Setwise and Filtered Gibbs Samplers for Teletraffic Analysis

LACHLAN L. H. ANDREW
Centre for Advanced Internet Architectures
Swinburne University of Technology, Australia
GUOQI QIAN
Department of Mathematics and Statistics
University of Melbourne, Australia
and
FELISA J. VÁZQUEZ-ABAD
Department of Computer Science
Hunter College of the City University New York, USA

A setwise Gibbs sampler (SGS) method is developed to simulate stationary distributions and performance measures of network occupancy of Baskett-Chandy-Muntz-Palacios (BCMP) telecommunication models. It overcomes the simulation difficulty encountered in applying the standard Gibbs sampler to closed BCMP networks with constant occupancy constraints. We show Markov chains induced by SGS converge to the target stationary distributions. This paper also investigates the filtered Gibbs sampler (FGS) as an efficient method for estimating various network performance measures. It shows that FGS's efficiency is considerable but may be improperly overestimated. A more conservative performance estimator is then presented.

Categories and Subject Descriptors: G.3 [**PROBABILITY AND STATISTICS**]: Probabilistic algorithms (including Monte Carlo), Queueing theory; C.2.1 [**Network Architecture and Design**]: Circuit-switching networks, Packet-switching networks

General Terms: Theory, Algorithms, Performance
Additional Key Words and Phrases: Gibbs Sampler; Markov chain Monte Carlo; Product form; Queueing networks

Email: lachlan@swin.edu.au, g.qian@ms.unimelb.edu.au, felisav@unimelb.edu.au.
While performing this work, Lachlan Andrew was with the Centre for Ultra-Broadband Information Networks (CUBIN), University of Melbourne, and with the Computer Science Department, California Institute of Technology. While performing this work, Felisa Vázquez-Abad was with CUBIN, with the Department of Mathematics and Statistics, University of Melbourne, and on leave from Department of Computer Science and Operations Research, University of Montreal, Montreal, Canada H3C 3J7. This work was supported in part by NSERC-Canada grant # WFA0184198 and by the Australian Research Council (ARC).

1. INTRODUCTION

Product form stationary distributions arise in many models for telecommunications systems. These models include the multi-class $M/G/k/k$ queues which model the traditional circuit switched telephone networks with fixed routing. They also include cellular networks with frequency-reuse constraints [Boucherie and Mandjes, 1998, Everitt and Macfadyen, 1983, Pallant and Taylor, 1995]; packet switched networks with fixed routing and effective bandwidth admission control [Berger and Whitt, 1998, Kelly, 1991] or with marking-based admission control [Kelly, Key, and Zachary, 2000]; and intelligent networks in which connections require a particular set of services for the duration of the call [Jordan, 1995]. Many other models with product form stationary distributions are listed in [Mitra and Morrison, 1994]. In particular, closed Baskett-Chandy-Muntz-Palacios (BCMP) networks [Baskett et al., 1975, Chao, Mayazawa, and Pinedo, 1999], which model packet switched networks with sliding window or token-based flow control [Reiser, 1979, Vázquez-Abad and Mason, 1999], have product form stationary distributions. The network models to be considered in this paper are presented in Section 2.

The importance of networks with product form stationary distributions has led to many techniques for their analysis [Ross, 1995]. A number of numeric measures may be calculated to assess the performance of the networks. These measures include the blocking probability of circuit switched networks, and mean queue lengths and throughputs of packet switched networks. Calculation of any such measure normally involves a normalising constant G in the stationary distribution as defined in equation (3). The normalising constant G, with various levels of difficulty, may be calculated by convolutional methods [Buzen, 1973, Coleman, Henderson, and Taylor, 1994], numerical inversion of generating functions [Choudhury, Leung, and Whitt, 1995] or by Monte Carlo integration [Boucherie and Mandjes, 1998, Ross, Tsang, and Wang, 1994].

As an alternative means which does not need to calculate G, Markov chain Monte Carlo (MCMC) simulation [Kendall, Wang, and Liang, 2005, Gamerman and Lopes, 2006, Neal, 2003] can be used to estimate the blocking probabilities in product form networks [Lassila and Virtamo, 1998a, Lassila and Virtamo, 1998b, Vázquez-Abad and Andrew, 2000]. In this paper we will investigate the performance of MCMC simulation in this estimation. In an MCMC simulation actual samples can be generated from the stationary distribution; thus they can be further used, say, for starting simulations to calculate other performance measures, as is done in [Conway and O'Brien, 1993]. An overview of MCMC simulation and one of its most fundamental methods, the Gibbs sampler, is to be given in Section 3.

The Gibbs sampler traverses the state space by modifying one component of the state vector at each step. As such it is not directly applicable to closed queueing networks, in which the sum of the state components is fixed, yielding solutions on a lower-dimensional manifold. The traditional solution is to remove one component from the state vector; but updates to each single component of the reduced state vector implicitly update the omitted component as well. In Section 4 we will propose a more flexible approach, the Setwise Gibbs Sampler (SGS) in which a subset of multiple components are updated simultaneously at each step. We have obtained conditions on the choice of subsets in SGS updates which ensure the convergence of MCMC simulation to the correct distribution.

In using simulated Gibbs samples to estimate performance measures of a network stationary distribution, conditioning or filtering is a useful technique to reduce the variance and improve the efficiency of various estimators. Conditioning is a generic term indicating

that unconditional expectations involved in a parameter of interest, when being estimated, are replaced by expectations conditional on statistics involving sufficient information of the parameter, rather than by the observed statistics directly. The filtered Gibbs sampler (FGS) [Vázquez-Abad and Andrew, 2000, Andrew and Vázquez-Abad, 2002] to be presented in Section 5 is an enhancement to the standard Gibbs sampler that implements the conditioning technique to the simulated estimates of network performance measures. No thorough performance analysis of FGS has yet appeared in the literature of MCMC simulation of network stationary distributions. This will now be done numerically in Sections 5.2 and 5.3, where an expression is also derived for the maximum benefit relative to the standard Gibbs sampler under the assumption of low network load. Note that neither FGS nor SGS is an improvement of the other; rather they are parallel enhancements to the standard Gibbs sampler.

Under a typical termination criterion, a Monte Carlo simulation will be terminated once a specified level confidence interval for the quantity being simulated is sufficiently short. But a computed confidence interval can be incorrectly too short because the variance involved is usually underestimated by standard methods, causing the simulation to terminate prematurely. This problem affects many filtering and importance sampling techniques. In Section 5.4 we will propose a more conservative variance estimator which allows simulations to be terminated at the appropriate time.

2. NETWORK MODEL

Consider the general BCMP model for a queueing network, introduced in [Baskett et al., 1975]. There are N service stations that may have single or multiple servers, and R classes of customers (that may possess different service requirements). A customer of class r that ends service at station i is routed to station j and given class q with probability $p_{(i,r),(j,q)}$ independently of the history of the process. Arrivals to service station i of class r customers from outside the network follow independent Poisson processes . The network may be as complicated as having some classes with zero external arrivals, so their behaviour is that of a closed network, while other classes sharing the network resources may have external arrivals and departures. The general model therefore considers the possibility that the routing matrix

$$P = \{p_{(i,r),(j,q)}\} \tag{1}$$

is not irreducible, but consists of m irreducible transition kernels. In this paper, transitions will not occur between classes, that is, P consists of $m = R$ submatrices, each of them irreducible, corresponding to the subspaces of the states (i, r) per customer class. This is the model for closed multiple-chain networks. Each subspace $S_r = \{(i, r) : i = 1, \ldots, N\}$, $r = 1, \ldots, R$ corresponds to either a closed or an open subsystem per class. Let $S = \cup_{r=1}^{R} S_r$ denote the complete set of indices (i, r). Because customers never change class here, for each subsystem, the *effective arrival rate* is the solution, $\{e_{j,q} : j = 1, \ldots, N; q = 1, \ldots, R.\}$, of the linear equations:

$$e_{j,q} = \lambda_{j,q} + \sum_{(i,r)\in S} e_{i,r} p_{(i,r),(j,q)} = \lambda_{j,q} + \sum_{(i,q)\in S_q} e_{i,q} p_{(i,q),(j,q)}, \tag{2}$$

where $\lambda_{j,q}$ is the external arrival rate to service station j of class q customers. If the subsystem S_q is closed, then $\lambda_{j,q} = 0$ for $j = 1, \ldots, N$ and the above linear system is

only defined up to a multiplicative constant. In that case one sets $\sum_{(j,q)\in S_q} e_{j,q} = 1$ and the factors are interpreted as the *relative number of visits to state* (j, q).

Service stations can be of different types. Denote by $G_{i,r}$ the service distribution of station i for class r. The *occupancy vector* of the whole network will be denoted by $\mathbf{n} = (n_{i,r} : i = 1, \ldots, N; r = 1, \ldots, R)$ indicating how many customers of each class r are in each station i. The aggregate occupancy of station i is $n_{i\bullet} = \sum_{r=1}^{R} n_{i,r}$. Denote by $1/\mu_{i,r}$ the mean service time of class r at service station i, and let $\rho_{i,r} = e_{i,r}/\mu_{i,r}$ be the *utilization factor* of the server/class pair (i, r). For single class networks or situations where customer class does not have effect, the second subscript r in $G_{i,r}$, $\mu_{i,r}$, $e_{i,r}$, $\lambda_{i,r}$, $\rho_{i,r}$, $n_{i,r}$ and other relevant quantities will be dropped for simplicity.

Service stations must be of one of the following types:

Type 1: First-come-first-served (FCFS), $G_{i,r} = G_i \sim \exp(\mu_i(n_i))$ for all customer classes (station may have one or several servers)

Type 2: Processor sharing, $G_{i,r}$ arbitrary, single server

Type 3: Infinite number of parallel servers, $G_{i,r}$ arbitrary

Type 4: Last-come-first-served (LCFS), $G_{i,r}$ arbitrary, single server.

