Mapping Bandwidth to Quality of Service

An Importance Sampling Based Traffic Engineering Approach

Benjamin Zhong Ming Feng[+], Changcheng Huang[+], Mike Devetsikiotis[*], Yanick Champoux[#]

[+]Department of Systems and Computer Engineering, Carleton University, Ottawa, Canada

[*]Department of Electrical and Computer Engineering, North Carolina State University, Raleigh, U.S.A

[#]Department of Research and Innovation, Alcatel, Ottawa, Canada

***Abstract*— This paper proposes a new traffic engineering approach: Importance Sampling based Traffic Engineering (ISTE). ISTE can map the bandwidth of a traffic flow to the Quality of Service (QoS) it can receive within a network. The proposed ISTE approach does not require extensive knowledge of the network internal details thus making it applicable to most large and complex networks. It can carry out the end-to-end QoS analysis of a network or carry out the performance analysis of a single network node. Even if there are multiple congested nodes in the network, the ISTE approach remains effective. This paper will show the ISTE approach, under self-similar [1] [2] traffic model, is capable of calculating the changes in the network QoS (e.g. Probability of Buffer Overflow) with respect to the changes in the bandwidths of the ingress network traffic flows. In the scenarios where several ingress traffic flows influence the QoS of the network, a more specialized technique called ISTE Alternating Twisting (ISTE-AT) is proposed. ISTE-AT makes the proposed ISTE approach even more powerful.**

Key words: ***Traffic engineering; Importance Sampling; QoS; bandwidth; probability of overflow; traffic mean rate; self-similar;***

I. Introduction

Network Quality of Service (QoS) has been a popular research topic in the recent years [3]. QoS is usually specified in terms of packet lose ratio, delay, delay jitter, utilization, etc. Since the packet loss in a network is mainly due to buffer overflow; therefore, this paper focuses on the probability of buffer overflow in a network.

In traffic engineering, it is key to know how the bandwidth of a traffic flow maps to the QoS it receives. Knowing the change in a network's QoS with respect to a change in the bandwidth of an incoming traffic flow is extremely helpful in optimizing the performance of a network to achieve its target QoS. Existing approaches, such as effective bandwidth [4], are applicable only to single network node and quickly become intractable with an end-to-end network path.

This paper uses the bandwidth (the mean rate) of the ingress traffic as the quantity to be mapped to the network QoS (the probability of overflow). The Importance Sampling based Traffic Engineering (ISTE) approach provides a simple and fast mapping of the bandwidths for the ingress flows to the network QoS. This mapping is effective in end-to-end network performance optimization, even if there are multiple congested nodes in the network. Under self-similar traffic model, the simulation results will show the ISTE approach is capable of accurately predicting the changes in the QoS if the bandwidth changes. For scenarios where the network probability of overflow is influenced by the bandwidths of more than one traffic flow, the ISTE-Alternating Twisting (AT) approach is proposed to handle such situations.

The rest part of the paper is organized as follows:

Section II is the overview of the Importance Sampling concept and its classical application. Section III explains this new concept of Importance Sampling based Traffic Engineering. The simulation framework and the simulation results of this paper are outlined in Section IV. Section V covers the special technique ISTE-AT. Finally, Section VI draws the conclusions of this paper.

II. Importance Sampling in Rare Event Simulation

We start with the concept of Importance Sampling[5] and its classical application before moving on to the new ISTE approach.

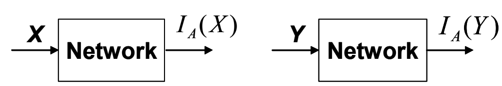

Figure 1. Network Event Occurences Under Different Input

As shown in Figure 1, the input random process *X(t)* (denoted using ***X***), with a mean value of *v*, represents the ingress network traffic. *X(t)* is the amount of traffic sent at time t in units of bytes, packets, or cells. Let *x* denote the random sample/trace *x(t)* of the input process ***X***.

