the network bandwidth is not wasted.

The queueing theory analytical approach is not a feasible solution, because the analysis is too complicated due to the long range dependent traffic model, the complex interconnectivity of the network nodes and the different queueing behaviors in each node. The effective bandwidth analytical approach may be a solution, because it can calculate the amount of traffic reduction needed, but the result is always conservative [7]. If the network is translated into a simulated network, a heuristic search based approach can be used. The mean input traffic rate can be first reduced by an amount Δ , followed by the execution of a simulation run. The events of packet loss in the network can be recorded by the measurement component. The analysis component checks how much the probability of loss has been reduced. If the target is still not been met, reduce the mean traffic rate further by Δ or Δ' . The amount of reduction in the next trial depends on the heuristic approach used.

As we reduce mean rate further, the probability of packet loss will also be reduced. Therefore, the event of packet loss becomes rarer. As a result, after each adjustment of

the mean rate, longer and longer waiting time may be required for enough overflow events to occur to obtain an accurate reading on the current probability of loss. It is therefore extremely time consuming.

1.2.2. Service Overlay Network

Figure 4 Service Overlay Network

Service Overlay Network (SON)'s goal is to provide end-to-end QoS to the end users [25]. Because an end-to-end path on the internet usually transverses many different network domains under different administrations, as a result it is very difficult to provide an end-to-end service guarantee. One proposed solution is the service overlay network, which has become a very popular topic in research recently. The Service Overlay Network provider will purchase bandwidth along with a predefined QoS guarantee from each individual network domain, and in turn provide end-to-end QoS to end users.

With the Service Overlay Network architecture, one approach the TE systems in the individual network domain can use to achieve their local end-to-end QoS guarantee is to control the traffic inputs of their network domain. The local end-to-end QoS refers to

the QoS guarantee from the ingress end of the network domain to the egress end of the network domain. The question of “how much adjustment is needed to meet the requirement?” occurs again. This is similar to the “Network Hot Spot” case, but in a network domain under Service Overlay Network, the end to end performance of an entire sub-network is more important than that of the performance of a single node. As mentioned before, the analysis based approaches do not work well when dealing with more than one node. Thus, analytical approaches will become very complex as mentioned in the Figure 2 example.

The Service Overlay Network provider does not know the internal details of each network domain. The individual network domain might be able to provide some statistical information on the local end-to-end performance of the domain or some statistical information on some of the hot spots in the domain, however, the SON provider will not know the interconnectivity of each node in the network domain nor the queueing behavior on each node in the network domain. Thus, each network domain looks like a black box to the SON provider.

Figure 5 Service Overlay Network's Black Box View

Figure 5 shows that the input to the black box is the input traffic of the network domain, and the output of the black box is that network domain's performance. The analysis based approach cannot work under this condition because it requires extensive knowledge of the inside of the network and the approach becomes intractable quickly

under LRD traffic. The heuristic based approach will work in this black box scenario. However, when the performance gets closer and closer to the performance target, each adjustment will result in longer and longer measurement time or simulation time, just like in the “Network Hot Spot” scenario.

[25] provides an analysis based approach for bandwidth provision in SON, but the analysis is based on the $(M / G / \infty)$ model. The analysis makes the assumption that the network is under traditional traffic model (Markovian) and the assumption that the network is a buffer-less network. This analysis is not very practical in a real network since [22] [23] [24] have shown that real internet traffic exhibit characteristics of self-similarity and it is impossible for a network to have an infinite number of packet processing servers.

1.2.3. Service Layer Agreement (SLA) Schemes

Service Layer Agreement is an agreement between the public backbone network owner and its Virtual Private Network (VPN) client. The agreement outlines the quality of service the public network will be providing to its users in terms of bandwidth, delay, delay jitters, latency, etc. There are three important types of service that could be provided to a VPN user under the MPLS VPN context: PIPE, HOSE, and the RANGE model.

In the PIPE model SLA [26], the pipe is a fixed capacity virtual circuit, and the bandwidth is guaranteed at any time.

Figure 6 PIPE SLA model

As shown in Figure 6 , the backbone network provides end to end connections between sites in the same VPN network through the publicly shared backbone network.

Figure 7 HOSE SLA model

In the HOSE model SLA [27], the end-to-end performance guarantee is from one common endpoint to a set of end points (connection from a common point to a “cloud”). As shown in Figure 7, the HOSE model is in contrast to the PIPE model where the guarantee is from one single end point to another single end point.

The agreement specifies the capacity required for the aggregated traffic from the set of end points (the cloud) to the common end point and the capacity required for the traffic from the common end point to the cloud.

Figure 8 RANGE SLA model

As shown in Figure 8, in the RANGE model SLA [28] [29], the VPN customer is provided with a choice of different groups of end-to-end bandwidth guarantees.

Depending on the group the user is in, the VPN service provider will give an assurance on the lower limit of the bandwidth and specify the maximum allowed bandwidth.

All three SLA models, regardless of whether point to point or point to cloud, all require end-to-end performance guarantees, yet, as mentioned before, the existing analytical approaches become intractable or too complex if the incoming traffic is modeled as self-similar random processes.

1.3. Related Work

The following are some previous work done in the same field as this thesis. Section 1.5 will show the contributions of this thesis by illustrating how the Importance Sampling based Traffic Engineering approach has overcome the limitations and shortcomings of the previous work.

1.3.1. Traditional Traffic Models (Markovian)

In [30], Ross presented mathematical tools for the analysis, design, and optimization of multi-service loss networks. Loss networks are networks that make sure there exist sufficient network resources when a call arrives. The target application is the ATM network. This book discusses the Erlang and Monte Carlo approximation of loss probability, routing and admission decision.

Neuts presented the matrix geometric results from advanced queueing theory topics in [31]. The discussion is based on the general properties of a class of Markov chains and random processes that have a matrix geometric invariant probability vector in the positive recurrence case.

Mischa Schwartz introduced a quantitative approach to the analysis of the broadband integrated networks in [32]. The book included performance modeling and performance analysis for ATM networks and an overview of queueing theories and analysis.

As can be seen, extensive work has been done for network performance optimization under traditional traffic models. However, as the network becomes more complex, the analysis, especially the end-to-end analysis, becomes very complicated [9] [30] [31]. Also, because the traffic distributions in real Internet traffic are actually self-similar [22][23][24], those traditional traffic model based approaches have limited application in real data networks.

1.3.2. Self-Similar Traffic Models

In [20], Norros described the parameters of the self-similar Gaussian traffic model, the properties of a queue with self-similar input and short term prediction.

In [21], Duffield and Lewis discussed the implications of self similar models for the problem of estimating loss probabilities in networks. The paper presented the tail asymptotics in a constant service rate queue whose input process has long-range dependence (self-similar) property.

The literature mentioned above is all focused on single queue behavior analysis, and makes the assumption that there are no buffer overflows on the nodes between the traffic source and the queue of interest. In practical traffic engineering scenarios, the end-to-end performance is sometimes of interest and it is very likely that there are buffer overflow occurrences in the upstream network nodes.

1.3.3. Importance Sampling

In [33], Yanick and Pierre used Importance Sampling to estimate packet loss ratio in an ATM switch. The ATM switch is modeled as a queueing network that is fed by several sources emitting fixed size packets (cells) according to a Markov-modulated ON/OFF process.

Huang also applied the Importance Sampling technique in to speed up the simulation for the estimation of the probability of buffer overflow in an ATM network in [34]. Huang used LRD traffic instead of the traditional SRD traffic. Importance Sampling is used in [34] only to speed up the simulation and is not used in traffic engineering to predict the behavior change of the network corresponding to changes in the network input.

1.4. Objectives

The objective of this thesis is to develop a method that can predict the network performance or behavior when the input of the network is change. Thus, it can be used in the optimization component of a TE system to find at least one course of action to meet the performance target. This method should be simple, fast, and applicable to large complex networks. It must remain effective when there are multiple congested node in the network, and it must be applicable in single node performance analysis as well as end-to-end network path performance analysis.

1.5. Contributions

This thesis introduces a brand new concept of Importance Sampling based Traffic Engineering (ISTE). The ISTE approach is a hybrid of the analysis based approach and the heuristic based approach. The ISTE approach is applicable to all networks as long as:

- 1) the network behavior, such as the probability of buffer overflow, the probability of packet loss, etc, can be described by stochastic events.
- 2) The network input, such as the network's input traffic stream, can be modeled as a stochastic process.
- 3) There is the capability to change the network input in terms of changing the attributes, such as the mean and the variance, of the input process.

It is well known that most communication networks can meet these three conditions.

This ISTE approach is faster, simpler, and more universally applicable than the existing traffic engineering approaches mentioned in Section 1.2 and Section 1.3. It can also carry out end-to-end network performance analysis under self-similar traffic model where many of the existing approaches cannot.

The ISTE approach can perform analysis on a network node even if there are buffer overflow occurrences in the node's upstream neighbors. This makes the ISTE approach more practical in real networks than other approaches since most other approaches make the assumption that there is no buffer overflow in the upstream nodes.

In addition to the general ISTE approach, we also propose an Alternating Twisting (AT) technique that can make ISTE more powerful when congestion is caused by more than one traffic stream. AT is an iterative process where each cycle tries to identify a dominating (one source that is primary responsible for the network behavior) traffic flow and then twist it until it is not dominating. Through a limited number of iterations, a pre-defined performance level can be quickly achieved.

Finally, under self-similar traffic model, the simulation results show that the ISTE approach is effective for the simulated network topologies.

1.6. Outline

Because Importance Sampling was originally used to speed up rare event simulations to obtain the simulation results faster, Section 2 will first discuss Importance Sampling's classical application to give the reader more background information. Then, the Importance Sampling (IS) theory which the ISTE approach is based on is explained, and finally, the Importance Sampling based traffic engineering concept is introduced. Section 3 will explain the simulation framework that was used to verify the effectiveness of the Importance Sampling based Traffic Engineering approach and the ISTE approach's advantages over the existing traffic engineering approaches. Section 4 will discuss the "alternating twisting" technique which is used in the ISTE approach when there are multiple traffic flows influencing the network behavior. Section 5 will draw the

conclusions of this thesis. Section 6 will outline some future research topics.

2. Importance Sampling

Important sampling [35] is originally used to speed up rare event simulations. Recall that rare events seldom occur, therefore, a simulation run will require a long measurement period to capture enough event occurrences [8] to obtain an accurate estimate on the probability of event occurrence.

The Importance Sampling technique changes the input of the system to make the rare events occur more frequently [33] [34] [36]. By doing so, the required measurement time is shortened. Importance Sampling tries to find the behavior of the system if the input were not changed and is applied into the system. Therefore, the measured probability of system event occurrences, under the new input scenario, is adjusted using a coefficient, called likelihood ratio, to keep the result un-biased.

2.1. Importance Sampling Theory

Assume:

- a) A is an event in a system. The event can be an instance of buffer overflow on a single network node, or a packet loss in a network.
- b) I_A is an indicator function. When $I_A = 1$, it implies that the event occurred and $I_A = 0$ implies event did not occur.
- c) X is a random process, with a mean value of ν , representing the input of a system; X is called the original process. [34].
- d) X is the only input of the system that is being changed, and other inputs remain the same.

