A New Method for Approximating Blocking Probability in Overflow Loss Networks

Eric W.M. Wong, Andrew Zalesky, Zvi Rosberg
and Moshe Zukerman

Abstract

In this paper, we present a new one-moment approximation for estimating steady-state blocking probability in overflow loss networks. Our approximation is based on regarding an overflow loss network as if it were operating under a fictitious preemptive priority regime. We show that for a certain pedagogical example, our approximation offers two advantages over Erlang's fixed point approximation (EFPA): 1) it can be computed recursively; and, 2) it yields a more accurate estimate of blocking probability. Based on empirical evidence, it is hypothesized that our approximation reduces the independence error as well as the Poisson error, which are inherent to EFPA. To demonstrate the versatility of our approximation, it is extended to the case of circuit-switched networks using alternative routing. Empirical results suggest that for a symmetric fully-meshed circuit-switched network, our approximation is more accurate than EFPA.

1 Introduction

Overflow loss networks form a large and important class of loss networks. They feature prevalently in stochastic models of many computer and telecommunications networks. The classic example is that of circuit-switched networks using alternative routing. Other examples include telephony call centers [4], optical networks [27], and multiprocessor systems with one redundant processor that can be used to alleviate congestion on active processors [10]. Roughly

* Eric Wong is with the Department of Electronic Engineering, City University of Hong Kong, Hong Kong SAR, China.
Andrew Zalesky and Moshe Zukerman are with the Centre for Ultra-Broadband Information Networks (CUBIN), Department of Electrical and Electronic Engineering The University of Melbourne, Melbourne, Vic. 3010, Australia.
Zvi Rosberg is with the Department of Communication Systems Engineering, Ben Gurion University, Beer-Sheva, 84105, Israel.

speaking, a loss network is classed as an *overflow* loss network if calls (jobs) that have been blocked at one server group are not simply blocked for good but are permitted in some circumstances to overflow to another server group.

Stochastic modeling of overflow loss networks is usually in terms of a multidimensional Markov process with finite state-space. Unlike many non-overflow loss networks, the state distribution generally does not admit a product-form solution. Although the state distribution can in principal be computed by numerically solving a set of balance equations, this approach must be ruled because the state-space is usually of an unmanageable dimension.

Approximations therefore play an crucial role in estimating steady-state blocking probability in overflow loss networks. The simplest yet crudest approach to estimating steady-state blocking probability in an overflow loss network proceeds via a one-moment approximation in which stream i is characterized solely in terms of its mean m_i. That is, m_i is the mean of the distribution of the number of busy servers on an infinite server group that is offered stream i, which is commonly referred to as the *offered intensity* of stream i. All streams offered to a common server group comprising N servers are pooled together to form a combined stream that offers an intensity of $\sum_i m_i$. The steady-state blocking probability perceived by the combined stream as well as each marginal stream i comprising the combined stream is estimated by $\mathbf{E}\Big(\sum_i m_i, N\Big)$, where

$$\mathbf{E}(a, N) = \frac{a^N}{N!}\left(\sum_{i=0}^{N}\frac{a^i}{i!}\right)^{-1}, \quad N \in \mathbb{N}, a \geq 0, \tag{1}$$

expresses the blocking probability in an $M/M/N/N$ queue offered intensity a, and is commonly referred as the Erlang B formula.

The overflow of each marginal stream i may then go on to offer an intensity of $m_i\mathbf{E}\Big(\sum_i m_i, N\Big)$ to a subsequent server group. Usually referred to as *Erlang's fixed-point approximation* (EFPA) in its most general form, this approximation was conceived by Cooper and Katz [2] in 1964 for the analysis of circuit-switched networks and has remained a cornerstone of network performance evaluation even to this day. See [1,5,9,15,14,19–21,24,25] and references therein for applications of EFPA.

It is well-known that EFPA may be inaccurate for *overflow* loss networks. The inaccuracy of EFPA in the context of overflow loss networks is usually attributable to two distinct sources of error:

(1) EFPA characterizes the traffic offered by any stream as if it were a Poisson process when in fact the traffic offered by an overflow stream is of

greater peakedness[1] relative to a Poisson process. This is referred to as the *Poisson error*.

(2) EFPA calculates the distribution of the number of busy servers on a server group as if it were mutually independent of any other server group when in fact they may be statistically dependent. This is referred to as the *independence error*.

Numerous approaches have been suggested to strengthen EFPA by combatting the presence of one or the other of these two errors. Strengthening EFPA to combat the Poisson error is usually accomplished by characterizing each stream in terms of its peakedness as well as its mean in an approach referred to as moment-matching. One of many examples is the work of Katz [13] in which EFPA is strengthened via the use of Wilkinson's method [23]. Another example is that of Kuczura and Bajaj [17]. Although combatting the independence error has not received as much attention, it is of no less importance, and was first considered by Holtzman [11]. (See the extended version of this paper [26] for a comprehensive survey of moment-matching approximations.)