When the input process ***X*** is applied to the network, the event of our interest (the event of packet loss, the event of buffer overflow, or the event of delay time violation) does not occur frequently. Define a set *A* such that *A* = {*w: an event of interest such as buffer overflow*}. $I_A(x)$ is a function that returns 1 if $x \in A$ and 0 if $x \notin A$. The expectation of I_A, $E_x\{I_A(X)\}$, is the probability of the event occurrence when the network is under the input process ***X***. This expectation can be calculated using the Monte Carlo [6] Estimator:

$$\hat{P}_{MC} = (\sum_{i=1}^{N} I_A(x_{(i)}))/N \qquad (1)$$

In equation (1), $x_{(i)}$ denotes the trace $x_{(i)}(t)$ of X generated on the i-th replication run. N is the total number of replication runs executed. At the end of i-th replication run, if the event of interest has occurred, $I_A(x_{(i)})$ is set to 1, else, it is set to 0.

The Monte Carlo estimator works only if every sample point is independent from each other. However, since this paper uses the self-similar traffic model, there is a strong correlation between the samples. To overcome this issue, instead of executing one long simulation run, several independent replication runs are executed, thus making the occurrence of event A independent in each replication. Thus, the probability of event occurrence can be calculated by taking the average number of replications with the event occurrence.

Because of the rare occurrences of the event when the network is under the input process ***X***, many replication runs are needed to capture enough occurrences in order to compute the probability of event occurrences. This usually results in long simulation time.

If the mean of the input process ***X*** is increased to v_o, a new random process ***Y*** is created. The amount of change, $|v-v_o|$, is called the twisted amount (m^*). Let y denote the random sample/trace $y(t)$ of the input process ***Y***.

Assume the network is under the input process ***Y***, and there are more event occurrences. It will require less replication runs to collect sufficient number of event occurrences. This change of the input process mean value to make the events occur more frequently is called "twisting". However, we are still only interested in the event occurrences when network is under input process ***X***. This is where Importance Sampling comes in.

Importance Sampling theory [5]states that:

$$\hat{P}_{IS} = (\sum_{i=1}^{N} I_A(y_{(i)})L_{(i)})/N \qquad (2)$$

where Equation (2) is an alternative way to calculate the probability of event occurrences when the network is under input process ***X*** even though we have applied the input process ***Y*** instead. Equation (2) is also called the Importance Estimator.

In equation(2), $y_{(i)}$ is the trace of the input process ***Y*** on the i-th replication run. $L_{(i)}$ is the likelihood ratio calculated for the i-th replication run. At the end of the i-th replication run, if the event of our interest has occurred, $I_A(y_{(i)})$ is set to 1 , otherwise to 0. Because the event occurs more frequently now, we require less replication runs to calculate the probability of event occurrences when the network is under input process ***X***. This is how Importance Sampling reduces the simulation time of the rare event simulation.

$L_{(i)}$ is called the likelihood ratio since it is the ratio between the distributions of the two input processes, ***X*** and ***Y***. The calculation of $L_{(i)}$ will require the sample trace $x_{(i)}$ along with input process parameters (mean, variance, correlation coefficients, etc.). In this paper, we use the self-similar Fractal Gaussian Noise (FGN) process for ***X*** and ***Y***. The formula for the likelihood ratio of the FGN processes can be found in [1].

For traffic engineering applications, the input traffic model is typically given. For online congestion control applications, traffic parameters can be estimated based on captured traces.

The two estimators (Monte Carlo Estimator and Importance Estimator) are trying to find the sample means of two random variables, $I_A(x)$ and $I_A(y)L(y)$, respectively. To know the accuracies of the estimators, the variances of those samples means can be calculated.

For the Monte Carlo Estimator, the variance [7] is:

$$\sigma^2 = (\sum_{i=1}^{N}(I_A(x_{(i)}) - \hat{P}_{MC})^2)/N/(N-1) \qquad (7)$$

The normalized variance is [7] $\sigma^2 / \hat{P}_{MC}$.

For the Importance Estimator, the variance is:

$$\sigma^2 = \sum_{i=1}^{N}(I_A(x_{(i)})L_{(i)} - \hat{P}_{IS})^2/N/(N-1) \qquad (8)$$

The normalized variance is $\sigma^2 / \hat{P}_{IS}$.

A large normalized variance implies a great degree of fluctuations in the sample mean value, which indicates the result from the point estimator is "noisy" and unreliable. Normalized Variance will play an important role in the ISTE-AT technique.