Therefore I_A is a function of X : $I_A(X)$. The expected probability of event A occurring is:

$$E_x\{I_A(X)\}$$

e) $f_X(x)$ is the joint pdf of the original process X , then [37]

$$E_x\{I_A(X)\} = \int I_A(t)f_X(t)dt \quad (1)$$

f) Y is another random process derived from $Y=X+m^*$, where m^* is a constant. Y is called the twisted process. [34]

g) The mean value of the twisted process Y is v_o , where $v_o=v+m^*$.

Adding a constant to a random process can be viewed as shifting the mean of that random process, or changing that random process' distribution. The process of changing the distribution of the original process X to obtain the new twisted process Y is called a change of measure. The change of measure is not limited to shifting the mean of a random process. There are other kinds of change of measure besides shifting the mean of a random process but they are not in the scope of this thesis.

The following are some terms defined in the context of Importance Sampling application in rare event simulation: [34]

- The shifting of the mean value of a random process, by a constant, to make the system events of interest to occur more often is called *twisting*.
- m^* is the “twisted amount”.
- Y is defined as the “twisted” input process of the system.
- X is defined as the “original” input process of the system.

It should be noted that the terms: “twisting”, “original input process”, and “twisted input process”, have a slightly different set of meanings in the context of Importance Sampling based Traffic Engineering. There will be more on this topic in Section 2.3.

Figure 9 Probability of Event Occurrence in a Simulated System

The probability of an event occurrence is usually estimated in the following manner in a simulated system [8]: As shown in Figure 9, a predefined number of independent replications will be executed for each simulation run. The probability of an event occurrence is the number of replications with the event occurrence divided by the total number of replications. If the event occurs rarely, in order to calculate the probability accurately, a large number of replication runs must be executed until a sufficient number of events occur.

If the twisted input process Y makes the event occur more frequently, fewer replications are required to capture sufficient number of event occurrences. This will

reduce the required simulation time. Keep in mind that the probability of an event occurrence, when the system is under the twisted input process Y , is observed, but only as an intermediate step. We are still only interested in “What if the original input process X is applied into the system, what will be the probability of the event occurrence in the system?”

By multiplying and dividing $f_Y(y)$ in (1):

$$E_X \{I_A(x)\} = \int I_A(t) \frac{f_X(t)}{f_Y(t)} f_Y(t) dt \quad (2)$$

where $f_Y(y)$ is the pdf of the twisted input process Y .

$$\text{Define } L(t) = \frac{f_X(t)}{f_Y(t)} \quad (3)$$

$L(t)$ is called the Likelihood ratio because it is a ratio of the distributions of the two random processes.

Substitute the likelihood ratio into (2):

$$E_X \{I_A(x)\} = \int I_A(t) L(t) f_Y(t) dt = E_Y \{I_A(y) L(y)\} \quad (4)$$

An important concept derived from (4):

$$E_X \{I_A(x)\} = E_Y \{I_A(y) L(y)\} \quad (5)$$

Figure 10 System Behavior Under Different Input Processes

Equation (5) provides the fundamental idea for the Importance Sampling theory. Figure 10 shows that under original input process X , the system will “produce” $I_A(X)$;

while under twisted input process Y , the system will “produce” $I_A(Y)$. With the combination of (5) and Figure 10, the following important conclusion is derived:

$I_A(y)$ is the occurrences of event A in the system with Y as the input. A trace of $I_A(y)$ can be easily captured (due to the high occurrence probability). With this captured trace of event A and the likelihood ratio, the expected probability of event A occurrences in the network, with X as the network input, can be found.

2.1.1. Monte Carlo Estimator and Importance Estimator

Figure 11 Event Occurrence Trace

If a trace of the event A occurrences (as seen in Figure 11) in the system can be captured, a point estimator called Monte Carlo [38] [39] estimator can be used to find

$E_X\{I_A(x)\}$:

$$\tilde{P}_X = \frac{\sum_{i=1}^{N_X} I_{Ai}}{N_X} \quad (6)$$

where

- N_X is the total number of replication runs executed when the system is under the original input process X .
- I_{Ai} is the event A occurrence indicator function. On the i -th replication run, I_{Ai} has a value of 1 if the event occurred; it has a value of 0 if the event did not occur.

In the Monte Carlo Estimator, I_{Ai} are the event A occurrences in the system when the original input process X is applied to the system.

According to (5), $E_X\{I_A(x)\}$ can also be determined by calculating $E_Y\{I_A(y)L(y)\}$.

As a result, another point estimator called Importance estimator [35] can also be used:

$$\tilde{P}_{X'} = \frac{\sum_{i=1}^{N_Y} I_{Ai} L_i}{N_Y} \quad (7)$$

Where

- N_Y is the total number of replications executed when the network is under the twisted input process Y .
- I_{Ai} is the event A occurrences in the network if Y is applied to the network.
- L_i is the likelihood ratio to keep the final result un-biased.

In contrast to the Monte Carlo estimator, I_{Ai} are the event A occurrences in the system when the twisted input process Y is applied to the system.

In a way, the two point estimators, Monte Carlo Estimator and Importance Estimator are used to find the sample mean values of two random variables, $I_A(\bullet)$ and $I_A(\bullet)L(\bullet)$, respectively, based on captured sample points I_{Ai} .

Figure 12 Importance Estimator Versus Monte Carlo Estimator

Figure 12 is a graphical comparison between the two estimators. The results from the two estimators will be the same or very close [34] [36] [33], but there are more overflow repetitions in the Importance Estimator scenario. This implies Importance Estimator will require less replications, thus, the simulation time of the Importance Estimator is shorter than that of Monte Carlo Estimator.

2.1.2. Likelihood Ratio

The following concepts regarding likelihood ratio are very important for Importance Sampling's application in rare event simulation and for its application in traffic engineering:

- 1) The Likelihood ratio plays a key role in Importance Sampling because it keeps the Importance Estimator result *un-biased*.
- 2) The likelihood ratio is the ratio of the two input process distribution functions.
- 3) Furthermore, when using “twisting” as the change of measure, the only difference between the distributions of the two input processes is their mean values.
- 4) The two mean values only differ by the twisted amount, therefore, the likelihood ratio can be considered as a function of the twisted amount.

This thesis uses the self-similar Fractal Gaussian Noise (FGN) process as the input process X , because the entire distribution of a Gaussian process can be defined with only its mean and variance. This makes the calculation for the likelihood ratio possible. Also because this thesis is focused on the topic of traffic engineering, and many studies have shown the real internet traffic exhibits self-similarity [22] [23] [24]; and also according to the central limit theorem [40], the aggregated traffic in the network core has Gaussian behavior. Therefore, modeling the original input process as an FGN process is a valid approach.

The equation for the likelihood ratio of FGN input processes can be found in [34]. However, the intermediate steps that derive the final likelihood ratio equation are absent. In the following paragraphs, we will fill in the missing steps in the derivation. In section 4.3, this derivation will be modified as we generalize the approach in [23] for our traffic engineering applications.

Assume the original input process X is a FGN process with a mean value of zero and correlation coefficients of ϕ_{jk} , then the conditional mean of X is [34]:

$$E_X(X_k | x_{k-1}, \dots, x_1) = \sum_{j=2}^k \phi_{kj}(x_{k-j}) \quad (8)$$

Since X 's mean value (v) is 0, according to $v_o = v + m^*$, $v_o = m^*$. Thus:

$$Y = X + m^* = X + v_o,$$

then

$$E_X(X_k | y_{k-1}, \dots, y_1) = \sum_{j=2}^k \phi_{kj}(y_{k-j}) = \sum_{j=2}^k \phi_{kj}(x_{k-j} + v_o) \quad (9)$$

Furthermore:

$$\begin{aligned} E_Y(Y_k | y_{k-1}, \dots, y_1) \\ &= (+v_o) + E_Y(Y_k - v_o | y_{k-1} - v_o, \dots, y_1 - v_o) \\ &= v_o + E_X(X_k | x_{k-1}, \dots, x_1) \\ &= v_o + \sum_{j=2}^k \phi_{kj}(x_{k-j}) \end{aligned} \quad (10)$$

The conditional variance of X and Y is the same since Y is the linearly shifted version of X [34]:

$$Var = \frac{1}{\sigma^2 \prod_{j=2}^k (1 - \phi_{jj}^2)} \quad (11)$$

Since the likelihood ratio is used in $E_Y\{\bullet\}$ to calculate $E_Y\{I_A(y)L(y)\}$, therefore

$$\begin{aligned} L(t) &= \frac{f_X(t)}{f_Y(t)} = \prod_{j=2}^t \frac{\exp(-(y_j - E_X(X_j | y_{j-1}, y_{j-2}, \dots, y_1))^2 / 2 \text{var}_X(Y_j | y_{j-1}, y_{j-2}, \dots, y_1))}{\exp(-(y_j - E_Y(Y_j | y_{j-1}, y_{j-2}, \dots, y_1))^2 / 2 \text{var}_Y(Y_j | y_{j-1}, y_{j-2}, \dots, y_1))} \\ &= \prod_{j=1}^t L_j \end{aligned} \quad (12)$$

Let the notion $E_X = E_X(X_j | y_{j-1}, y_{j-2}, \dots, y_1)$ and let the notation of

$E_Y = E_Y(Y_j | y_{j-1}, y_{j-2}, \dots, y_1)$ to increase readability of the rest of the derivation.

(12) becomes:

$$L_j = \frac{\exp(-(y_j - E_X)^2 / 2 \text{var})}{\exp(-(y_j - E_Y)^2 / 2 \text{var})} \quad (13)$$

$$= \exp((- (x_j - E_X)^2 / 2 \text{var}) - (- (y_j - E_Y)^2 / 2 \text{var}))$$

$$= \exp((- y_j^2 + 2 y_j E_X - E_X^2 + y_j^2 - 2 y_j E_Y + E_Y^2) / 2 \text{var}))$$

$$= \exp\left(\frac{2 y_j E_X - E_X^2 - 2 y_j E_Y + E_Y^2}{2 \text{var}}\right)$$

$$= \exp\left(\frac{2 y_j (E_X - E_Y) - (E_X^2 - E_Y^2)}{2 \text{var}}\right)$$

$$= \exp\left(\frac{2 y_j (E_X - E_Y) - (E_X + E_Y)(E_X - E_Y)}{2 \text{var}}\right)$$

$$= \exp\left(\frac{y_j (E_X - E_Y)}{\text{var}} - \frac{(E_X + E_Y)(E_X - E_Y)}{2 \text{var}}\right)$$

$$\text{Let } \theta = \frac{(E_X - E_Y)}{\text{var}} \quad M = \exp\left(\frac{(E_X + E_Y)\theta}{2}\right) \quad (14)$$

(13) becomes:

$$L_j = \frac{\exp(y_j \theta)}{M} \quad (15)$$

Substitute (9) and (10) into (14), then

$$\theta = \frac{(E_X - E_Y)}{\text{var}} = \frac{v_0 (\sum_{j=2}^k \phi_{kj} - 1)}{\sigma^2 \prod_{j=2}^k (1 - \phi_{jj}^2)} \quad M = \exp\left(\frac{(2 \sum_{j=2}^k \phi_{kj} x_{k-j} + v_0 (1 + \sum_{j=2}^k \phi_{kj}))\theta}{2}\right) \quad (16)$$

From equation (16), different values of v_o will result in different values of the likelihood ratio. This property is crucial to Importance Sampling-based Traffic Engineering which will be discussed later. It is also important to note that there is a x_{k-j} term in (16) which implies a trace of the original input process X is required. However, the actual input of the network is the twisted input process Y , therefore all the traces captured are for the process Y . The trace of the input process X cannot be captured directly, but it can be extracted from the trace of the process Y based on the numerical relation of $Y=X+v_o$.

