Efficient Rare Event Simulation Using DPR for Multidimensional Parameter Spaces

Ahmet A. Akyamaç 1, Zsolt Haraszti 2, J. Keith Townsend 1,3

Abstract

Performance constraints in communication systems and networks require the error or fault probabilities to be very low. Analytical and numerical models are often too restrictive and Monte Carlo simulation is computationally prohibitive for low probabilities. Importance sampling (IS) has been used to estimate rare event probabilities but is increasingly difficult to apply to complex systems where there is an indirect relationship between the inputs and the system response. Splitting is an alternative rare event simulation method that can overcome some of the difficulties associated with IS techniques, and has produced satisfactory results for systems for which the splitting parameter space is one dimensional. However, when the efficient simulation of the rare event requires that splitting is controlled by multiple system parameters, incorrect choice of the splitting parameters can lead to estimates with poor statistical accuracy. In this paper, we show that the important regions of the multidimensional parameter space can be correctly and easily identified when the system parameters are selected depending to the nature of the rare event. The equalization of the raw or unweighted steady state probabilities within the important region yields statistically accurate rare event estimates. The probability equalization can be achieved using the readily available Direct Probability Redistribution (DPR) method, by mapping the multidimensional parameter space to a one dimensional DPR control parameter. In this paper, we discuss alternative ways to perform this mapping, and demonstrate multidimensional splitting using two illustrative example systems.

I. Introduction

Performance constraints in communication systems and networks require the error or fault probabilities to be very low. For example, in Asynchronous Transfer Mode (ATM) networks, the cell loss and delay probabilities are typically required to be on the order of $10^{-9}$ and less. Thus, an important phase in the design of systems and networks involves the determination of such rare event probabilities. Due to the complexity of the systems involved, analytical and numerical models are often too restrictive and Monte Carlo (MC) simulation is computationally infeasible for low probabilities. Importance sampling (IS) has been used to estimate rare event probabilities in communication systems and networks (see [1] for a survey on the application of IS to communication systems and networks). In very complex systems, it is often difficult to determine effective IS methods due to the increasingly indirect relationship between the modifiable parameters and the system response, which could lead to poor statistical accuracy in the estimates. Due to this complexity, reevaluation for changing system topologies involves reimplementing the IS method for each topology. Furthermore, if the relationship between the modifiable parameters and the system response is not properly identified, improper use of IS could result in the phenomenon of underestimation [2]. Thus, it is desirable to be able to employ

1Center for Advanced Computing and Communication, Dept. of Electrical and Computer Engineering, North Carolina State University, Box 7914, Raleigh, NC, USA 27695-7914. Tel: +1 919 515 7353, Fax: +1 919 515 2285, Email: {aaakyama, jkt}@eos.ncsu.edu.
2Applied Research Lab - Switchlab, Ericsson Telecom AB, S-12625 Stockholm, Sweden. Tel: +46 8 719 2499, Fax: +46 8 719 6677, Email: Zsolt.Haraszti@ericsson.com.
3Corresponding author.
4This work was supported by the Center for Advanced Computing and Communication, North Carolina State University, NC, USA and Ericsson Telecom AB, Stockholm, Sweden.
a rare event simulation technique which is easy to implement and which can directly influence the system response without suffering from poor statistical accuracy and underestimation.

Splitting can be viewed as an alternative approach to rare event simulation and can potentially overcome some of the difficulties associated with IS. Splitting has been considered in early works [3] and has recently been used in the context of queueing networks [4], [5], [6], [7], [8], [9], [10], [11]. The main notion behind splitting is to partition the system states into subsets. When a subset is entered during system simulation, numerous retrials are initiated with the current subset as the starting or entry state of the system. Essentially, the main system trajectory is split into a number of subtrajectories. The number of subtrajectories spawned after visiting each subset is determined by the splitting or oversampling factors. The splitting method involves the selection of the subsets and oversampling factors, where the subsets are indexed such that the higher subsets correspond to the rare events. Splitting increases the relative frequency of the rare events by placing an emphasis on higher subsets. Efficient splitting can be achieved by equalizing the steady state probabilities of the subsets under splitting [5], [6], [9], [10]. A method of splitting called RESTART [4], [5] assumes the subsets are nested, and allows for only single subset transitions in a trajectory, and assumes integer oversampling factors. A more general technique we apply here, called direct probability redistribution (DPR) [10] requires the subsets to be disjoint (but not necessarily nested), and accounts for multiple subset transitions. The DPR technique can be used to estimate steady state probabilities in finite state Markovian or semi-Markovian systems, and has been successfully applied to performance estimation in mobile wireless systems [11] and in ATM systems [10], including ATM systems with internal flow control and systems with multiple hops, for which traditional IS techniques are difficult to implement.

Existing splitting techniques typically use one system parameter to control the splitting (e.g., the queue length). Thus, the state space is partitioned into subsets according to a single splitting parameter that is often a directly observable system parameter. However, there are systems for which efficient splitting cannot be achieved without using multiple system parameters when constructing splitting subsets. These systems can be characterized in two ways: the rare events are related to multiple system parameters, or the typical trajectories the system takes to the rare events are defined through multiple system parameters. The time-reversal two node Jackson network with a two dimensional system space, for which single-parameter splitting failed to yield statistical estimates in [8], falls into the first category. We consider two illustrative example systems, which fall into either of the above categories.