THEOREM 1 BCMP. *[Baskett et al., 1975] Let* $\mathbf{n}_i = (n_{i,1}, \ldots, n_{i,R})$ *denote the occupancy vector at station* i *(implying* $\mathbf{n} = (\mathbf{n}_1, \ldots, \mathbf{n}_N)$*). Then the stationary distribution of the network occupancy has the product form:*

$$\pi(\mathbf{n}) \equiv \pi(\mathbf{n}_1, \ldots, \mathbf{n}_N) = \frac{1}{G}\, d(\Omega) \prod_{i=1}^{N} g_i(\mathbf{n}_i), \tag{3}$$

where:

—if i *is of type 1, then* $g_i(\mathbf{n}_i) = n_{i\bullet}! \left(\frac{1}{\mu_i}\right)^{n_{i\bullet}} \prod_{r=1}^{R} \frac{e_{i,r}^{n_{i,r}}}{n_{i,r}!}$,

—if i *is of type 2 or 4, then* $g_i(\mathbf{n}_i) = n_{i\bullet}! \prod_{r=1}^{R} \frac{\rho_{i,r}^{n_{i,r}}}{n_{i,r}!}$,

—if i *is of type 3, then* $g_i(\mathbf{n}_i) = \prod_{r=1}^{R} \frac{\rho_{i,r}^{n_{i,r}}}{n_{i,r}!}$.

Here Ω *denotes the state space of the occupancy vector* $\mathbf{n}$*,* $d(\Omega)$ *is a function of the external arrival rates such that* $d(\Omega) = 1$ *when the whole network is closed, and* G *is the* normalising constant*, chosen so as to make* $\sum_{\mathbf{n}\in\Omega} \pi(\mathbf{n}) = 1$.

Note that $g_i(\mathbf{n}_i)$ can be written as

$$g_i(\mathbf{n}_i) = h_i(n_{i\bullet}) \prod_{r=1}^{R} \frac{\rho_{i,r}^{n_{i,r}}}{n_{i,r}!}, \tag{4}$$

where $h_i(n_{i\bullet}) = 1$ if station i is of type 3 (which we will call IS — infinite server station), and $h_i(n_{i\bullet}) = n_{i\bullet}!$ otherwise.

For a single class closed network implying $n_{i,r} \equiv n_i \equiv \mathbf{n}_i = n_{i\bullet}$, a considerable simplification follows: let T_1 be the subset of all stations that are of type 1, 2 or 4, and T_2

be the set of the (remaining) stations which are of type 3, then:

$$\pi(\mathbf{n}) = \frac{1}{G} \prod_{i \in T_1} \rho_i^{n_i} \prod_{i \in T_2} \left(\frac{\rho_i^{n_i}}{n_i!} \right). \tag{5}$$

2.1 Circuit switched networks

Circuit switched networks may be described by a BCMP model. In a circuit switched network, where for simplicity of presentation is assumed to be open with type 3 servers and have a single customer class, the N service stations model distinct routes through the network and n_i is the number of calls currently using route i. If the network can support a particular combination of calls, then it can also support any subset of those calls. Thus for any feasible occupation vector $\mathbf{n} = (n_1, \ldots, n_N) \in \Omega$, we have $\{\mathbf{n}' : n_i' \leq n_i\} \subseteq \Omega$, where '$\leq$' is taken componentwise.

The feasible region, Ω, is often of the form

$$\Omega = \{\mathbf{n} \in \mathbb{N}^N : A\mathbf{n}^t \leq \mathbf{C}\} \tag{6}$$

(but [Jordan, 1995, Kind, Niessen, and Mathar, 1998] give exceptions). Here $A = [a_{ji}] \in \{0,1\}^{L \times N}$ (or more generally $\mathbb{N}^{L \times N}$) specifies the number of channels required by route i on link j ($i = 1, \cdots, N; j = 1, \cdots, L$), and $\mathbf{C} = (C_j) \in \mathbb{N}^L$ is a vector of the numbers of channels available on each link.

Because the model corresponds to a single class open network of type 3 servers, the form of the marginal densities $g_i(n_i)$ in (3) is

$$g_i(n_i) = \left(\frac{\rho_i^{n_i}}{n_i!} \right).$$

Let B be the network blocking probability. A feasible state, $\mathbf{n}$, is a blocking state for route i if one more call on route i would lead to an infeasible state. The set of blocking states for route i, $i = 1, \ldots, N$, is

$$\mathcal{B}_i = \left\{ \mathbf{n} \in \Omega : \exists j, a_{ji} + (A\mathbf{n}^t)_j > C_j \right\}. \tag{7}$$

Let $B_i = \mathsf{P}(\mathbf{n} \in \mathcal{B}_i)$ be the blocking probability of route i. Writing $\lambda = \sum_{i=1}^N \lambda_i$ for the total external arrival rate gives

$$B = \sum_{i=1}^{N} \left(\frac{\lambda_i}{\lambda} \right) B_i. \tag{8}$$

2.2 Window flow control

In contrast to a circuit switched network, a packet switched communication network with window flow control can be modelled by a closed BCMP queueing networks in the following sense [Reiser, 1979]. Each connection on the communication network is regarded as a class. And packets or acknowledgements in transit in the network are regarded as customers. Customers can also be used to represent packets received but not acknowledged, or packets within the current transmit window which have not yet been transmitted. (With greedy sources and fast receivers, the latter two cases are not encountered.) The number of customers of each class is equal to the size of the window, which is assumed constant. Store-and-forward switches are represented as FCFS nodes corresponding to the service

stations in the BCMP model, and transmission delays can be modelled by IS nodes with constant service times. The routing of customers through the queueing network is the same as that of packets through the communication network, and in this paper will be assumed to be deterministic.

For these networks,

$$\Omega = \left\{ \mathbf{n} : \sum_{i=1}^{N} n_{i,r} = C_r \text{ for all } r \right\},$$

where C_r is the constant number of customers (packets) on connection r, which is equal to the window size for the corresponding connection.

Measures of interest in packet networks include overflow probabilities (the probabilities that the buffer occupancies exceed a certain threshold), mean queue lengths and throughputs. In general the performance of the network will be of the form

$$B = \sum_{i=1}^{N} w_i \, B_i,$$

for some weight factors w_i and local performance functions $B_i = \mathsf{E}[b_i(\mathbf{n})]$. The sample performance b_i is a local function of the occupancy of station i, and the expectation is with respect to π. This is clearly the case for the three performance measures mentioned above, with throughputs calculated by applying Little's law to an estimate of the idle time of each queue.

3. MARKOV CHAIN MONTE CARLO SIMULATION

Evaluating blocking probabilities using (3) and (8) directly is a difficult numerical problem even for networks with realistic sizes of N and R. Moreover, in many cases, it is not sufficient to know the blocking probability, and it is desirable to sample from the distribution itself (see for example [Conway and O'Brien, 1993]). In [Vázquez-Abad and Andrew, 2000] a wavelength-division-multiplexing (WDM) network was studied. A typical WDM backbone network may have over $m = 20$ nodes and $C = 32$ or more wavelengths. The simplest approach is to calculate the normalising factor G, where the sums are over the space Ω, and then explicitly sum (3) over all states $\mathbf{n} \in \mathcal{B}_i$. The number of routes is $N = m^2/2 + o(m^2)$, and for densely connected networks, the number of states is $\mathcal{O}(C^N)$. Thus computing G directly takes of the order of $C^{m^2/2}$ multiplications. For a modest network of $m = 10$ nodes with $C = 8$ wavelengths, this requires around $8^{45} \approx 10^{40}$ multiplications, taking 10^{21} years on a 1 Tflops computer.

Monte Carlo techniques bridge the gap between exact algorithms [Buzen, 1973, Choudhury, Leung, and Whitt, 1995, Coleman, Henderson, and Taylor, 1994] and approximations [Knessl and Tier, 1998, Mitra and Morrison, 1994, Mitra, Morrison, and Ramakrishnan, 1999]. They allow a quantifiable tradeoff between computational time and accuracy, while being conceptually simple.

This section presents the construction of a "surrogate" Markov chain $\{X_k : k = 1, 2, \ldots\}$ of the occupancy vector $\mathbf{n}$ with state space Ω whose steady state probabilities are given exactly by π in (3). That is,

$$\forall\, \mathbf{n} \in \Omega \quad \lim_{k \to \infty} \mathsf{P}(X_k = \mathbf{n}) = \pi(\mathbf{n}). \tag{9}$$

The methods underlying such construction are called Markov chain Monte Carlo (MCMC) (see [Brémaud, 1999]). Then B can be estimated from T samples generated from the Markov chain as $\hat{Y}(T) = (1/T)\sum_{i=1}^{T} y(X_i)$ for any function $y(\cdot)$ with $\mathsf{E}[y(X)] = B$.

Define the relative mean squared error as $\mathsf{Var}[\hat{Y}(T)]/B^2$. An estimate of a given relative mean square error can be obtained faster by either decreasing the CPU time required to evaluate $y(X)$, or by using an estimator, y, of B with reduced variance and reducing T. This tradeoff is quantified by the *relative efficiency* defined by

$$\mathcal{E}_r(\hat{Y}) = \lim_{T\to\infty} \frac{B^2}{\mathsf{CPU}[\hat{Y}(T)]\mathsf{Var}[\hat{Y}(T)]},$$

where $\mathrm{CPU}[\hat{Y}(T)]$ denotes the average CPU time of the simulation that produces the T samples.

Note that it is not necessary for the T replications to be independent. However, if there is significant positive correlation between them, then $\mathsf{Var}[\hat{Y}(T)]$ may be very much larger than $\mathsf{Var}[\hat{Y}(1)]/T$, which would have resulted from independent samples. Thus, in addition to having the desired steady state distribution, a good surrogate process should have a smaller (or slightly negative) correlation between successive states than the simple arrival/departure process. This can reduce the variance of the final estimate of the blocking probability by orders of magnitude.

One of the frequently used MCMC methods is the Gibbs sampler [Brémaud, 1999, Fishman, 1996, Ross, 1997]. Section 3.1 in the following describes the standard Gibbs sampler. Sections 4 and 5 then present two enhancements: the *setwise* Gibbs sample, which extends the range of networks which can be analysed, and *filtered* Gibbs sampler, which improves the efficiency of the network performance estimator.

3.1 The standard Gibbs sampler

The Gibbs sampler applies to multi-dimensional state spaces. The key principle is that each transition in the surrogate Markov chain updates only one component, selected either deterministically or randomly, and the associated transition probability is proportional to the readily derived stationary conditional probability for that component given the current values of all other components. This is clearly ideally suited to product form distributions, where each such conditional probability has a very simple form. It is the Gibbs sampler's ability to make large changes to each component, reducing the correlation between samples generated, that leads to its greater efficiency than direct simulation of the arrival and departure of calls.