Equation (16) shows that some of the attributes of the input process, such as the correlation coefficient, are needed to calculate the likelihood ratio. In real-life systems, those attributes have to be estimated from the system input measurements. However, because in the simulation framework of this thesis, the simulation program can access the FGN process generator; therefore, those attributes will be directly retrieved from the generator during the simulation runs to calculate the likelihood ratio.

2.1.3. Normalized Variance

The point estimators, the Monte Carlo Estimator and the Importance Estimator, are used to find the sample mean of two random variables based on the captured sample points. The variance for the sample mean from the Monte Carlo Estimator is [8]

$$\sigma^2 = \left(\sum_{i=1}^N (I_{A_i} - \tilde{P}_X)^2 \right) / N / (N - 1) \quad (17)$$

and its normalized variance is defined as [34]

$$\frac{\sigma^2}{\tilde{P}_X} \quad (18)$$

Similarly for sample mean from Importance Estimator, its variance is defined as [8]

$$\sigma^2 = \sum_{i=1}^N (I_{A_i} L_i - \tilde{P}_X')^2 / N / (N - 1) \quad (19)$$

and its normalized variance is defined as [34]

$$\frac{\sigma^2}{\tilde{P}_X'} \quad (20)$$

A large normalized variance of the Importance Estimator or the Monte Carlo Estimator implies a great degree of fluctuations in the sample mean value, which indicates the result from the point estimator is “noisy” and unreliable. From the simulation results in Section 3.3 and from [34], normalized variance usually increases as the value of the twisted amount increases. Intuitively, this means that if the twisted process Y is “too different” from the original process X , the Importance Estimator result which is based on the traces captured from the network under the input process Y , is not accurate anymore. “Too different” means the difference between the mean value of the twisted input process Y and mean value of the original input process X is very large.

It is desirable to minimize the variance. Many variance reduction techniques [35] use different kinds of “change of measure” to introduce correlation (dependencies) in the sample data to reduce the variance. Importance Sampling is one type of variance reduction technique. The two processes X and Y are correlated since Y is just a linearly shifted version of X . There are other kinds of “change of measure”, besides twisting, that can be used to reduce the variance, but they are not in the scope of this thesis.

The threshold value of the normalized variance, for the sample mean to be acceptable, is very subjective. Based on heuristic results (Section 3.3 and from [34]), this thesis sets the acceptable normalized variance threshold of 0.1 in some cases. Normalized Variance will also play an important role in the “alternating twisting” method in Section 4.

2.2. Importance Sampling in Rare Network Events Simulation

If the system under simulation is a network, then,

- i. The input traffic of the network can be treated as the system input process X .
- ii. The event of buffer overflow is treated as the system event of interest (treated as event A).
- iii. The probability of buffer overflow can be treated as the system behavior.

Because X is used to model the incoming traffic flow to a network, shifting its mean positively (creating the twisted traffic input process Y) is the most intuitive way of increasing the mean rate of the input traffic, which indicates more traffic will enter the network. By applying the twisted input traffic process Y into the network, the events of buffer overflow will occur more often. Therefore, the simulation time for capturing event A 's occurrences in the network with Y as the network input traffic process is much shorter than the simulation time for capturing event A 's occurrence in the network with X as the network input traffic process. According to equation (5), if there are captured traces of overflow events from the network when the network is under the twisted input traffic process Y , with the help of the likelihood ratio, the probability of buffer overflow

in the network under the original input traffic process X can be determined. Therefore, Importance Sampling helps shortening the simulation time of rare event simulations.

Because the likelihood ratio only depends on the input traffic process distributions, and because the overflow occurrence indicator function I_{Ai} could be easily measured from the network without the knowledge of the internal details of the network, such as the network internal interconnectivity and the queueing behaviors of network nodes, therefore:

- 1) Importance Sampling is applicable in any kind of network.
- 2) Importance Sampling is applicable to network node events or end-to-end network events since I_{Ai} can represent event occurrences on a single network node as well as end-to-end events occurrences belonging to an entire network.

Section 2.3 will show that two features are very important in Importance Sampling's application in traffic engineering.

Even though this thesis uses network incoming traffic stream as the system input and probability of buffer overflow as the system behavior, other network inputs and network behavior can also be used, as long as the behavior can be described using stochastic events and the network input can be modeled as a stochastic process.

Figure 13 Importance Sampling in Rare Event Simulation Summary

Figure 13 summarizes the Importance Sampling operation. The twisted input traffic process (Y) is applied into the network instead of the original input traffic process (X). *The original input process is the initial on-hand process that is supposed to be applied to the network but the event occurrences due to initial input process will be rare.* The twisted input is the result of the linear shifting of the original/current input. The twisted input is applied to the network to make network events occur more frequently.

The overflow occurrence trace and the twisted input traffic trace are recorded from the network. The captured traffic trace and the twisted amount are then used to calculate the likelihood ratio (using equation 16). The measured probability of event is adjusted (according to equation 7) using the likelihood ratio to find the probability of buffer overflow if the network was under the original/current input traffic stream.

One way to interpret the application of Importance Sampling in speeding up rare event simulation is that we start with a known “original” input process X with a fixed mean value of v . We try to find an input process Y with mean of v_0 that will make

overflow events occur more often. Importance Sampling based Traffic Engineering is a “reverse” of this concept.

2.3. Importance Sampling Based Traffic Engineering

Everything so far in Section 2 has been about the existing application of Importance Sampling in simulation. This section focuses on the new idea of Importance Sampling based Traffic Engineering (ISTE). The concept of Importance Sampling in Traffic Engineering can be viewed as the “reverse” of the IS concept in its classical application in rare events simulation: the twisted input process Y with mean of ν_o is known and therefore ν_o is fixed. The mean (ν) of the original input process X is changed. In the previous section, ν is known and fixed while ν_o is changed. We try to find the original input process X such that when X is applied into the network, the probability of overflow in the network meets a predefined requirement. Since Y is just a linearly shifted or twisted version of X , instead of fix ν and adjust ν_o , like in the classical application of Importance Sampling, we now fix ν_o and change ν .

In the ISTE approach, the twisted input (Y) is applied into the network. The “twisted” input process is modeling the network input that is currently being applied to the network. The “original” input process (X) is unknown, or more precisely, the mean value (ν) of the “original” input process is unknown. It is assumed that the mean value of some unknown “original” process has been shifted and the “twisted” process (the current network input process) is the result. The distributions of the “original” input process and the “twisted” input process are different only in their means values. The difference between the two mean values is the twisted amount. By using different values for the

twisted amount (m^*), a number of “original” input processes, with different mean values (v), can be created. This thesis will show that ISTE can, correspondingly, find the probabilities of event occurrences under those different “original” input processes. Reducing the mean value (v_o) of the current input process will transform the current input one of those “original” input process; this implies that *ISTE can predict the network behavior, in terms of probability of event occurrences, when the network input changes*. All the network inputs mentioned in this thesis can be modeled as a stochastic process, and when “the network input changes”, it is referring to the change of the input process’ statistical parameters, such as its mean value. It is important to note that the terms “original input” and “twisted input” all have new meanings now, in the context of Importance Sampling based Traffic Engineering. For the rest of the thesis, “original input” and “twisted input” are referenced in the context of Importance Sampling based Traffic Engineering.

In the case of optimizing the network performance to meet a predefined target, ISTE checks which of the created “original” input processes, or more precisely, which mean value (v), can be applied into the network so the network performance can meet the performance target. Once that “original” process is found, the current network input mean value must be reduced to that of the “original” process for the network performance to satisfy the requirement.

Figure 14 Importance Sampling based Traffic Engineering Process

Figure 14 is a graphical view of the ISTE approach. The incoming traffic stream to the network is treated as the network input, and the event of buffer overflow is the network event of interest and it is used to describe the network behavior. The ISTE approach starts with the capturing of a trace of the current network input traffic flow and a trace of buffer overflow occurrences currently in the network. The number of replication runs used in the simulation run is also recorded. Equation (7) indicates that

$$\tilde{P}_X' = \frac{\sum_{i=1}^{N_Y} I_{A_i} L_i}{N_Y} \text{ Where}$$

- N_Y is the total number of replications executed when the network is under the twisted input traffic process Y .
- I_{A_i} is the overflow event occurrences in the network if the twisted input process Y is applied to the network.

- L_i is the likelihood ratio to keep the final result un-biased.

The “twisted” input process models the traffic flow currently being applied to the network. Therefore, I_{Ai} is the overflow occurrence trace that was captured at the beginning of the ISTE process. As a result, the term I_{Ai} in equation (7) is fixed. The N_Y term in (7) is also captured and therefore it is fixed. The only term remaining in (7) that is not fixed is the likelihood ratio L_i .

A trace of the “original” input traffic process and the twisted value (m^*) is required to calculate the likelihood ratio according to (16). The “original” input process is unknown, thus, a trace of the “original” input process does not exist. Since $Y=X+m^*$, a trace of the “original” input traffic (X) can be extracted by subtracting the twisted amount (m^*) from the captured trace of the “twisted” input traffic process (Y). Because the “twisted” traffic process’s trace is captured and is fixed, therefore, the trace of the extracted “original” traffic process can be considered to be a function of the twisted amount (m^*). Furthermore, the likelihood ratio can now be considered as a function of the twisted amount. Since the likelihood ratio is the only variable term in (7), the probability of buffer overflow in the network, when some unknown “original” input process is applied, is a function of the twisted amount.

The conclusion is that the probability of buffer overflow, when the “original” input traffic process is applied to the network, can be made into anything desired by setting the twisted amount to the appropriate value. No more simulation is required since everything needed to solve equation (7) have been collected or has been set.

Since $v_o = v + m^*$, when using different values of twisted amount (m^*), the mean rate (v) of the “original” traffic process will take on different values. As a result, $\tilde{P}_X$ in equation (7) represents the probability of buffer overflow when the network is under different “original” network traffic processes (X), each with different mean values (v). By reducing the current input traffic (Y)’s mean rate to match the mean rate of any one of those “original” input processes, the current input process *becomes* that specific “original” input process since the two processes only differ by their mean values. Therefore, (7) can predict the network behavior, in terms of the probability of buffer overflow, when the mean rate of the network input traffic changes. As stated in Section 1.2, the optimization component of any traffic engineering systems needs the capability to predict, as precisely as possible, how the network is going to behave under changing network input. Because the ISTE approach can provide this capability, therefore, ISTE can be used to implement the optimization component of any traffic engineering system.

Just like Importance Sampling’s application in speeding up rare event simulation, the ISTE process can be used with other network inputs (besides the network’s incoming traffic flow) and other network behavior (besides the probability of buffer overflow), as long as the particular network behavior can be described using stochastic events and the particular network input can be modeled as a stochastic process.

The process of calculating the probability of event occurrence, after the network input mean value has been reduced by a specific twisted amount, is called twisting. Twisting, in the context of Importance Sampling’s application in rare event simulation, is the shifting of the mean value of the input random process by a constant (Twisted Amount) in order to make the event of interest to occur more often. For the rest of the

thesis, twisting is referenced in the context of Importance Sampling based Traffic Engineering. For a given network block, twisting predicts the behavior or performance of the network block as the input to the network block is changed.