In this paper, we present a technique to implement multidimensional splitting. To use our technique to obtain efficient and accurate statistical estimates of rare event probabilities, it is necessary to perform the following: to select system parameters as splitting parameters that properly represent the rare event; to identify which regions of the multidimensional parameter space are closely related
to the occurrence of the rare event; and to set the splitting factors such that the raw steady state probabilities in the important region are equalized. We use the term *multidimensional direct probability redistribution* to refer to the DPR simulation that performs the above three actions. Multidimensional DPR is implemented by mapping the multidimensional parameter space to the subset indices that control DPR simulation. We discuss alternative ways to perform this mapping and show that if the mapping is not performed compliant to the typical trajectories the system follows to the rare events, the resulting estimates will not be statistically accurate and may even exhibit the phenomenon of underestimation.

We demonstrate multidimensional DPR with two illustrative example systems, the first of which is the same two node Jackson network used in [8], and the second of which is a rate-controlled (leaky bucket policed) ATM switch. We show in both cases that improper mapping of the multidimensional parameter space to the subset indices results in estimates with poor statistical accuracy, and sometimes in underestimation. We point out that, for these systems, the above problems occur when the system trajectories that result from an improper choice of subsets are not the same as the typical trajectories that lead to the rare event under normal circumstances. This behavior has been observed before in earlier works on IS [2] and more recently in works on splitting [8].

We demonstrate that, in the case of splitting, the onset of poor statistical estimates and underestimation coexists with other simpler to detect symptoms, such as the lack of convergence of the mean estimate and the non-decreasing relative error. Thus, we present guidelines on how to detect the improper selection of splitting parameters and multidimensional mapping. For the two systems considered, we then proceed to identify different splitting parameters that include the typical trajectories that lead to the rare events. We also address other issues such as the determination of the oversampling factors and the DPR-specific iterative exploration process used in the initial phase of the DPR simulation.

The remainder of this paper is organized as follows: In Section II, we give an overview of the DPR simulation technique, followed by a description and discussion of the calculation of the oversampling factors in Section III. Multidimensional DPR and the mapping of multiple splitting parameters to the DPR subset indices is presented in Section IV. The DPR simulation methodology and results for the two-node tandem Jackson network and the rate-controlled ATM switch are presented in Sections V and VI, respectively. We present concluding remarks in Section VII.

II. The DPR Simulation Technique

The DPR simulation technique was developed in detail in [10]. Here, we summarize some key results and notations that will be used in the remainder of the paper. Note that this section introduces DPR for a one-dimensional indexed subset partition. Multidimensional DPR will be described in the subsequent sections.
Let the state space of the system be denoted by $\mathcal{S}$ and assume there are $n$ states, such that $|\mathcal{S}| = n$, where $|\cdot|$ denotes the cardinality of the set. The system state is defined by a set of system parameters that are not necessarily statistically independent. For example, for a single queue fed by Markovian sources, the queue length, source states and server state would be a set of system parameters. The individual states are indexed from $\mathcal{S} = \{1, \ldots, n\}$. Let $V_0, V_1, V_2, \ldots, V_k \in \mathcal{S}$ represent the system evolution, i.e. the states that are visited in successive observation points. Using DPR, the set $\mathcal{S}$ is partitioned into $m$ disjoint, nonempty, indexed subsets $\mathcal{S}_1, \ldots, \mathcal{S}_m$. The partitioning is uniquely defined by a subset indicator function:

$$\Gamma(V_k) \in [1, m]: \quad \Gamma(V_k) = i \iff V_k \in \mathcal{S}_i$$

(1)

The macro-evolution of the system can thus be represented by the series $\Gamma(V_0), \Gamma(V_1), \Gamma(V_2), \ldots$

Let $\mu = \{\mu_1, \ldots, \mu_m\}$ denote the so-called oversampling vector, which defines the retrial or oversampling factors $\mu_i$ for each subset $\mathcal{S}_i$. We assume, without loss of generality, that $\mu_1 = 1$ and $\mu_1 \leq \mu_2 \leq \ldots \leq \mu_m$. If this condition is not satisfied, the subset indices described by $\Gamma(\cdot)$ can be reordered according to $\mu$ and normalized to form a set of indices described by a new subset indicator function $\Gamma'(\cdot)$ such that $\mu'_1 = 1$ and $\mu'_1 \leq \mu'_2 \leq \ldots \leq \mu'_m$. Subsequently, $\Gamma'(\cdot)$ would replace $\Gamma(\cdot)$ as the subset indicator function. Using DPR, each subset $\mathcal{S}_i$ is visited $\mu_i$ times more often than it would have been using MC simulation [10]. Denoting the steady state probabilities of the original and $m$-partitioned system by $\pi = \{\pi_1, \ldots, \pi_n\}$ and $\pi^{(m)} = \{\pi_1^{(m)}, \ldots, \pi_n^{(m)}\}$, respectively, the following relationship holds between $\pi_l$ and $\pi_l^{(m)}$:

$$\pi_l^{(m)} = \Phi \cdot \mu_{\Gamma(l)} \pi_l, \quad \text{for } 1 \leq l \leq n$$

(2)

where $\Phi$ is a normalization constant, $\Phi = 1/\sum_{l=1}^{n} \mu_{\Gamma(l)} \pi_l$.

