Conditional Importance Sampling and its Application to ATM Switch Analysis

Ahmet A. Akyamaç
J. Keith Townsend
Center for Advanced Computing and Communication
Department of Electrical and Computer Engineering
North Carolina State University
Box 7914
Raleigh, N.C. 27695-7914

Abstract

In communication systems and networks, important performance measures include the error or fault probabilities related to the inability of the system to sustain a desirable operating state. These probabilities are typically low, in the range of 10^{-6} or less. In many cases, it is not possible to find analytical expressions for such probabilities. Monte Carlo simulation quickly becomes intractable due to the low probabilities involved, although Importance Sampling (IS) techniques have been used as a means of increasing simulation efficiency. Parametric IS methods are not very effective in cases where the input processes are characterized by uniform input distributions (e.g., random delays), which arise frequently in communication systems and networks. In this paper, we present a conditional IS scheme for systems with input processes that can be characterized by uniform input distributions. The scheme adaptively modifies an initial biasing strategy as samples are taken and also incorporates a problem specific component that enables the algorithm to be applied to a diverse set of problems. The overall approach is more effective than parametrically biasing the uniform input distributions. We use the conditional biasing algorithm to estimate rare jitter probabilities in ATM switches for CBR sources multiplexed with heterogeneous background VBR and CBR sources. For the experimental systems considered, we observe that the improvement in simulation efficiency is inversely proportional to the probability being estimated. [1]

[1]This work was supported by the Center for Advanced Computing and Communication, North Carolina State University.

1 Introduction

In communication systems and networks, some important performance measures are the error or fault probabilities related to the inability of the system to sustain a desirable operating state. Examples include bit error probabilities, cell loss probabilities and excessive delay and jitter probabilities. These probabilities are typically low, in the range of 10^{-6} or less. In many cases, it is not possible to find analytical expressions for such probabilities. Monte Carlo (MC) simulation quickly becomes intractable due to the low probabilities involved. One alternative is to use Importance Sampling (IS) as a means of improving simulation efficiency without loss of validity.

Importance Sampling simulation involves modifying or "biasing" the input distributions in a known way to cause more errors or faults in the system. The final result is properly weighted to obtain an unbiased estimate. Several methods of IS biasing have been used in the literature. Variance Scaling [1] and Mean Translation [2] are examples of methods which parametrically modify the input distributions. Large Deviations Theory has also been used to arrive at optimal biasing parameters [3] and generally involves numerical solutions. IS biasing methods have also been developed for Markov Chains [4, 5]. For complex systems for which an optimal bias cannot be found analytically, stochastic methods have been used to arrive at near-optimum bias points [6, 7]. However, the above methods which involve the modification of the parameters of the input distributions are not very effective when these distributions are uniform, which arise frequently in communication systems and networks.

In this paper, we present a conditional Importance Sampling scheme for systems with input processes characterized by uniform input distributions (e.g., random delays). After selecting an initial biasing strategy, this scheme employs a conditional biasing algorithm which adaptively modifies the biasing strategy as samples are taken, thus increasing efficiency and simulation speed. The overall scheme also employs a problem-dependent function to select the initial and modified biasing methods. This problem specific component allows the algorithm to be suitably adapted to the requirements of a diverse set of problems that are characterized by uniform input distributions. The combined approach is more effective than parametrically modifying the uniform distributions since it allows for noncontiguous sample sets. Contrary to the contiguous sets required for parametrically biasing the uniform distributions, the noncontiguous sets are smaller and target important events more efficiently.

To demonstrate the technique, we use conditional biasing to estimate rare jitter probabilities in ATM switches with constant bit rate (CBR) traffic multiplexed with heterogeneous background traffic which can consist of variable bit rate (VBR) traffic and other CBR traffic.

After presenting some assumptions and a general Importance Sampling problem definition, we introduce the conditional biasing algorithm. Subsequently, we present the application of the conditional biasing algorithm to estimate rare jitter probabilities in ATM switches. This paper, in effect, generalizes the IS scheme presented in [8], which was used to estimate rare jitter probabilities for the case of CBR traffic multiplexed with homogeneous background VBR traffic. The scheme presented here can be used to estimate rare jitter probabilities for the case of CBR traffic multiplexed with heterogeneous background traffic, which can consist of different types of VBR and CBR sources. The IS scheme presented here incorporates conditional biasing and is more efficient than the one used in [8]. For the experimental systems considered, for which the background traffic consists of different combinations of VBR and CBR traffic, the improvement in simulation efficiency (speedup over Monte Carlo simulation) is observed to be inversely proportional to the probability being estimated.

2 Assumptions

We assume that the system consists of generic discrete elements (typically inputs) θ. In the setting of communication systems and networks, these elements can refer to pulses, cells, bursts, customers etc. For every element θ, there is an associated random variable X, which is uniformly distributed over the sample set $\mathcal{S}$. We refer to X as the corresponding "entries" of θ. In the setting of communication systems and networks, X can typically refer to the arrival positions of the elements. The set $\mathcal{S}$ is discrete and finite, with cardinality $|\mathcal{S}|$. There is a norm $\|\cdot\|$ defined on $\mathcal{S}$, which induces a total ordering $'\prec'$ such that if $x, y \in \mathcal{S}$, then $x \prec y$ if and only if $\|x\| < \|y\|$. Furthermore, $\forall x, y \in \mathcal{S}$, $x \neq y$, either $x \prec y$ or $y \prec x$. The elements $x, y \in \mathcal{S}$ are realizations of X. We will consider $x, y \in \mathcal{S}, x \prec y$ to be contiguous if and only if $\nexists z \in \mathcal{S}$ such that $x \prec z \prec y$. A subset $\mathcal{T}$ of $\mathcal{S}$ is contiguous in $\mathcal{S}$ if and only if for any elements x, y contiguous in $\mathcal{T}$ and for which $x \prec y$, $\nexists z \in \mathcal{S} \setminus \mathcal{T}$ such that $x \prec z \prec y$.

The randomness in the system can then be described by the collection $< \theta, X >$. We consider the estimation of small probabilities (rare event probabilities) which indicate important events. For brevity, we use the term "stressed condition" to refer to the situation which occurs when the elements θ are forced into subsets that cause important events. Typically, the sets that result in the occurrence of important events are not known *a priori* and the elements are forced into larger subsets that are known to contain the important events. For example, in communication systems and networks, probabilities such as packet loss, high delay, high interference etc. are all due to a stressed condition of the system. This stressed condition can be typically described through the collection $< \theta, X >$ by a metric $d(\cdot)$ defined on $\mathcal{S}$.

3 Importance Sampling

For the given system, assume that the number of elements θ is fixed at N. Then, the randomness can be concisely represented by a vector $\underline{X}$ which corresponds to N different entries of the inputs θ. Let Ω be the sample space of all vectors $\underline{X}$. Since there are N elements which have entries uniformly distributed on the set $\mathcal{S}$, $\Omega = \mathcal{S}^N$ and the cardinality of Ω is given by $|\Omega| = |\mathcal{S}|^N$. The set Ω contains the realizations of $\underline{X}$ and represents all statistical outcomes of the system. Rare events correspond to small (possibly disjoint) subsets of Ω.

Assume we are interested in a function $g(\underline{X})$ which involves rare important events. Then, an important probability for the system is given by:

$$P = \frac{\mathrm{E}\{g(\underline{X})\}}{\mathrm{E}\{h(\underline{X})\}} \tag{1}$$

Usually, $h(\underline{X})$ does not correspond to rare events and is either deterministic or can be estimated by conventional Monte Carlo techniques. Furthermore, $h(\underline{X})$ can be estimated independently of $g(\underline{X})$ [9]. Let the subset of Ω that contains all important events be denoted by Ω_g.

For $g(\underline{X})$, we have the following:

$$\mathrm{E}\{g(\underline{X})\} = \sum_{\underline{x}\in\Omega} g(\underline{x}) f_{\underline{X}}(\underline{x}) \tag{2}$$

where $f_{\underline{X}}(\underline{x})$ is the pmf of $\underline{X}$ and Ω is the space of all vectors $\underline{x}$. Using Importance Sampling, we modify or bias the pmf $f_{\underline{X}}(\underline{x})$ to $f^*_{\underline{X}^*}(\underline{x}^*)$ and form the following:

$$\mathrm{E}^*\{g(\underline{X})\} = \sum_{\underline{x}^*\in\Omega^*} g(\underline{x}^*) f^*_{\underline{X}^*}(\underline{x}^*) w(\underline{x}^*) \tag{3}$$

where Ω^* is the space of all biased vectors $\underline{x}^*$ and

$$w(\underline{x}^*) = \frac{f_{\underline{X}}(\underline{x}^*)}{f^*_{\underline{X}^*}(\underline{x}^*)} \tag{4}$$

is the vector-dependent weight function. Note that $\mathrm{E}^*\{g(\underline{X})\} = \mathrm{E}\{g(\underline{X})\}$. Here, $\Omega^* \subset \Omega$ and $f^*_{\underline{X}^*}(\underline{x}^*) > 0$ whenever $f_{\underline{X}}(\underline{x}^*) > 0$ for (3) to be valid. This condition is equivalent to $\Omega^* \supset \Omega_g$.