In summary, the Importance Sampling based Traffic Engineering approach works because the Importance Sampling theorems outlined in Section 2.1 are still applicable in ISTE. As long as the ISTE process applies the “twisted” input into the network to calculate the probability of event occurrence if the “original” input was applied to the network, equation (5) guarantees $E_X \{I_A(x)\} = E_Y \{I_A(y)L(y)\}$, where X is the “original” input process and Y is the “twisted” input process.

2.3.1. ISTE Implementation

The ISTE process can be implemented by implementing equation (7) and equation (16) in hardware or software. The operations involved in equation (7) and (16) is not too complex, thus, the implementation process should be straightforward.

However, these two equations can not explicitly find the exact value for the twisted amount (m^*) if given a target network performance. The ISTE analysis in this thesis is implemented in such way that different amount of reduction in the mean rate of the current input traffic process (different values of the twisted amount) is used to calculate different probabilities of occurrences, until one of these reduction amount brings the probability to, or close to, the desired level.

Amount of reduction	Log(Probability of event Occurrence)
0	-1.1
0.1	-2.3
0.2	-3.9
0.3	-4.1
....	

Table 1 Network Behavior Extrapolation Using ISTE

ISTE needs to capture a trace of the event occurrences currently in the network together with a trace of the current input process to the network. After the initial simulation run, each mean value reduction trial is pure off-line calculations (7) (16) which requires no additional simulation run.

In a real network, assume traffic shapers are stationed on the edge of the network, the ISTE approach can act as the feedback mechanism to shape the ingress traffic in order to improve the network performance. ISTE can observe the current network behavior and predict how the mean rate of the incoming traffic should be shaped in order to meet the target. This information is then used to control the traffic shapers at the edge of the network.

2.3.2. Advantages of Importance Sampling Based Traffic Engineering

Some of the advantages of the ISTE approach:

1) From equation (7): $\tilde{P}_X' = \frac{\sum_{i=1}^{N_Y} I_{A_i} L_i}{N_Y}$. None of the terms in equation (7) depends on the

internal details of the network, such as the network nodes' queueing behavior, the network's interconnectivity, etc. Every term is either captured (N_Y, I_{A_i}), calculated from captured traces (L_i), or variable set by the user (twisted amount). The traces are either observable network behavior or external inputs to the network, all of which can be

captured directly without worrying about the internal details and internal activities of the network. This allows ISTE to be applied to large complex networks where queueing theory calculations become extremely cumbersome.

2) Because the ISTE analysis does not care about the internal details of the network, it can still work even if there are buffer overflow occurrences in the upstream neighbor nodes. Effective Bandwidth based analysis and queueing theory analysis all have to assume there is no buffer overflow in the upstream network node.

3) The term I_{Ai} can represent a network event on a network node or an end-to-end event of a network; thus, ISTE can be used in end-to-end traffic engineering which effective bandwidth calculations cannot be used.

4) As mentioned at the end of Section 2.3.1, ISTE only needs one simulation run to capture the current network input process's trace and the event occurrences trace of events currently happening in the network; yet, in each step of the heuristic search based approach, one simulation runs is required after each network input adjustments . Furthermore, events of interest in the current network always occur more frequently than they will be after the network performance gets close to the performance target. The heuristic based approach will require longer and longer measurement time after each adjustment as it gets closer and closer to the target, because event occurrences becomes rarer and rarer. For the ISTE approach, the simulation run is used to capture the event occurrences currently in the network, which means ISTE does not require a long measurement/trace capturing period.

5) As shown by the simulation results in Section 3.3, the ISTE approach works under Self-Similar traffic model.

2.3.3. Normalized Variance in ISTE

For each probability of overflow predicted, we can calculate the normalized variance for that predicted probability. Looking at equation (19):

$$\sigma^2 = \sum_{i=1}^N (I_{A_i} L_i - \tilde{P}_X')^2 / N / (N - 1)$$

$\tilde{P}_X'$ is the predicted probability of overflow when the network is under some specific “original” input traffic process. I_{A_i} is the captured overflow event trace when the network is under the “twisted” input traffic process. L_i is the likelihood ratio used to calculate $\tilde{P}_X'$. N is the number of replication runs used on the network when the “twisted” traffic is applied. Since all the terms are known in (19), σ^2 can be calculated. The normalized variance can then be calculated using equation (20):

$$\frac{\sigma^2}{\tilde{P}_X'}$$

Remember that the normalized variance is an indication of how reliable or how “noisy” the estimated or predicted result is. As the simulation result in Section 3.3 will show, as the twisted amount (m^*) gets larger, the normalized variance of the prediction also gets bigger. This implies the prediction becomes noisier as the twisted amount gets bigger. This is intuitively understandable, as the “original” traffic process under prediction gets further and further away from the current traffic input (twisted), the accuracy of the prediction will get worse. “Further away” implies the difference between the mean rate of the original traffic process and the twisted traffic process is large.

Therefore, the further the current network input traffic is twisted, the worse the prediction result will get.

Figure 15 Normalized Variance in ISTE

If the “original” traffic process of interest is “too far away” to be predicted accurately, the mean rate of the “twisted” or the current traffic process can be changed to be closer to the mean rate of the “original” process of interest, then, a new simulation run can be executed to recapture the new traces of the overflow event and new input traffic. The twisting process starts again from that point, as shown in Figure 15. The normalized variance will, again, start from a small value and gradually increase as twisting continues. Hopefully, the “original” input traffic process of interest can be reached before the normalized variance becomes too big again. This technique will be very important in the “Alternating Twisting” method in Section 4.

3. Simulation

The advantages of ISTE over the analysis based approach have been described in the previous sections. The purpose of the simulation framework presented in this section is to verify that the network behavior predicted by the ISTE analysis actually matches the measured network behavior.

When a heuristic simulation based approach is used to optimize a network's performance, after each adjustment, a simulation run or a measurement run is required to collect sample points (buffer overflow event occurrences) to find the new network performance. The Monte Carlo Estimator is the most commonly used estimator to measure the network performance. As a result, in a simulated network, after each adjustment the Heuristic based approach makes, a Monte Carlo simulation run is needed to verify the new network performance.

For simplicity, in the comparison of the simulation time from the ISTE approach and that of the Heuristic based approach, the optimal case of the Heuristic approach is used. In the optimal case, the target performance requirement is achieved in the first trial. This optimal case is rare in the real Heuristic based searches: any search usually needs to make several adjustments or trials to achieve the target. Therefore, if it can be proven that the even though the results from the two approaches are similar, yet the required simulation time of the ISTE approach is shorter than even that of the optimal case of the Heuristic based approach, the advantage of the ISTE approach over the Heuristic based approach can be clearly demonstrated.

The simulation strategy in this thesis is to use the ISTE analysis to predict the network performances under different network input scenarios; then, one Monte Carlo simulation run is executed to verify each prediction. The reason that only one Monte Carlo simulation run is used to verify each prediction is that: the heuristic search usually involves several Monte Carlo simulation runs: one simulation runs after each adjustment to check if the goal is achieved. However, in the optimal case, the target is reached in the first adjustment; therefore, only one Monte Carlo simulation run is required. Another part of the simulation strategy is that in the cases that there are more than one queue in the network, the simulation framework make sure that there are always overflows in all the queues in the network, this is to verify that the ISTE analysis still works when there are overflow events in upstream network nodes. The Monte Carol simulation results are treated as benchmark results representing the actual performance measured from the network.

3.1. Simulation Model

In the simulation framework, three basic network topologies are used. Since all complex networks are combinations of these three topologies, the simulation results and conclusions with these basic network topologies are applicable to all networks.

Figure 16 Network Simulation Topologies

The Single Queue scenarios are simple cases to do a “sanity” check on the ISTE approach to make sure it works. The Tandem Queue scenarios are used to verify that the ISTE approach works in end-to-end performance analysis and in single node analysis if there are buffer overflow in the upstream queue. The Split traffic Tandem Queue introduces multiple traffic sources, traffic stream split cases, and traffic stream merge cases to make sure the ISTE approach is effective in complex networks.

The ISTE approach does not care about the queueing behavior of the network nodes, except in the sense of what “causes” important events, but for the purpose of simplicity, all the network queues in the simulations are modeled as FIFO. The input traffic streams of the network are modeled as Fractal Gaussian Noise Self Similar processes because the aggregated traffic flows in network core usually exhibit Gaussian property [40], and real internet traffic exhibits strong characteristics of self-similarity [22] [23] [24]. In all simulation scenarios, all traffic sources in the network send traffic to the same destination. In the scenarios based on the Split Traffic Tandem Queue topology in Figure 12, Source B is used to model the background traffic stream and Source A is

treated as the primary traffic source for the network. There are some special Split Traffic Tandem Queue topology scenarios in which the behavior of the network is influenced mainly by the background traffic source instead of the primary traffic source.

The Lindley equation, $Q_k = (Q_{k-1} + X_k - \mu)^+$ [34], is used to model the behavior of queues in the network. X_k is the amount of traffic arrived at k -th time unit (the time unit could be in seconds, milliseconds, days, or month), and μ is the service rate of the queue. The unit of simulation time is seconds, thus the mean value of the input traffic process is the amount of traffic per second; if a queue size exceeds its buffer size at the stop time of a replication run, that replication run is considered to have had an event occurrence. The event indicator function for that replication is set at a value of 1.

In summary, the simulation used in this thesis is discrete time simulation.

3.2. Simulation Setup

The ISTE approach can be used with any network input which can be modeled as a stochastic process to extrapolate any network behavior that can be modeled as stochastic events. In the simulation setup, the network input is the incoming traffic flow to the network. The network behavior to be optimized is the probability of buffer overflow of the end-to-end network, or the probability of buffer overflow on a single network node. Probability of buffer overflow in a simulation run is defined as the ratio of the number of replication runs with the buffer overflow divided by the total number of replication runs.

For each basic network topology, several scenarios are created each with different buffer sizes, service rates, input traffic process Hurst parameters, traffic split ratios, etc. Many analysis based approaches make the assumption of no overflow in the upstream nodes. In this thesis' simulation, for the scenarios with more than one queue, the

upstream queue always has some buffer overflow occurrences, which makes the scenarios more realistic. This is different from all other Importance Sampling research work which always assume only one queue in a network is congested. For the topologies with multiple queues, some scenarios are configured to have one queue with more overflows while other scenarios will make sure that the other queue has more overflows. This is to make sure, when doing end-to-end ISTE analysis, the result is not biased toward any one of the queues.

Figure 17 Simulation Runs and Measurements

Figure 17 shows the different scenarios, different simulation runs, and different measurements in our simulation setups. Each scenario has two simulation runs and each

simulation run has many independent replications. The number of replications is set from 1000 to 60,000,000 in different simulation runs depending on the overflow occurrence frequency.

As mentioned in Section 3.2, the simulation for each scenario is composed of two simulation runs. The first simulation run is the ISTE simulation run. The ISTE simulation run is used to capture the current input traffic trace along with a trace of the event occurrences currently in the network. It does not physically change the network input to optimize the network performance, instead, the ISTE analysis will calculate the change needed to be made in the network input to optimize the network performance. As mentioned in Section 3.2, the second simulation run is the Monte Carlo simulation run. The Monte Carlo simulation run verifies that the actual network's performance, after making those input changes, matches the performance predictions made by the ISTE analysis.