In this paper, we present a new approximation for estimating steady-state blocking probability in overflow loss networks, which is fundamentally different from EFPA and its strengthened formulations. It is based on regarding an overflow loss network as if it were operating under a fictitious preemptive priority regime. In this fictitious regime, each stream is classified according to the number of server groups at which it has sought to engage a server but found all servers busy; that is, the number of times it has overflowed. The key is to suppose a stream that has overflowed n times is given strict preemptive priority over a stream that has overflowed m times, $n < m$. As it shall be seen, this preemptive priority regime gives priority to streams that most closely resemble a Poisson stream (peakedness closest to unity) and least violate the independence assumption.

A model of a simple overflow loss network shall be defined in the next section, which shall serve as an pedagogical example to facilitate the presentation of our approximation. It shall then be empirically shown that EFPA as well as its strengthened formulations yield a poor estimate of steady-state blocking probability for this model. In Section 3, the new approximation shall be introduced, some of its numerical properties shall be addressed and we shall conjecture that it yields a more accurate estimate of steady-state blocking probability than EFPA. Section 4 shall describe the rationale underpinning our new approximation, while Section 5 shall exhibit the versatility of our

[1] The peakedness of a stream is defined as the variance-to-mean ratio of the distribution of the number of busy servers on an infinite server group to which the stream is offered and is usually denoted by Z. The peakedness of a Poisson stream is unity, while the peakedness of an overflow stream is always greater than unity.

approximation by considering its extension to circuit-switched networks using alternative routing. Empirical results shall be presented that suggest for a symmetric fully-meshed circuit-switched network, our approximation is more accurate than EFPA.

2 An Overflow Loss Network Model

We consider the following simplified model of an overflow loss network that arose during the study of a video-on-demand distributed-server network. The network comprises N cooperative and identical servers. Calls[2] initiated by users are offered to each server according to an independent time-homogeneous Poisson processes of intensity a. A call that arrives at a busy server overflows to one of the other $N-1$ servers with equal probability and without delay. A call continues to overflow as such until either: it encounters an idle server in which case it engages that server until its service period is complete; or, it has sought to engage all N servers exactly once but found all N servers busy in which case it is blocked and never returns. The search for an idle server is conducted instantly and referred to as a *random hunt*. Service periods are independent and identically distributed according to an exponential distribution with normalized unit mean.

An n-call is defined as a call that overflows n times before engaging the $(n+1)$th server of its random hunt. Strictly speaking, an n-call is therefore a call that is *carried* by the $(n+1)$th server of its random hunt; however, allowing a slight abuse of this definition, an n-call shall often be spoke of in reference to a call that is *offered* to the $(n+1)$th server of its random hunt. When speaking of an n-call, it should be clear from the context whether this call is offered or carried. According to this definition, an N-call is a call that is blocked and cleared. Let it be clarified that each of the N servers is offered: calls initiated by users (exogenous calls), which have been defined as 0-calls; and, calls that were originally 0-calls but have overflowed n times to become n-calls, $n > 0$.

This is a simplified model of a video-on-demand distributed-server network that fails to capture many features. These include the added complication of media content varying from location-to-location as well as the presence of certain media that is in disproportionately high demand. However, the purpose of this model is to serve as a pedagogical example to facilitate the presentation of our approximation.

[2] A call in the context of a video-on-demand system is a request to download media content from a server. See the article by Deloddere *et al.* [7] for a further introduction to video-on-demand.

2.1 Steady-State Blocking Probability

Let $\{c_1, c_2, \ldots, c_K\}$ represent a time-contiguous sequence of calls for some K, where $c_i \in \{0\text{-call}, 1\text{-call}, \ldots, \text{N-call}\}$. The most important performance measure that is to be considered is *steady-state blocking probability*, which is defined as

$$P = \frac{\sum_{i=1}^{K} \mathbf{1}\{c_i = N\text{-call}\}}{K}, \quad K \longrightarrow \infty, \tag{2}$$

where $\mathbf{1}(\cdot)$ denotes the indicator function. Henceforth, when making reference to a probability or a time, it is the steady-state probability and a time in steady-state that is assumed.

This model of a distributed-server network can be regarded as an $M/M/N/N$ queue that is offered an intensity of Na. This allows for exact calculation of blocking probability using the Erlang B formula as $P = \mathbf{E}(Na, N)$. Therefore, $\mathbf{E}(Na, N)$ provides a benchmark to gauge the error in estimating blocking probability via EFPA. An easily computable benchmark is one of the incentives for resorting to such a simplified model.

2.2 Erlang's Fixed Point Approximation

At any time instant, server i is either busy or idle. Let X_i be a random variable such that $X_i = 1$ if server i is busy and $X_i = 0$ if server i is idle. Let $\mathbf{X} = (X_1, \ldots, X_N) \in \{0, 1\} \times \cdots \times \{0, 1\}$ and

$$b_i(x) = \mathbf{P}(X_i = x), \quad x \in \{0, 1\}. \tag{3}$$

The correlation error is a result of treating the random variables $X_1, \ldots, X_N$ as if they were independent, and thus writing

$$\mathbf{P}(\mathbf{X} = \mathbf{x}) = \prod_{i=1}^{N} b_i(x), \quad \mathbf{x} \in \{0, 1\} \times \cdots \times \{0, 1\}. \tag{4}$$

All N servers are stochastically equivalent in the sense that $b_i(x) = b_j(x)$, $i, j = 1, \ldots, N$. This is because the random hunt ensures that the intensity of n-calls offered to server i is the same as the intensity of n-calls offered to server j. The subscript i in $b_i(x)$ is therefore suppressed, and server i may refer to any of the N servers.