The key to applying Importance Sampling is to determine the biased pmf and the weight function. In this paper, we present a conditional biasing scheme for the case where the vector $\underline{X}$ consists of independent and identically distributed uniform random variables on $\mathcal{S}$.

4 Conditional Biasing

Using Monte Carlo simulation, vectors are independently and uniformly sampled from the set Ω. Sampling a vector corresponds to selecting entries from the set $\mathcal{S}$ for each of the N elements. For each sampled vector $\underline{x}$, the indicator function $g(\underline{x})$ indicates whether or not the important event has occurred. Monte Carlo simulation terminates when a certain number of hits have been achieved, where a hit refers to a sample for which $g(\underline{x}) = 1$. Hence, Monte Carlo simulation results in the following estimator for $g(\underline{X})$:

$$\hat{\mathrm{E}}\{g(\underline{X})\} = \frac{1}{N_{MC}} \sum_{s=1}^{N_{MC}} g(\underline{x}_s) \tag{5}$$

where N_{MC} is the total number of Monte Carlo simulation runs and $\underline{x}_s$ is the vector sampled at run s. For practical systems of interest and for rare event probabilities, the space Ω is large compared to Ω_g and achieving a specific number of hits requires excessively many simulation runs. In fact, for an acceptable accuracy, the number of Monte Carlo simulation trials required is inversely proportional to the probability being estimated.

The set Ω_g, in its most general sense, is a collection of M (possibly disjoint) subsets of Ω as follows:

$$\Omega_g = \Omega_1 \cup \Omega_2 \cup \cdots \cup \Omega_M \tag{6}$$

Each of the subsets Ω_i contains realizations of $\underline{X}$ that cause rare events. Note that due to the rare nature of the important events, $|\Omega_g| \ll |\Omega|$. Monte Carlo simulation is intractable since it samples from the set Ω. The rare events correspond to a stressed condition of the elements θ. The significance of the stressed condition is problem-specific and can be characterized for each problem by a generic function $\alpha(\underline{X}, d(\cdot), \prec)$. The function $\alpha(\cdot)$ is an indicator function of a stressed condition for the realization $\underline{X}$, where the stressed condition is defined through the metric $d(\cdot)$ and the ordering $\prec$. The stressed conditions defined by $\alpha(\cdot)$ are necessary for rare events, but typically not every stressed condition causes a rare event. Thus, it is very difficult to analytically identify the sets Ω_i.

Using conditional IS biasing, we sample the individual components of $\underline{X}$ so as to force the elements θ into a stressed condition defined by $\alpha(\cdot)$. The biasing scheme is conditional in that the entry of element n is sampled depending on the entries of elements $1, \ldots, n-1$. The conditional biasing is performed so as to force the elements θ to remain in a stressed condition. Conditional IS results in a biased sample space Ω^* which contains all realizations of $\underline{X}$ for which $\alpha(\underline{X}, d(\cdot), \prec) = 1$ and hence contains the set Ω_g. Thus, we have that $\Omega_g \subset \Omega^* \subset \Omega$. A good IS scheme must satisfy $|\Omega^*| \ll |\Omega|$. Importance Sampling simulations result in the following estimator for $g(\underline{X})$:

$$\hat{\mathrm{E}}^*\{g(\underline{X})\} = \frac{1}{N_{IS}} \sum_{s=1}^{N_{IS}} g^*(\underline{x}_s^*) w(\underline{x}_s^*) \tag{7}$$

where $g^*(\cdot)$ is the indicator function under the biased pmf and N_{IS} is the number of IS runs. Here, $\underline{x}^*$ indicates a sample under the biased pmf.

For any given event, there is a minimum number of elements that must be present in the system for that event to occur. Let this minimum number of elements be denoted by N_{min}. Unless $N \geq N_{min}$, no important events occur.

Let $r = N - N_{min}$ denote the number of elements above the minimum required to potentially cause an important event. Hence, there are r degrees of freedom since these extra r elements do not necessarily have to contribute to an important event.

Then, the conditional biasing algorithm proceeds as follows:

Conditional Biasing Algorithm:

- Set the Importance Sampling weight to $w(\underline{x}^*) = 1$.
- Sample $r = N - N_{min}$ elements from the entire set $\mathcal{S}$. Since this is the same as Monte Carlo sampling, the IS weight remains unchanged. This step is required since for a

valid IS estimate, all vectors $\underline{x}$ that result in important events must be included in the biased distribution and there are r degrees of freedom. This step does not reduce the efficiency of IS, rather it is required to generate a valid IS scheme.

- Depending on the relative metric distances of the entries $x_0^* \ldots x_{r-1}^*$ and the function $\alpha(\cdot)$, determine subset $\mathcal{T}_r$ of $\mathcal{S}$ from which element r can be chosen and potentially cause an important event. This constitutes the basis of the conditional biasing scheme since the biased distribution for the entry of element r is chosen conditioned on the entries of the previously sampled elements. The criteria to generate the subset $\mathcal{T}_r$ is problem-dependent and should not omit any important events. In the conditional biasing scheme, $\mathcal{T}_r$ can also represent a noncontiguous set. All entries in $\mathcal{T}_r$ are equiprobable and element r is sampled uniformly over $\mathcal{T}_r$. This approach is more effective than parametrically biasing the uniform distribution since it allows for noncontiguous sets. If the uniform distributions were parametrically biased, we could only modify the parameters $[a, b]$ of the uniform distributions. Important events typically occur in noncontiguous sets and since biasing cannot exclude important events, the ranges $[a, b]$ must include all noncontiguous sets that cause important events and other nonimportant elements to make $[a, b]$ contiguous, thus reducing efficiency.

- If regular Monte Carlo simulation were used to sample for element r, the sampling would be performed from the entire set $\mathcal{S}$. Hence, the IS weight for element r is calculated as $w(x_r^*) = \frac{|\mathcal{T}_1|}{|\mathcal{S}|}$ and the overall weight is updated as follows: $w(\underline{x}^*) \leftarrow w(\underline{x}^*) \times \frac{|\mathcal{T}_1|}{|\mathcal{S}|}$.

- For elements l such that $r \leq l \leq N$:
  - Generate $\mathcal{T}_l$ from $\alpha(x_0^*, \ldots, x_{r-1}^*, x_r^*, \ldots, x_l^*, d(\cdot), \prec)$.
  - Sample element l uniformly from $\mathcal{T}_l$.
  - $w(\underline{x}^*) \leftarrow w(\underline{x}^*) \times \frac{|\mathcal{T}_l|}{|\mathcal{S}|}$.

Proposition: For the conditional biasing scheme discussed above, $w(\underline{x}^*) = f_{\underline{X}}(\underline{x})/f^*_{\underline{X}^*}(\underline{x}^*)$. Furthermore, for the IS estimator using the conditional biasing scheme, $\mathrm{E}\{\hat{\mathrm{E}}^*\{g(\underline{X})\}\} = \mathrm{E}\{g(\underline{X})\}$.
Proof: If Monte Carlo simulation were used, the vector $\underline{x}^*$ would be sampled uniformly from the set $\mathcal{S}$. Then, we would have:

$$f_{\underline{X}}(x_1^*, x_2^*, \ldots, x_N^*) = \frac{1}{|\mathcal{S}|^N} \tag{8}$$

To apply Importance Sampling, we use the conditional biasing scheme. Hence, the pmf under the IS scheme can be represented as follows:

$$f^*_{\underline{X}^*}(x_1^*, x_2^*, \ldots, x_N^*) = f^*_{\underline{X}^*}(x_1^*) f^*_{\underline{X}^*}(x_2^*|x_1^*) \ldots f^*_{\underline{X}^*}(x_N^*|x_1^*, x_2^*, \ldots, x_{N-1}^*) \tag{9}$$

But according to the conditional biasing scheme, the first $r = N - N_{min}$ elements are sampled uniformly from the set $\mathcal{S}$ and the remaining elements l for which $r+1 \leq l \leq N$ are sampled over subsets $\mathcal{T}_l$. Hence, the IS pmf can also be written as:

$$f^*_{\underline{X}^*}(x^*_1, x^*_2, \ldots, x^*_N) = \underbrace{\frac{1}{|\mathcal{S}|} \cdot \frac{1}{|\mathcal{S}|} \cdots \frac{1}{|\mathcal{S}|}}_{\text{r elements}} \cdot \frac{1}{|\mathcal{T}_r|} \cdot \frac{1}{|\mathcal{T}_{r+1}|} \cdots \frac{1}{|\mathcal{T}_N|} \tag{10}$$