The Monte Carlo simulation is also treated as the optimal case of the heuristic search based approach where the heuristic search is able to achieve the target in one adjustment. Thus, comparing the simulation time of ISTE and that of the Monte Carlo Simulation is the same as comparing the simulation time of ISTE and that of the optimal heuristic search case.

In different simulation runs, different measurements are taken. Measurements include the first queue overflow probability, the second queue overflow probability if it exists, and the end-to-end overflow probability. The end-to-end overflow event occurs when either an overflow event occurs in Q1 or an overflow event occurs in Q2. Since the

normalized variance is an indicator on the reliability of the ISTE predictions, it is also recorded.

By trying different stop times for the replication runs, and by using simulation results from [34], it has been observed that the network usually enters a steady state after 100 time unit: after the replication run's stop time has been increased to 100 time unit, the probability of overflow usually does not change much even if the stop time is increased further. Therefore, the first 100 time unit of each replication run are treated as the warm-up period. The stop time for each replication is set to be 500 time unit so that the stop time is well after the warm up period and the network has had enough time being in the steady state.

Figure 18 Input Traffic Process Shift/Twist

The random input process generated by an existing FGN traffic generator has zero mean and unity variance. To save time from developing a new traffic generator for the simulations, an FGN process with zero mean is applied into the network as the current network input. ISTE predicts how the network will behave if the mean value of the current network input is reduced. As a result, the mean values of the FGN processes, after the adjustments are made, are not positive, as shown in Figure 18. Because the Lindley

equation $Q_k = (Q_{k-1} + X_k - \mu)^+$ is used to model the queue behavior, the queue size depends on the difference between the incoming cell rate and the service rate; therefore, a negative mean input process is valid when the Monte Carlo simulation run is used to verify the adjustment recommended by the ISTE approach indeed produces the expected network performance.

All simulations are executed on a Dell Workstation with Intel Xeo 1.7 GHz CPU and 523 MB of Ram.

3.3. Simulation Results

This Section only shows the result of one scenario from each network topology, see appendix A, B, C for complete simulation results of the single queue, tandem queue, traffic splitting tandem queue scenarios, respectively.

The title of each simulation result figure contains the configuration of the simulation scenario. Some of the configuration parameters are in abbreviation form; the following table explains what each parameter is.

Notation	Explanation
B1	Size of buffer in Queue 1
B2	Size of buffer in Queue 2
C1	Service rate in Queue 1
C2	Service rate in Queue 2
H1	Hurst parameter of input traffic stream A
H2	Hurst parameter of input traffic stream A
R1	Percentage of traffic entering Queue 2 from Queue 1
R2	Percentage of traffic
IS	Importance Sampling based Traffic Engineering Prediction
MC	Monte Carlo Result (Heuristic Search)

Table 2 Notations for Simulation Result Figures

3.3.1. Probability of Overflow

Figure 19 Single Queue Probability of Overflow Result

The probability of overflow is usually defined in terms of $\log_{10}()$, therefore the probability of overflow in the figures are in the logarithm scale. The 95% confidence interval is plotted in the probability of overflow figures. Most of the analysis based approach can work in single queue scenario. The single queue scenarios are used here as “sanity check” for the ISTE process. From Figure 19, one can see that ISTE prediction closely match the actual measured probability of overflow.

Figure 20 Tandem Queue, end-to-end Probability of Overflow Result

The tandem queue scenarios are more interesting. Here, the ISTE analysis is used to predict the end-to-end probability of overflow in the network. As it can be seen in Figure 20, the prediction is fairly accurate. For each different ingress traffic mean rate, the ISTE approach predicts the probability of overflow. Afterward, the ingress traffic is adjusted to that mean rate, and a simulation run is executed to measure the probability of overflow. The ISTE prediction is very reliable.

Figure 21 Tandem Queue, Q1 Probability of Overflow Result

Figure 22 Tandem Queue, Q2 Probability of Overflow Result

In this particular scenario, the ISTE approach was not only used to predict the end to end probability of overflow, but it is also used to predict the probability of overflow in

Q1 and Q2, individually. As it can be seen in Figure 21 and Figure 22, even though Q1 is congested (with overflow), the ISTE prediction on the second queue is still very accurate.

In this case shown in Figure 21, there are more buffer overflow occurrences in Q2. There are other scenarios where there are more buffer overflow occurrences in Q1. The results in Appendix B show that the ISTE prediction is also accurate in those scenarios. This supports our claim that the ISTE predicted probability of buffer overflow is not biased towards any one of the queue in the network.

Figure 23 Split Traffic Tandem Queue, end-to-end Probability of Overflow

The traffic split tandem queue scenario includes multiple traffic sources, traffic stream merge, and traffic stream split cases. As it can be seen in Figure 23, the ISTE analysis is still very accurate in predicting the end to end probability of overflow.

Figure 24 Split Traffic Tandem Queue, Q1 Probability of Overflow

Figure 25 Split Traffic Tandem Queue, Q2 Probability of Overflow

Figure 24 and Figure 25 show the ISTE predictions are also accurate in the probability of overflow of each individual queue. Note that both buffers have

congestions, thus, there are multiple “hot spots” in the network. According to these results, the ISTE approach remains effective.

Because all the complex network topologies can be broken down into elements of split traffic tandem queue topology, the simulation results show that ISTE approach should be effective in almost any network topology.

3.3.2. Accuracy of the ISTE Prediction

Section 3.3.1 has only given a graphical estimate of how accurate the ISTE predicted probability of overflow is. To describe the accuracy in a more rigorous manner, a percentage of error is defined:

$$\frac{|MC_result - ISTE_result|}{|MC_result|} 100\% \quad (21)$$

(21) shows the difference between the performance estimated by ISTE and the performance result measured by the Monet Carlo simulation run.

Figure 26 Percentage of Difference for Single Queue Scenario

Figure 26 shows the discrepancy between the predicted probability of overflow and measured probability of overflow is less than 1.6% in the single queue scenarios.

Figure 27 Percentage of Difference for End-to-end Performance Prediction, Tandem Queue Scenario

Figure 28 Percentage of Difference for Q1 Performance Prediction, Tandem Queue Scenario

Figure 29 Percentage of Difference for Q2 Performance Prediction, Tandem Queue Scenario

Figure 27, Figure 28, and Figure 29 show the errors between the predicted values and the measured values are also less than 3% in the tandem queue case.

Figure 30 Percentage of Difference for End-to-End Performance Prediction, Split Tandem Queue Scenario

Figure 31 Percentage of Difference for Q1 Performance Prediction, Split Tandem Queue Scenario

Figure 32 Percentage of Difference for Q2 Performance Prediction, Split Tandem Queue Scenario

Figure 30, Figure 31, and Figure 32 show the percentage of difference between the predicted behavior and measure behavior is less than 5%. The figures are some erratic

behavior in terms of the discrepancy between the ISTE prediction and the actual measured results. This fluctuation should be due to the stochastic nature of buffer overflow in the network.

From the result all the different simulation scenarios, (including those in the Appendix), the difference between the ISTE prediction and actual measured result is less than 5%, except in a few rare cases. This shows the ISTE approach is accurate enough to be acceptable as a valid traffic engineering approach.

3.3.3. Required Simulation Time

Figure 33 Required Simulation Time Comparison, Single Queue Scenario

Figure 34 Required Simulation Time Comparison, Tandem Queue Scenario

Figure 35 Required Simulation Time Comparison, Split Traffic Tandem Queue Scenario

Figure 33, Figure 34, and Figure 35 show the simulation time required the optimal Heuristic based approach (Monte Carlo) versus the ISTE approach for all three network topologies. The simulation time is recorded in units of minutes. Because of the huge

difference in the simulation time of ISTE and that of Monte Carlo, the simulation times are plotted in the logarithm scale.

Section 3.3.2 stated the difference between the ISTE prediction and Monte Carlo measurement is less than 5%; Yet, it only took the ISTE approach minutes to obtain those results while it took optimal heuristic approach days to obtain these similar results. This clearly demonstrates the advantage the ISTE approach has over the Heuristic approach.

3.3.4. Normalized Variance Results

Figure 36 Normalized Variance for Single Queue Scenario

Figure 37 Normalized Variance for Tandem Queue Scenario

Figure 38 Normalized Variance for Tandem Queue Scenario

Figure 36, Figure 37, and Figure 38 show the Normalized Variance for the single queue, tandem queue, split traffic tandem queue, scenarios. The Monte Carlo process' normalized variance is relatively flat because the number of repetition runs increases to

keep up with the diminishing occurrences of buffer overflow. The normalized variance of the ISTE twisting process, as expected, increases as the mean of the input process becomes more negative (as twisted amount m^* gets larger). This shows the ISTE prediction becomes noisier as the twisted amount gets large. However, by setting a target probability around $10^{-5.5}$ and 10^{-6} , the normalized variance did not get too big and the prediction's accuracy did not deteriorate too much (see simulation result in 3.3.1 and Section 3.3.2) when the target probability is reached.

4. Alternating Twisting

An interesting phenomenon occurred in the simulation when there was an unexpected scenario where ISTE prediction on probability of overflow became very different from the probability measured by Monte Carlo simulation. Further analysis indicates it is the issue of ISTE, which was overcome by the development of the method called “alternating twisting”.

4.1. Dominating Source Twisting Limitation

The ISTE limitation scenario is the one with traffic split tandem topology. The topology consists of two traffic sources as shown in Figure 16. As mentioned in Section 3.2, Source A is the primary traffic source responsible for the network behavior while source B acts as the background source. However, in this ISTE limitation scenario, where $H1 = 0.6$, $H2=0.8$, $B1=100$, $B2=50$, $C1=0.03$, $C2= 0.8$, $R1=0.8$, $R2=1$, the Hurst parameter ($H1$) of source A is much smaller than the Hurst parameter ($H2$) of source B. This implies the traffic from source B is more bursty, and as a result of that, instead of source A being responsible for the overflows in Q2, the background source, source B, is now responsible for most of the overflows in Q2. Dominating source is the source which is the primary cause of the overflow in the network or in a network node.

Figure 39 ISTE Limitation Scenario 1 Result, twisting the non-dominating source.

Figure 40 ISTE Limitation Scenario 1 Result, twisting the dominating source.

As shown by Figure 39 and Figure 40, if the ISTE analysis predicts the behavior of Q2 by “twisting” source A, the predicted result does not match the actual queue

behavior measured from Monte Carlo simulation. It is obvious that the ISTE result is wrong, because it is easy to see that since source A is not responsible for the overflow in Q2, reducing its mean rate should not reduce the probability of overflow.

Instead, if the mean rate of the dominating background source B is twisted, Figure 39 indicates the ISTE results are indeed accurate. This shows the limitation of Importance Sampling in Traffic Engineering: ISTE can only predict the behavior of the network or the network node by “twisting” the source that is taking dominating responsibility for the behavior of the network or the network node. Otherwise, the results will become very noisy and therefore not useful anymore. In another words, the traffic trace captured must be from the source that is causing the occurrences of event of interest in the network. This is intuitively understandable since it is impossible to make predictions regarding a phenomenon based on a quantity that has no (or little) relationship to that phenomenon.