By definition, 0-calls arriving at a server form a Poisson stream that offers an intensity of a. However, n-calls, $n > 0$, arriving at a server form a stream that is of greater peakedness than a Poisson stream. The Poisson error is a result of characterizing this stream as if it were a Poisson stream but with reduced intensity. The factor by which intensity is reduced is determined by taking into account all stochastic permutations in which an n-call is offered to a server and weighting each permutation by its probability of occurrence. It can be verified that n-calls arriving at a server offer an intensity of

$$\begin{aligned} a(n) &= \sum_{i_1,\ldots,i_n \neq i} a\big(b_{i_1}(1),\ldots,b_{i_n}(1)\big)\frac{(N-n-1)!}{(N-1)!} \\ &= ab(1)^n, \quad n = 0,\ldots,N-1, \end{aligned} \tag{5}$$

where the sum $\sum_{i_1,\ldots,i_n \neq i}$ is to be understood as the sum over all $(N-1)!/(N-n-1)!$ permutations of $(i_1,\ldots,i_n)$ such that $i_1,\ldots,i_n \in \{1,\ldots,N\}-\{i\}$. For example, a 1-call is offered to server i if a 0-call is blocked at any of the other $N-1$ servers, which occurs with probability $b(1)$, and then has server i listed as the second server in its random hunt, which occurs with probability $1/(N-1)$. There are no other permutations in which a 1-call is offered to server i, hence $a(1) = ab(1)(N-1)/(N-1)$.

Independence error has been admitted in writing (5) because each of the $N-1$ marginal streams that offer n-calls, $n > 0$, to a server are pooled together to form a combined stream that offers an intensity of $a(n)$, which is an approximation given that any two of these marginal streams may be mutually dependent.

A combined Poisson stream that offers an intensity of $\sum_{n=0}^{N-1} a(n)$ to a server is then formed by pooling together each of these $N-1$ marginal streams formed by n-calls, $n = 1,\ldots,N-1$, together with the stream formed by 0-calls.

To surmise, there are two 'levels' of pooling that take place. Foremost, each of the $N-1$ marginal streams that offer n-calls, $n > 0$, to server i are pooled together to form a combined stream that offers an intensity of $a(n)$ to server i, and then each of these $N-1$ combined streams are in turn pooled together with the stream formed by 0-calls, to form a combined stream that offers an intensity of $\sum_{n=0}^{N-1} a(n)$ to server i.

We digress for a moment to refine the definition of independence error. Independence error is attributable to two doings:

C.1) Treating the random variables $X_1,\ldots,X_N$ as if they were independent, and thus writing (4). This is referred to as *independence error 1.*

C.2) Pooling together each of the $N-1$ marginal streams formed by n-calls, $n = 1,\ldots,N-1$, together with the stream formed by 0-calls, to form a

combined stream that offers an intensity of $\sum_{n=0}^{N-1} a(n)$. This is referred to as *independence error 2*.

Therefore, even if the random variables $X_1, \ldots, X_N$ were in fact independent, correlation error 2 would still be present.

Returning from this digression, according to EFPA

$$b(1) = \mathbf{E}\left(\sum_{n=0}^{N-1} a(n), 1\right) = \frac{\sum_{n=0}^{N-1} a(n)}{1 + \sum_{n=0}^{N-1} a(n)} \tag{6}$$

and $b(0) = 1 - b(1)$.

Substituting (5) into (6) gives the fixed-point equation

$$b = \frac{a \sum_{n=0}^{N-1} b^n}{1 + a \sum_{n=0}^{N-1} b^n} = a - ab^N \tag{7}$$

in which a and N are given, and $b = b(1)$ is to be determined. The sequence $\{b_i\}_{i=0}^{\infty}$ that is generated by iterating according to the fixed-point mapping $b_{i+1} = a - ab_i^N$ for an initial estimate $b_0 \in [0, 1]$ may diverge.

An alternative is to rewrite (7) as the polynomial $f : [0, 1] \to [-a, 1]$ defined by $f(b) = ab^N + b - a$ and consider the equation $f(b) = 0$. The equation $f(b) = 0$ has a unique solution on its domain $b \in [0, 1]$.

Lemma 1 *For $a \geq 0$ and $N \in \mathbb{N}$, the equation $f(b) = 0$ has a unique solution.*

Proof: See the extended version [26]. ■

Newton's method of iteration is well suited to calculating the unique solution of $f(b) = 0$. Let $b_0 \in [0, 1]$ be an initial estimate and b_i the estimate at iteration i. Then

$$b_{i+1} = b_i - \frac{f(b_i)}{f'(b_i)} = \frac{a(1 + Nb_i^N - b_i^N)}{aNb_i^{N-1} + 1}. \tag{8}$$

Newton's method is that it is guaranteed to converge to the unique solution of $f(b) = 0$ provided $b_0 = 1$.