During DPR simulation, a trajectory is split whenever a transition to a higher indexed subset occurs, e.g., the system makes a transition from subset $\mathcal{S}_i$ to subset $\mathcal{S}_j$ such that $i < j$. The expected number of new subtrajectories generated upon such a transition is $\mu_j / \mu_i - 1$. The life of the new sub-trajectories is restricted by the following ticketing mechanism: At its creation, each subtrajectory is assigned a ticket, $\Omega$, which is an integer number such that $i < \Omega \leq j$. A sub-trajectory is terminated when it attempts to make a transition to a subset $\mathcal{S}_l$ for which $l < \Omega$. Tickets are assigned randomly to the new trajectory with a distribution that maintains the proper oversampling ratios in every subset. According to DPR theory [10] and assuming the above $\mathcal{S}_i \rightarrow \mathcal{S}_j$, $i < j$ transition, such a distribution is:

$$\text{Prob} \{\Omega = l \mid \mathcal{S}_i \rightarrow \mathcal{S}_j\} = \begin{cases} \frac{\mu_i - \mu_{i-1}}{\mu_j / \mu_i} & \text{if } i < l \leq j, \\ 0 & \text{otherwise}. \end{cases}$$

Due to the oversampling, every state of subset $\mathcal{S}_i$ is visited, in the asymptotic sense, $\mu_i$ times more often than in an MC simulation run [10]. Thus, unbiased estimates can be obtained when the event
procedure simulate \((k, V_k, \Omega, \mu, h, e, r, w)\)

/* Initial call: \(k = 0, V_0: \) arbitrary, \(\Omega = 1\), \(h = e = 0\), \(r = w = 0\) */

while \(k \leq k_{\text{max}}\) do

\(V_{k+1} := \text{transition}(V_k)\) /* Make state transition(s) */

if \(\Gamma(V_{k+1}) < \Omega\) then /* Investigate resulting subset indicator */

return /* Return if indicator smaller than ticket */

\(h_{\Gamma(V_{k+1})} := h_{\Gamma(V_{k+1})} + 1\) /* Bookkeeping for the new state */

\(e_{\Gamma(V_{k+1})} := e_{\Gamma(V_{k+1})} + 1/\mu_{\Gamma(V_{k+1})}\) /* Weighted subset counter */

\(r := r + \Lambda(V_k)\) /* Bookkeeping for rare events */

\(w := w + \Lambda(V_k)/\mu_{\Gamma(V_k)}\) /* Weighted event counter */

if \(\Gamma(V_{k+1}) > \Gamma(V_k)\) then /* Conduct retrials if indicator larger than previous */

\(n_{\text{done}} := 1\) /* Number of trajectories done */

for \(i = \Gamma(V_k) + 1\) to \(\Gamma(V_{k+1})\) do /* One loop for each type of ticket */

\(n := \) an occurrence of RV \(Y\) such that */ Set desired number of traj's */

\(n \in \mathbb{N}^+ \text{ and } E[G] = \mu_i/\mu_{\Gamma(V_k)} - n_{\text{done}}.\) /* ...issued with ticket \(\Omega = i\) */

for \(j = 1\) to \(n\)

\(h_{\Gamma(V_{k+1})} := h_{\Gamma(V_{k+1})} + 1\) /* Bookkeeping for extra trajectory */

\(e_{\Gamma(V_{k+1})} := e_{\Gamma(V_{k+1})} + 1/\mu_{\Gamma(V_{k+1})}\) /* Bookkeeping for rare events */

\(r := r + \Lambda(V_k)\) /* Weighted event counter */

\(w := w + \Lambda(V_k)/\mu_{\Gamma(V_k)}\)

simulate \((k + 1, V_{k+1}, i, h, e, r, w)\) /* Trajectory with given ticket*/

end

end

\(n_{\text{done}} := \mu_i/\mu_{\Gamma(V_k)}\)

end

end

Fig. 1. Pseudo-code of the recursive simulation controller.

counters in the simulation are updated by weighting by a factor of \(1/\mu_i\) when in subset \(S_i\), or \(1/\mu_{\Gamma(V)}\) in general, where \(V\) is the current state of the system.

The pseudo-code in Fig. 1 summarizes the simulation procedure. In Fig. 1, the counter vector \(h\) represents the number of unweighted or raw hits per subset, the weighted counter vector \(e\) represents the hits per subset weighted by the oversampling factors, the counter \(r\) represents the number of raw hits of the rare event (referenced by the rare event indicator function \(\Lambda(\cdot) \in \{0, 1\}\)), the weighted counter \(w\) represents the rare event hits weighted by the oversampling factor of the current subset \(\Gamma(V_k)\), and \(k_{\text{max}}\) represents the maximum simulation time.

Note that the above DPR simulation mechanism does not require the rare event region (i.e., the state space region where the rare event occurs) to be located in a specific subset. In other words, \(\Gamma(\cdot)\) and \(\Lambda(\cdot)\) are not necessarily related, even though their mutual relationship may have a strong effect on the efficiency of the DPR simulation. This is particularly observable in systems for which the splitting parameter space is multidimensional. For multidimensional splitting parameter spaces, the
one dimensional subset indicator function provides a means of applying DPR and, as will be seen in Sections V and VI, this flexibility allows solutions for cases for which the rare event function fails to work as an efficient indicator function.

III. THE OVERSAMPLING FACTORS

As will be shown in Sections V and VI, the determination of the subsets depends on the system and the desired performance measure. Once the subsets are determined, the oversampling vector $\mu$ has to be chosen prior to performing DPR simulations.