To make sure this limitation is not an isolated incident; another scenario is used to verify this limitation. Let this scenario be “ISTE limitation scenario 2” where $H1=0.6$ and $H2=0.6$, $B1=25$, $B2=50$, $C1=0.2$, $C2=0.1$, $R1=0.11$, $R2=1$. In this scenario, the Hurst Parameters of the two traffic source are the same. However, with $R1=0.11$, 89% of traffic from Q1 (from source A) was diverted off, as a result, the overflows in Q2 are mainly due to background source B.

Figure 41 ISTE Limitation Scenario 2 Result, twisting the non-dominating source

Figure 42 ISTE Limitation Scenario 2 Result, twisting the dominating source

Again, predictions on Q2's probability from twisting source A are incorrect. If we twist the dominating source A to predict the probability of Q1, Figure 41 and Figure 42 show the prediction is accurate.

This dominating source twisting issue on ISTE hinders the application of ISTE on real networks, because in real complex networks, it is difficult to know which traffic source is the dominating cause for the overflow on a single node or the end-to-end overflow of the network. One possible solution to find the dominating source is to simply increase/decrease the mean rate of each traffic source (one by one) and observe the change in the queue/network overflow occurrence. (For more details, see Chong's work in [41] [42] [43]). However, the amount of reduction in the mean rate is hard to determine. It will usually require several trials. All the traffic sources need to be given the same amount of reduction in order to find the dominating source. If the initial reduction amount is too small, none of the traffic source will have an effect on the queue overflow, and a different reduction amount has to be used and the process has to be repeated. If there are many traffic sources in the network and the overflow event is rare, after each adjustment on each traffic source, a long waiting time will be required to determine if there is a change in the overflow occurrences. The process is extremely time-consuming. "Alternating Twisting" (AT) is a new method developed to overcome the dominating source limitation. It is more rigorous and systematic than the solution just mentioned. Alternating Twisting can find the dominating source and optimize the performance at the same time. AT determines the dominating source based on calculations instead of measurements. As a result, AT doesn't have to try different amount of mean rate reduction and wait for the result. The amount of reduction required is calculated. According to the simulation results in Section 4.4, AT has indeed solved the dominating source problem of the ISTE approach.

Alternating Twisting is developed based on a phenomenon observed while doing simulation on the two ISTE limitation scenarios.

Figure 43 ISTE Limitation Scenario 1, Normalized variance.

For ISTE limitation scenario 1, instead of checking the probability of overflow, the normalized variance is observed. The normalized variance of the ISTE approach when twisting the non-dominating source A (results inaccurate) and the normalized variance of ISTE when twisting the dominating source B (result accurate) are compared in Figure 43. As it can be seen, the normalized variance increases much faster when twisting source A than when twisting source B.

Figure 44 ISTE Limitation Scenario 2, Normalized variance

To verify this phenomenon, the normalized variance for ISTE limitation scenario 2 is plotted. Figure 44 shows that when ISTE is twisting the non-dominating source A to estimate the probability of overflow in Q2, the normalized variance increase much faster than the normalized variance when ISTE is twisting source A to predict the probability of overflow in Q1 (where source A is the dominating source).

In conclusion, Normalized Variance increases much faster when twisting a non-dominating source than when twisting a dominating source. Intuitively, because the twisting of the non-dominating source can not predict the network behavior that the source is not responsible for, its predication becomes noisy, causing the normalized variance to increase. As mentioned in Section 2.1.3, the normalized variance is an indicator on the degree of variation in the sample mean, where larger variation indicates the sample mean is unreliable.

This implies the normalized variance is an indicator on whether the source being twisted is the dominating source or not. Therefore, when the traffic source is no longer

the dominating source, the ISTE process can move on to twisting the next traffic source. Alternating Twisting is based on this idea.

4.2. Threshold Based Alternating Twisting

Figure 45 Alternating Twisting

“Threshold based Alternating Twisting” sets a threshold value on the normalized variance. If the normalized variance is below the threshold, the source being twisted is still considered as the dominating source (a source can become a non-dominating source after its mean has been reduced too much). If the normalized variance exceeds the threshold, the source is no longer considered as the dominating source, it becomes a non dominating source. Recall from Section 2.3, “twisting” is a prediction and calculation process; it does not actually reduce the mean rate of the source. “Twisting” instead predicts the probability of overflow if the mean rate was reduced by a specific amount. In the “twisting” process, if the normalized variance exceeds the threshold for a certain amount of reduction, it means that if we actually made such reduction in the mean rate, the source will no longer be the dominating source.

If it is estimated from twisting that after the source mean rate has been reduced such that it is not the dominating source anymore, the target network/node probability is still not met, the source mean rate is reduced to the point so it is not the dominating source anymore as predicted by the twisting process, and the twisting is *continued* on the next traffic source in the network. This repeats until eventually the expected probability target is achieved by reducing the mean rates across all the traffic sources on the network. Thus, the “Alternating Twisting” method actually alternatively twists the traffic sources on the network to reduce the probability of overflow to an acceptable level.

```

/*
Prob_overflow: probability of overflow predicted by ISTE
Target_Prob_overflow: probability of overflow to be achieved
*/
i=0;
while (Prob_overflow > target_Prob_overflow) {
    while (norm_variance < threshold)
        && (Prob_overflow < target_Prob_overflow)
    {
        twisted_amount += delta;
        /* twist further since we did not reach the target probability of
        overflow in the last try*/

        prob_overflow= ISTE_twisting( source[i], twisted_amount);
        /* using ISTE to predict the probability of overflow if traffic
        source i 's mean rate is reduced by the twisted_amount*/

        norm_variance = find_variance( source[i],twisted_amount);
        /* find the normalized variance of the predicted probability of
        overflow. Normalized Variance is used to see if the mean rate
        is reduced by the twisted amount, will the source still be the
        dominating source? */
    }
    if (prob_overflow < target_Prob_overflow) {
        reduce_traffic( source[i],twisted_amount);
        i = i +1;
        if (i == MAX_SOURCES) {
            /* we have twisted all the sources, start again with the first
            source*/
            i = 0;
        }
        rerun_simulation();
        /* rerun simulation since event occurrence trace will be
        different after we reduce the incoming traffic */
    } else {
        break;
    }
    /* we have achieved our target, we know how much (the current

```

```

        value in twisted_amount) to reduce the mean rate by to obtain
        the target*/
    }
}

```

Figure 46 AT-ISTE algorithm C-code

At the beginning of each twisting process, after switching to a new source, a simulation run must be executed again to recapture the event occurrence trace. This is due to the fact that the previous source has reduced its mean rate, which means the event occurrence trace could be different. As stated in Section 2.3.3, once the simulation run is re-executed to recapture the traffic trace and the event trace, the normalized variance of the ISTE prediction will become small again, as indicated in Section 2.1.3. It will increase gradually as the twisting continues, but hopefully, the target can be reached before the normalized variance exceeds the threshold again. If threshold is crossed again, another simulation run is required. Figure 46 shows the C-code for the Alternative Twisting (AT) algorithm.

Previously, it was indicated that only one simulation run is required in the ISTE approach in comparison to the multiple simulation runs in the heuristic search based approaches, alternating twisting actually will require more than one simulation run. As it will be shown in Section 4.4, even though it is hard to “twist” one source all the way to reach the target probability, without its normalized variance exceeding the threshold, it can still usually get a “big range”, as it may not be able to reach the target, but it can get closer to it, before its normalized variance crosses the threshold. Therefore only a few additional simulation runs are required to achieve the target probability, in contrast to the many simulation runs, where one run is needed after each adjustment, required for the

heuristic search. Based on the simulation results in Section 3.3, we use a threshold value of 0.1 as an example.

4.3. Likelihood Ratio in Alternating Twisting

The likelihood ratio formula in Section 2.1 is valid if the original input process has a zero mean. However, in Alternating Twisting, the original input traffic process may not have a mean of zero: Assume the behavior of a queue is not dominated by a single traffic source, but rather two traffic sources. By reducing the mean rate of traffic source A, traffic source B becomes the dominated source. Assume the target probability has not been reached after the mean rate of source A is reduced, thus traffic B source mean rate also has to be reduced, which may make source A the dominating source. When the mean of source B is reduced, the target probability is not reached, but source A does become the dominating source. Source A needs to be twisted again. Because the mean rate of traffic source A was reduced once before, its mean is no longer zero. The twisting process requires the calculation of the likelihood ratio, yet the formula for likelihood ratio in Section 2.1 no longer works. A more general formula (the mean of the original input process is not equal to zero) has to be derived (as one of the contribution of this thesis):

If X is a FGN random process with mean of one, a unit variance, and correlation coefficients of ϕ_{jk} , then the conditional mean of X is:

$$E_X(X_k | x_{k-1}, \dots, x_1) = \sum_{j=2}^k \phi_{kj}(x_{k-j}) [34]$$

Let $Y=X+v$, and Y 's pdf is $f(t, v)$

Let $Z=X+v_0$, and Z 's pdf is $f(t, v_0)$

Y is the unknown “original” input process need to be determined. (It is equivalent to the “ X ” process in Section 2.1.) Z is the “twisted process that will be applied into the network to capture the traffic and event occurrence trace (it is equivalent to the ‘ Y ’ process in Section 2.1.)

$$E_X(X_k | x_{k-1}, \dots, x_1) = \sum_{j=2}^k \phi_{kj}(x_{k-j}) \quad (22)$$

$$E_X(X_k | y_{k-1}, \dots, y_1) = \sum_{j=2}^k \phi_{kj}(y_{k-j}) = \sum_{j=2}^k \phi_{kj}(x_{k-j} + v) \quad (23)$$

$$E_X(X_k | z_{k-1}, \dots, z_1) = \sum_{j=2}^k \phi_{kj}(z_{k-j}) = \sum_{j=2}^k \phi_{kj}(x_{k-j} + v_0) \quad (24)$$

$$\begin{aligned} & E_Z(Z_k | z_{k-1}, \dots, z_1) \\ &= (+v_0) + E_{Y-v}(Z_k - v_0 | z_{k-1} - v_0, \dots, z_1 - v_0) \\ &= v_0 + E_X(X_k | x_{k-1}, \dots, x_1) \\ &= v_0 + \sum_{j=2}^k \phi_{kj}(x_{k-j}) \end{aligned} \quad (25)$$

$$\begin{aligned} & E_Y(Y_k | z_{k-1}, \dots, z_1) \\ &= (+v) + E_{Y-v}(Z_k - v | z_{k-1} - v, \dots, z_1 - v) \\ &= v + E_{X+v_0-v}(X_k + v_0 - v | x_{k-1} + v_0 - v, \dots, x_1 + v_0 - v) \\ &= v + \sum_{j=2}^k \phi_{kj}(x_{k-j} + v_0 - v) \end{aligned} \quad (26)$$

The conditional variances of X , Y , Z are the same: $Var = \frac{1}{\sigma^2 \prod_{j=2}^k (1 - \phi_{jj}^2)}$ (27)

Similar to the proof in Section 2.1:

$$\begin{aligned}
 L(t, v_o, v) &= \frac{f(t, v)}{f(t, v_o)} = \prod_{j=2}^t \frac{\exp(-(z_j - E_Y(Y_j | z_{j-1}, z_{j-2}, \dots, z_1))^2 / 2 \text{var}_Y(Z_j | z_{j-1}, z_{j-2}, \dots, z_1))}{\exp(-(z_j - E_Z(Z_j | z_{j-1}, z_{j-2}, \dots, z_1))^2 / 2 \text{var}_Z(Z_j | z_{j-1}, z_{j-2}, \dots, z_1))} \\
 &= \prod_{j=1}^t L_j
 \end{aligned} \tag{28}$$