Lemma 2 *The sequence $\{b_i\}_{i=0}^{\infty}$ that is generated by iterating according to $b_{i+1} = b_i - f(b_i)/f'(b_i)$ for $b_0 = 1$ converges to the unique solution of $f(b) = 0$.*

Proof: See the extended version [26]. ■

Of interest is estimating the distribution of n-calls. Since $b(1)$ is an estimate of the probability that a call encounters a busy server listed at *any* position of its random hunt, and $b(0) = 1 - b(1)$ is an estimate of the probability that a call encounters an idle server, the density function of n-calls is given by

$$\begin{aligned} h(n) &= \mathbf{P}(c = n\text{-call}) \\ &= \begin{cases} b(1)^n b(0), & n = 0, \ldots, N-1, \\ b(1)^N, & n = N, \\ 0, & n \neq 0, \ldots, N, \end{cases} \end{aligned} \tag{9}$$

where c denotes an arbitrary call. It follows that

$$P = \mathbf{P}(c = N\text{-call}) = h(N). \tag{10}$$

2.3 Strengthened Formulations of Erlang's Fixed-Point Approximation

Approaches to strengthening EFPA by combatting the Poisson error usually entail constructing a better estimate of the blocking probability perceived by an overflow stream than simply approximating it as if it were a Poisson stream. Several such higher-moment approximations are surveyed in the extended version [26]. Some of them shall be considered in this paper in terms of the model of the distributed-server network.

Let the pair $\Big(a(n), v(n)\Big)$ be a two-moment characterization of the stream formed by n-calls offered to a server of the distributed server network, where $a(n)$ and $v(n)$ is the mean (intensity) and variance of the number of busy servers on a infinite trunk group offered this stream.

Based on the same rationale leading to (5), n-calls arriving at a server offer an intensity of

$$a(n) = \begin{cases} a, & n = 0, \\ a(b_0 \cdots b_{n-1}), & n = 1, \ldots, N-1, \end{cases} \tag{11}$$

where b_n is the blocking probability perceived by an n-call arriving at a server.

Calculating the variance of the stream formed by n-calls, $n > 0$, arriving at a server proceeds by borrowing some ideas inherent to the equivalent random method [23]. (For 0-calls, the variance is equal to the mean.) In particular,

$v(n)$, $n > 0$, is estimated by regarding $\Big(a(n), v(n)\Big)$ as the overflow stream of a fictitious trunk group comprising x servers that is offered a Poisson stream of intensity a. To calculate the so-called equivalent random parameter x, the approach prescribed by Jagerman [12] can be used. Upon calculating the equivalent random parameter x, $v(n)$ is estimated such that

$$v(n) = a(n)\left(1 - a(n) + \frac{a}{x + 1 - a + a(n)}\right). \tag{12}$$

A combined stream that is characterized by $\Big(\sum_{n=0}^{N-1} a(n), \sum_{n=0}^{N-1} v(n)\Big)$ is then formed by pooling together each of the N marginal streams $\Big(a(n), v(n)\Big)$ formed by n-calls, $n = 0, \ldots, N-1$, offered to a server. Using Hayward's method [8] to estimate the blocking probability perceived by this combined stream, say p, gives

$$p = \mathbf{E}\left(M/Z, 1/Z\right), \tag{13}$$

where $M = \sum_{n=0}^{N-1} a(n)$, $V = \sum_{n=0}^{N-1} v(n)$ and $Z = V/A$ is the mean, variance and peakedness of this combined stream. Hence, the mean of the distribution of the number of busy servers on an infinite trunk group offered the overflow of this combined stream is Mp.

All that remains is the task of apportioning this combined overflow stream Mp to each of the N marginal streams $\Big(m(n), v(n)\Big)$, $n = 0, \ldots, N-1$. Although a great many empirical formulae have been prescribed for this purpose, in this paper, Mp shall be apportioned in proportion to $v(n)/V$. That is, the portion of Mp owing to marginal stream $\Big(m(n), v(n)\Big)$ is $Mpv(n)/V$. Then according to this apportionment, the blocking probability perceived by n-calls offered to a server is given by

$$b_n = \frac{Mpv(n)}{a(n)V}, \quad n = 0, \ldots, N-1. \tag{14}$$

Equations (11), (12), (13) and (14) define a set of fixed-point equations. Presupposing that a solution of this set of fixed-point equations does indeed exist, it may be determined by writing an appropriate fixed-point mapping and iterating until an error criterion is met. [3]

[3] Although there is no certainty that the sequence generated by iterating converges, divergence is rare in practice and can often be overcome by periodically re-initializing with a convex combination of the most recent iterations.

Table 1
Formulations of EFPA

EFPA	One-moment formulation
EFPA 2M	Two-moment formulation (Hayward's method)
EFPA BPP	EFPA strengthened via BPP approximation
EFPA IPP	EFPA strengthened via IPP approximation

Given that the sequence generated by iterating converges to a solution $b_0, \ldots, b_{N-1}$, P is estimated according to (10) after appropriately rewriting the density function (9).

There is a wealth of other higher-moment approximations that may be used to strengthen EFPA. The precision of two others shall be tested empirically, as well as the afore described two-moment approximation, though this paper shall avail itself of describing their intricacies.

The first is a three-moment approximation suggested by Kuczura [16] in which a server is approximated as an $IPP/M/1/1$ queue that is offered a combined stream characterized by its first three moments.