By the definition of the IS weight and from (8) and (10), we have the following:

$$w(\underline{x}^*) = \frac{f_{\underline{X}}(\underline{x}^*)}{f^*_{\underline{X}^*}(\underline{x}^*)} = \underbrace{1 \cdot 1 \cdots 1}_{\text{r elements}} \cdot \frac{|\mathcal{T}_r|}{|\mathcal{S}|} \cdot \frac{|\mathcal{T}_{r+1}|}{|\mathcal{S}|} \cdots \frac{|\mathcal{T}_N|}{|\mathcal{S}|} \tag{11}$$

But, (11) is the same as the weight calculated in the conditional biasing scheme. Furthermore, due to the independence of the IS experiments and from (7), the expected value of the IS estimator is given by:

$$\mathrm{E}\{\hat{\mathrm{E}}^*\{g(\underline{X})\}\} = \mathrm{E}\{g^*(\underline{x}^*)w(\underline{x}^*)\} \tag{12}$$

Note that $g^*(\underline{x}^*)f^*_{\underline{X}^*}(\underline{x}^*) = g(\underline{x}^*)f_{\underline{X}}(\underline{x}^*)$. Hence, using (11), the expected value of the IS estimator can be rewritten as:

$$\mathrm{E}\{\hat{\mathrm{E}}^*\{g(\underline{X})\}\} = \mathrm{E}\{g(\underline{X})\} \tag{13}$$

Thus, we establish that the conditional biasing algorithm generates correct IS weights. Furthermore, the IS estimator using the conditional biasing scheme is unbiased.

5 Estimation of Jitter in ATM Switches

5.1 System Description

We consider an ATM switch, a model of which is shown in Fig. 1. The traffic into the switch consists of 53-byte ATM cells and is characterized by the ATM Forum standardized traffic descriptors consisting of the peak cell rate $\hat{\lambda}$ (Mbps), sustained cell rate $\bar{\lambda}$ (Mbps) and maximum burst length $\hat{B}$ cells at the peak rate [10]. The switch has N_P input ports and N_P output ports and the number of connections is $N_C = N_P$. The output ports contain buffers of infinite capacity. Each connection is routed uniformly and instantaneously to one of the output ports through a nonblocking shared bus, and we analyze a single output port (to which N_C connections are routed) to be indicative of the performance of all output buffers.

There are C classes of traffic. We consider the jitter for a "target" single-source CBR class which is defined by the triplet $(\hat{\lambda}_{\text{CBR}}, \bar{\lambda}_{\text{CBR}}, \hat{B}_{\text{CBR}})$. By convention, we assume the target CBR class to be class 1. Here, we define jitter by the ATM Forum standardized 1-Point Cell Delay Variation (CDV) definition [10]. We consider dispersion [10] as a measure of jitter. The target

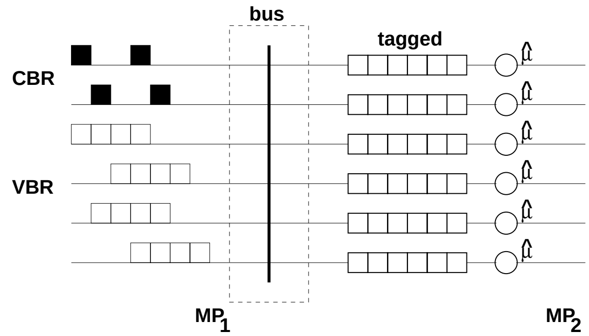

Figure 1: The ATM Switch Model.

CBR class is multiplexed with heterogeneous background traffic, which can consist of different VBR and CBR classes with triplets $(\hat{\lambda}_i, \bar{\lambda}_i, \hat{B}_i)$ for $2 \leq i \leq C$. Of the background traffic, we define the "dominant" VBR class as the VBR class for which $\hat{\lambda}_{\mathrm{VBR}} = \max_{2 \leq i \leq C} \left\{\hat{\lambda}_i\right\}$ and $\hat{B}_{\mathrm{VBR}} = \max_{\left\{i | \hat{\lambda}_i = \hat{\lambda}_{\mathrm{VBR}}\right\}} \left\{\hat{B}_i\right\}$ and we assume that there is a single dominant VBR class. For the dominant VBR class, the triplet is given by $(\hat{\lambda}_{\mathrm{VBR}}, \bar{\lambda}_{\mathrm{VBR}}, \hat{B}_{\mathrm{VBR}})$ and by convention, we assume the dominant VBR class to be class 2. The remaining background CBR and VBR classes are referred to as the nondominant classes.

Each class has N_i connections, thus $N_{\mathrm{CBR}} + N_{\mathrm{VBR}} + \sum_{i=3}^{C} N_i = N_C$. For classes $i \neq 2$, we assume that $\hat{\lambda}_{\mathrm{VBR}} = k_i \hat{\lambda}_i$, where $k_i \in \mathcal{Z}^+$ and $k_i \geq 2$. For the VBR classes, we make the worst case assumption that the traffic arrives according to the *greedy* pattern where in the ON period, a source from VBR class i generates traffic at the peak rate of $\hat{\lambda}_i$ for a burst duration of $\hat{B}_i$, and the OFF period is required to average out to the mean rate of $\bar{\lambda}_i$. Note that it is also required that $\sum_{i=1}^{C} \bar{\lambda}_i \cdot N_i < \hat{\mu}$.

For the simulation model, we normalize with respect to the peak rate of the dominant VBR source, $\hat{\lambda}_{\mathrm{VBR}}$. This results in a slotted-time system where one slot corresponds to the transmission of a single 53-byte ATM cell at the rate of $\hat{\lambda}_{\mathrm{VBR}}$. After normalization, the equivalent arrival rate of the dominant VBR source is 1 cell/slot and that of all other classes $i \neq 2$ is $\hat{\lambda}_i / \hat{\lambda}_{\mathrm{VBR}} = 1/k_i$ cells/slot. All classes i, $1 \leq i \leq C$ are periodic with the periods given by:

$$
\begin{aligned}
T_{\mathrm{VBR}} &= \frac{\hat{\lambda}_{\mathrm{VBR}} \hat{B}_{\mathrm{VBR}}}{\bar{\lambda}_{\mathrm{VBR}}} \\
T_{\mathrm{CBR}} &= \frac{\hat{\lambda}_{\mathrm{VBR}} \hat{B}_{\mathrm{CBR}}}{\bar{\lambda}_{\mathrm{CBR}}} = \frac{k_1 \hat{\lambda}_{\mathrm{CBR}}}{\bar{\lambda}_{\mathrm{CBR}}} = k_1 \\
T_i, 3 \leq i \leq C &= \frac{\hat{\lambda}_{\mathrm{VBR}} \hat{B}_i}{\bar{\lambda}_i} = \frac{k_i \hat{\lambda}_i}{\bar{\lambda}_i} \qquad (14)
\end{aligned}
$$

Hence, the entire input traffic has a super-period of $T_{\mathrm{SP}} = \operatorname{lcm}_{1 \leq i \leq C} \{T_i\}$. We assume

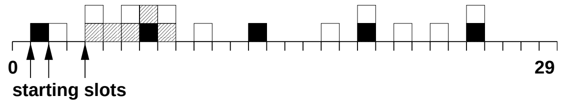

Figure 2: Slotted-time example pattern for CBR/VBR traffic, one period.

T_{VBR} to be integer multiples of k_i, $1 \leq i \leq C$, $i \neq 2$ so that $T_{\text{VBR}} = T_{\text{SP}}$. After normalization, the equivalent service rate is given by $\mu = \hat{\mu}/\hat{\lambda}_{\text{VBR}}$ cells/slot. An example of one period of the resulting slotted-time pattern is plotted in Fig. 2 for $\hat{\lambda}_{\text{VBR}} = 6$ cells/slot, $\bar{\lambda}_{\text{VBR}} = 1$ cell/slot, $\hat{B}_{\text{VBR}} = 5$ cells (plaid), $\bar{\lambda}_{\text{CBR}} = \hat{\lambda}_{\text{CBR}} = 1$ cell/slot and $\hat{B}_{\text{CBR}} = 1$ cell (black) and $\hat{\lambda}_3 = 3$ cells/slot, $\bar{\lambda}_3 = 2$ cells/slot and $\hat{B}_3 = 5$ cells (white), where the target CBR connection starts at slot 1, the dominant VBR connection starts at slot 4 and the nondominant VBR connection starts at slot 2.