For better readability,

Let $E_{v_o} = E_Z(Z_j | z_{j-1}, z_{j-2}, \dots, z_1)$ and $E_v = E_Y(Y_j | z_{j-1}, z_{j-2}, \dots, z_1)$

(28) becomes:

$$\begin{aligned}
 L_j &= \frac{\exp(-(z_j - E_v)^2 / 2 \text{var})}{\exp(-(z_j - E_{v_o})^2 / 2 \text{var})} \\
 &= \exp((- (z_j - E_v)^2 / 2 \text{var}) - (- (z_j - E_{v_o})^2 / 2 \text{var})) \\
 &= \exp((- z_j^2 + 2 z_j E_v - E_v^2 + z_j^2 - 2 z_j E_{v_o} + E_{v_o}^2) / 2 \text{var}) \\
 &= \exp\left(\frac{2 z_j E_v - E_v^2 - 2 z_j E_{v_o} + E_{v_o}^2}{2 \text{var}}\right) \\
 &= \exp\left(\frac{2 y_j (E_v - E_{v_o}) - (E_v^2 - E_{v_o}^2)}{2 \text{var}}\right) \\
 &= \exp\left(\frac{2 y_j (E_v - E_{v_o}) - (E_v - E_{v_o})(E_v + E_{v_o})}{2 \text{var}}\right) \\
 &= \exp\left(\frac{y_j (E_v - E_{v_o})}{\text{var}} - \frac{(E_v - E_{v_o})(E_v + E_{v_o})}{2 \text{var}}\right)
 \end{aligned} \tag{29}$$

$$\text{Let } \theta = \frac{(E_v - E_{v_o})}{\text{var}} \quad M = \exp\left(\frac{\theta(E_v + E_{v_o})}{2}\right) \tag{30}$$

(29) becomes:

$$L_j = \frac{\exp(z_j \theta)}{M} \tag{31}$$

Substitute (25) and (26) into (30)

$$\theta = \frac{(E_Y - E_Z)}{\text{var}} = \frac{(v - v_0) + \sum_{j=2}^2 \phi_{kj}(v_0 - v)}{\text{var}} \quad (32)$$

$$M = \exp\left(\frac{\theta(E_Y + E_Z)}{2}\right) = \exp\left(\frac{\theta(v + v_0 + 2\sum_{j=2}^2 \phi_{kj}(x_{k-j}) + \sum_{j=2}^2 \phi_{kj}(v_0 - v))}{2}\right) \quad (33)$$

Since equations (32) (33) is a more general form of (16), by setting $v=0$, (32) and (33)

$$\text{become: } \theta = \frac{v_0(\sum_{j=2}^2 \phi_{kj} - 1)}{\sigma^2 \prod_{j=2}^2 (1 - \phi_{jj}^2)} \quad M = \exp\left(\frac{(2\sum_{j=2}^2 \phi_{kj}x_{k-j} + v_0(1 + \sum_{j=2}^2 \phi_{kj}))\theta}{2}\right)$$

which is the same as (16). This shows (32) and (33) are correct.

4.4. Alternating Twisting Simulation Results

Alternating Twisting will find out how much reduction in the mean rates is needed in each traffic source to achieve a performance target. Then, Monte Carlo simulation is executed to verify if the target is actually met.

Alternating Twisting Simulation Scenario 1: Split Traffic Tandem Queue, Q2, H1=0.6 H2=0.8 B1=100 B2=50 C1=0.03 R1=0.8 R2=1. Target: $\log(\text{Probability of overflow})=-6$

SRC_A initial mean	0.00000	
SRC_B initial mean	0.00000	
Q2		
mean rate (Traffic Source A)	Probability of Overflow	Normalized Variance
0	-1.76700	0.00570
-0.05	-1.87470	0.00820
-0.1	-1.98800	0.02320
-0.15	-2.10670	0.09020
-0.2	-2.23920	0.26540

SRC_A initial mean	-0.15000	
SRC_B initial mean	0.00000	
Q2		
mean rate (Traffic Source B)	Probability of Overflow	Normalized Variance
0	-2.03010	0.00110
-0.1	-2.39870	0.00110
-0.2	-2.79680	0.00130
-0.3	-3.22300	0.00190
-0.4	-3.67550	0.00290
-0.5	-4.1531	0.0052
-0.6	-4.6548	0.0099
-0.7	-5.1805	0.0186
-0.8	-5.7311	0.0337
-0.81	-5.7876	0.0356
-0.82	-5.8443	0.0376
-0.83	-5.9014	0.0397
-0.84	-5.9587	0.0419
-0.85	-6.0162	0.0442
-0.86	-6.0741	0.0466
-0.87	-6.1322	0.049

Figure 47 Alternating Twisting Simulation Scenario 1

Alternating Twisting Simulation Scenario 1 is actually ISTE Limitation Scenario 1. Alternating Twisting method is applied here to make sure it can really resolve the limitation of the ISTE mentioned in Section 4.1. In this scenario, Source B is clearly the

dominating source for Q2. The source B twisting process is able to reach the target probability without crossing the normalized variance threshold. According to the results from alternation twisting, when the mean of SRC A is reduced by 0.15 units/sec and the mean of SRC B is reduced by 0.86 units/sec, the probability, in logarithm scale, of overflow in Q2 should be -6.07. Using Monte Carlo simulation to verify the result, the logarithm scale overflow probability in Q2 after these reductions is -6.25297. The ISTE prediction is off by about 2.8%, which is tolerable.

Figure 48 Probability of Overflow Trajectory in Alternating Twisting Simulation Scenario 1

Figure 48 shows the probability of overflow trajectory predicted by the Alternating Twisting process. We can see how the probability of overflow in Q2 will change if we reduce the mean rates of source A and source B.

Alternating Twisting Simulation Scenario 1 has Q2 completely dominated by the source B. Alternating Twisting Simulation Scenario 2 has Q2 equally dominated by sources A and B. Thus, the two sources become the dominating source in turns. As the sources are reduced alternatively, the queue that will be more congested (having more overflow events than the other) will be Q1 or Q2 alternatively.

Alternating Twisting Simulation Scenario 2: Split traffic tandem queue, Q2, H1=0.6 H2=0.6 B1=25 B2=50 C1=0.2 C2=0.1 R1=0.11 R2=1; Target: log(Probability of overflow)=-6.3

SRC_A initial mean	0.00000	
SRC_B initial mean	0.00000	
Q2		
mean rate (Traffic Source A)	Probability of Overflow	Normalized Variance
0	-1.72350	0.00520
-0.05	-1.82090	0.00750
-0.1	-1.91620	0.02050
-0.15	-2.02010	0.07000
-0.2	-2.15030	0.19130

SRC_A initial mean	-0.15000	
SRC_B initial mean	0.00000	
Q2		
mean rate (Traffic Source B)	Probability of Overflow	Normalized Variance
0	-1.9469	0.0044
-0.05	-2.5806	0.0054
-0.1	-3.2792	0.0135
-0.15	-4.0104	0.0545
-0.2	-4.7573	0.1776

SRC_A initial mean	-0.15000	
SRC_B initial mean	-0.15000	
Q2		
mean rate (Traffic Source A)	Probability of Overflow	Normalized Variance
-0.15	-4.1288	0.0022
-0.2	-4.2389	0.0032
-0.25	-4.3527	0.0077
-0.3	-4.4884	0.0211
-0.35	-4.6728	0.0514
-0.4	-4.9334	0.1029

SRC_A initial mean	-0.15000	
SRC_B initial mean	-0.35000	
Q2		
mean rate (Traffic Source B)	Probability of Overflow	Normalized Variance
-0.15	-4.5477	0.0118
-0.2	-5.4255	0.0155
-0.25	-6.3417	0.0323

Figure 49 Alternating Twisting Simulation Scenario 2

According to the results from alternating twisting, if the mean of source A is reduced by 0.35 units/sec and the mean of source B is reduced by 0.25 units/sec, the logarithm scale probability of overflow in Q2 should be -6.3417. Monte Carlo simulation shows that the logarithm scale probability of overflow in Q2 after these reductions is -6.37. The ISTE prediction is off by 0.4%. The results strongly support that Alternative Twisting prediction is accurate in this scenario.

Figure 50 Probability of Overflow Trajectory in Alternating Twisting Simulation Scenario 2

Figure 50 shows that source A and source B are “alternatively” twisted to reach the target probability of overflow.

Alternating Twisting Simulation Scenario 3 is to make sure Alternating Twisting is effective for predicting end-to-end network behavior.

Alternating Twisting Simulation Scenario 3: split traffic tandem queue, end-to-end, H1=0.7
H2=0.7 B1=50 B2=100 C1=0.3 C2=0.1 R1=0.5 R2=1; Target: $\log_{10}(\text{Probability of Overflow})=-6.0$

SRC_A initial mean	0.00000	
SRC_B initial mean	0.00000	
end-to-end		
mean rate (Traffic Source B)	Probability of Overflow	Normalized Variance
0	-1.2165	0.0008
-0.1	-1.5953	0.0016
-0.2	-1.8515	9.60E-03
-0.3	-1.9963	0.0675
-0.4	-2.15	0.1289

SRC_A initial mean	0.00000	
SRC_B initial mean	-0.30000	
end-to-end		
mean rate (Traffic Source A)	Probability of Overflow	Normalized Variance
0	-1.8586	0.0036
-0.1	-2.5481	0.0044
-0.2	-3.3296	8.70E-03
-0.3	-4.195	0.0524
-0.4	-5.1498	0.1177
-0.5	-6.2088	0.1586

SRC_A initial mean	-0.30000	
SRC_B initial mean	-0.30000	
end-to-end		
mean rate (Traffic Source B)	Probability of Overflow	Normalized Variance
-0.3	-4.1192	0.0263
-0.4	-4.1601	0.0703
-0.5	-3.9578	0.3102

SRC_A initial mean	-0.30000	
SRC_B initial mean	-0.40000	
end-to-end		
mean rate (Traffic Source A)	Probability of Overflow	Normalized Variance
-0.3	-4.0757	0.006
-0.32	-4.2237	0.0061
-0.34	-4.3692	0.0065
-0.36	-4.5124	0.0073
-0.38	-4.6533	0.0084
-0.4	-4.7921	0.0101
-0.42	-5.2781	0.0212
-0.44	-5.4588	0.0224
-0.46	-5.6391	0.0246
-0.48	-5.8187	0.0282
-0.5	-5.9974	0.0335
-0.52	-6.1751	0.0414

Figure 51 Alternating Twisting Simulation Scenario 3

According to the result from alternating twisting, if the mean of source A is reduced by 0.5 unit/sec and the mean of source B is reduced by 0.4 unit/sec, the logarithm scale end-to-end probability of overflow should be is -5.9974. The Monte

Carlo Simulation shows the logarithm scale end-to-end probability of overflow is -6.065. It is obvious that the predicted value of alternating twisting is extremely close to the value measured by the Monte Carlo simulation. Figure 52 gives a graphical view of how the end-to-end probabilities of overflow will change as the mean rates of the traffic sources are reduced.