The second is a two-moment approximation suggested by Delbrouck [6] in which each of the N streams formed by n-calls, $n = 0, \ldots, N-1$, offered to a server are characterized via a Pascal distribution. To remain consistent with literature, this approximation shall be implemented exactly as described in Section IV of [6] and referred to as the Bernoulli-Poisson-Pascal (BPP) approximation.

2.4 Empirical Quantification of Error

It is timely as this stage to gauge the error in estimating blocking probability via EFPA and its strengthened formulations. An experiment was conducted in which the intensity offered to each of ten servers is varied over the range $[0.2, 1]$. The error in estimating blocking probability relative to the benchmark provided by the Erlang B formula $P = \mathbf{E}(Na, N)$ is plotted in Fig. 1. Table 1 defines each formulation of EFPA considered. Relative error is defined in the usual way as the ratio $(\tilde{x} - x)/x$, where $\tilde{x}$ is an estimate of x.

It was found that the numerical stability of EFPA IPP was poor and often several re-initializations were required to ensure convergence of the sequence generated by iterating, especially for low intensities. Estimates provided by EFPA IPP for $a < 0.3$ do not feature in Fig. 1 for this reason.

Based on the outcome of this experiment, it is evident that EFPA yields an estimate of blocking probability that is probably too inaccurate for most pur-

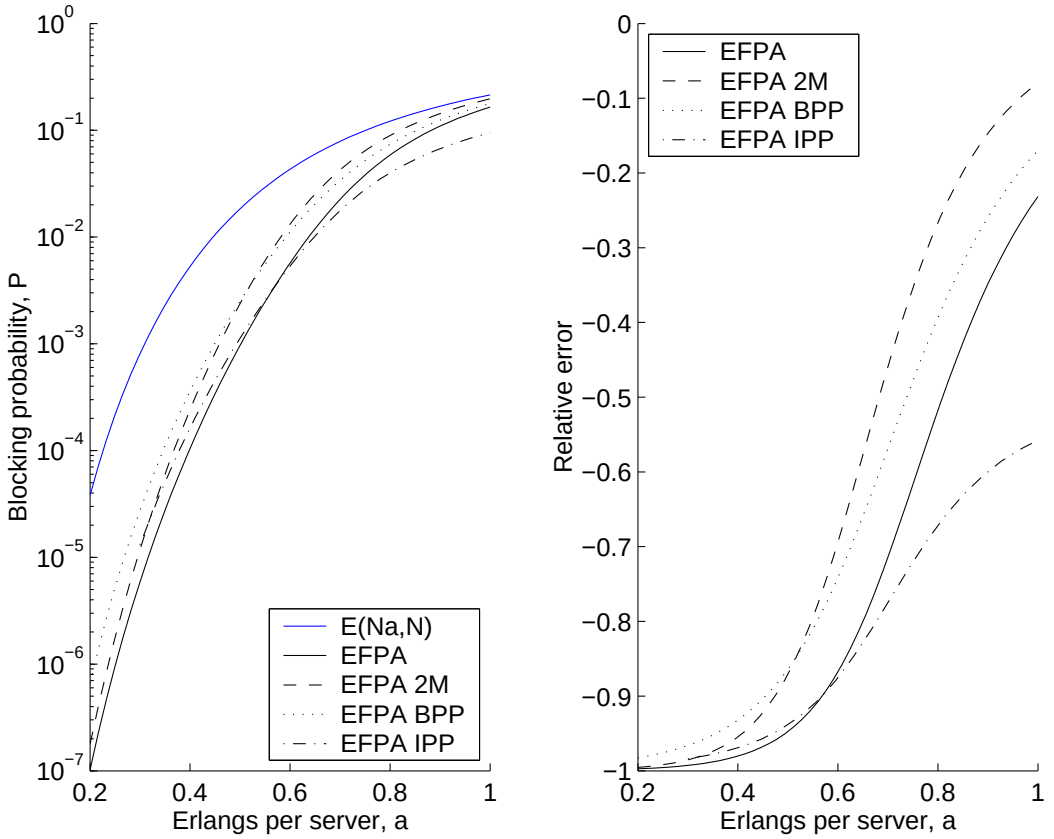


Fig. 1. Gauging the relative error in estimating blocking probability via EFPA and its strengthened formulations for $N = 10$

poses. Although strengthening EFPA via higher-moment approximations may offer a marginal reduction in error (reduction in relative error does not exceed 0.12), this reduction is hardly justified in consideration of the computational burden in dealing with additional moments.

3 The New Approximation

As a recap of the previous section, it may be said that a model of a distributed-server network was defined and it was then empirically found that the error in estimating blocking probability for this model using EFPA or its strengthened formulations is probably too high, thus driving the need for a better approximation.

To avoid a possible mix-up with a second model, also of the same distributed-server network, that shall be presented in this section, the model that was defined in the previous section is referred to as the *true model* (TM) henceforth. Furthermore, the estimate of P given by (10) is written as $\tilde{P}_{M_T}$ to reflect that it is derived from the TM.