To present the algorithms for generating state X_{k+1} from X_k for the occupancy vector $\mathbf{n}$ we introduce the following notation. First rewrite $\mathbf{n}$ as $X = (X(1), \ldots, X(NR))$ which is a vector in $\mathbb{N}^{NR}$. It is straightforward to see that $n_{i,r} = X(N(r-1)+i)$. Then define:

$$X(-i) = (X(1), \ldots, X(i-1), X(i+1), \ldots, X(NR)),$$

which is a vector in $\mathbb{N}^{NR-1}$, missing component i. A realization of X is $x \in \mathbb{N}^{NR}$ and $x(-i)$ is similarly defined as $X(-i)$. We also similarly define $X_k(-i)$. Given any $x \in \Omega$ and an index $1 \le i \le NR$, the notation $\pi(\cdot|x(-i))$ is used for the stationary conditional

probability of the ith component given all the others:

$$\begin{aligned}\pi(y|x(-i)) &= \mathsf{P}(X(i) = y|X(-i) = x(-i)) \\ &= \frac{\pi(x_i(y))}{\sum_{x(i)=0}^{C_i^*(x)} \pi(x)},\end{aligned}$$

where $x_i(y)$ denotes the vector x with the scalar y replacing $x(i)$, and $C_i^*(x)$ is the state dependent upper bound for $x(i)$ such that all states in the sum in the denominator lie in Ω.

A *Gibbs Update* is a rule for generating X_{k+1} from X_k, of the form:

(1) Select a coordinate $\sigma_k \in \{1, \dots, NR\}$, independent of X_k.
(2) Set $X_{k+1}(-\sigma_k) = X_k(-\sigma_k)$ and take $X_{k+1}(\sigma_k) \sim \pi(\cdot|X_{k+1}(-\sigma_k))$.

For example, if σ_k are i.i.d. random variables then $\{X_k\}$ forms a Markov chain, while if $\sigma_k = k (\text{mod } \ NR)$, then $\{(X_k, \sigma_k)\}$ forms a Markov chain, as does every NRth sample, $\{X_{NRk}\}$. The key property of Gibbs updates is that if X_k is distributed according to π (denoted $X_k \sim \pi$) then $X_{k+1} \sim \pi$. In other words, the target probability is stationary for the Gibbs sampler.

Once a stationary Markov chain $\{X_k : k = 1, \dots, T\}$ has been constructed by the Gibbs sampler, it can be used to estimate those network performance measures such as the blocking probability etc.. We illustrate this by recalling the model of the circuit-switched network considered in Section 2.1, where $R = 1$ and $\pi(\cdot|X_{k+1}(-\sigma_k))$ is a one dimensional Poisson distribution truncated by (6) where $\mathbf{n}$ corresponds to X. For each $1 \le i \le N$, let

$$P_i(g) = \sum_{d=0}^{g} \frac{\rho_i^d}{d!} \quad g = 1, 2, \dots. \tag{10}$$

Let $Z_j(X) = C_j - \sum_{c \in L_j} a_{jc} X(c)$ be the number of free channels on link j in state X, where $L_j = \{i : a_{ji} \neq 0\}$ is the set of all routes being used on link j. At every step k, let $i = \sigma_k$ and let

$$C_i(X_k) = \min_{j: i \in L_j} (Z_j(X_k)/a_{ji} + X_k(i)) \tag{11}$$

be the maximum allowable number of connections using route i given $X_k(-i)$. Then the required conditional probability satisfies $\mathsf{P}(X_{k+1}(i) \le g) = P_i(g)/P_i(C_i(X_k)), g = 0, \dots, C_i(X_k)$.

Since, as $k \to \infty$, $X_k \sim \pi$, it is possible to estimate B_i by $(1/T)\sum_{k=1}^{T} \mathbf{1}_{\{X_k \in \mathcal{B}_i\}}$, where $\mathbf{1}_{\{A\}} = 1$ if A is true, 0 otherwise. However since updates to component/route i' only change $\mathbf{1}_{\{X_k \in \mathcal{B}_i\}}$ when i and i' share a link, evaluating this sum involves significant unnecessary computation at each step k for all routes that do not share a link with the current updated route. Having evaluated $C_i(X_k)$ and X_{k+1}, it is easy to calculate $\mathbf{1}_{\{X_{k+1} \in \mathcal{B}_i\}} = \mathbf{1}_{\{X_k(i) = C_i(X_k)\}}$ for the component i which is updated at iteration k. Thus B_i can be estimated by

$$Y_i(T) = \frac{1}{T(i)} \sum_{k=1}^{T} y_i(X_k) \mathbf{1}_{\{\sigma_k = i\}} \tag{12}$$

where $y_i(X_k) = \mathbf{1}_{\{X_{k+1} \in \mathcal{B}_i\}}$, and $T(i) = \sum \mathbf{1}_{\{\sigma_k = i\}}$ counts the number of iterations where $\sigma_k = i$. These *local estimates* converge to B_i at rate $\mathcal{O}(T^{-1/2})$ as T increases.

4. SETWISE GIBBS SAMPLER

For a distribution on a manifold, it is often impossible to update one coordinate at a time. For example, in a closed network such as a window flow controlled packet switched network, if only one occupation number, n_σ, is to be updated, the requirement that the number of customers of each class in the network remains constant means that the next state must equal the previous state. The next state still satisfies $X_{k+1} \sim \pi$, since $X_k \sim \pi$, but the process is no longer ergodic. We now propose the *setwise Gibbs sampler* (SGS), which restores ergodicity. We consider only the closed networks to focus our presentation. The SGS algorithm can be seen as an efficient special case of the Generalised Gibbs Sampler [Liu and Sabatti, 2000].

4.1 SGS algorithm

Recall the notation $S_r = \{(i,r), i = 1,\dots,N\}$ defined in Section 2 which gives the indices of the occupancy vector $X \in \Omega$ (another notation is $\mathbf{n} \in \Omega$) corresponding to the customers in class r ($r = 1,\dots,R$). Let $\mathbf{s}_r = \{(s_1,r),\dots,(s_j,r)\}$, $2 \le j \le N$, denote a subset of S_r. Define $X(-\mathbf{s}_r)$ (or $Y(-\mathbf{s}_r)$) to be a subvector of $X \in \Omega$ ($Y \in \Omega$) removing those components indexed by $\mathbf{s}_r$. We will consider a Gibbs-style update of $X(\mathbf{s}_r)$, those components of X indexed by $\mathbf{s}_r$, in constructing a Markov chain $\{X_k, k = 1,2,\dots\}$ for $X \in \Omega$. But first note that the constant occupancy constraint for class r says that the sum $C_r = n_{1,r} + \cdots + n_{N,r} \equiv \sum_{i=1}^{N} X(N(r-1)+i)$ remains constant. Also for $X, Y \in \Omega$ the statement $X(-\mathbf{s}_r) = Y(-\mathbf{s}_r)$ implies

$$\frac{\pi(Y)}{\pi(X)} = \frac{\pi_{\mathbf{s}_r}(Y(\mathbf{s}_r) \,|\, X(-\mathbf{s}_r))}{\pi_{\mathbf{s}_r}(X(\mathbf{s}_r) \,|\, X(-\mathbf{s}_r))}, \tag{13}$$

where $\pi_{\mathbf{s}_r}$ is the conditional distribution of the occupancy subvector indexed by $\mathbf{s}_r$ given $X(-\mathbf{s}_r)$.

The general scheme for a *Setwise Gibbs Update* is a pre-specified set, $\mathfrak{S} = \{\mathbf{s}_r : \mathbf{s}_r \subseteq S_r, r \in (1,2,\dots,R)\}$, and a rule for generating X_{k+1} from X_k for X, of the form:

(1) Select an index set $\mathbf{s}_r^{(k)} \in \mathfrak{S}$, independent of X_k.

(2) Set $X_{k+1}(-\mathbf{s}_r^{(k)}) = X_k(-\mathbf{s}_r^{(k)})$ and take $X_{k+1}(\mathbf{s}_r^{(k)}) \sim \pi_{\mathbf{s}_r^{(k)}}(\cdot | X_{k+1}(-\mathbf{s}_r^{(k)}))$.

The selection of $\mathbf{s}_r^{(k)}$ may be deterministic or random, but each set $\mathbf{s}_r \in \mathfrak{S}$ is assumed to be selected an infinite number of times with probability 1 as $k \to \infty$. Also assume that each S_r can be covered by a sequence of $\mathbf{s}_r$ in $\mathfrak{S}$. This ensures that all the components of $X_k(S_r)$ for each r communicate. Further, note that SGS becomes a standard Gibbs sampler when j in the definition of $\mathbf{s}_r$ equals 1.

To further illustrate the SGS we now take on a case where each set $\mathbf{s}_r \in \mathfrak{S}$ is of the form $\mathbf{s}_r = \{(s_1,r),(s_2,r)\}$. Rather than prohibiting an update, the occupancy constraint now helps by substituting the generation of a two-dimensional random variable with that of a one-dimensional one. This gives updates of the form $n_{s_1,r} \equiv X(N(r-1)+s_1) = Q$ and $n_{s_2,r} \equiv X(N(r-1)+s_2) = C_r - \sum_{i:(i,r)\notin\mathbf{s}_r} n_{i,r} - Q \stackrel{\text{denote}}{=} C_{r,\mathbf{s}_r} - Q$ where Q is a random variable having the distribution determined by $\pi_{\mathbf{s}_r}(\cdot \,|\, X(-\mathbf{s}_r))$. For a closed class of a BCMP network, it can be derived from (3) and (4) that

$$\mathsf{P}(Q=q) \propto \frac{h_{s_1}(n'_{s_1\bullet}+q)}{q!} \frac{h_{s_2}(n'_{s_2\bullet}+C_{r,\mathbf{s}_r}-q)}{(C_{r,\mathbf{s}_r}-q)!} \left(\frac{\rho_{s_1,r}}{\rho_{s_2,r}}\right)^q, \tag{14}$$

where $n'_{s_1\bullet} = \sum_{j\neq r} n_{s_1,j}$ and $n'_{s_2\bullet} = \sum_{j\neq r} n_{s_2,j}$. Note that if nodes (stations) s_1 and s_2 are both single-class nodes ($n'_{s_1\bullet} \equiv n'_{s_2\bullet} \equiv 0$) of a type other than type 3 (IS), then the distribution of Q becomes a truncated geometric distribution. If one of the two nodes is instead of type 3 (IS), then a truncated Poisson distribution results. And if both of the nodes are of type 3 (IS), then a binomial distribution results. All these distributions can be efficiently generated using standard random number generating programs with minor modifications. In principle, SGS can be applied to any general BCMP network model; and the associated transitional distribution in step 2 in SGS can be derived from (3). However, the derivation can sometimes be analytically very complicated. In these situations, other more feasible MCMC algorithms are expected to be developed and will not be pursued here.