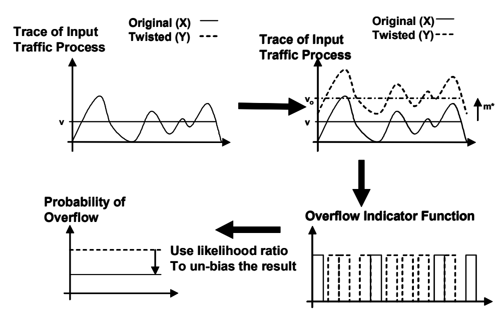

Figure 2. Importance Sampling in Rare Event Simulation Summary

Figure 2 summarizes the Importance Sampling operation. The twisted input process (***Y***) is applied into the network instead of the original input process (***X***). The event occurrence trace and the twisted input processes trace are recorded from the network. The captured traffic trace and the twisted amount are then used to calculate the likelihood ratio. The measured probability of event occurrences is adjusted using the likelihood ratio to find the probability of event occurrences when the network is under the original input process (***X***).

III. IMPORTANCE SAMPLING BASED TRAFFIC ENGINEERING

When Importance Sampling is applied in rare event simulation, we know the value of v (the mean value of the original input process ***X***) and we try to find a v_o (the mean value of the twisted input process ***Y***) that will make the event occur more frequent. ISTE is the "reverse" of this operation. We start with v_o and change the value of v.

Instead of treating the current input to the network as the original input process ($\boldsymbol{X}$), we treat it as the twisted input process ($\boldsymbol{Y}$). In another word, we assume the current input is the result of the twisting of some "original" process we do not know yet. Our goal is to create different "original" processes and predict the probabilities of overflow when these processes are applied to the network. In this paper, we let the probability of buffer overflow be the QoS of interest, thus, we let the event of buffer overflow be the event of our interest. The ingress traffic flows are treated as the network inputs; as a result, $\boldsymbol{Y}$ will produce more overflows in the network than $\boldsymbol{X}$ since v_o is greater than v.

The only thing different between the twisted input ($\boldsymbol{Y}$) and the original input ($\boldsymbol{X}$) is their mean value and the difference is the twisted amount (m^*). By trying different values of m^*, we can actually "create" a number of possible "original" processes.

Equation (2) calculates the probability of buffer overflow if the original input ($\boldsymbol{X}$) is applied. Because the current input to the network is treated as the twisted input process ($\boldsymbol{Y}$), $I_A(y_i)$ is the event occurrences currently in the network. These occurrences can be easily recorded, thus, the $I_A(y_i)$ term in (2) is fixed. N is the number of independent replication runs executed. It is also easily recorded and therefore the N term in (2) is also fixed. The only variable term left in (2) is $L_{(i)}$. The likelihood ratio is a ratio between the two distributions of $\boldsymbol{X}$ and $\boldsymbol{Y}$. The only thing different between the two distributions are their mean values, and the difference is the twisted amount (m^*). Therefore, the likelihood ratio is a function of the twisted amount, which in turn implies the probability of overflow, when the original input ($\boldsymbol{X}$) is applied, is also a function of the twisted amount.

As mentioned before, different values of twisted amount represent different original input processes ($\boldsymbol{X}$), each with a different mean value. Equation (2) is calculating the network probability of overflow when a traffic flow with a specific mean rate/bandwidth is applied to the network. This is how the ISTE approach provides a mapping of the bandwidth (ingress traffic mean rate) to the QoS (Probability of Overflow) of the network.

As we have shown, the ISTE approach does not require the internal details of the network. It only requires the capturing of the event occurrences currently in the network and the traces of the current input processes. This makes ISTE ideal for large and complex networks.

The event of interest in the ISTE approach could be a buffer overflow of a specific node, an end-to-end network path, or a VPN, etc. Thus, the ISTE approach could be used for end-to-end network QoS optimization as well as single network node performance analysis.