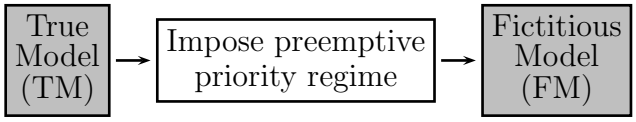


Fig. 2. A conceptual depiction of the TM and FM convention

Consider a second model that does not reflect the physical characteristics of the distributed-server network as accurately as the TM. This second model is constructed simply by imposing a preemptive priority regime on the TM. In this preemptive priority regime, each stream is classified according to the number of servers which it has sought to engage but found busy; that is, the number of times it has overflowed. A stream that has overflowed n times is given strict preemptive priority over a stream that has overflowed m times, $n < m$, given that both streams compete for a common server.

Hence, an n-call that arrives at a server that is engaged by an m-call, $n < m$, is given the right to preempt the m-call and seize the server for itself. This m-call that is preempted must then re-consult its random hunt and seek to engage a server that it has not yet sought to engage. Given that an idle server is found, the service period begins anew irrespective of the service time accumulated at prior servers at which it was preempted. A call is blocked if it has sought to engage all N servers exactly once, but has been unable to engage a server for its entire service period. Owing to the fact that each stream is classified according to the number of times it has overflowed, this preemptive priority regime is referred to as *overflow priority classification* (OPC).

Imposing the preemptive priority regime defined by OPC on an instance of the TM gives rise to an instance of a new model which is referred to as the *fictitious model* (FM) for the reason that it wrongly characterizes the distributed-server network as if the preemptive priority regime defined by OPC were in place. A depiction of this TM and FM convention is shown in Fig. 2.

Although reasoning for defining OPC is desperately in need, it has been deferred until Section 4 in favor of pursuing a course that leads as quickly as possible to the new approximation that forms the crux of this paper. A succinct description of this new approximation is as follows:

> *Given an instance of the TM, impose on it the preemptive priority regime defined by OPC to yield the corresponding FM and apply EFPA to the FM to generate the FM estimate* $P \approx \tilde{P}_{M_F}$.

This constitutes the *OPC approximation* (OPCA) of which EFPA is an integral part, but is in contrast to EFPA proper in which EFPA or one of its strengthened formulations is applied directly to TM to generate the TM estimate $P \approx \tilde{P}_{M_T}$.

The TM estimate $\tilde{P}_{M_T}$ is calculated as given by (10), while the FM estimate is derived by applying EFPA to the FM and shall be the theme of the next subsection.

3.1 The Fictitious Model

Let $X_i = n$ if server i is busy with and n-call and $X_i = -1$ if server i is idle. Let $\mathbf{X} = (X_1, \ldots, X_N) \in \{-1, 0, \ldots, N-1\} \times \cdots \times \{-1, 0, \ldots, N-1\}$ and rewrite (3) such that

$$b_i(x) = \mathbf{P}(X_i = x), \quad x \in \{-1, 0, \ldots, N-1\}.$$

As before, the random variables $X_1, \ldots, X_N$ are treated as if they were independent and thus (4) holds except the state-space must be enlarged to $\mathbf{x} \in \{-1, 0, \ldots, N-1\} \times \cdots \times \{-1, 0, \ldots, N-1\}$. Owing to the same rationale described in Subsection 2.2, all N servers are stochastically equivalent and thus it can be written that $b(n) = b_i(n)$.

Parallel to the reasoning leading to (5), n-calls arriving at a server offer an intensity of

$$a(n) = \begin{cases} a, & n = 0, \\ a\Big(b(0) \cdots b(n-1)\Big), & n = 1, \ldots, N-1. \end{cases} \tag{15}$$

The stream formed by n-calls, $n > 0$, arriving at a server is characterized as if it were a Poisson stream of intensity $a(n)$, though higher-moment characterizations are a plausible refinement. Hence, the blocking probability perceived by an n-call seeking to engage a server is $b(n)$.

The preemptive priority regime defined by OPC awards highest priority to 0-calls. A 0-call is therefore oblivious to the existence of n-calls, $n > 0$, and only perceives the existence of other 0-calls. It follows that

$$b(0) = \mathbf{E}\Big(a(0), 1)\Big). \tag{16}$$

A 1-call is oblivious to the existence of n-calls, $n > 1$; however it may be preempted by a 0-call that competes for a common server. The blocking probability perceived by a 1-call is equal to the ratio given by the intensity of the stream formed by 2-calls to the intensity of the stream formed by 1-calls.

Taking this ratio gives

$$b(1) = \frac{a(2)}{a(1)} = \frac{\mathbf{E}\Big(a(0)+a(1),1\Big)\Big(a(0)+a(1)\Big) - a(1)}{a(1)}.$$

And in general,

$$\begin{aligned} b(n) &= \frac{a(n+1)}{a(n)} \\ &= \frac{\mathbf{E}\Big(\sum_{i=0}^{n} a(i),1\Big)\sum_{i=0}^{n} a(i) - \sum_{i=1}^{n} a(i)}{a(n)}, \end{aligned} \tag{17}$$

for all $n = 0, \ldots, N-1$, where $a(N)$ is defined as the intensity of the stream formed by calls that are blocked and cleared.

A desirable property is that the blocking probabilities $b(0), \ldots, b(N-1)$ can be computed recursively in $O(N)$. This recursion is more desirable than solving the fixed-point equation given by (7), and then dealing with concerns regarding the existence and uniqueness of a fixed-point as well as convergence of iteration.