The SGS updates are closely related to the true network dynamics; instead of customers moving from one queue to the next one at a time, groups of customers move in batches between queues on their route which need not be consecutive. This extra flexibility reduces the correlation between successive estimates of the quantity of interest (such as queue length or link utilisation), which improves the efficiency of the estimation.

THEOREM 2. *Consider a setwise Gibbs sampler for simulating a Markov chain $\{X_k, k = 1, 2, \ldots\}$ for occupancy vector in a closed BCMP network model. Assume all components of the occupancy vector indexed by S_r, i.e. customer class r, communicate in SGS updates. Also assume the network routing matrix $P = \{p_{(i,r),(j,s)}\}$ is irreducible when restricted to any customer class r. Then the Markov chain $\{X_k, k = 1, 2, \ldots\}$ simulated by the SGS converges to the occupancy stationary distribution defined in Theorem 1.*

This will be proved with the help of the following lemma.

LEMMA 1. *Consider an occupancy vector $\mathbf{n} \in \Omega$ in a closed BCMP network model. Let $\{F, V\}$ be a partition of S_r, the index subset corresponding to customer class r, such that V can be covered by a route of elements of $\mathfrak{S}$ and the cardinalities $|F| \geq 0$ and $|V| \geq 2$. Assume all components of $\mathbf{n}$ indexed by S_r communicate. Then for any j such that $(j, r) \in V$, and target value $t_j \in \{0, \ldots, \bar{C}_F\}$ with $\bar{C}_F = C_r - \sum_{k:(k,r)\in F} n_{k,r}$, it is possible under the randomized setwise Gibbs sampler to attain a state, $\mathbf{m}$ from $\mathbf{n}$, where the occupancy components not indexed by V will be unchanged, $\mathbf{m}(-V) = \mathbf{n}(-V)$, and where the occupancy components indexed by (j, r) equals the target value $m_{j,r} = t_j$.*

PROOF. Call the occupancy components indexed by V "variable" and those by F "fixed". Consider an arbitrary $i_1 \neq j$ satisfying $(i_1, r) \in V$. Note that $n_{i_1,r} + n_{j,r} \leq \bar{C}_F$ is always true. If $n_{i1,r} + n_{j,r} = t_j$, then the setwise Gibbs sampler has a positive probability to reach the target state in one step, which chooses and changes the occupancies at (i_1, r) and (j, r) so that station j has exactly t_j customers of class r by moving all components at (i_1, r).

Next consider the case that no such i_1 exists, but that $n_{j,r} < t_j \leq \bar{C}_F$. we now argue that SGS has a positive probability to transfer $t_j - n_{j,r}$ customers from variable components into the component indexed by (j, r) in a finite number of steps. First, with positive probability SGS updates will select in a finite number of steps $k \geq 2$ components indexed by $\{(i_1, r), \ldots, (i_k, r)\} \subseteq V$, such that $\{(i_1, r), \ldots, (i_k, r)\}$ can be covered by a subset of elements of $\mathfrak{S}$ and

$$n_{i_1,r} + \cdots + n_{i_{k-1},r} + n_{j,r} < t_j, \quad n_{i_1,r} + \cdots + n_{i_k,r} + n_{j,r} \geq t_j.$$

Because all occupancy components indexed by V communicate, again with positive probability the SGS updates will transfer all $n_{i_1,r}$ customers of class r at station i_1 to i_2 without changing any other occupancy components, will then transfer all $n_{i_1,r} + n_{i_2,r}$ customers at i_2 to i_3, and so on until transfer all $n_{i_1,r} + \cdots + n_{i_{k-1},r}$ customers at i_{k-1} to i_k, and transfer only $t_j - n_{j,r}$ customers at i_k to station j. Eventually, $t_j - n_{j,r}$ customers of class r in the variable components can, with positive probability, be transferred to the component indexed by (j, r) with no change in the fixed components.

Finally, if $n_{j,r} > t_j$, then there exists an i_1 such that $(i_1, r) \in V$ and $\{(i_1, r), (j, r)\} \in \mathfrak{S}$. It can be seen that the SGS updates have positive probability to transfer $n_{j,r} - t_j$ customers of class r at station j to station i_1, so that exactly t_j customers of class r are left at station j. This establishes the lemma. ◁

PROOF OF THEOREM 2. Let $P(\mathbf{s}) = (p_{\mathbf{m},\mathbf{n}}(\mathbf{s}))$ be the transition kernel in Ω when all, but only, those coordinates indexed by $\mathbf{s} \in \mathfrak{S}$ are updated according to the Gibbs sampling strategy:

$$p_{\mathbf{m},\mathbf{n}}(\mathbf{s}) = \pi_{\mathbf{s}}(\mathbf{n}(\mathbf{s}) \,|\, \mathbf{m}(-\mathbf{s}))\mathbf{1}_{\{\mathbf{n}(-\mathbf{s})=\mathbf{m}(-\mathbf{s})\}},$$

It is easily shown [Fishman, 1996, sec. 5.15, 16] that the target distribution π is stationary for $P(\mathbf{s})$, that is, $\pi P(\mathbf{s}) = \pi$. Indeed, for any $\mathbf{n} \in \Omega$, from (13)

$$\begin{aligned}
&\sum_{\mathbf{m}\in\Omega} \pi(\mathbf{m}) p_{\mathbf{m},\mathbf{n}}(\mathbf{s}) \\
&\quad = \sum_{\mathbf{m}\in\Omega} \pi(\mathbf{m}) \pi_{\mathbf{s}}(\mathbf{n}(\mathbf{s}) \,|\, \mathbf{m}(-\mathbf{s}))\mathbf{1}_{\{\mathbf{m}(-\mathbf{s})=\mathbf{n}(-\mathbf{s})\}} \\
&\quad = \sum_{\mathbf{m}\in\Omega} \pi(\mathbf{m}) \left(\frac{\pi(\mathbf{n})}{\pi(\mathbf{m})}\right) \pi_{\mathbf{s}}(\mathbf{m}(\mathbf{s}) \,|\, \mathbf{m}(-\mathbf{s}))\mathbf{1}_{\{\mathbf{m}(-\mathbf{s})=\mathbf{n}(-\mathbf{s})\}} \\
&\quad = \pi(\mathbf{n}) \sum_{\mathbf{m}\in\Omega} \pi_{\mathbf{s}}(\mathbf{m}(\mathbf{s}) \,|\, \mathbf{m}(-\mathbf{s}))\mathbf{1}_{\{\mathbf{m}(-\mathbf{s})=\mathbf{n}(-\mathbf{s})\}}.
\end{aligned}$$

For any $\mathbf{n} \in \Omega$, $\sum_{\mathbf{m}\in\Omega} \pi_{\mathbf{s}}(\mathbf{m}(\mathbf{s}) \,|\, \mathbf{m}(-\mathbf{s}))\mathbf{1}_{\{\mathbf{m}(-\mathbf{s})=\mathbf{n}(-\mathbf{s})\}} = 1$ because the conditional probability satisfies the law of total probability on the set of coordinates $\mathbf{s}$. So $\pi(\mathbf{n}) = \sum_{\mathbf{m}\in\Omega} \pi(\mathbf{m}) p_{\mathbf{m},\mathbf{n}}(\mathbf{s})$, as required.

Because π is stationary under $P(\mathbf{s})$ for all $\mathbf{s} \in \mathfrak{S}$, it suffices now to ensure that the successive iterations of a Gibbs sampler will produce an *ergodic* chain, that is, one for which all states are reachable, so that the limit distribution will be the target one: $\lim_{k\to\infty} \mathsf{P}(X_k = \mathbf{n}) = \pi(\mathbf{n})$ for all $\mathbf{n} \in \Omega$. For this, it is sufficient to show that, from any state, $\mathbf{n}$, which is recurrent under true process, any state, $\mathbf{m}$, reachable in one step under true process is reachable under the SGS.

According to the definition of SGS and the condition on P, the (two) components in which $\mathbf{m}$ and $\mathbf{n}$ differ must be in the same irreducible block of the routing matrix, P. Without loss of generality, label the components in the maximal such irreducible block as $(1, r), \ldots, (N, r)$. By hypothesis, all occupancy components in this irreducible block communicate in SGS updates. We show now that starting from $\mathbf{n}$ there is a path of the randomized setwise Gibbs sampler that has positive probability and reaches $\mathbf{m}$ in finite time. That is, we will show how to perform a series of positive probability Gibbs updates that will change the occupancy from $n_{i,r}$ to $m_{i,r}$ for all $i = 1, \ldots, N$.

First, from Lemma 1 it is always possible to construct an update that reaches with

positive probability a state with the target occupancy $m_{N,r}$ of class r at the last station, by changing one or several of the other occupancies of the same class r. Next, reach a state where this occupancy $m_{N,r}$ remains the same but station $N-1$ reaches the target value $m_{N-1,r}$ of class r. By continuing in this fashion it is straightforward that $\mathbf{m}_r = (m_{1,r}, \ldots, m_{N,r})$ is reachable from $\mathbf{n}_r = (n_{1,r}, \ldots, n_{N,r})$ using the setwise Gibbs sampler, by Lemma 1. Thus reachability of the whole state space follows from the randomisation in the updates: the Gibbs sampler will choose the next class r' to update at random, and then chooses one set $\mathbf{s}_{r'} \in \mathfrak{S}$ for the update, also at random. ◁

COROLLARY 1. *Lemma 1 and Theorem 2 also hold for SGS with updates in a deterministic order.*

PROOF. With non-zero probability, intervening updates have no impact. ◁

As with the analysis of the circuit switched model (where there is a single customer class), it is possible to estimate B_i consistently using the fact that the chain satisfies (9), so that

$$B_i = \lim_{T\to\infty} \frac{1}{T} \sum_{k=1}^{T} b_i(X_k), \quad i = 1, 2, \ldots, N.$$

Clearly, for the mean queue length where $b_i(\mathbf{n}) = R^{-1} \sum_{r=1}^{R} n_{i,r}$, if coordinate $N(r-1)+i$ is not updated at iteration k then $X_k(N(r-1)+i) = X_{k+1}(N(r-1)+i)$ and it contributes nothing to the estimate to add this sample: on the contrary, it increases computational effort. Thus, similarly to (12), we use instead the localised estimator of B_i:

$$Y_i(T) = \frac{1}{T(i)} \sum_{k=1}^{T} b_i(X_k) \mathbf{1}_{\left\{i:\exists r \text{ s.t. } (i,r)\in \mathbf{s}_r^{(k)}\right\}},$$

where $T(i) = \sum_{k=1}^{T} \mathbf{1}_{\left\{i:\exists r \text{ s.t. } (i,r)\in \mathbf{s}_r^{(k)}\right\}}$.