The random component of this model is found in the connection starting-slots. Hence, the elements θ are the connections and the entries $\underline{X}$ correspond to the starting-slots of the connections. In a vector $\underline{X}$, the first entry corresponds to the starting-slot of the target CBR connection, the next N_{VBR} entries correspond to the starting-slots of the dominant VBR connections and the remaining entries correspond to the starting-slots of the other classes. Thus, we write:

$$\underline{x} = [c_1, v_1, v_2, \ldots, v_{N_{\text{VBR}}}, x_1^3 \ldots, x_{N_3}^3, \ldots, x_1^C, \ldots, x_{N_C}^C] \tag{15}$$

where c refers to the target CBR class, v refers to the dominant VBR class and x^i refers to class i for $3 \leq i \leq C$. Once the starting-slots have been selected, the system behaves deterministically and periodically. For this case, the sample spaces for the entries of each class are given by:

$$\begin{aligned} \mathcal{S}_1 &= [0, \ldots, T_{\text{CBR}} - 1] \\ \mathcal{S}_2 &= [0, \ldots, T_{\text{VBR}} - 1] \\ \mathcal{S}_i, 3 \leq i \leq C &= [0, \ldots, T_i - 1] \end{aligned} \tag{16}$$

The sample space for $\underline{X}$ is then given by:

$$\begin{aligned} \Omega &= \mathcal{S}_1^{N_{\text{CBR}}} \times \mathcal{S}_2^{N_{\text{VBR}}} \times \mathcal{S}_3^{N_3} \times \cdots \times \mathcal{S}_C^{N_C} \\ &= \Omega_1 \times \Omega_2 \times \Omega_3 \times \cdots \times \Omega_C \end{aligned} \tag{17}$$

where Ω_1 and Ω_2 correspond to the independent sample spaces for the CBR and VBR starting-slots, respectively and Ω_i, $3 \leq i \leq C$ correspond to the independent sample spaces for the remaining classes. Note that $|\Omega| = (T_{\text{CBR}})^{N_{\text{CBR}}} \cdot (T_{\text{VBR}})^{N_{\text{VBR}}} \cdot (T_3)^{N_3} \cdots (T_C)^{N_C}$.

In a superperiod T_{SP}, the total number of target CBR pairs is given by:

$$N_{\text{pairs}} = \frac{T_{\text{SP}}}{T_{\text{CBR}}} \tag{18}$$

For a given set of system parameters and for a given dispersion value δ, the maximum number of target CBR pairs that can be dispersed is given by J_{max}. Since there is only one target CBR source and due to the periodicity of the system, we can assume, without loss of generality, that $c_1 = 0$ in (15). As in [8], let n_{DP_j} be the number of vectors that result in exactly j pairs that are dispersed by more than δ in steady state. Then, the dispersion probability is given by:

$$P_{DP} = \sum_{j=0}^{J_{max}} \left(\frac{j}{N_{\text{pairs}}} \right) p_{DP_j} \tag{19}$$

where p_{DP_j} is the probability that exactly j pairs are dispersed by more than δ in steady state and is given by $p_{DP_j} = n_{DP_j}/|\Omega|$. We rewrite (19) as follows:

$$P_{DP} = \frac{\mathrm{E}\{g(\underline{X})\}}{\mathrm{E}\{h(\underline{X})\}} = \frac{\sum_{j=0}^{J_{max}} j p_{DP_j}}{N_{\text{pairs}}} \tag{20}$$

Note that (20) is in the form of (1), where $\mathrm{E}\{h(\underline{X})\} = N_{\text{pairs}}$. Thus, we only estimate $\mathrm{E}\{g(\underline{X})\}$. Here, $\mathrm{E}\{g(\underline{X})\}$ is similar to the multinomial formulation found in [8] where the events fall in categories or "bins" 0 to J_{max} depending on the number of pairs dispersed. The exhaustive solution to (19) or (20) is intractable due to the large size of the set Ω.

5.2 Monte Carlo Simulation

Using Monte Carlo simulation, we can form an unbiased estimate of $\mathrm{E}\{g(\underline{X})\}$ as follows:

$$\hat{\mathrm{E}}\{g(\underline{X})\} = \sum_{j=0}^{J_{max}} j \hat{p}_{DP_j} \tag{21}$$

as in (5), where $\hat{p}_{DP_j}$ are the estimates of the individual probabilities p_{DP_j} as in [8] and are given by:

$$\hat{p}_{DP_j} = \frac{1}{N_{MC}} \sum_{s=1}^{N_{MC}} g_j(\underline{x}_s) \tag{22}$$

where N_{MC} is the total number of Monte Carlo simulation runs and $g_j(\underline{x}_s)$ is an indicator for j cells being dispersed by more than δ for sampled vector $\underline{x}_s$. The variance for the Monte Carlo simulation is estimated as follows [8]:

$$\hat{\sigma}^2(\hat{\mathrm{E}}\{g(\underline{X})\}) = \sum_{j=0}^{J_{max}} j^2 \hat{\sigma}^2(\hat{p}_{DP_j}) \tag{23}$$

where the variance estimates for the individual bins are given by:

$$\hat{\sigma}^2(\hat{p}_{DP_j}) = \frac{1}{N_{MC}(N_{MC}-1)} \sum_{s=1}^{N_{MC}} (g_j(\underline{x}_s) - \hat{p}_{DP_j})^2 \tag{24}$$

Note here the estimate of P_{DP} is given by

$$\hat{P}_{DP} = \frac{\hat{\mathrm{E}}\{g(\underline{X})\}}{N_{\text{pairs}}} \tag{25}$$

and that the estimator variance of P_{DP} is given by

$$\hat{\sigma}^2(\hat{P}_{DP}) = \frac{\hat{\sigma}^2(\hat{\mathrm{E}}\{g(\underline{X})\})}{N^2_{\text{pairs}}} \tag{26}$$

5.3 Importance Sampling Using Conditional Biasing

In ATM networks, the probability of large dispersion is expected to be very low, typically on the order of 10^{-6} and below. Monte Carlo simulation is not feasible at these probabilities. Hence, we use Importance Sampling (IS) to increase the efficiency of the simulations.

The set of vectors that result in at least one pair being dispersed by more then δ is given by Ω_g. For realistic systems, $|\Omega_g| \ll |\Omega|$. We do not know Ω_g *a priori* and hence sample from a set Ω^* as in Section 4, for which $\Omega^* \supset \Omega_g$ and for which $|\Omega^*| \ll |\Omega|$. We form the IS estimate as in (21) as follows:

$$\hat{\mathrm{E}}^*\{g(\underline{X})\} = \sum_{j=0}^{J_{max}} j\hat{p}^*_{DP_j} \tag{27}$$

and the individual estimates $\hat{p}^*_{DP_j}$ are found as follows:

$$\hat{p}^*_{DP_j} = \frac{1}{n_j}\sum_{s=1}^{n_j} g^*_j(\underline{x}^*_s)w^*_j(\underline{x}^*_s) \tag{28}$$

where $g^*_j(\underline{x}^*_s)$ is the indicator function for event $g(\cdot)$ under the biased pdf and $w^*_j(\underline{x}^*_s)$ is the IS weight function for the vector $\underline{x}^*_s$.

The variance of the IS estimate is found as in (23), where the individual estimates $\hat{\sigma}^2(\hat{p}^*_{DP_j})$ are calculated as follows:

$$\hat{\sigma}^2(\hat{p}^*_{DP_j}) = \frac{1}{n_j(n_j-1)}\sum_{s=1}^{n_j}(g^*_j(\underline{x}^*_s)w^*_j(\underline{x}^*_s) - \hat{p}^*_{DP_j})^2 \tag{29}$$

where n_j is the number of IS runs for bin j. The confidence intervals are generated as in [8] using the result that the confidence interval for a weighted sum of individual multinomial probabilities follows a χ^2 distribution [11]. The improvement factors (speedup over Monte Carlo simulation) are calculated as in [8] as follows:

$$\hat{R}_{net} = \frac{\sum_{j=0}^{J_{max}} j^2\hat{p}^*_{DP_j}(1-\hat{p}^*_{DP_j})/n_j}{\sum_{j=0}^{J_{max}} j^2\hat{\sigma}^2(\hat{p}^*_{DP_j})} \tag{30}$$

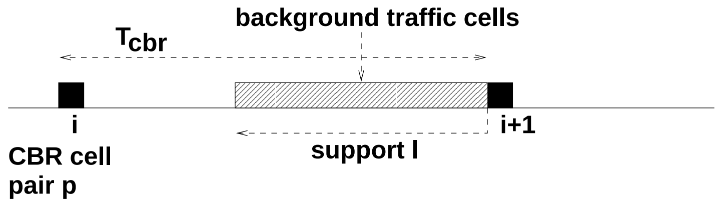

Figure 3: An illustration of supports for a CBR pair.