Lemma 3 *Given $a > 0$, the blocking probability perceived by an n-call can be computed in $O(n)$ via the recursion*

$$A_n = \begin{cases} a, & n = 0, \\ A_{n-1} + a - \frac{A_{n-1}}{1+A_{n-1}}, & n > 0, \end{cases} \tag{18}$$

and then

$$b(n) = \frac{A_{n+1} - A_n}{A_n - A_{n-1}}, \quad n > 0. \tag{19}$$

Proof: The proof follows after substitution of (15) into (17) and is presented in the extended version [26]. ■

Analogous to (9), the density function of n-calls can be written as

$$h(n) = \mathbf{P}(c = n\text{-call}) \tag{20}$$
$$= \begin{cases} 1 - b(0), & n = 0, \\ \Big(b(0) \cdots b(n-1)\Big) & \\ \times \Big(1 - b(n)\Big), & n = 1, \ldots, N-1, \\ \Big(b(0) \cdots b(N-1)\Big), & n = N, \\ 0, & n \neq 0, \ldots, N, \end{cases}$$

and $\tilde{P}_{M_F} = \mathbf{P}(c = N\text{-call}) = h(N)$.

3.2 Toward a Proven Superior Estimate of Blocking Probability

It has already been said that one desirable property of OPCA is that the FM estimates can be computed in $O(N)$ via the simple recursion given by (18). As shall be conjectured, the key advantage of OPCA is that

$$|\tilde{P}_{M_F}(a, N) - P(a, N)| \leq |\tilde{P}_{M_T}(a, N) - P(a, N)|, \tag{21}$$

for all $a \geq 0$. Hence, OPCA yields a more accurate estimate of blocking probability than EFPA.

Before arriving at this conjecture, the first step shall be to express $\tilde{P}_{M_T}(a, N)$ and $\tilde{P}_{M_F}(a, N)$ in terms of the same function. To set apart notation that is common to both models, the subscripts M_T and M_F are used henceforth. For example, $a_{M_T}(n)$ is in reference to the TM and is as given by (5), while $a_{M_F}(n)$ is in reference to the FM and is as given by (15). Let

$$\nu_M(n) = \sum_{i=0}^{n} a_M(i), \quad M \in \{M_T, M_F\}, \tag{22}$$

which is the sum of the intensities offered by $(0, \ldots, n)$-calls arriving at a server.

Lemma 4 *For $M \in \{M_T, M_F\}$,*

$$\tilde{P}_M(a, N) = 1 - \frac{\nu_M(N-1)\Big(1 - \mathbf{E}(\nu_M(N-1), 1)\Big)}{a}. \tag{23}$$

Proof: For $M = M_T$, according to (6) and (10),

$$\tilde{P}_{M_T} = \mathbf{E}\Big(\nu_{M_T}(N-1), 1\Big)^N. \tag{24}$$

Using (7) in (24) gives

$$\begin{aligned}\tilde{P}_{M_T} &= \frac{a - \mathbf{E}\Big(\nu_{M_T}(N-1), 1\Big)}{a} \\ &= \frac{a - \nu_{M_T}(N-1)\left(1 - \frac{\nu_{M_T}(N-1)}{1+\nu_{M_T}(N-1)}\right)}{a},\end{aligned}$$

after which the required result follows from the fact that $\mathbf{E}(\alpha, 1) = \alpha/(1+\alpha)$, $\alpha \geq 0$.

For $M = M_F$, according to (17),

$$\begin{aligned}\tilde{P}_{M_F} &= \frac{a_{M_F}(1)\cdots a_{M_F}(N)}{a_{M_F}(0)\cdots a_{M_F}(N-1)} = \frac{a_{M_F}(N)}{a} \\ &= \frac{\nu_{M_F}(N) - \nu_{M_F}(N-1)}{a}.\end{aligned} \tag{25}$$

Let $\boldsymbol{\Upsilon}(\alpha) = \alpha\mathbf{E}(\alpha, 1)$. Note that for $n > 0$,

$$\begin{aligned}a_{M_F}(n) &= \nu_{M_F}(n) - \nu_{M_F}(n-1) \\ &= \boldsymbol{\Upsilon}\Big(\nu_{M_F}(n-1)\Big) - \boldsymbol{\Upsilon}\Big(\nu_{M_F}(n-2)\Big),\end{aligned} \tag{26}$$

where $\nu_{M_F}(-1) = 0$. Substituting (26) into (22) gives rise to a telescoping sum that results in the recursion

$$\nu_{M_F}(n) = a + \boldsymbol{\Upsilon}\Big(\nu_{M_F}(n-1)\Big), \quad n > 0. \tag{27}$$

To arrive at the required result, (27) is used in (25) giving

$$\begin{aligned}\tilde{P}_{M_F} &= \frac{a + \boldsymbol{\Upsilon}\Big(\nu_{M_F}(N-1)\Big) - \nu_{M_F}(N-1)}{a} \\ &= 1 - \frac{\nu_{M_F}(N-1)\Big(1 - \mathbf{E}(\nu_{M_F}(N-1), 1)\Big)}{a}.\end{aligned}$$

■

Before reaching the main conjecture, a further lemma followed by a minor conjecture shall be required. This lemma shows that the FM estimate of blocking probability $\tilde{P}_{M_F}(a, N)$ is an upper bound for the TM estimate of blocking

probability $\tilde{P}_{M_T}(a,N)$, while the minor conjecture states that the benchmark provided by the Erlang B formula $P(a,N) = \mathbf{E}(Na,N)$ is an upper bound for $\tilde{P}_{M_F}(a,N)$. Therefore, $\tilde{P}_{M_F}(a,N)$ is sandwiched in between the benchmark provided by the Erlang B formula and $\tilde{P}_{M_T}(a,N)$.