If the variance of the occupancy of either component (i,r) or (j,r) is very small, then the state will usually not change significantly when $\mathbf{s}_r^{(k)} = \{(i,r),(j,r)\}$. In BCMP networks, this typically occurs when the expected occupancy is low. Since the resulting correlation in performance estimates will reduce the efficiency, it is advisable to group components together with others of similar expected occupancy.

Each component can be grouped with arbitrarily many other components. In the extreme case, one component could be selected and grouped with every other component of the same class, giving updates a star topology. This can be viewed as converting the closed network into an open network with one fewer dimension, and then using the standard Gibbs sampler, as mentioned in Section 1. However, if the selected component has very little variance, then consecutive estimates can be very highly correlated, as noted in the previous paragraph. Additionally, the original problem may have some mathematical symmetry which can be exploited to simplify the implementation of the sampler; singling out one component to be grouped with every other component may break that symmetry.

4.2 An application of SGS

The setwise Gibbs sampler will be demonstrated by investigating the impact of delay on the utilisation of a window flow control network. To understand this model, consider the

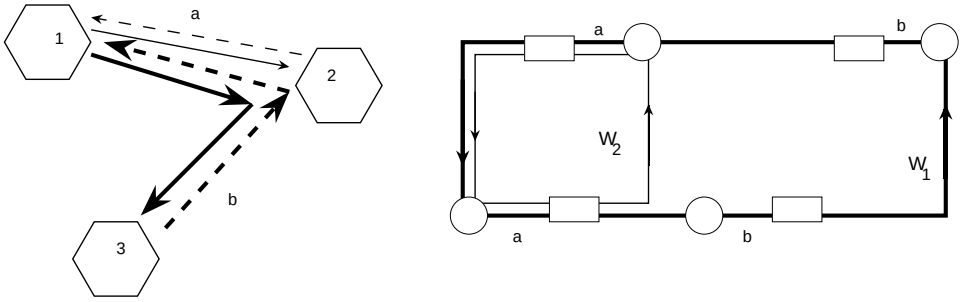


Fig. 1. Two connections for three store-and-forward nodes. Left: the system. Right: the closed network model; the circles denote FCFS queues (packet buffers) and the rectangles denote IS queues with deterministic service times (transmission delays)

case illustrated in Figure 1 to the left, where two possible connections are depicted, one sending packets from node 1 to node 3 via node 2, and the other one only from node 1 to 2. Consider, for example, the class representing the first connection (going to node 3). Packets are processed at node 1 at a certain service rate called the "transmission rate", after which they are sent to node 2 along the link, which takes a fixed amount of time known as the "propagation delay" δ. Next, they receive service at node 2 and are routed towards node 3, where they arrive after δ units of time. Once serviced at node 3 they are released, and an acknowledgement is sent back along the same route in opposite direction, following transmission and propagation delays until they arrive back at the originating node of the connection (this path is shown in dashed arrows in the figure to the left).

A simplified model of the transmission control protocol (TCP) for flow control establishes a window size W_r, $r = 1, 2$ for each connection, and works as follows. Packets at node 1 are sent while there are less than W_1 packets sent for which no acknowledgment has yet been received. As soon as there are W_1 unacknowledged packets the source is stopped until an acknowledgement is received. While the connection is active, there are always packets to be sent, and it is straightforward to see that after an initial transient, the number of packets plus the number of acknowledgements within the system at any given time is always exactly W_1. This is why a connection can be modelled as a closed subnetwork. The closed network model associated with the example is depicted to the right of Figure 1. Each station (link), i, in the model corresponds to either a store and forward node (FCFS) with iid serivce times following distribution G_i, or a propagation delay (IS) where the server is of type 3 with deterministic service times δ. The two classes of customers in the network correspond to the two connections of the example.

The data rate of any given connection is the inverse of the maximal service rate along the (closed) path, which corresponds to the maximum transmission rate. The window size for a connection is set at four times the number of hops in the path. Using a fixed window size, W_r, this models a situation where the window uses the maximum buffer space allowed by the receiver, and cannot increase as the propagation delay grows.

The standard ARPA2 topology, with 21 nodes and 26 links, was used in our experiments. For this network the transmission rate of all FCFS stations is assumed constant and it is expressed in units of $1/\delta$. Studying the impact of delay in network utilization is equivalent to studying the proportion of idle time as a function of the transmission rate (in units of reciprocal propagation time). Application of the SGS matches similar queues

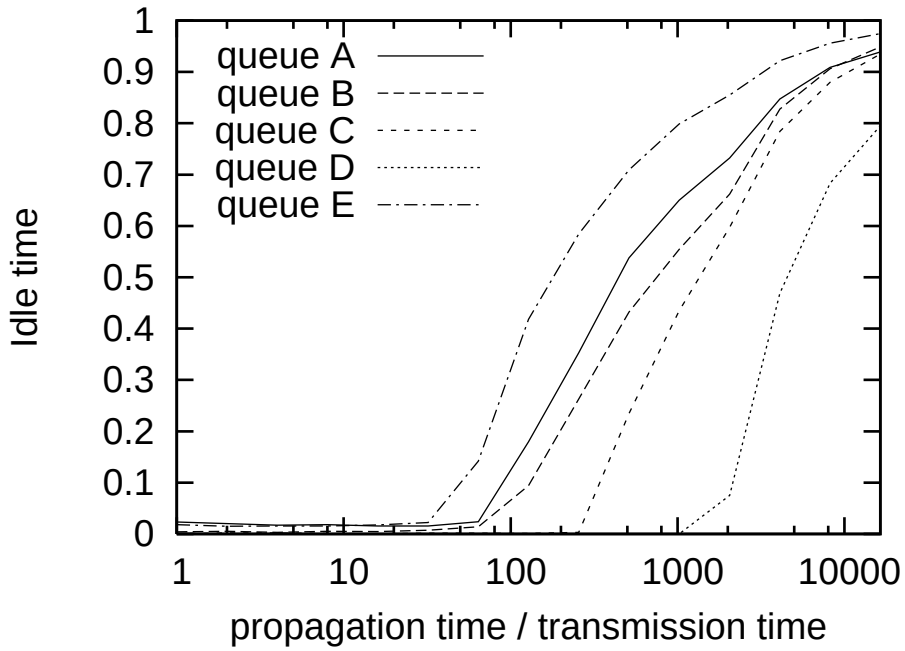


Fig. 2. Fraction of time queues are idle in ARPA2 network as a function of data rate, expressed as mean propagation time normalised by transmission time. This network of FCFS and IS queues requires a technique such as SGS for its analysis.

in the pairs $\mathbf{s} \in \mathfrak{S}$, that is, consecutive updates consider pairs of IS-IS or FCFS-FCFS queues in the network. For each path, at least one IS-FCFS pair and one FCFS-IS are also included in $\mathfrak{S}$ to ensure SGS updates corresponding to each irreducible block of the routing matrix P communicate. Because this network consists of a mixture of FCFS and IS nodes, these results, shown in Figure 2, could not be generated by, for example, Buzen's algorithm [Buzen, 1973].

5. FILTERED GIBBS SAMPLER

Consider a Markov chain $\{X_k\}$ and an estimator

$$\bar{B}_T = \frac{1}{T}\sum_{k=1}^{T} b(X_k),$$

for a sample performance b. The method of *filtered Monte Carlo* is based on conditioning at each stage [Ross, 1997, sec. 8.3]:

$$\bar{B}'_T = \frac{1}{T}\sum_{k=1}^{T} \mathsf{E}[b(X_{k+1})|X_k].$$

It is expected that conditioning would reduce estimator's variance, i.e, $\mathsf{Var}[\bar{B}'_T] \leq \mathsf{Var}[\bar{B}_T]$. Filtered Monte Carlo is closely related to "inverse convolution" [Lassila and Virtamo, 2000].

The *Filtered Gibbs Sampler* (FGS) combines the filtering with the distribution of the estimation via the local estimates as follows.

Consider a network model with a single class of customers. Suppose the Gibbs updates of the corresponding Markov chain $\{X_k\}$ use the set of components $\{\sigma_k\}$, with a deterministic assignment of period p that updates every coordinate at least once in p iterations. The FGS estimator based on b is of the form:

$$\hat{Y}(T) = \sum_{i=1}^{N} \hat{Y}_i(T) = \frac{1}{T} \sum_{i=1}^{N} \sum_{k=1}^{T} \left(\frac{\nu(i)}{p} \right) y_{\sigma_k,F}(X_k), \tag{15}$$

where $y_{i,F}(x) = \mathsf{E}[b(X_{k+1}) \mid X_k]\mathbf{1}_{\{\sigma_k=i\}}$ and $\nu(i) > 0$ is the number of times that coordinate i is updated in one period. Since $T(i)/T \to \nu(i)/p$ as $T \to \infty$ where $T(i)$ is the number of times coordinate i is updated in the chain $\{X_k, k = 1, \ldots, T\}$, it follows that under the FGS, $\hat{Y}(T) \to B = \mathsf{E}(\bar{B}_T)$ [Vázquez-Abad and Andrew, 2000].

Applying this approach to the circuit switched network, where B is assumed to be the network blocking probability, it requires evaluation of the conditional probabilities:

$$\begin{aligned} \mathsf{P}(X_{k+1} \in \mathcal{B}_i | X_k) &= \frac{P_i(C_i(X_k)) - P_i(C_i(X_k) - 1)}{P_i(C_i(X_k))} \\ &\equiv g(C_i(X_k); \rho_i) \end{aligned} \tag{16}$$

where $P_i(\cdot)$ are given in (10) and $C_i(X_k)$ is given in (11). When it is feasible to precompute $g(\cdot;\cdot)$, calculation of the probabilities is as simple as reading a table. This is the case when there is a small number of distinct loads, ρ_j, in the network. Following this discussion, an FGS estimator of the blocking probability B is

$$\hat{Y}(T) = \frac{N}{T} \sum_{k=1}^{T} \left(\frac{\lambda_{\sigma_k}}{\lambda} \right) y_{\sigma_k,F}(X_k), \tag{17}$$

where $y_{i,F}(x) = g(C_i(x); \rho_i) = \mathsf{P}(X_{k+1} \in \mathcal{B}_i | X_k = x)$.