Provided that the splitting parameters have been properly selected to represent the typical paths the system takes to reach the rare events, efficient and accurate estimates of the rare event probabilities can be obtained by a DPR setting that equalizes the visiting probabilities of the subsets. This can be achieved by an oversampling vector $\mu$ that is chosen such that the resulting unweighted or raw subset probabilities are roughly equalized. We do not claim such a setting is optimal or asymptotically efficient, but it is supported by empirical studies of diverse cases, such as the examples considered in Sections V and VI. Similar observations were also made in [5], [6], [9], [10].

Equalized hit probabilities of the subsets can be achieved by using DPR and setting the factors in $\mu$ according to the inverse of the estimated steady state probabilities. Denoting the latter by $P_1, \ldots, P_m$, where $P_i = e_i / \sum_{i=1}^{m} e_i$, $\mu$ can be constructed such that $\mu_k = (\max_i P_i)/P_k$ for $1 \leq i \leq m$. Although this represents a tautology, we use the concept of iterative subset exploration to estimate such an oversampling vector $\mu$. We start with a simple MC simulation (corresponding to a DPR simulation with $\mu=1$) which provides relatively accurate estimates of the high probability subsets. Using only the high subset probabilities that satisfy a loose confidence constraint, we produce the first oversampling vector. Typically, assuming the subset indicator function is properly chosen, a subsequent run with the new oversampling vector will give further details about the higher subsets and can be used to generate the next oversampling vector. We repeat the above iterative exploration process until the proper oversampling vector which covers all subsets is obtained.

A pseudo-code that summarizes the iterative subset exploration is given in Fig. 2, where $\mu^i$ represents the oversampling vector at iteration $i \geq 0$, the function call $\text{simulate}(k, V_k, \Omega, \mu, h, e, r, w)$ represents a DPR simulation as in Fig. 1, and the function call $\text{reorder}(\mu, \Gamma(\cdot), h, e, r, w)$ represents the reordering described in Section II that results in increasing oversampling factors for higher subsets. In addition, $h_{\text{min}}$ represents the loose confidence constraint such that only those subsets that have higher than $h_{\text{min}}$ hits affect the calculation of the oversampling factors. Note that since the $\text{reorder}(\cdot)$ function changes the subset indicator function $\Gamma(\cdot)$ and the oversampling vector $\mu$, the subset counter vectors $h, e$, and the rare event counters $r, w$ are also reordered to correspond to the new subset indicator function. This reordering is inherent to the $\text{reorder}(\cdot)$ function and is not explicitly shown in Fig. 2. As we will
show in Sections V and VI, the success or failure of the exploration process is strongly related to the subset indicator function and, as such, can be regarded as a first indicator of the effectiveness of the choice of the subset indicator function.

An alternative approach would be to use the results of the initial MC simulation and assume that the tail is exponentially distributed to identify the oversampling factors for the higher subsets. The exponential tail assumption holds in most cases where the tail is a large deviations (LD) tail. However, some finite state systems may not have an LD tail or may not reach the LD tail within the subsets of interest (e.g., the feedback system considered in [10]). The exponential tail assumption would not be appropriate for such cases. The iterative subset exploration method provides a tail-independent means of generating oversampling factors.

**IV. Multidimensional DPR**

Multidimensional DPR involves three steps. The first two steps are the determination of the multiple splitting parameters, and the identification of the regions of the splitting parameter space that are closely related to the occurrence of the rare event. The third is the mapping of the multiple splitting parameters to the one dimensional subset indicator function for use in DPR simulations.
A. Multidimensional Splitting Parameter Space

There are systems that require the use of multiple system parameters when constructing efficient splitting subsets. These systems can be characterized in at least one of two ways: the rare events are related to multiple system parameters, or the typical trajectories the system takes to the rare events are defined through multiple system parameters.

To this end, let \( \omega \) be the number of splitting parameters. Thus, the splitting parameter space is \( \omega \)-dimensional. Note that \( \omega \) is not required to be the same as the dimensionality of the state space \( S \) of the system. Let \( \mathbf{X} = (X^1, \ldots, X^\omega) \) represent the splitting parameter space, the vector \( \mathbf{x} = (x^1, \ldots, x^\omega) \) represent a realization of the splitting parameters and let \( \mathbf{x}_k \) represent the splitting parameters at the observation instant \( k \). Each splitting parameter \( x^i, 1 \leq i \leq \omega \) can take on values \( x^i \in X^i \), where \( |X^i| = d^i \). For example, \( X^i \) could represent the queue length of a queue of size \( K \), where \( d^i = K + 1 \).

Thus, the number of possible realizations is given by \( \prod_{i=1}^{\omega} d^i \). Define \( I(\mathbf{x}) \in \{0, 1\} \) as an indicator function of whether or not the realization \( \mathbf{x} \) contributes to the rare event, i.e. \( I(\mathbf{x}) = 1 \) indicates that \( \mathbf{x} \) is a realization of system parameter for which either a rare event occurs or that is on a trajectory that leads to a rare event. Thus, \( I(\cdot) \) defines a region in the \( \omega \)-dimensional parameter space that contributes to the rare event. Obviously, \( I(\mathbf{x}) \equiv 1 \) would include all realizations that contribute to the rare event, but as we show in the Jackson network example in Section V, we can improve the efficiency of the rare event simulation by selecting an \( I(\cdot) \) that is more restrictive while maintaining the realizations that include the rare event and the dominant paths to the rare event.