Figure 52 Probability of Overflow Trajectory in Alternating Twisting Simulation Scenario 3

Figure 53 Individual Queue Behavior during AT

In Figure 53, as the mean rate of each of the traffic source is reduced one at a time, the probability of overflow in the individual queues change. Initially, there were more overflows in Q2 than in Q1. As the Alternating Twisting continues, Q1 becomes the queue with more overflow events. Since both source A and source B contributes to the overflow in Q2, regardless which mean rate is reduced, Q2's probability of overflow will

always decrease. This is reason why Q2 always have a lower probability as the AT process continues.

5. Conclusions

Importance Sampling based Traffic (ISTE) engineering approach can act as an excellent optimization component in any traffic engineering system. It is a simple and effective way of predicting how the behavior of the network will change if network inputs are changed. The simulation results in the thesis verified that the ISTE analysis is accurate in predicting the probability of overflow in a network with self-similar input traffic under different input traffic mean rates. Other metrics, such as packet loss ratio, probability of latency time violation, was not tried. They are topics for future research.

For complex networks, ISTE is more applicable than any analysis based approach because the ISTE approach does not require the knowledge of the internal details or internal actives (whether there are overflows in upstream nodes) of a network. The ISTE approach can also perform end-to-end performance predictions which some of the analytical approaches can not perform. The simulation results in the thesis prove that the ISTE approach only needs a few *minutes* to obtain the same results that the Heuristic based approach must spend *days* to obtain.

The Alternating Twisting method is developed to overcome the dominating source limitation of Importance Sampling based Traffic Engineering. The normalized variance in the twisting process is a good indicator of whether a traffic source is responsible for the behavior of a network node or a network. This discovery alone is extremely useful in congestion control. A simple way of resolving congestion is to find the traffic source(s) that are mainly responsible for the congestion (high overflow probability of a node), and force that source to reduce its traffic. In the process of developing Alternating Twisting,

a more general equation to calculate the likelihood is derived, that could be useful in the future if Importance Sampling is applied in other fields of research.

6. Future Work

Recall that ISTE should work regardless of the network behavior (as long as it can be modeled to include stochastic events) and regardless of the network input (as long as it can be modeled as a stochastic process). This thesis has only simulated the probability of overflow of a network and the input traffic process; other network behavior and inputs should also be simulated. On the same note, ISTE should also be applicable regardless of the network internal details, therefore simulation with more complex network topologies and different queueing behavior should be carried out. Eventually, ISTE should be implemented on a real network to test its effectiveness.

The mathematical and logical operations required to implement ISTE are simple, thus ISTE can be implemented in hardware using VHDL, Verilog, FPGA to make the traffic engineering system's response time better.

Appendix A. Complete Single Queue Simulation Results

Single Queue Scenario 1 Simulation Results

Single Queue Scenario 2 Simulation Results

Single Queue Scenario 3 Simulation Results

Single Queue Scenario 4 Simulation Results

Appendix B. Complete Tandem Queue Simulation Results

Tandem Queue Scenario 1 Simulation Results

Tandem Queue Scenario 2 Simulation Results

Tandem Queue Scenario 3 Simulation Results

Tandem Queue Scenario 4 Simulation Results

Tandem Queue Scenario 5 Simulation Results

Tandem Queue Scenario 6 Simulation Results

Appendix C. Complete Traffic Splitting Tandem Queue Simulation Results

Reference

- [1] D. O. Awduche, A. Chiu, A. Elwalid, I. Widjaja, and X. Xiao, "Overview and Principles of Internet Traffic Engineering", IETF Internet Draft, May 2002.
- [2] D. O. Awduche, B. Jabbari, "Internet traffic engineering using multi-protocol label switching (MPLS)", *Computer Networks, The International Journal of Computer and Telecommunications Networking Computer Networks and ISDN Systems*, 40 (2002) 111–129, 2002
- [3] K. Park and W. Willinger (eds.). "Self-Similar Network Traffic and Performance Evaluation", Wiley-Interscience, 2000.
- [4] J. Strauss, D. Katabi, F. Kaashoek, "A Measurement Study of Available Bandwidth Estimation Tools", *ACM SIGCOMM Internet Measurement Conference '03*, Miami, Florida, October 2003
- [5] R. D. Callaway, M. Devetsikiotis, and C. Kan, "Design and Implementation of Measurement-Based Resource Allocation Schemes Within the Real-time Traffic Flow Measurement Architecture" in *Proc. IEEE International Conference on Communications (ICC 2004)*, June 2004
- [6] K. Papagiannakix, R. Cruz, C. Diot, "Network Performance Monitoring at Small Time Scales", *ACM SIGCOMM Internet Measurement Conference '03*, Miami, Florida, October 2003
- [7] F. Hacıomeroglu, "On-line Measurement-based Capacity Allocation Schemes", A dissertation submitted to the Graduate Faculty of North Carolina State University, 2003
- [8] J. Banks, J. S. Carson, II, B. L. Nelson, David M. Nicol, "Discrete Event System Simulation, Third Edition", Prentice Hall, 2001, page 204-445, page 410-411, page 430
- [9] <http://www.m-w.com/cgi-bin/dictionary?book=Dictionary&va=trial+error>
- [10] W. L. Winston, M. Venkataramanan, "Introduction to Mathematical Programming", 4th edition, Thomson Brooks/Cole, 2003, page 692-697, 744-745
- [11] W. L. Winston, M. Venkataramanan, "Introduction to Mathematical Programming", 4th edition, Thomson Brooks/Cole, 2003, page 703-706
- [12] J.T. Lewis, N. O'Connell and R. Russell, "An introduction to large deviations for teletraffic engineers", Dublin Institute for Advanced Studies, November 1997
- [13] F.P. Kelly. "Stochastic Networks: Theory and Application", Oxford University Press, 1996, pages 141-168

- [14] C. Courcoubetis, V. A. Siris, and G. Stamoulis, "Application of the many sources asymptotic and effective bandwidths to traffic engineering", *Telecommunication Systems*, 12(2-3):167-191, 1999
- [15] C. Courcoubetis and R. Weber, "Buffer overflow asymptotics for a buffer handling many traffic sources.", *Journal of Applied Probability*, 33:886-903, 1996
- [16] R.J. Gibbens and P.J. Hunt. "Effective bandwidths for the multi-type uas channel. Queueing Systems", *ACM Trans*, Volume 9 , Issue 1-2 , October 1991, page 17-28
- [17] Norros I. "On the use of fractional Brownian motion in the theory of connectionless networks. *IEEE Journal of Selected Areas in Communications*, 13(6):953-962, 1995
- [18] R. Guerin, H. Ahmadi, and M. Naghshineh, "Equivalent capacity and its application to bandwidth allocation in high-speed networks.", *IEEE Journal of Selected Areas in Communications*, 7(7):968-981, 1991
- [19] <http://mathworld.wolfram.com/QueueingTheory.html>, Wolfram Research, 2004
- [20] I. Norros, "On the use of fractional Brownian motion in the theory of connectionless networks", *Selected Areas in Communications, IEEE Journal on* , Volume: 13 , Issue: 6 , 1995, Pages:953 – 962
- [21] N.G. Duffield, J.T. Lewis, N. O'Connell, R. Russell, F. Toomey, "Predicting quality of service for traffic with long-range fluctuations" *Communications*, 1995. ICC 95 Seattle, Gateway to Globalization, 1995 *IEEE International Conference on* , Volume: 1 , 18-22 June 1995 Pages:473 - 477 vol.1
- [22] W.E. Leland, M.S. Taqqu, W. Willinger, and D.V. Wilson, "On Self-Similar Nature of Ethernet Traffic (Extended Version)". *ACM/IEEE Transactions on Networking*, 2(1):1-15, Feb. 1994
- [23] V. Paxson and S. Floyd. "Wide Area Traffic: The failure of Poisson Modeling *IEEE/ACM Transactions on Networking*, 3(3):226-244, 1995
- [24] M.E. Crovella; A. Bestavros, "Self-similarity in World Wide Web traffic: evidence and possible causes", *Networking, IEEE/ACM Transactions on Networking*, Volume: 5 , Issue: 6 , Dec. 1997 Pages:835 – 846
- [25] Z. Duan, Z. Zhang, Y. T. Hou "Service Overlay Networks: SLAs, QoS, and Bandwidth Provisioning", *Networking, IEEE/ACM Transactions on* , Volume: 11 , Issue: 6 , Dec. 2003 Pages:870 - 883, 2003
- [26] N.G. Duffield, P. Goyal, A. Greenberg, et al, "draft-duffield-vpn-qos-framework-00.txt", *IETF Draft*, 1998

- [27] N.G. Duffield P. Goyal, et al, "Resource Management with Hoses: Point to Cloud Services for VPN", IEEE/ACM. Transaction on Networking, Vol 10, No 5, October 2002
- [28] I. Khalili et al, "A Range-Based SLA and Edge Driven Virtual Core Provisioning in Diffserv-VPNs", IEEE-LCN, 2001
- [29] I. Khalil, et al, "Edge Provisioning and Fairness in VPN-Diffserv Networks", IEEE-ICCN, 2000
- [30] K. Ross, "Multiservice Loss models for Broadband Telecommunication Networks", Springer-Verlag, 1995.
- [31] M. F. Neuts, "Matrix-Geometric Solutions in Stochastic Models: An Algorithmic Approach ", Johns Hopkins University Press, 1981.
- [32] M. Schwartz, "Broadband Integrated Networks", Prentice Hall PRT, 1996
- [33] P. L'ECUYER, Y. Champoux, "Estimating Small Cell Loss Ratio in ATM Switches via Importance Sampling", ACM Transactions on Modeling and Computer Simulation, Vol. 11, No. 1, January 2001, Pages 76-105
- [34] C. Huang, "Long Range Dependent Traffic: Modeling, Simulation, and Congestion Control", Doctor of Philosophy thesis, Carleton University, Canada, 1997
- [35] R. Y. Rubinstein, B. Melamed "Modern Simulation and Modeling", John Wiley & Sons Inc, 1998, Pages 100-101, Pages 89-108
- [36] M. Falkner, M. Devestsikiotis, I. Lambadaris. "Fast simulation of networks of queues with effective and decoupling bandwidths.", ACM Trans. Model. Computer Simulation 9,1 (Jan. 1999), 45-48
- [37] A. L. Garcia, "Probability and Random Processes for Electrical Engineering", 2nd edition, Addison Wesley Longman, 1994, Pages 119-126
- [38] G.S. Fishman, "Monte Carlo: Concepts, Algorithms, and Applications", Springer-Verlag, New York, NY, 1996.
- [39] <http://mathworld.wolfram.com/MonteCarloMethod.html>, Wolfram Research, 2004
- [40] A. L. Garcia, "Probability and Random Processes for Electrical Engineering", 2nd edition, Addison Wesley Longman, 1994, Pages 280-288
- [41] G. Wu, E. K. P. Chong, and R. L. Givan, "Burst-level congestion control using hindsight optimization," IEEE Transactions on Automatic Control, special issue on Systems and Control Methods for Communication Networks, vol. 47, no. 6, pp. 979--991, June 2002.

[42] E. K. P. Chong, "A recursive approach to stochastic optimization via infinitesimal perturbation analysis," in Proceedings of the 33rd IEEE Conference on Decision and Control, Lake Buena Vista, Florida, December 14--16, 1994, pp. 1984--1989 (Invited Paper).

[43] E. K. P. Chong and S. H. Žak, "An Introduction to Optimization", New York, NY: John Wiley and Sons, Inc., 1996