Note that in addition to estimating blocking probabilities, with a suitable choice of function g, other performance statistics may be estimated, such as mean queue size.

Unlike most exact techniques whose complexity is $\mathcal{O}(C)$ when the number of channels on each link is a constant C and the numbers of routes and links are fixed, the complexity per iteration of the FGS is $\mathcal{O}(1)$ as the capacity per link increases, assuming the time to generate a single random number is independent of C. However, its primary strength is that it is $\mathcal{O}(N \max_i |L_i|)$ as the numbers of routes and links increase, where L_i is the number of links used by route i, and N is the number of routes. The complexity of all known exact methods is exponential in the number of links.

5.1 Networks used for testing FGS

The performance of FGS will be presented in Sections 5.2 and 5.3, where it is tested on the following circuit switched network topologies:

(a) Mesh-torus: a rectangular grid with each node connected to four neighbours, wrapping at the edges. Components of the state vector $\mathbf{n}$ are the numbers of current calls on a route. In the experiments, the load on all routes was equal. Static shortest path routing ensured a constant number of routes used each link.

(b) Cellular: Spatial reuse constraints in cellular networks with dynamic channel assignment produce "cliques" of cells with a maximum aggregate number of calls [Everitt and Macfadyen, 1983]. These cliques are analogous to links, while cells correspond to routes. The networks considered here employ a hexagonal grid of cells, and cliques consist of groups of three mutually adjacent cells.

In addition to these, FGS was tested on closed BCMP networks such as the packet switched networks with window flow control, where it was shown to cause minimal improvement. We recommend the use of SGS but not FGS for such networks, and so the specific numerical results have been omitted for brevity.

5.2 Correlation

For a single random variable, $\mathsf{Var}[Y] = \mathsf{Var}[\mathsf{E}[Y|Z]] + \mathsf{E}[\mathsf{Var}[Y|Z]]$, and conditioning always entails a variance reduction. However, it is not always the case for Markov chains that conditioning always reduces the variance of the estimator, due to the correlation structure [Ross, 1997, sec. 8.3]. Explicitly, for the estimators $\bar{B}_T$ and $\bar{B}'_T$ at the beginning of Section 5,

$$\mathsf{Var}[\bar{B}_T] = \frac{1}{T}\mathsf{Var}[b(X_2)] + \frac{2}{T^2}\sum_{j=1}^{T-1}\sum_{k=1}^{T-j}\mathsf{Cov}[b(X_j), b(X_{j+k})] \tag{18}$$

$$\begin{aligned}\mathsf{Var}[\bar{B}'_T] &= \frac{1}{T}\mathsf{Var}[\mathsf{E}\{b(X_2)|X_1\}] \\ &\quad + \frac{2}{T^2}\sum_{j=1}^{T-1}\sum_{k=1}^{T-j}\mathsf{Cov}[\mathsf{E}\{b(X_{j+1})|X_j\}, \mathsf{E}\{b(X_{j+k+1})|X_{j+k}\}].\end{aligned} \tag{19}$$

Although the first term in the righthand side of (19) is smaller than the corresponding term in (18), the second term in the righthand size of (19) may be larger than that in (18). Consequently, it may not always be true that $\mathsf{Var}[\bar{B}'_T] \leq \mathsf{Var}[\bar{B}_T]$.

The variance $\mathsf{Var}[\bar{B}'_T]$ can be estimated using batch means (grouping runs of K samples to obtain approximately independent estimates (e.g. [Alexopoulos and Seila, 1998])). The impact of the correlation can be quantified by the ratio of $\mathsf{Var}[\bar{B}'_T]$ to $\mathsf{Var}[b(X_1)]/T$, the variance estimated by treating individual samples as independent.

Figure 3 shows the results of using batches of size $K = 3 \times 10^6$ (10000 for each of the 300 routes) in a 5×5 mesh-torus, for both the FGS and the standard Gibbs sampler. (Note that these only show the impact of correlation, and do not compare the actual variances of FGS and the standard Gibbs sampler.) These results show that the covariance term has minimal impact except when blocking is very high. This justifies ignoring its effect in arguing that filtering should reduce the variance of the estimated blocking probability. However, when blocking is high, the variance of the final blocking estimator using FGS is up to an order of magnitude higher than would be predicted by treating samples as independent. Since this does not occur without filtering, the benefit due to filtering would be overestimated in the case of high blocking if batch means were not used. This effect is greatest for networks with many channels per link, as they have a higher occupancy per channel for a given blocking probability, due to increased trunking efficiency.

Figure 3 suggests that, for high blocking, the true variance of the standard Gibbs sampler is actually less than would be predicted by treating samples as independent. This indicates a negative correlation between samples, but the reason for this is unclear.

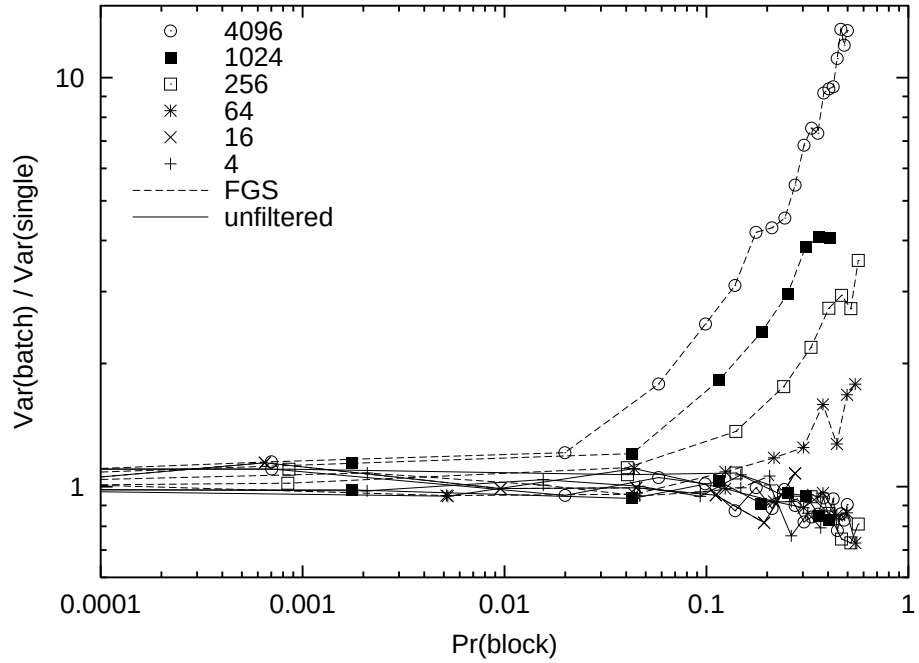


Fig. 3. Ratio of variance of a 4-hop route estimated by batch means, $K = 3 \times 10^6$, divided by $\mathsf{Var}[b(X_1)]/T$. 5×5 mesh-torus network, 4 to 4096 channels per link.

5.3 Improvement due to filtering

Numerical results show that filtering causes negligible increase in efficiency for closed BCMP networks. However the gains can be quite significant for $M/G/k/k$ networks, such as the circuit switched networks described in Section 2.1. This will now be quantified.

Consider a single link of C channels, used by N routes of load ρ each, and assume that the blocking probability, B, is low. As was demonstrated in Section 5.2, for low blocking the variance of FGS is dominated by the variance of each update, rather than the covariance introduced by the Markov structure. Let

$$D_A = \sum_{j=0}^{C} A^j/j!$$

and note that for $B << 1$ (small A or large C), $D_A \approx e^A$. Denote the Erlang loss function by

$$E_k(\rho) = \frac{\rho^k/k!}{\sum_{j=0}^{k} \rho^j/j!}.$$

The blocking probability of the link is $B = E_C(N\rho)$, and the variance of the Gibbs sampler estimator is $B - B^2$.

For low B, the occupancy of the N routes is well approximated by independent Poisson variables. Each FGS update will see the link filled with the aggregate of the $N - 1$ other

routes, which is Poisson with rate $(N-1)\rho$. Thus with probability

$$\frac{((N-1)\rho)^{C-j}/(C-j)!}{D_{(N-1)\rho}},$$

the FGS estimate is $E_j(\rho)$. Thus

$$\begin{aligned}&\mathsf{Var}[FGS] + B^2\\ &= \sum_{j=0}^{C} \frac{((N-1)\rho)^{C-j}/(C-j)!}{D_{(N-1)\rho}} \left(\frac{\rho^j/j!}{\sum_{k=0}^{j}\rho^k/k!}\right)^2,\end{aligned}$$

and

$$\begin{aligned}&\frac{\mathsf{Var}[FGS] + B^2}{\mathsf{Var}[GS] + B^2}\\ &= \frac{D_{N\rho}}{D_{(N-1)\rho}} \sum_{j=0}^{C} \frac{C!}{(C-j)!}\frac{(N-1)^{C-j}}{N^C}\frac{\rho^j/j!}{(\sum_{k=0}^{j}\rho^k/k!)^2}\\ &= e^{\rho}\left(\frac{N-1}{N}\right)^C \sum_{j=0}^{C}\binom{C}{j}\left(\frac{1}{N-1}\right)^j \frac{E_j(\rho)}{\sum_{k=0}^{j}\rho^k/k!} \qquad (20)\\ &\to \left(\frac{N-1}{N}\right)^C \text{ as } \rho \to 0. \qquad (21)\end{aligned}$$

This analysis extends easily to unequal loads.

Figure 4 shows the increase in the relative efficiency of the FGS compared to a standard Gibbs sampler for 5×5 and 200×200 cellular networks ($N = 3$) and 5×5 and 7×7 mesh-torus networks ($N = 15, 42$). The improvements are very similar for both cellular networks, while the improvements differ for the two mesh-torus networks. This is because cellular networks have $N = 3$ cells per clique, while the values of N differ greatly for the mesh-tori.