5.4 Importance Sampling Method

Dispersion occurs when the difference between the departure times of two consecutive target CBR cells (one target CBR pair), $D_{i+1} - D_i$, exceeds the specified jitter δ. The dispersion is caused by the arrival of background cells between the two target CBR cells.

In [8], the background traffic was composed of homogeneous VBR sources. It was determined that dispersion occurred due to the arrival of VBR cells in certain "supports", which indicate the time slot after which the VBR cells affect the dispersion of the CBR cells. The support structure for a given pair of target CBR cells i and $i+1$ is shown in Fig. 3. Due to the bursty nature of the VBR cells, the supports for a single pair were defined in the range $1 \leq s \leq T_{\text{CBR}} + \hat{B}_{\text{VBR}} - 1$. For each support s, the following parameters were defined in [8], which were essential to the biasing procedure:

- $V_{min}(s)$, the minimum number of cells required in the support s to cause CBR cells i and $i+1$ to be dispersed.
- $V_{max}(s)$, the maximum number of cells (from the homogeneous VBR sources) that can arrive in support s. Unless $V_{max}(s) \geq V_{min}(s)$, dispersion is not possible for support s.
- $N_{min}(s)$, the minimum number of VBR bursts necessary in support s to cause dispersion. This is calculated as $N_{min}(s) = \left\lceil V_{min}(s) / \min\{s, \hat{B}_{\text{VBR}}\} \right\rceil$.
- $V_{\text{extra}}(s)$, the "extra" number of VBR cells required to cause dispersion for support s. By definition, for support s to cause dispersion, a VBR burst must be present at the first slot of the support. $V_{\text{extra}}(s)$ specifies the extra number of cells necessary in support s and is given by $V_{\text{extra}}(s) = V_{min}(s) - \min\{s, \hat{B}_{\text{VBR}}\}$.
- $r(s)$, the number of additional VBR bursts present in the system above the minimum required, where $r(s) = N_{\text{VBR}}(s) - N_{min}(s)$.
- $f_0(s), f_1(s), \ldots, f_r(s)$, which are the distances (as described in [8]) within which the VBR bursts must arrive to cause dispersion. In the support s, there have to be at least $N_{min}(s)$ bursts arriving within $f_0(s)$ slots of each other or $N_{min}(s)+1$ bursts arriving within $f_1(s)$ slots of each other etc. for dispersion to occur.
- $\mathcal{F}(s)$, a flag indicating the presence of a VBR burst at the first slot of support s. Note that $\mathcal{F}(s) = 1$ is required for support s to cause dispersion.

Using the above parameters for all supports s and all pairs p, $1 \leq p \leq N_{\text{pairs}}$, the IS algorithm developed in [8] was used to sample starting-slot vectors and generate IS weight factors. Note that in the case of homogeneous background VBR sources, the support structure is identical for all CBR pairs p.

In this paper, the background traffic can consist of a number of VBR and CBR sources, which include the dominant VBR source. We will only bias the dominant VBR connection and sample the remaining connections uniformly over their respective sample periods, since the dominant VBR source is the most effective source that contributes to the dispersion of target CBR cell pairs.

Define the norm $||\cdot||$ on $\mathcal{S}_2$ such that for $x \in \mathcal{S}_2$, $||x|| = x$. The ordering $\prec$ is defined such that $x \prec y$ if and only if $x < y$. Here, we define the metric $d(x,y) = \min\{|x-y|, T_{\text{SP}}-|x-y|\}$. This metric compensates for the inherent periodicity of the system. If Monte Carlo simulation were used, the pdf for vector $\underline{x}^*$ would be given by:

$$f_{\underline{X}}(\underline{x}^*) = \left[\frac{1}{T_{\text{CBR}}}\right]^1 \cdot \left[\frac{1}{T_{\text{VBR}}}\right]^{N_{\text{VBR}}} \cdot \prod_{i=3}^{C} \left[\frac{1}{T_i}\right]^{N_i} \tag{31}$$

The IS biasing procedure results in a biased pdf $f^*_{\underline{X}^*}(\underline{x}^*)$.

The conditional biasing algorithm essentially uses the same parameters defined in the above parameter list. However, the parameters are defined separately for each pair p and support s. The algorithm initially samples all nondominant connections $3 \leq i \leq C$ over their respective sample spaces $\mathcal{S}_i$. Subsequently, the algorithm calculates the following parameters:

- $\hat{V}(s,p)$, the number of nondominant cells that are present in support s for pair p.
- $\Lambda(s,p)$, a flag indicating whether or not there are nondominant cells present at the first slot of support s at pair p.

The parameters for the conditional biasing algorithm are then computed as follows:

- $V_{min}(s,p)$: Compute $V_{min}(s)$ as in [8]. Since the nondominant cells present in the support contribute to the minimum number of cells required, $V_{min}(s,p) = V_{min}(s) - \hat{V}(s,p)$.
- $V_{max}(s,p)$: Here, $V_{max}(s,p)$ is the maximum number of dominant VBR cells that can arrive at support s for pair p. Thus, $V_{max}(s,p) = N_{\text{VBR}} \cdot \min\{s, \hat{B}_{\text{VBR}}\}$.
- $N_{min}(s,p)$: Here, $N_{min}(s,p)$ is the minimum number of dominant VBR bursts required at support s for pair p to cause dispersion. It is computed as $N_{min}(s,p) = \left\lceil V_{min}(s,p)/\min\{s, \hat{B}_{\text{VBR}}\}\right\rceil$.
- $V_{\text{extra}}(s,p)$: This is computed for the dominant VBR source as before, $V_{\text{extra}}(s,p) = V_{min}(s,p) - \min\{s, \hat{B}_{\text{VBR}}\}$.
- $r(s,p)$: This is also computed as before, $r(s,p) = N_{\text{VBR}}(s,p) - N_{min}(s,p)$.

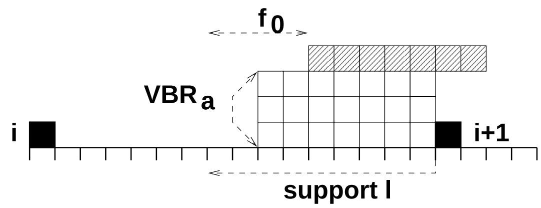

Figure 4: Calculation of the distance f_0 for $V_{min}(s,p) > 0$ and $\Lambda(s,p) = 1$.

- $\mathcal{F}(s,p)$: Defined similarly as above.

The overall minimum number of dominant VBR sources required to cause dispersion is then given by $N_{min}^{over} = \min_s \min_p\{N_{min}(s,p)\}$. Hence, the overall number of extra dominant VBR sources present is given by $r^{over} = N_{\mathrm{VBR}} - N_{min}^{over}$. For each support s and pair p, the distances $f_i(s,p)$ depend on $V_{\mathrm{extra}}(s,p)$ and $V_{min}(s,p)$ and are computed according to one of the following categories:

- If $V_{min}(s,p) \leq 0$, this means that the cells of the nondominant connections present at support s for pair p are sufficient to cause the desired target dispersion δ. This means that all dominant VBR connections must be sampled over $\mathcal{S}_2$.

- If $V_{\mathrm{extra}}(s,p) \leq 0$, a dominant burst at support s for pair p provides at least the necessary number of cells required to cause the desired target dispersion. Set $\mathcal{F}(s,p) = 1$. Since $V_{\mathrm{extra}}(s,p) \leq 0$, we have $N_{min}(s,p) = 1$ and $r(s,p) = N_{\mathrm{VBR}} - 1$. Calculate $f_0(s,p) = s - (\min\{s, \hat{B}_{\mathrm{VBR}}\} - |V_{\mathrm{extra}}(s,p)|)$ and calculate $f_1(s,p), \ldots, f_{r(s,p)}(s,p)$ according to the "normal" calculation described below.

- If $V_{min}(s,p) > 0$ and $\Lambda(s,p) = 1$, then the support s for pair p is defined by the presence of nondominant cells at the first slot. Hence, a VBR burst is not required at the first slot. Set $\mathcal{F}(s,p) = 1$. Define $V_a(s,p)$ as the maximum number of aligned dominant VBR bursts that can contribute to the target dispersion. The calculation of V_a for this case is illustrated in Fig. 4.