We hypothesize that if \( \mathbf{x} \) and \( I(\mathbf{x}) \) are selected such that \( \mathbf{x} \) contains all system parameters that are required to identify the important rare events and \( I(\mathbf{x}) = 1 \) for all rare events and trajectories that lead to the rare events (i.e., \( I(\cdot) \) includes all rare events and trajectories that lead to the rare events), then applying DPR such that the raw visiting probabilities are equalized among the realizations \( \mathbf{x} \) for which \( I(\mathbf{x}) = 1 \) results in efficient and statistically accurate rare event probability estimates. Although we do not present a formal proof of this hypothesis, it is strongly supported by the examples in Sections V and VI.

B. Mapping of Splitting Parameters to Subset Indices

In order to apply DPR to the multidimensional splitting problem, we map the multiple splitting parameters to a one dimensional subset indicator function to be used in DPR simulations. From Section IV-A, the number of possible realizations of the splitting parameters is:

\[
\sum_{x^1=1}^{d^1} \sum_{x^2=1}^{d^2} \cdots \sum_{x^\omega=1}^{d^\omega} I(x^1, x^2, \ldots, x^\omega)
\]

Thus, a one-to-one mapping can be achieved by simply considering \( m = \sum_{x^1=1}^{d^1} \cdots \sum_{x^\omega=1}^{d^\omega} I(x^1, \ldots, x^\omega) \) subsets, where each realization \( \mathbf{x} \) for which \( I(\mathbf{x}) = 1 \) is mapped to a distinct subset. Since DPR
accounts for multiple subset transitions [10] and the reordering of subsets (described in Section II) guarantees a proper oversampling vector $\mu$, this mapping will result in subsets that are suitable for DPR simulation. Note, however, that such a mapping will also result in a large number of subsets. For certain systems, a more efficient mapping can be found that reduces the number of subsets.

To this end, define $P(x)$ as the steady state probabilities of the realizations $x$. Since the equalization discussed in Section IV-A is achieved by assigning oversampling factors to the realizations $x$ that are inversely proportional to $P(x)$, the realizations with nearly identical oversampling factors can be clustered to form one subset. Thus, a more efficient mapping would identify realizations $x$ with (almost) equal probabilities $P(x)$ and cluster them to form one DPR subset. Essentially, the subsets would be formed as isoprobabilistic clusters of realizations $x$.

The Jackson network in Section V is an example for which isoprobabilistic clustering can be implemented, whereas the rate-controlled ATM network in Section VI is an example for which the one-to-one mapping has to be implemented.

V. TWO-NODE TANDEM JACKSON NETWORK

For the first case, we consider the two-node tandem Jackson network shown in Fig. 3. The input into the network is Poisson distributed and has a rate $\lambda$. As indicated in [8], application of IS to this system can be difficult. The first queue is of infinite size and has an exponential server with rate $C_1$, which feeds into the second queue, also of infinite size, which has an exponential server with rate $C_2$.

![Fig. 3. The two-node tandem Jackson network.](image)

We consider the rare event of interest to be the probability $\gamma_k = \text{Prob}(Q_1 \geq k, Q_2 \geq k)$, where $Q_1$ and $Q_2$ represent the queue length in the first and second queues, respectively. Note that the state of the system is defined by $V = (Q_1, Q_2, \alpha_1, \alpha_2)$, where $\alpha_i$ is an indicator of whether server $i$ is busy. We select the splitting parameter space defined by $X = (Q_1, Q_2)$, which contains the rare events and trajectories that lead to the rare events.

Setting $\lambda = 1$, $C_1 = 2$, and $C_2 = 3$, we form estimates $\hat{\gamma}_k$ of $\gamma_k$ for $k \in \{1, 60\}$. We first show an example of multidimensional mapping which results in an ineffective subset indicator function (also used in [8]). In this case, the poor statistical results are manifested as underestimation, as also noted in [8]. We observe the onset of underestimation early on in the simulation by simple to detect symptoms. Subsequently, we use a more refined mapping that is based on typical trajectories that lead to rare events, which results in a subset indicator function that produces statistically accurate results.
A. Example of an Ineffective Indicator Function

The straightforward mapping for this system uses the rare event of interest. At any point \((Q_1, Q_2)\) in the two-dimensional splitting parameter space, we define \(\Gamma(Q_1, Q_2) = \min\{Q_1, Q_2\} + 1\). This is essentially the same subset indicator function used in [8].

We applied the iterative subset exploration method to generate the oversampling factors. However, despite numerous iterations, the oversampling factors for the low probability subsets could not be obtained because the simulation results were concentrated at the higher probability subsets. This was the first indication of the problematic nature of the chosen subset indicator function. Nevertheless, due to the exponential tail of the Jackson network, we used the exponential tail method described in Section III to generate the oversampling vector.

Using the above oversampling vector, we initially ran 100 sets of DPR simulations, each having a duration of \(10^6\) simulation slots. The mean estimates \(\hat{\gamma}_k\) for 100 sets are shown as solid lines in Fig. 4 along with the analytical results, and the relative half-size 99% confidence intervals are shown as solid lines in Fig. 5. The entire simulation took approximately 24 minutes on an IBM RS/6000 workstation.