Since CPU time is approximately unchanged by filtering, the ratio of relative efficiencies is given by $\mathsf{Var}[FGS]/\mathsf{Var}[GS]$, which is bounded between 1 and the expression in (20). As suggested by (20), the gain in relative efficiency due to conditioning increases as the capacity of the links increases. It is a minimum in the range of blocking probabilities which are of greatest interest, around 10^{-2} to 10^{-3}. However, even in this range the gains are substantial for networks with many channels per link.

5.4 Confidence intervals

It is important to know when to terminate a simulation. A typical criterion is achieving a sufficiently small confidence interval, based on the estimated variance of the probability estimator. Traditional Monte Carlo techniques yield Bernoulli outcomes for probability estimates. If the probability to be estimated is so small that no "successes" have been observed, the variance estimate is zero. This invalidates the resulting confidence interval, but such vanishing confidence intervals are easily identified and discarded, allowing the simulation to continue.

In contrast, techniques such as importance sampling and the FGS produce samples from an unknown and highly skewed distribution. This makes it possible to underestimate the

variance of the estimator by orders of magnitude if insufficient samples are taken, causing premature termination of the simulation. Figure 5 shows the estimate of blocking after each iteration, and also the value ("traditional upper") which is usually used as the upper limit of a confidence interval, i.e., the estimated mean plus twice the estimated standard deviation. After a small number of samples, this "2σ" upper limit is below the true mean value for much more than 2.5% of the time (which it would be in the Gaussian case), and is ineffective as a confidence bound.

To see why this occurs, consider the terms in

$$B_i = \sum_{j=0}^{C} g(j;\rho_i)\mathsf{P}(C_i(X) = j), \tag{22}$$

with $g(j;\rho_i)$ defined by (16). Without filtering, $y_j(X) = 1$ if $C_i(X) = X(i)$ and 0 otherwise. If at least one non-zero sample is generated then the variance estimator would, with high probability, be of the correct order of magnitude. If all samples are 0, it is clear that the sample variance (zero) is not a true indication of the error. However, this is not the case for highly skewed continuous distributions. There are many non-zero terms in (22) which have a high probability, but make very little contribution to the sum due to small values of $g(j;\rho_i)$. Thus if the sample size is too small, the sample mean and variance can be very much smaller than the ensemble values, without any tell-tale zeros to indicate their unreliability. For the FGS to be of practical value, it is necessary to be able to detect when an estimate is statistically unreliable.

For a better indication of the accuracy of the result, consider the individual terms ("partial expectations") of (22). Figure 6 plots these terms against the cumulative probability for a 37-cell cellular network with 64 channels and 12 Erlangs per cell, as studied in [Vázquez-Abad, Andrew, and Everitt, 2002]. (As $\mathsf{P}(C_i < j)$ is monotonic in j, the horizontal axis is simply a non-linear scale for j.)

Since $g(j;\rho_i)$ is known, it suffices to estimate $\mathsf{P}(C_i(X) = j)$, or those for which $g(j;\rho_i)\mathsf{P}(C_i(X) = j)$ is a significant fraction of B. Because these terms decay rapidly for $j < \text{argmax}_j(g(j;\rho_i)\mathsf{P}(j))$, as seen in Figure 6, it is possible to determine by inspection when all "significant" terms have been estimated with sufficient confidence.

To quantify this, assume that the sample contains enough points to capture the peak of the probability distribution, which requires orders of magnitude less data than capturing the peak of the partial expectation. (Note the different scales in Figure 6.) Let m be the smallest value such that $\mathsf{P}(C_i = m)$ can be reliably estimated from the sample, and for $j \geq m$, let $p_{i,j}$ be the sample estimate of $\mathsf{P}(C_i = j)$. For $j < m$, conservatively approximate the tail as $\mathsf{P}(C_i = j) \approx p_{i,j} \equiv p_{i,m}\Delta_i^{j-m}$, where Δ_i is fitted to the sample data. In this paper,

$$\Delta_i = \sqrt[h]{\frac{\sum_{j=0}^{h-1} p_{i,m+j}}{\sum_{j=0}^{h-1} p_{i,m+h+j}}},$$

where m is estimated by the smallest value of $C_i(X)$ observed more than once in the simulation, and h is such that $m + 2h$ is the fourth smallest such value.

Ignoring correlations (Section 5.2), the variances of the estimates $p_{i,j}$ based on T sam-

ples, and $\mathsf{Var}[\hat{Y}_i(T)]$ can then be approximated by

$$\begin{aligned} \hat{V}_i(j) &= p_{i,j}(1-p_{i,j})/T, \\ \hat{V}_i &= \sum_{j=0}^{C} (g(j;\rho_i))^2 \hat{V}_i(j). \end{aligned} \tag{23}$$

The curve "conservative" in Figure 5 plots $\hat{B} + 2\sqrt{\hat{V}}$ where $\hat{B} = \hat{Y}(T)$ is a weighted sum of $\hat{Y}_i(T)$'s and $\hat{V}$ is the estimated variance of $\hat{B}$ using $\hat{V}_i$'s. It is clearly overly conservative for very small sample sizes, since $p_{i,j}$, $j < m$, are very conservative. However, if the sample is large enough for B to be suitably accurate, then the bound becomes usably tight.

6. CONCLUDING REMARKS

The Gibbs sampler has been applied to the broad class of BCMP queueing networks, including both closed queueing networks and $M/G/k/k$ networks as important special cases. For closed BCMP networks, the setwise Gibbs sampler (SGS) has been proposed, and the Markov chain generated by SGS for simulating the network occupancies has been proved to converge to the network stationary distribution. For $M/G/k/k$ networks, the filtered Gibbs sampler (FGS) is used which not only outperforms the standard Gibbs sampler, but also has its relative efficiency grow with problem size and improve under heavy load. However, filtering in FGS provides relatively little reduction in variance for closed BCMP networks using window flow control, when the load per route is very low.

ACKNOWLEDGMENT

The authors wish to thank the editor and three anonymous referees whose comments and suggestions have led to a significant improvement of the presentation of this paper.

References

Alexopoulos, C., and A. Seila. 1998. Output data analysis. In *Handbook of Simulation*, ed. J. Banks, chapter 7, 225–272. New York, NY: John Wiley and Sons.

Andrew, L. L. H., and F. J. Vázquez-Abad. July, 2002. Filtered Gibbs sampler for estimating blocking in product form networks. In *Proc. IASTED Wireless and Optical Communications*, 527–532, Banff, Canada.

Baskett, F., K. M. Chandy., R. R. Muntz., and F. G. Palacios. 1975. Open, closed, and mixed networks of queues with different classes of customers. *J. ACM*, 22(2):248–260.

Berger, A. W., and W. Whitt. 1998. Effective bandwidths with priorities. *IEEE/ACM Trans. Networking*, 6(4):447–460.

Boucherie, R. J., and M. Mandjes. 1998. Estimation of performance measures for product form cellular mobile communications networks. *Telecommunication Systems*, 10:321–354.

Brémaud, P. 1999. *Markov Chains, Gibbs Fields, Monte Carlo Simulation, and Queues*. Texts in Applied Mathematics, 31, New York, NY: Springer.

Buzen, J. P. 1973. Computational algorithms for closed queueing networks with exponential servers. *Comm. ACM*, 16:527–531.

Chao, X., M. Mayazawa., and M. Pinedo. 1999. *Queueing Networks: Customers, Signals and Product Form Solutions*. Chichester, UK: John Wiley and Sons.

Choudhury, G. L., K. K. Leung., and W. Whitt. 1995. An inversion algorithm to compute blocking probabilities in loss networks with state-dependent rates. *IEEE/ACM Trans. Networking*, 3(5):585–601.

Coleman, J. L., W. Henderson., and P. G. Taylor. 1994. A convolution algorithm for calculating exact equilibrium distributions in resource allocation problems with moderate user interference. *IEEE Trans. Commun.*, 42(2/3/4):1106–1111.

Conway, A. E., and D. E. O'Brien. 1993. Hybrid analysis of response time distributions in queueing networks. *IEEE Trans. Commun.*, 41(7):1091–1101.

Everitt, D., and N. W. Macfadyen. 1983. Analysis of multicellular mobile radiotelephone systems with loss. *Br. Telecom Technol. J.*, 1(2):37–45.

Fishman, G. S. 1996. *Monte Carlo: Concepts, Algorithms, and Applications*. New York, NY: Springer-Verlag.

Gamerman, D., and H. F. Lopes. 2006. *Markov Chain Monte Carlo*. Chapman Hall.

Jordan, S. 1995. A continuous state space model of multiple service, multiple resource communication networks. *IEEE Trans. Commun.*, 43(2/3/4):477–484.

Kelly, F. P. 1991. Effective bandwidths at multi-class queues. *Queue. Syst.*, 9:5–16.

Kelly, F. P., P. B. Key., and S. Zachary. 2000. Distributed admission control. *IEEE J. Select. Areas Commun.*, 18(12):2617–2628.

Kendall, W. S., J. S. Wang., and F. Liang. 2005. *Markov Chain Monte Carlo: Innovations And Applications*. World Scientific Publishing Company.

Kind, J., T. Niessen., and R. Mathar. 1998. Theory of maximum packing and related channel assignment strategies for cellular radio networks. *Math. Meth. Op. Res.*, 48(1):1–16.

Knessl, C., and C. Tier. 1998. Asymptotic approximations and bottleneck analysis in product form queueing networks with large populations. *Perf. Eval.*, 33(4):219–248.

Lassila, P., and J. Virtamo. 2000. Nearly optimal importance sampling for Monte Carlo simulation of loss systems. *ACM Trans. Model. Comput. Simul.*, 10(4):326–347.

Lassila, P. E., and J. T. Virtamo. 1998a. Efficient Monte Carlo simulation of product form systems. In *Proc. Nordic Teletraffic Seminar (NTS) 14*, 355–366, Copenhagen, Denmark. Available from `http://keskus.hut.fi/tutkimus/cost257/publ/efmcsim.pdf`.

Lassila, P. E., and J. T. Virtamo. 1998b. Variance reduction in Monte Carlo simulation of product form systems. *Electron. Lett.*, 34(12):1204–1205.

Liu, J. S., and C. Sabatti. 2000. Generalised Gibbs sampler and multigrid Monte Carlo for Bayesian computation. *Biometrika*, 87(2):353–369.

Mitra, D., and J. A. Morrison. 1994. Erlang capacity and uniform approximations for shared unbuffered resources. *IEEE/ACM Trans. Networking*, 2(6):558–570.