  We calculate $V_a(s,p) = \left\lfloor \frac{V_{min}(s,p)}{\min\{s,\hat{B}_{\mathrm{VBR}}\}} \right\rfloor$ resulting in

$$f_0(s,p) = s - \left\lfloor \frac{V_{min}(s,p) - V_a(s,p)\min\{s, \hat{B}_{\mathrm{VBR}}\}}{N_{min}(s,p) - V_a(s,p)} \right\rfloor \tag{32}$$

  The distances $f_1(s,p), \ldots, f_{r(s,p)}(s,p)$ are calculated according to the normal procedure in [8]:

$$f_{r,r>0}(s,p) = \begin{cases} s & \text{if } f_0(s,p) \geq s - r \\ s-1 & \text{otherwise} \end{cases} \tag{33}$$

- If $V_{min}(s,p) > 0$ and $\Lambda(s,p) = 0$, a VBR burst is required at the first slot. For this case, the calculation of $V_a(s,p)$ is illustrated in Fig. 5 and results in $V_a(s,p) = \lfloor V_{\mathrm{extra}}(s,p)/\min\{s, \hat{B}_{\mathrm{VBR}}\}\rfloor$.

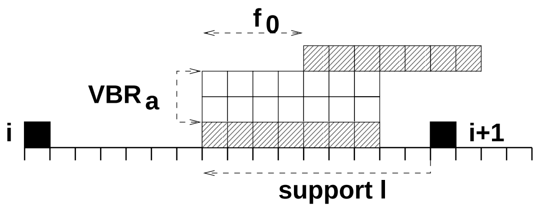

Figure 5: Calculation of the distance f_0 for $V_{min}(s,p) > 0$ and $\Lambda(s,p) = 0$.

Thus, as in [8], we obtain:

$$f_0(s,p) = s - \left\lfloor \frac{V_{\text{extra}}(s,p) - V_a(s,p)\min\{s, \hat{B}_{\text{VBR}}\}}{N_{min}(s,p) - 1 - V_a(s,p)} \right\rfloor \tag{34}$$

The distances $f_1(s,p), \ldots, f_{r(s,p)}(s,p)$ are calculated according to the normal procedure given above.

The conditional biasing scheme samples r^{over} VBR starting-slots $v_1^*, \ldots, v_{r^{over}}^*$. For connection l, $r^{over} + 1 \le l \le N_{\text{VBR}}$, v_l^* is sampled conditioned on the previous samples of all nondominant classes and the dominant VBR class using the following stressed condition indicator function:

For $0 \le i \le r(s,p)$;

$$\alpha_{s,p,i}(x_1^3, \ldots, x_{N_C}^C, v_1^*, \ldots, v_{l-1}^*, d(\cdot), \prec) = \begin{cases} 1 & \text{if } \exists \text{ a } \mathcal{G}(N_{min}(s,p)+i) \subset \{v_1^*, \ldots, v_{l-1}^*\} \ni \\ & d\left(\max\limits_{\prec}\{v \in \mathcal{G}\}, \min\limits_{\prec}\{v \in \mathcal{G}\}\right) \le f_i(s,p) \\ 0 & \text{otherwise} \end{cases}$$

$$\alpha(x_1^3, \ldots, x_{N_C}^C, v_1^*, \ldots, v_{l-1}^*, d(\cdot), \prec) = \bigvee_{s=1}^{T_{\text{CBR}}+\hat{B}_{\text{VBR}}-1} \bigvee_{p=1}^{N_{\text{pairs}}} \bigvee_{i=0}^{r(s,p)} \alpha_{s,p,i} \tag{35}$$

where a group $\mathcal{G}(N_{min}(s,p)+i)$ is a set of any $N_{min}(s,p)+i$ elements, $\max\limits_{\prec}$ and $\min\limits_{\prec}$ denote the maximum and minimum under the ordering $\prec$, respectively, and $\bigvee$ denotes the logical *or* operator. In essence, the above stressed condition indicator function $\alpha(\cdot)$ returns a 1 if the conditions for the target dispersion have been satisfied and returns a 0 otherwise. If the stressed condition indicator function $\alpha(\cdot)$ returns a 1, all remaining VBR bursts are sampled over $\mathcal{S}_2$.

Depending on the connections sampled previously, connection v_l^* may not need to be sampled from a restricted set. If the remaining $N_{\text{VBR}} - l$ connections are sufficient to cause a dispersion greater than the target value, then v_l^* should be sampled over $\mathcal{S}_2$ for the IS scheme to be valid. This condition is checked using the following indicator function:

$$\text{For } 0 \leq i \leq r(s,p) \; ;$$
$$\beta_{s,p,i}(x_1^3, \ldots, x_{N_C}^C, v_1^*, \ldots, v_{l-1}^*, d(\cdot), \prec) = \begin{cases} 1 & \text{if } \mathcal{B}(f_i(s,p)) + N_{\text{VBR}} - l \geq N_{min}(s,p) + i + 1 \\ 0 & \text{otherwise} \end{cases}$$

$$\beta(x_1^3, \ldots, x_{N_C}^C, v_1^*, \ldots, v_{l-1}^*, d(\cdot), \prec) = \bigvee_{s=1}^{T_{\text{CBR}}+\hat{B}_{\text{VBR}}-1} \bigvee_{p=1}^{N_{\text{pairs}}} \bigvee_{i=0}^{r(s,p)} \beta_{s,p,i} \tag{36}$$

where $\mathcal{B}(f_i(s,p))$ denotes the number of bursts within a distance of $f_i(s,p)$ of support s for pair p. In essence, the above indicator function $\beta(\cdot)$ returns a 1 if the next burst is not necessarily required to cause dispersion beyond the target value since there are sufficiently many bursts remaining to be sampled. The indicator function $\beta(\cdot)$ is not related to the stressed condition indicator function $\alpha(\cdot)$, but also contributes to the conditional nature of the algorithm.

After the parameters for each support s and pair p are calculated, the set $\mathcal{T}(s,p)$ from which the next dominant VBR burst l is to be sampled for support s and pair p is given by:

$$\mathcal{T}_l(s,p) = \begin{cases} [0, T_{\text{SP}}] & \text{if } \alpha(\cdot) = 1 \text{ or } \beta(\cdot) = 1 \\ \left[T_{\text{CBR}} - s + (p-1)T_{\text{CBR}}, T_{\text{CBR}} - s + (p-1)T_{\text{CBR}} + f_{r(s,p)}(s,p)\right] & \\ \qquad \text{if } \alpha(\cdot) = 0,\ \beta(\cdot) = 0 \text{ and } V_{max}(s,p) \geq V_{min}(s,p) & \\ \emptyset & \text{otherwise} \end{cases} \tag{37}$$

Then, the overall region from which the next dominant VBR burst l is to be sampled is given by:

$$\mathcal{T}_l = \bigcup_{s=1}^{T_{\text{CBR}}+\hat{B}_{\text{VBR}}-1} \bigcup_{p=1}^{N_{\text{pairs}}} \mathcal{T}_l(s,p) \tag{38}$$

Note that the set $\mathcal{T}_l$ need not be contiguous. The IS weight for VBR burst l is then calculated as follows:

$$w^*(v_l^*) = \frac{|\mathcal{T}|}{|\mathcal{S}_2|} \tag{39}$$

A pseudocode of the Importance Sampling Algorithm that incorporates Conditional Biasing follows:

<u>*simulation_procedure:*</u>

- $w = 1$. `/* initialize IS weight */`
- `unmark MC.` `/* initialize Monte Carlo flag */`
- `for (`s`=1 ; `s` <= `$T_{\text{CBR}} + \hat{B}_{\text{VBR}} - 1$` ; `$s$`++)`
  - `calculate `$V_{min}(s)$. `/* initial cells required */`
- `for (`i`=3 ; `i` <= `C` ; `i`++)`

- sample $v^i_1, \ldots, v^i_{N_i}$ uniformly over $\mathcal{S}_i$. /* nondominant connections */
- $v_i = (v^i_1, \ldots, v^i_{N_i})$.