It can be observed from Fig.’s 4 and 5 that the results are not statistically accurate, as apparent from the difference in the estimates \(\hat{\gamma}_k\) with the analytical results for \(\gamma_k\) and the increasing confidence intervals for \(k \geq 10\).

![Fig. 4. Mean estimates \(\hat{\gamma}_k\) generated by DPR using subset indicator function \(\Gamma(Q_1, Q_2) = \min\{Q_1, Q_2\} + 1\).](image)

We subsequently increased the duration of the DPR simulations to \(10^7\) slots. The mean estimates \(\hat{\gamma}_k\) and confidence intervals are shown as dashed lines in Fig.’s 4 and 5, respectively. Note that the simulation results obtained by increasing the simulation time to \(10^7\) slots are inconsistent with the ones obtained by using \(10^6\) slots. Furthermore, increasing the simulation time did not decrease the confidence intervals. The behavior of the simulation results and confidence intervals show the
phenomenon of underestimation and are further indications of the problematic nature of the subset indicator function.

To better understand the problem behind the chosen subset indicator function, we present the histogram of the visited states under a DPR simulation in Fig. 6, where the points with a high number of hits are lightly shaded. It can be observed from the histogram in Fig. 6 that the choice of \( \Gamma(Q_1, Q_2) = \min\{Q_1, Q_2\} + 1 \) as the subset indicator function stresses the part of the state space for which \( Q_1 \) and \( Q_2 \) are approximately equal. Most of the hits are concentrated along the line \( Q_1 = Q_2 \) and the regions closely surrounding it. This is mainly due to the fact that the subset indicator increases when both queue lengths increase simultaneously. Thus, the DPR simulation reaches the higher indexed (rare) subsets mainly through trajectories that are close to the line \( Q_1 = Q_2 \).

In Fig. 7, we plot the the typical paths followed by the system to rare event regions \( E_k \overset{\text{def}}{=} \{Q_1 \geq k, Q_2 \geq k\} \) resulting from an MC simulation run on the same setup as in Fig. 6. We consider a region

---

**Fig. 5.** Estimated half size confidence intervals (for 99% confidence level) normalized to the mean estimates, generated by DPR using subset indicator function \( \Gamma(Q_1, Q_2) = \min\{Q_1, Q_2\} + 1 \).

**Fig. 6.** Histogram of visited states by applying \( \min(Q_1, Q_2) + 1 \) as subset indicator (made from \( 10^7 \) samples).
Fig. 7. A typical path that reaches $E_k \equiv \{Q_1 \geq k, Q_2 \geq k\}$ from $(0, 0)$ for $k = 6$.

$E_6$ and demonstrate the preferred path of the system by an arrow in Fig. 7. This suggests that higher subsets are reached through paths for which the first queue builds up to a certain level, and for which the second queue builds up while the first one is draining. This is consistent with the observations made in [8]. Note that these paths are not reflected in the trajectories that use $\Gamma(Q_1, Q_2) = \min\{Q_1, Q_2\} + 1$ as the subset indicator function.

The chosen subset indicator function reinforces numerous trajectories that do not coincide with the natural trajectories of the system that lead to the rare events. Thus, the DPR simulation is not computationally efficient and results in underestimation, as also indicated in [8]. On the other hand, we were able to identify problems with the chosen indicator function both during the iterative subset exploration process and the simulation process by observing the probability estimates and the confidence intervals.

**B. Refined Subset Indicator Function**

For the Jackson network, the two dimensional probability mass function in the splitting space is an inclined plane (in the logarithmic sense, and with different slopes in each dimension). Consequently, the isoprobabilistic clusters are lines, which can be represented in one dimension using a mapping of $(Q_1, Q_2)$ to a subset indicator function of the form $\Gamma(Q_1, Q_2) = \lfloor \alpha Q_1 + \beta Q_2 + 1 \rfloor$ such that $\alpha/\beta < 1$. The ratio $\alpha/\beta$ determines which queue is stressed more and the ratio being less than unity places more stress on the second queue. The individual values of $\alpha$ and $\beta$ determine the resolution of the subsets in the low probability region.

Using preliminary simulation runs to equalize the stress on the two queues, we generated a ratio of $\alpha/\beta = 1/1.5$ and chose the following mapping:

$$\Gamma(Q_1, Q_2) = \left\lfloor \frac{Q_1 + 1.5Q_2}{2} \right\rfloor + 1$$  (3)
It is not surprising that this ratio is the same as the ratio of the service rates. The iterative subset exploration method proved to be successful for all subsets using the above refined subset indicator function.

Similar to the case in Section V-A, we ran two sets of simulations, the first one consisted of 100 batches of $10^6$ simulation slots and the second one consisted of 100 batches of $10^7$ slots. The mean estimates $\hat{\gamma}_k$ and confidence intervals are plotted in Fig.'s 8 and 9, respectively. The simulations took 53 min. and 523 min., respectively, on an IBM RS/6000 workstation.