Mitra, D., J. A. Morrison., and K. G. Ramakrishnan. 1999. Optimization and design of network routing using refined asymptotic approximations. *Perf. Eval.*, 36-37:267–288.

Neal, R. M. 2003. Slice sampling. *Ann. Stat.*, 31(3):705–767.

Pallant, D. L., and P. G. Taylor. 1995. Modeling handovers in cellular mobile networks with dynamic channel allocation. *Operations Research*, 43(1):33–42.

Reiser, M. 1979. A queueing network analysis of computer communication networks with window flow control. *IEEE Trans. Commun.*, 27:1199–1209.

Ross, K. W. 1995. *Multiservice loss models for broadband telecommunication networks*. Berlin, Germany: Springer-Verlag.

Ross, K. W., D. H. K. Tsang., and J. Wang. 1994. Monte Carlo summation and integration

applied to multiclass queuing networks. *J. ACM*, 41(6):1110–1135.

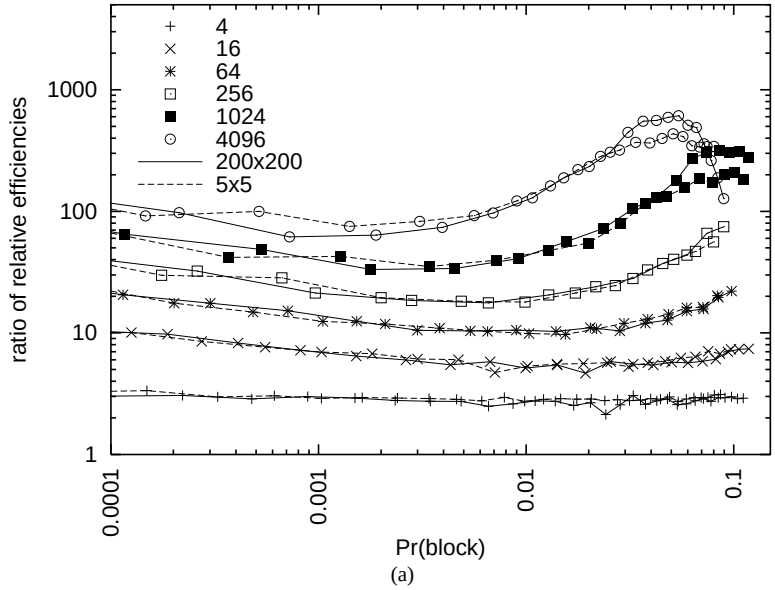
Ross, S. M. 1997. *Simulation*. 2nd ed. Boston, MA: Academic Press.

Vázquez-Abad, F., and L. G. Mason. 1999. Decentralized adaptive isarithmic flow control for high speed data networks. *Operations Research*, 47(6):928–942.

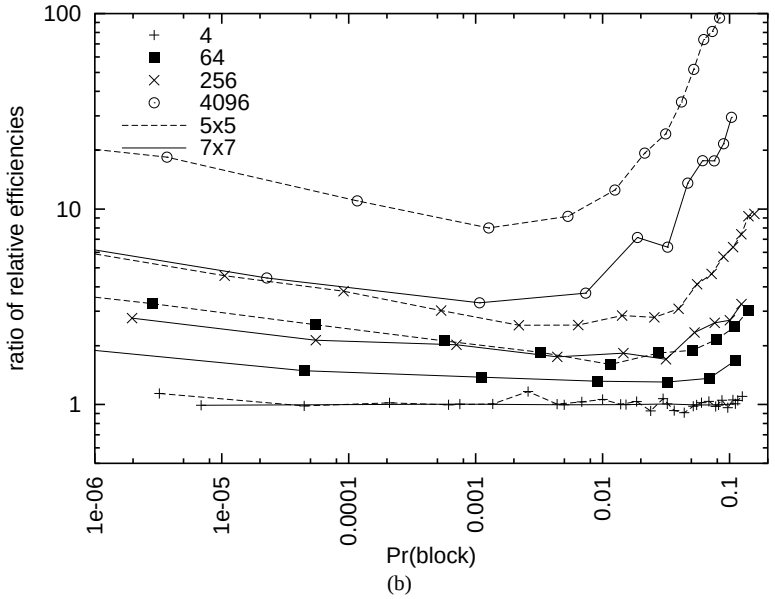
Vázquez-Abad, F. J., and L. L. H. Andrew. May, 2000. Filtered Gibbs sampler for estimating blocking probabilities in WDM optical networks. In *Proc. 14th European Simulation Multiconference*, ed. D. Landeghem, 548–555, Ghent, Belgium. Society for Computer Simulation.

Vázquez-Abad, F. J., L. L. H. Andrew., and D. Everitt. 2002. Estimation of blocking probabilities in cellular networks with dynamic channel assignment. *ACM Trans. Model. Comput. Simul.*, 12:54–81.

Received ...

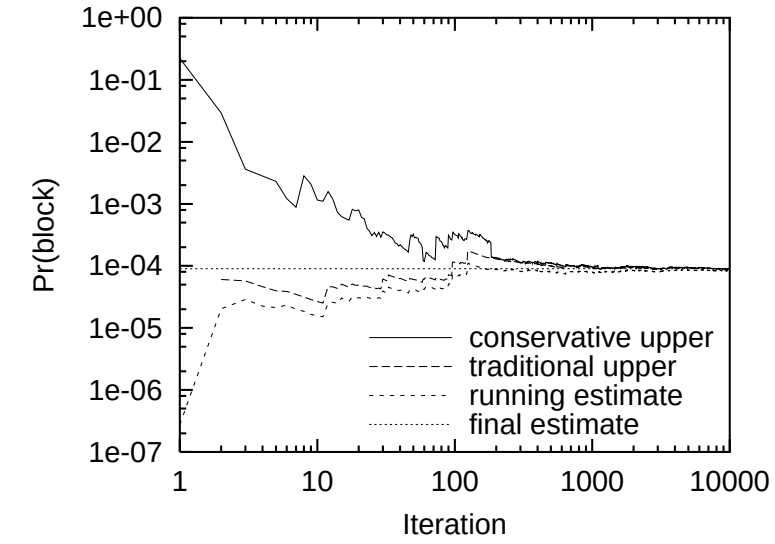


(a)

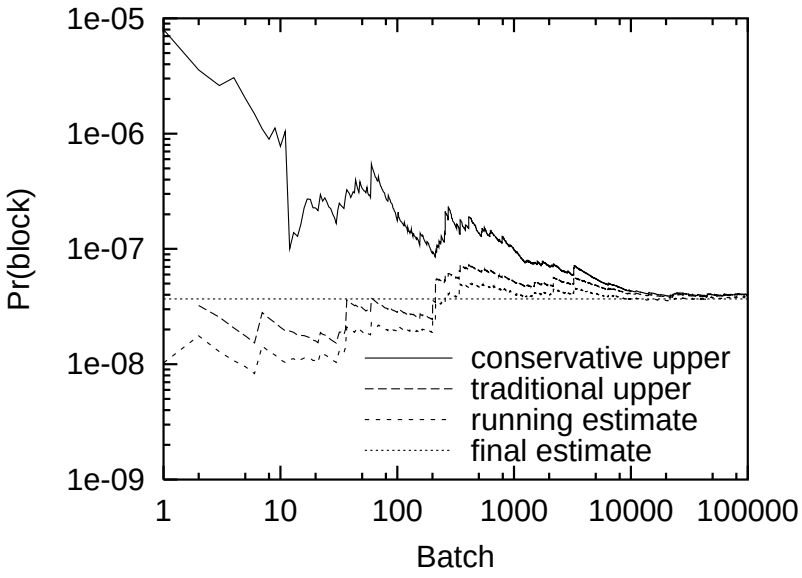


(b)

Fig. 4. Ratio of efficiency of filtered to standard Gibbs sampler with 4 to 4096 channels per link for (a) 5×5 and 200×200 cellular (b) 5×5 and 7×7 mesh-torus networks.



(a)



(b)

Fig. 5. Estimated upper bounds on B: $\hat{B} + 2\sigma$ and the conservative estimator of (23) for a 3×3 mesh with 64 channels per link. (a) 13 Erlangs per route, simple variance (b) 10 Erlangs per route, batches of 100

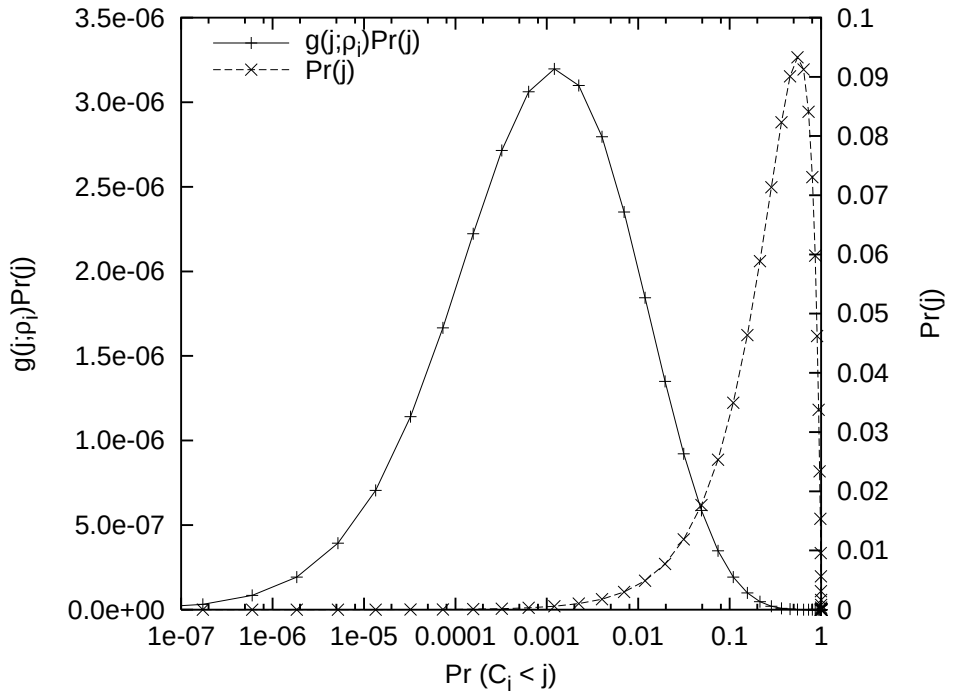


Fig. 6. Comparison of partial expectations and state probabilities.