- for (p=1 ; p <= N_{pairs} ; p++) /* all pairs */
  - for (s=1 ; s <= $T_{\text{CBR}} + \hat{B}_{\text{VBR}} - 1$; s++) /* all supports */
    - calculate $\hat{V}(s,p)$, $\Lambda(s,p)$. /* from nondominant connections */
    - $V_{min}(s,p) \leftarrow V_{min}(s) - \hat{V}(s,p)$. /* reduction in cells required */
    - if $V_{min}(s,p) < 0$,
      - mark MC. /* set the MC flag */
    - calculate $N_{min}(s,p)$, $r(s,p)$. /* biasing parameters */
    - calculate $f_0(s,p), \ldots, f_{r(s,p)}(s,p)$.
- $r^{over} \leftarrow N_{\text{VBR}} - \min_s \min_p \{N_{min}(s,p)\}$. /* degrees of freedom */
- if ($r^{over} < 0$) exit. /* not enough VBR connections */
- $v_2 = (\texttt{zeros}(N_{\text{VBR}}))$. /* initialize dominant VBR slots */
- if MC, /* if MC, sample all bursts over $\mathcal{S}_2$ */
  - for (l=1 ; l <= N_{VBR} ; l++)
    - sample $v_2(l)$ uniformly over $\mathcal{S}_2$.
  - return $(0, v_2, v_3, \ldots, v_C)$, $w = 1$. /* return IS weight of 1 */
  - exit.
- else, /* if not MC, continue with algorithm */
  - for (l=1 ; l <= r^{over} ; l++) /* the first r^{over} bursts are MC */
    - sample $v_2(l)$ uniformly over $\mathcal{S}_2$.
  - call *update_procedure* $(v_2, \mathcal{T}_l)$. /* generates sample set */
  - for (l=$r^{over}+1$; l <= N_{VBR} ; l++) /* sample other bursts */
    - sample $v_2(l)$ uniformly over $\mathcal{T}_l$.
    - $w \leftarrow w \times |\mathcal{T}_l| / |\mathcal{S}_2|$.
    - call *update_procedure* $(v_2, \mathcal{T}_l)$. /* update after every burst */
  - return $(0, v_2, v_3, \ldots, v_C)$, w. /* return the vector and IS weight */
  - exit.

update_procedure$(v_2, \mathcal{T}_l)$:

- for (p=1 ; p <= N_{pairs} ; p++) /* all pairs */
  - for (s=1 ; s <= $T_{\text{CBR}} + \hat{B}_{\text{VBR}} - 1$; s++) /* all supports */

Source	$\hat{\lambda}$, Mbps	$\bar{\lambda}$, Mbps	$\hat{B}$, cells	T, slots
CBR-t	5	5	1	1
VBR-d1	500	10	20	1000
VBR-d2	500	25	50	1000
VBR-d3	200	5	20	800
CBR-n1	50	50	1	1
CBR-n2	10	10	1	1
VBR-n1	100	10	5	50
VBR-n2	50	25	2	4

Table 1: Input parameters for the delay jitter probabilities.

- $*$ `calculate` $\alpha(\cdot)$, $\beta(\cdot)$.
- $*$ `if` $\alpha(\cdot) = 1$,
  - $\cdot$ `mark MC.` `/* set the MC flag */`
  - $\cdot$ $\mathcal{T}_l(s,p) = [0, T_{\text{SP}}]$.
- $*$ `else if` $\beta(\cdot) = 1$,
  - $\cdot$ $\mathcal{T}_l(s,p) = [0, T_{\text{SP}}]$. `/* only for the next burst */`
- $*$ `if` $V_{max}(s,p) \geq V_{min}(s,p)$,
  - $\cdot$ $\mathcal{T}_l(s,p) = \left[T_{\text{CBR}} - s + (p-1)T_{\text{CBR}}, T_{\text{CBR}} - s + (p-1)T_{\text{CBR}} + f_{r(s,p)}(s,p)\right]$.
- $*$ `else`,
  - $\cdot$ $\mathcal{T}_l(s,p) = \emptyset$.

- $\mathcal{T}_l \leftarrow \bigcup_s \bigcup_p \mathcal{T}_l(s,p)$.
- `return` $\mathcal{T}_l$. `/* the set to be sampled from */`

5.5 Experimental Results

For the delay jitter probabilities, we consider the sources for which the input parameters are listed in Table 1. We generated results for 3 systems, labeled 1 through 3, which represent different combinations of background sources. The system parameters and derived parameters for these systems are listed in Table 2. In Tables 1 and 2, "t" stands for target, "n" stands for nondominant and "d" stands for dominant.

System 1 consists of the target CBR connection CBR-t, multiplexed with background traffic consisting of the dominant VBR source VBR-d1 and the nondominant VBR source VBR-n1. The dispersion or 1-Point CDV probabilities are plotted in Fig. 6 for 9 fixed dominant VBR-d1 sources and 1 and 5 nondominant VBR-n1 sources, resulting in utilizations of 0.68 and 0.94, respectively. The target jitter value is varied from 100% of the CBR period

System	Sources	$\hat{\mu}$, Mbps	T_{SP}, slots	μ, cells/slot
1	CBR-t, VBR-d1, VBR-n1	155	1000	0.31
2	CBR-t, VBR-d2, CBR-n1, VBR-n2	622	1000	1.244
3	CBR-t, VBR-d3, CBR-n2	155	800	0.775

Table 2: System parameters and derived parameters.

to 550% of the CBR period. The solid curves in Fig. 6 are the dispersion probabilities and the dashed curves are the improvement factors. The simulation stopping conditions were set at 25-50 hits per bin. The 95% confidence intervals are shown as bars and vary due to the different stopping conditions. As expected, the dispersion probability increases as more background nondominant VBR sources are added. However, this difference decreases as the target jitter value is decreased. As can be observed from Fig. 6, the improvement factors (simulation speedup) are inversely proportional to the probability being estimated.

System 2 consists of the target CBR connection CBR-t, multiplexed with background traffic consisting of the dominant VBR source VBR-d2, the nondominant CBR source CBR-n1 and the nondominant VBR source VBR-n2. The 1-Point CDV probabilities are plotted in Fig. 7 for 10 fixed dominant VBR-d2 sources and two configurations of nondominant sources. The configuration with 1 VBR-n2 source and 1 CBR-n1 source has a utilization of 0.53 and the configuration with 5 VBR-n2 sources and 3 CBR-n1 sources has a utilization of 0.85. The target jitter value is varied from 100% of the CBR period to 350% of the CBR period. The difference between the two dispersion estimates appears almost constant until 300% and increases after 300% dispersion. Again, it can be observed from Fig. 7 that the improvement in simulation efficiency is inversely proportional to the probability being estimated.

For System 3, the target CBR-t connection is multiplexed with dominant VBR sources VBR-d3 and nondominant CBR sources CBR-n2. The results are plotted in Fig. 8 for 10 dominant VBR-d3 sources with 1 nondominant CBR-n2 source and 7 nondominant CBR-2 sources, resulting in utilizations of 0.42 and 0.81, respectively. Results similar to the previous cases are observed. However, as expected, more nondominant CBR sources are necessary to cause an increase in the dispersion as compared to nondominant VBR sources. From Fig. 7, we once again observe that the simulation speedup is inversely proportional to the probability being estimated.

6 Conclusion

In this paper, we presented an Importance Sampling scheme for systems with input processes characterized by uniform distributions, which uses a conditional biasing algorithm. After selecting an initial biasing strategy, the conditional biasing algorithm adaptively modifies the biasing strategy as samples are drawn, thus increasing simulation efficiency. The overall

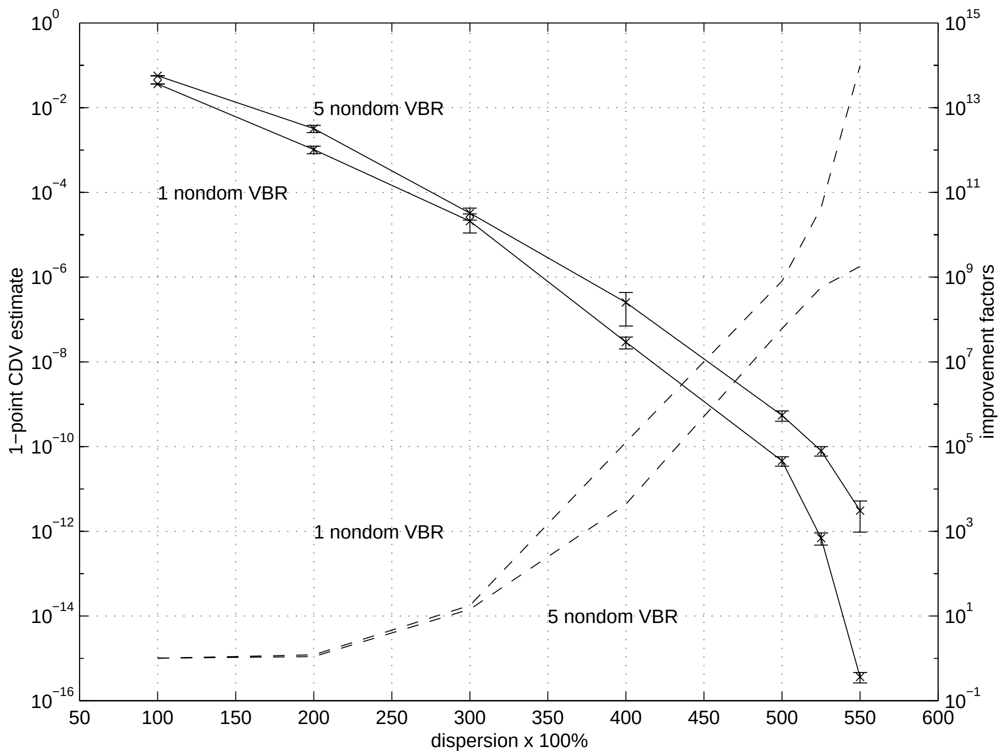

Figure 6: Dispersion results for System 1 with 9 dominant VBR-d1 sources.