![Fig. 8. Mean estimates $\hat{\gamma}_k$ generated by DPR using subset indicator function $\Gamma(Q_1, Q_2) = (Q_1 + 1.5Q_2)/2 + 1$.](image1)

![Fig. 9. Estimated half size confidence intervals (for 99% confidence level) normalized to the mean estimates, generated by DPR using subset indicator function $\Gamma(Q_1, Q_2) = (Q_1 + 1.5Q_2)/2 + 1$.](image2)

As can be observed from Fig. 8, both the short simulations (solid line) and long simulations (dashed line) result in consistent estimates which are almost identical to the analytical results. As expected, the longer simulation run produces more accurate estimates, which is also apparent from the confidence intervals in Fig. 9.
The stable behavior of the iterative subset exploration method, the consistent simulation estimates and the confidence intervals are all indications that the refined subset indicator is a proper selection and does not result in underestimation. Observation of the histogram of the visited states under DPR simulation shown in Fig. 10 reveals that the refined subset indicator function includes the trajectories that correspond to the typical paths the system takes to reach the low probability regions.

![Histogram of visited states](image)

**Fig. 10.** Histogram of visited states with applying \((Q_1 + 1.5Q_2)/2 + 1\) as subset indicator (made from \(10^7\) samples).

### VI. Rate-Controlled ATM Switch

We consider the ATM switch with the leaky bucket rate controllers shown in Fig. 11. There are \(N\) independent homogeneous input sources for which we define a triplet \((p, m, b)\), where \(p\) is the peak cell arrival rate, \(m\) is the mean or sustained cell arrival rate and \(b\) is the mean burst length at the peak rate. We assume the entire system is normalized with respect to the peak rate \(p\) and thus consider time to be discrete, where one time slot corresponds to the arrival time of a single cell at the peak rate \(p\). With this normalization, we assume that, for each source, cells arrive according to an interrupted Bernoulli process (IBP). The IBP sources can be in one of two states \(s \in \{0, 1\}\) and have a mean rate of \(m\) and a mean burst length of \(b\).

Each source is policed by a leaky bucket rate controller as shown in Fig. 11. The leaky bucket controller consists of an arrival queue and a token queue. The token queue is of size \(\sigma\) and tokens arrive at a rate of \(\rho\), corresponding to a discrete-time \((\sigma, \rho)\) shaper [12]. The arrival queue is assumed to be sufficiently large such that cell loss at the arrival queue is negligible and peak rate control is omitted since the model is normalized with respect to the peak rate. The cells exiting the shapers enter an ATM switch with a shared bus switch architecture and are switched to the output queue, which has a finite size of \(K\) and a normalized service rate of \(C = 1\). We consider the queue length probability mass function \(\gamma_x = \text{Prob}\{Q = x\}\), which is rare for high values of \(x\). The state of the system can be defined by \(V = (s_1, \ldots, s_N, A_1, \ldots, A_N, B_1, \ldots, B_N, Q)\), where \(s_i\), \(A_l\) and \(B_l\) are the source state,
Fig. 11. ATM switch with leaky bucket rate controllers.

arrival queue length and token queue length of connection \( l \), respectively, and \( Q \) is the output queue length. For this system, we select the splitting parameter space defined by \( X = (B_1, \ldots, B_N, Q) \), which contains the rare events and trajectories that lead to the rare events.

Setting \( p = 1, m = 0.05, b = 10 \) for \( N = 8 \) sources and \( \sigma = 5, \rho = 0.05 \) for the leaky bucket shapers and \( K = 40 \) for the output queue, we form estimates \( \hat{\gamma}_x \) for \( x \in \{1, 40\} \). As in Section V, we first use a mapping that results in an ineffective subset indicator function, which also leads to poor statistical accuracy. We show that the problems with the indicator function can be detected by the same symptoms used in Section V. We subsequently use a more refined mapping, again based on the observation of typical trajectories that lead to rare events, which results in a subset indicator function that produces statistically accurate estimates.

A. Example of an Ineffective Subset Indicator Function

For the rate-controlled ATM switch configuration, the straightforward mapping is of the form \( \Gamma(B_1, \ldots, B_N, Q) = Q \), which focuses entirely on the output queue and is based on the rare event of interest.

As was the case in Section V, we encountered difficulties in the iterative subset exploration process, indicating that the chosen subset indicator function is problematic. Nevertheless, we used the best possible version of the oversampling factors obtained from the iterative process. The best version is considered to be the iteration that penetrates the highest subset.

Using the oversampling vector, we ran 50 sets of two types of DPR simulations, with durations of \( 10^6 \) and \( 10^7 \) slots, respectively. The mean estimates \( \hat{\gamma}_x \) for 50 sets are shown as solid lines for \( 10^6 \) runs.
and dashed lines for $10^7$ runs in Fig. 12. The relative half-size 99% confidence intervals are shown in Fig. 13. The simulations took 86 minutes and 13 hours, respectively, on an Intel Pentium Pro PC running FreeBSD Unix.

![Graph showing mean estimates and confidence intervals](image)

**Fig. 12.** Mean estimates $\hat{\gamma}_x$ generated by DPR using subset indicator function $\Gamma(B_1, \ldots, B_N, Q) = Q$.

![Graph showing normalized confidence intervals](image)

**Fig. 13.** Estimated half size confidence intervals (for 99% confidence level) normalized to the mean estimates, generated by DPR using subset indicator function $\Gamma(B_1, \ldots, B_N, Q) = Q$.

Even though analytical results are not readily available for this system, we can conclude from Fig. 12 that the estimates are not consistent. This is due to the fact that, as seen in Fig. 12, the longer simulation resulted in a probability curve that significantly deviates from the curve obtained from the shorter simulation runs at low probabilities. Moreover, as seen in Fig. 13, the increase in the number of simulation runs did not result in an observable improvement in the confidence intervals.