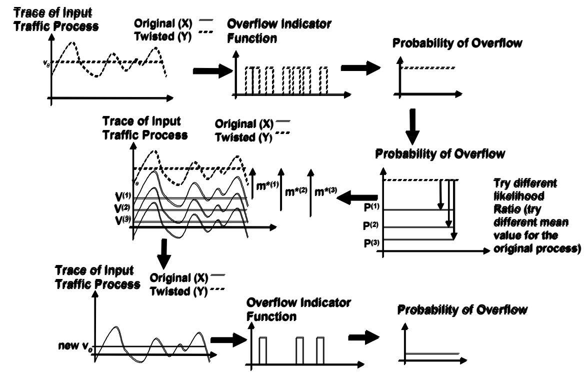

Figure 3. Importance Sampling Based Traffic Engineering Summary

Figure 3 is a graphical view of the ISTE approach. We start with the capturing of the event occurrences currently in the network and the trace of the current network input. By setting the twisted value (m^*) to 0, the Importance Estimator becomes a Monte Carlo Estimator [6] and gives the current probability of overflow in the network. We then try different values for the twisted amount ($m^{*(1)}$, $m^{*(2)}$, $m^{*(3)}$,…), thus we will calculate different probabilities of overflow ($P^{(1)}$,$P^{(2)}$,$P^{(3)}$,…) corresponding to network probabilities of overflow under different inputs each with different mean values ($v^{(1)}$, $v^{(2)}$, $v^{(3)}$,…). The Importance Sampling theory promises that when those input processes are applied to the network, the probabilities of overflow will be the probabilities we have calculated.

IV. SIMULATION

Figure 4 is the network topology used in verifying the effectiveness of the ISTE approach. It consists of a tandem queue, two traffic sources with traffic split after the first queue and traffic merge at the second queue.

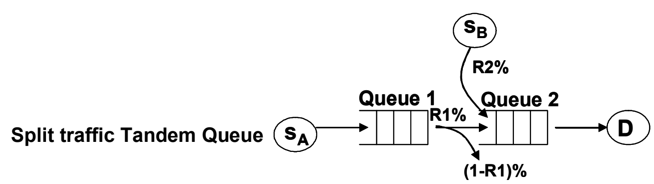

Figure 4. Network Topology

This topology may be simple but all the complex network topologies are some combination of this topology. Traffic Source A models the main ingress network traffic flow while source B models the background traffic flows from other sources in the network.

The Lindley equation [8] is used to model the queue behavior.

$$Q_k = (Q_{k-1} + X_k - \mu)^+ \tag{9}$$

Assume time slot based measurement is used, under the Lindley equation, the queue size depends on the difference between the instantaneous incoming traffic rate (X_k) at time slot k and the queue service rate (μ). At the end of each replication run, we check if the queue size has exceeded the buffer size. If the buffer size has been exceeded, that replication run is considered as a replication run with a buffer overflow event.

After the ISTE approach calculates the different probabilities of overflow if the mean rate of incoming traffic flow is reduced by different amounts, a simulation run is executed for each reduction and a Monte Carlo Estimator is used to measure the actual probability of overflow after the reduction has been made. The measured result is compared with the calculated result to verify the bandwidth to QoS mapping is correct.

Because the self-similar traffic model is used, each traffic source has a Hurst parameter. The Hurst parameter indicates the degree of self-similarity for traffic flows that exhibit long range dependency. The larger the Hurst parameter, the more bursty the traffic is. More bursty traffic source usually generates more faulty events in the network than non-bursty traffic source.

95% confidence intervals are plotted in the graphs. Here are some notations for reading the graphs for our simulation results:B1:Size of buffer in Queue 1 (units/sec). B2:Size of buffer in Queue 2 (units/sec). C1: Service rate in Queue 1 (units/sec). C2:Service rate in Queue 2 (units/sec).H1: Hurst parameter of traffic source A. H2:Hurst parameter of traffic source B. R1: Percentage of traffic entering Queue 2 from Queue 1. R2: Percentage of traffic entering Queue 2 from Source B. IS: ISTE, Probability of Overflow Prediction. MC: Monte Carlo Simulation, Actual Probability of Overflow.

As shown in Figure 5, the predictions made by ISTE are very close to the actual probabilities of overflows, when the bandwidth of the ingress traffic flow has been reduced by different amounts, measured by the Monte Carlo simulations. Therefore, ISTE's mapping of bandwidth (traffic source mean rate) to the network QoS (Probability of Overflow) is very accurate.