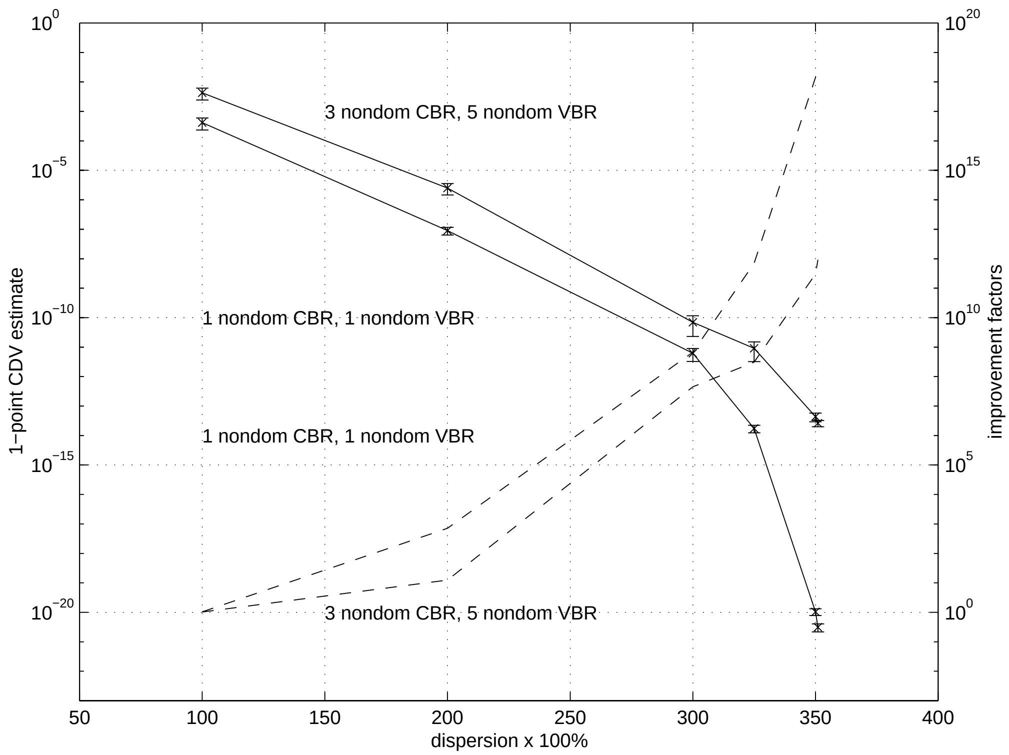

Figure 7: Dispersion results for System 2 with 10 dominant VBR-d2 sources.

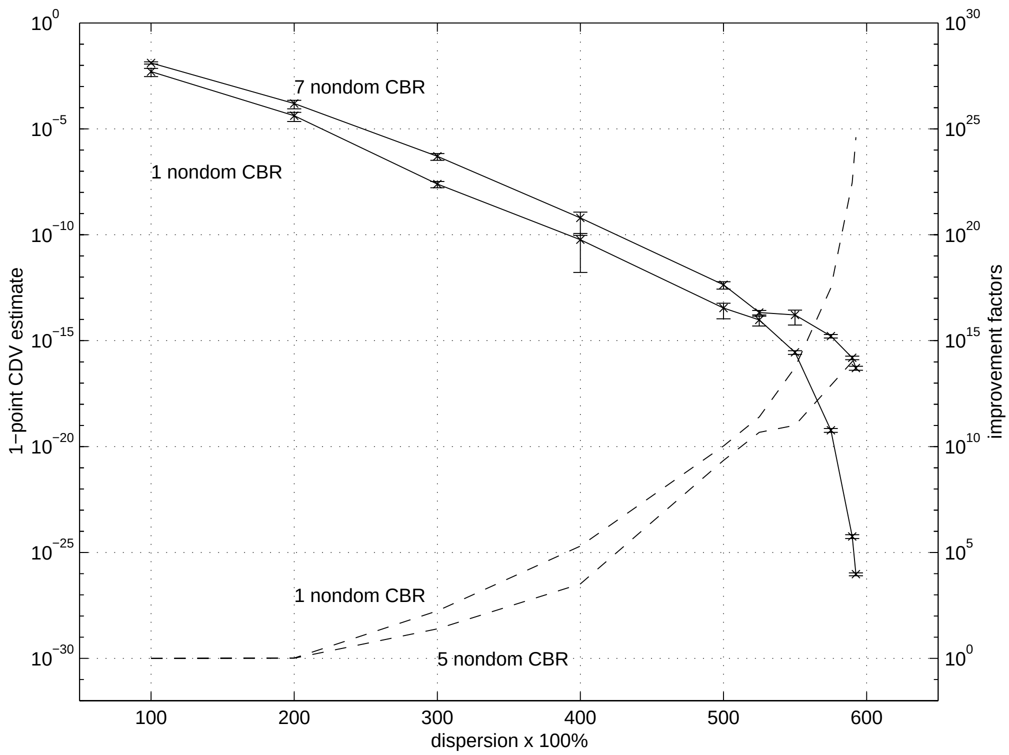

Figure 8: Dispersion results for System 3 with 10 dominant VBR-d3 sources.

scheme also employs a problem-dependent function to select the initial and modified biasing strategies. This problem-specific component allows the algorithm to be suitably adapted to the requirements of a diverse set of problems that are characterized by uniform input distributions. The combined scheme provides for a more effective approach than parametrically biasing the uniform input distributions. To demonstrate the technique, we used conditional biasing to estimate rare jitter probabilities in ATM switches with constant bit rate (CBR) traffic multiplexed with heterogeneous background traffic which can consist of variable bit rate (VBR) traffic and other CBR traffic. For the experimental systems considered, for which the background traffic consisted of different combinations of VBR and CBR traffic, the improvement in simulation efficiency was observed to be inversely proportional to the probability being estimated.

References

[1] K. S. Shanmugan and P. Balaban. A Modified Monte-Carlo Simulation Technique for the Evaluation of Error Rate in Digital Communication Systems. *IEEE Trans. Commun.*, COM-28(11):1916–1924, Nov. 1980.

[2] D. Lu and K. Yao. Improved Importance Sampling Technique for Efficient Simulation of Digital Communication Systems. *IEEE J. Select. Areas Commun.*, 6(1), Jan. 1988.

[3] John S. Sadowsky and J. A. Bucklew. Large Deviations Theory Techniques in Monte Carlo Simulation. In *Proc. of the Winter Simulation Conference*, Washington, DC, 1989.

[4] P. W. Glynn and D. L. Iglehart. Importance Sampling for Stochastic Simulations. *Management Science*, 35(11):1367–1392, Nov. 1989.

[5] A. Goyal, P. Shahabuddin, P. Heidelberger, V. F. Nicola, and P. W. Glynn. A Unified Framework for Simulating Markovian Models of Highly Dependable Systems. *IEEE Trans. Computers*, 41(1):36–51, Jan. 1992.

[6] M. Devetsikiotis and J. K. Townsend. An Algorithmic Approach to the Optimization of Importance Sampling Parameters in Digital Communication System Simulation. *IEEE Trans. Commun.*, 41(10):1464–1473, Oct. 1993.

[7] M. Devetsikiotis and J. K. Townsend. Statistical Optimization of Dynamic Importance Sampling Parameters for Efficient Simulation of Communication Networks. *IEEE/ACM Trans. Networking*, 1(3):293–305, June 1993.

[8] A. A. Akyamac and J. K. Townsend. Efficient Simulation of Rare Jitter Probabilities in ATM switches. To appear in *IEEE Global Telecom. Conf., GLOBECOM '97*.

[9] A. Goyal, P. Heidelberger, and P. Shahabuddin. Measure Specific Dynamic Importance Sampling for Availability Simulations. In *Proc. 1987 Wint. Simul. Conf.*, pages 351–357, 1987.

[10] ATM User-Network Interface Specification, Version 3.0. The ATM Forum, September 10, 1993. The ATM Forum.

[11] R. Z. Gold. Tests Auxiliary to χ^2 Tests in a Markov Chain. *Ann. Math. Statist.*, Vol. 34:56–74, 1963.