The above simulation observations, along with the difficulties encountered in the iterative subset exploration process, indicate the problematic nature of the chosen subset indicator function. For this
system, MC simulation trials indicate that the sum of the available tokens in the system affects the typical paths to the rare events. We observed that the higher subsets are reached through paths for which the token queues build up to a certain level, then begin to empty resulting in an increase in the output queue length. As can be observed from the histogram of the visited states under DPR simulation in Fig. 14, these paths are not reflected in the trajectories that use \( \Gamma(B_1, \ldots, B_N, Q) = Q \) as the subset indicator function. In Fig. 14, the points with a higher number of hits are lightly shaded. It can be observed from the histogram in Fig. 14 that the choice of \( \Gamma(B_1, \ldots, B_N, Q) = Q \) as the subset indicator function does not properly stress the token queues and results in most of the hits being concentrated in regions for which the token queues are low.

### B. Refined Subset Indicator Function

The probability surface for this system is not a plane and thus, the isoprobabilistic clusters are not lines, as was the case in Section V. Since there is no obvious mapping from the two dimensional splitting state space to a one dimensional subset index, we use a one-to-one mapping as described in Section IV-B. However, in reference to the observations made from the MC trials, we use the sum of the available tokens, \( \sum_{i=1}^{N} B_i \), rather than the tokens available to the individual connections, since the increase in queue length occurs depending on the sum of the tokens available in the leaky buckets. The initial indexing of the subsets can be chosen arbitrarily maintaining a one-to-one relationship between a point in the parameter space and a subset index, since this indexing will potentially change for each step of the iterative subset exploration method which will reorder the subsets indices such that \( \mu_1 \leq \ldots \mu_m \) for the new subset indices, as described in Section III. For example, we can use the
following mapping:

\[ \Gamma(B_1, \ldots, B_N, Q) = 41 \star \left( \sum_{i=1}^{N} B_i \right) + Q \]  

(4)

The iterative subset exploration using the above initial subset indicator function was successful.

Using the oversampling vector from the iterative subset exploration, we again ran two sets of simulations, consisting of 50 batches each of $10^6$ and $10^7$ simulation slots, respectively. The mean estimates \( \hat{\gamma}_x \) and confidence intervals are plotted in Fig.'s 15 and 16, respectively. The simulations took 92 min. and 13.5 hours, respectively, on an Intel Pentium Pro PC running FreeBSD Unix.

Fig. 15. Mean estimates \( \hat{\gamma}_x \) generated by DPR using the refined subset indicator function.

Fig. 16. Estimated half size confidence intervals (for 99% confidence level) normalized to the mean estimates, generated by DPR using the refined subset indicator function.

As seen from Fig. 15, the simulation runs are consistent in that the longer simulation refined the results of the shorter simulation for lower probabilities and coincided with the shorter simulation
for higher probabilities. The confidence intervals in Fig. 16 indicate that the longer simulation run produced more accurate estimates, which is a further indication that underestimation did not occur. Furthermore, note that the refined subset indicator function generated results for \( x \leq 30 \) within a 100\% relative confidence limit with \( 10^7 \) runs, whereas the straightforward indicator function was only able to generate results for \( x \leq 14 \) within the same confidence limit and for the same number of runs.

The histogram of the visited states under DPR simulation shown in Fig. 17 reveals that the refined subset indicator function includes the typical paths to the rare events, as described above.

![Histogram of visited states](image)

**Fig. 17.** Histogram of visited states by applying the refined subset indicator function.

**VII. Concluding Remarks**

In this paper, we presented a splitting simulation technique for systems with multidimensional parameter spaces. The technique, called multidimensional DPR, consists of two steps. The first of these is the identification of the splitting parameters, and the second is the mapping of the multidimensional parameter space to a one dimensional subset indicator function for use in DPR simulations.

Using our approach to obtain efficient and accurate statistical estimates of rare event probabilities, it is necessary to perform the following: to select system parameters that properly represent the rare event; to identify which regions of the multidimensional parameter space are closely related to the occurrence of the rare event; and to set the splitting parameters such that the raw steady state probabilities in the important region are equalized. We also discussed alternative ways to perform the mapping of the multidimensional parameter space to a one dimensional subset indicator function and showed that if the mapping is not compliant to the typical trajectories the system follows to the rare events, the resulting estimates will not be statistically accurate and may even exhibit the phenomenon of underestimation. We used the multidimensional DPR splitting method to analyze two
example systems, the first of which is a two node Jackson network, and the second of which is a rate-controlled (leaky bucket policed) ATM switch. We observed, in both cases, that improper selection of the mapping results in poor statistical accuracy. For the first system, we also observed the problem of underestimation. We pointed out that, for these systems, the above problems occurred when the system trajectories that result from an improper mapping are not the same as the typical trajectories that lead to rare events under normal circumstances.

However, we demonstrated that the onset of poor statistical estimates and underestimation coexists with other simpler to detect symptoms, such as the lack of convergence of the mean estimate and the non-decreasing relative error. An advantage of the DPR technique is that it is flexible enough to be quickly modified according to the underlying rare event mechanism of the investigated problem. Thus, if a mapping is determined to be unsatisfactory, a relatively small effort is required to modify the DPR simulation program to reflect a different choice of subsets. We subsequently applied different mappings for the two systems which included the typical trajectories that lead to the rare events, which resulted in statistically accurate simulations.

References