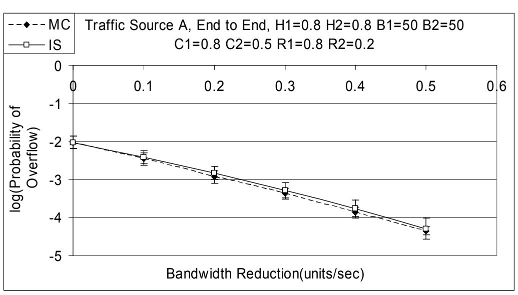

Figure 5. Probability of Buffer Overflow

The scenarios are configured such that there is congestion in both Q1 and Q2 at all time. As it can be seen, the ISTE approach is still effective. Many of the existing analytical approaches have to make the assumption that there is only a single static congested point in the network. This shows that ISTE is more practical than those other approaches. For more simulation results, see [9]

V. ALTERNATING TWISTING

While testing the ISTE approach, there were some scenarios in which the ISTE approach did not work well. They are the scenarios where the background source B generates so much traffic that the network overflows are caused by both source A and source B. This has made the prediction by twisting only source A more noisy, as shown in Figure 8.

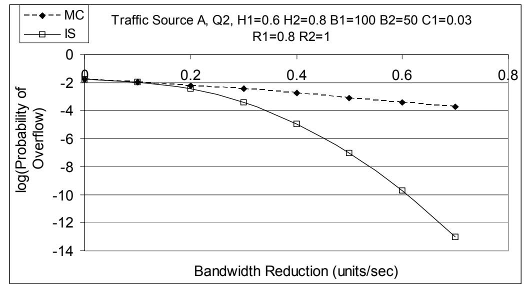

Figure 6. Overflow Due to Both Traffic Sources

This phenomenon is understandable because source A did not cause most of the overflows in the network in the first place. Importance Sampling does not help when the twisted process has little or no relationship with the event of interest.

A special technique, Alternating Twisting (ISTE-AT) is developed and ISTE-AT is able to solve this issue.

ISTE-AT is based on the following observation: When we twist the traffic source which is responsible for the events of the buffer overflows, its normalized variances increase much slower than the normalized variances increase when we twist the traffic source which is NOT responsible for the buffer overflow as shown in Figure 9.

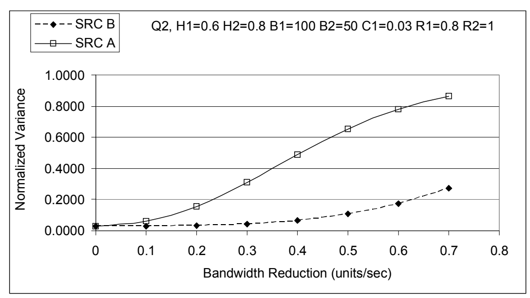

Figure 7. Normalized Variance Comparison

Therefore we can say that, when the normalized variance of the twisting becomes too large or too "noisy", it means that, after the mean rate of the source being reduced to that level, the source is no longer the dominating (responsible for the network overflow) source.

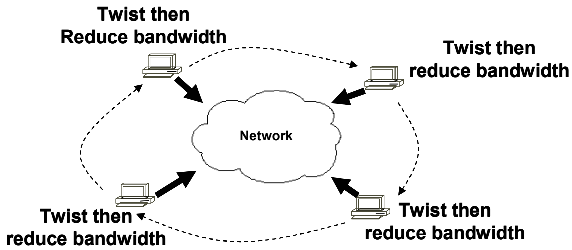

Figure 8. ISTE-Alternating Twisting

Figure 10 shows how ISTE-AT works. ISTE-AT first starts with one traffic source, and twists it to the point that it is no longer the dominating source. Then we actually reduce the mean rate (bandwidth) of that source to that level and rerun the simulation until the variance is acceptable again. We then select the next dominating source and apply twisting to this source. We repeat this process until we achieve our target probability of overflows.

In the case that there are large numbers of traffic sources in the network, the ISTE-AT is still scalable. The calculations can be taken place locally at each source or the ingress points of the network. The statistics of network traffic from each source can be easily collected and calculated locally. [9] shows the equation to calculate the likelihood ratio L_i can be easily implemented in a recursive manner. It consists of only normal mathematical operations such as add, multiplication, exponential, etc. There are total 7 add/subtraction operations, 3 division operation, 11 multiplication operation, and 2 exponential operation. Assume each operation takes 10 cycles, the total calculation for each traffic source will take 230 cycles. With processor at 1.7 GHz these days, with even 10,000 sources in the network, the total calculation will take less than 1 ms.

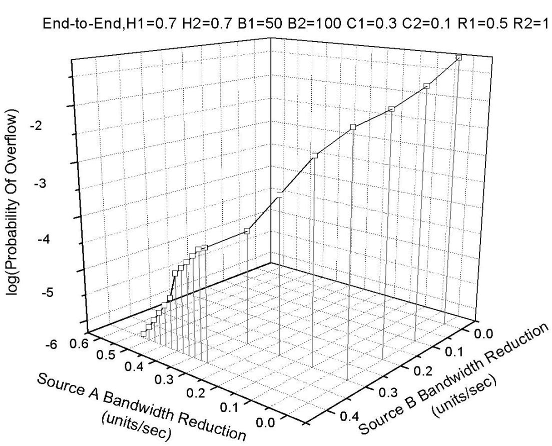

Figure 9. ISTE-AT Prediction on Probability of Overflow Trajectory

Figure 11 is the predictions made by ISTE-AT on how the probability of overflows will change as the bandwidths of the two traffic sources are reduced. It started with twisting source B, and then it switched to twisting source A, and then back to source B, etc. From the twisting results, if the mean rate of source A is reduced by 0.5 unit/sec and the mean rate of source B is reduced by 0.4 unit/sec, the logarithm scale end-to-end probability of overflow should be -5.9974. The measured result from the Monte Carlo Simulation shows the logarithm scale of the actual end-to-end probability of overflow, after these reductions are made, is -6.065. Since the predictions and the measured results are so close, it shows the IST-AT prediction is accurate.

VI. CONCLUSION

The ISTE approach proposed in this paper is simple, fast, and does not require intimate knowledge of the internals of a network. Thus, it is applicable in large and complex networks. From the simulation results of this paper, it is shown that ISTE can indeed offer a mapping from the bandwidth used by ingress traffic flows to the network QoS. The ISTE approach is effective in end-to-end performance analysis, as well as in single node performance analysis, under the self-similar traffic model. Even in scenarios where there are multiple congested nodes in the network, the ISTE prediction is still accurate. The simulation results also show that the ISTE-AT technique is effective when the network QoS is heavily influenced by more than one traffic flow.

REFERENCES

[1] C. Huang, “Long range dependent traffic: modeling, simulation, and congestion control“, Doctor of Philosophy thesis, Carleton University, Canada, 1997

[2] V. Paxson and S. Floyd. “Wide area traffic: The failure of Poisson modeling” IEEE/ACM Transactions on Networking, 3(3):226-244, 1995

[3] D. O. Awduche, A. Chiu, A. Elwalid, I. Widjaja, and X. Xiao, “Overview and principles of Internet traffic engineering”, IETF RFC 3272, 2002

[4] F. Haciomeroglu, “On-line measurement-based capacity allocation schemes”, Master of Science Thesis, North Carolina State University, 2003

[5] R. Y. Rubinstein, B. Melamed “Modern Simulation and Modeling” , John Wiley & Sons Inc, 1998, pp. 100-101

[6] G.S. Fishman, “Monte Carlo: concepts, algorithms, and applications”, Springer-Verlag, New York, NY, 1996.

[7] J. Banks, J. S. Carson, II, B. L. Nelson, D. M. Nicol, “Discrete Event System Simulation, Third Edition” , Prentice Hall, 2001, pp. 410-411

[8] J.W. Cohen. “The Single Server Queue”, North-Holland, 1982

[9] B. Feng, “Importance Sampling Based Internet Traffic Engineering Under Self Similar Traffic Model”, Master of Applied Science Thesis, Carleton University, 2005