Lemma 5 *For all $a \geq 0$ and $N \in \mathbb{N}$,*

$$\tilde{P}_{M_T}(a,N) \leq \tilde{P}_{M_F}(a,N).$$

Proof: A simple rearrangement of Lemma 4 gives

$$\tilde{P}_M(a,N) = 1 - \frac{\nu_M(N-1)}{a\Big(1+\nu_M(N-1)\Big)}, \quad M \in \{M_T, M_F\}.$$

Hence, it suffices to show that $\nu_{M_F}(N-1) \leq \nu_{M_T}(N-1)$. Induction shall be used to show

$$\nu_{M_F}(n) \leq \nu_{M_T}(N-1), \quad n = 1, \ldots, N-1.$$

According to (5),

$$\nu_{M_T}(N-1) = a \sum_{i=0}^{N-1} \mathbf{E}\Big(\nu_{M_T}(N-1),1\Big)^i \tag{28}$$

and explicitly writing out the first few terms of (27) gives

$$\begin{aligned}
\nu_{M_F}(n) &= a + \boldsymbol{\Upsilon}\Big(\nu_{M_F}(n-1)\Big) \\
&= a + \mathbf{E}\Big(\nu_{M_F}(n-1)\Big)\Big(a + \boldsymbol{\Upsilon}\Big(\nu_{M_F}(n-2)\Big)\Big) \\
&= a \sum_{i=0}^{n} \prod_{j=1}^{i} \mathbf{E}\Big(\nu_{M_F}(i-j)\Big),
\end{aligned} \tag{29}$$

where a null product is unity. For the base case $n = 1$, an immediate consequence of (28) and (29) is

$$\begin{aligned}
\nu_{M_F}(1) &= a + a\mathbf{E}(a,1) \\
&\leq a + a\mathbf{E}\Big(\nu_{M_T}(N-1),1\Big) \leq \nu_{M_T}(N-1).
\end{aligned}$$

The inductive hypothesis is that $\nu_{M_F}(k) \leq \nu_{M_T}(N-1)$ for all $k < n \leq N-1$. Using the inductive hypothesis and because $\mathbf{E}(\alpha,1)$ is monotonically increasing, it follows that

$$
\begin{aligned}
\nu_{M_F}(n) &= a\sum_{i=0}^{n}\prod_{j=1}^{i}\mathbf{E}\Big(\nu_{M_F}(i-j)\Big) \\
&= a + a\mathbf{E}(a,1) + a\mathbf{E}(a,1)\mathbf{E}\Big(\nu_{M_F}(1)\Big) + \cdots \\
&\leq a + a\mathbf{E}\Big(\nu_{M_T}(N-1)\Big) \\
&\quad + a\mathbf{E}\Big(\nu_{M_T}(N-1)\Big)^2 + \cdots \\
&= a\sum_{i=0}^{n}\mathbf{E}\Big(\nu_{M_T}(N-1),1\Big)^i \leq \nu_{M_T}(N-1)
\end{aligned}
$$

Since the base case is true and the inductive step is true, $\nu_{M_F}(n) \leq \nu_{M_T}(N-1)$ is true for all $n \leq N-1$. It may be noted that the case of $n = 0$ follows trivially since $\nu_{M_T}(0) = \nu_{M_F}(0) = a$. ■

Conjecture 1 *For all $a \geq 0$ and $N \in \mathbb{N}$,*

$$\tilde{P}_{M_F}(a,N) \leq P(a,N), \tag{30}$$

where $P(a,N) = \mathbf{E}(Na,N)$ is the benchmark provided by the Erlang B formula.

At first glance, it is reasonable to suggest that a proof of Conjecture 1 would proceed via induction on N; however, after pursing this course, it does not seem to result in anything revealing. Methodical experimentation has yet to reveal a case indicating the falseness of Conjecture 1.

Conjecture 2 *For all $a \geq 0$ and $N \in \mathbb{N}$,*

$$\tilde{P}_{M_T}(a,N) \leq \tilde{P}_{M_F}(a,N) \leq P(a,N).$$

By a sandwiching argument, Conjecture 2 states that $\tilde{P}_{M_F}(a,N)$ is a more accurate estimate of blocking probability than $\tilde{P}_{M_T}(a,N)$. Conjecture 2 could immediately be formulated as a theorem if a proof of Conjecture 1 was forthcoming.

3.3 Empirical Quantification of Error

Conjecture 2 falls short of quantifying the reduction in error that is achieved in estimating $P(a,N)$ via $\tilde{P}_{M_F}(a,N)$ rather than via $\tilde{P}_{M_T}(a,N)$. To gauge this reduction in error, an experiment is considered.

The same experiment described in Subsection 2.4 was repeated, but this time the FM estimate given by OPCA was generated. The error in estimating

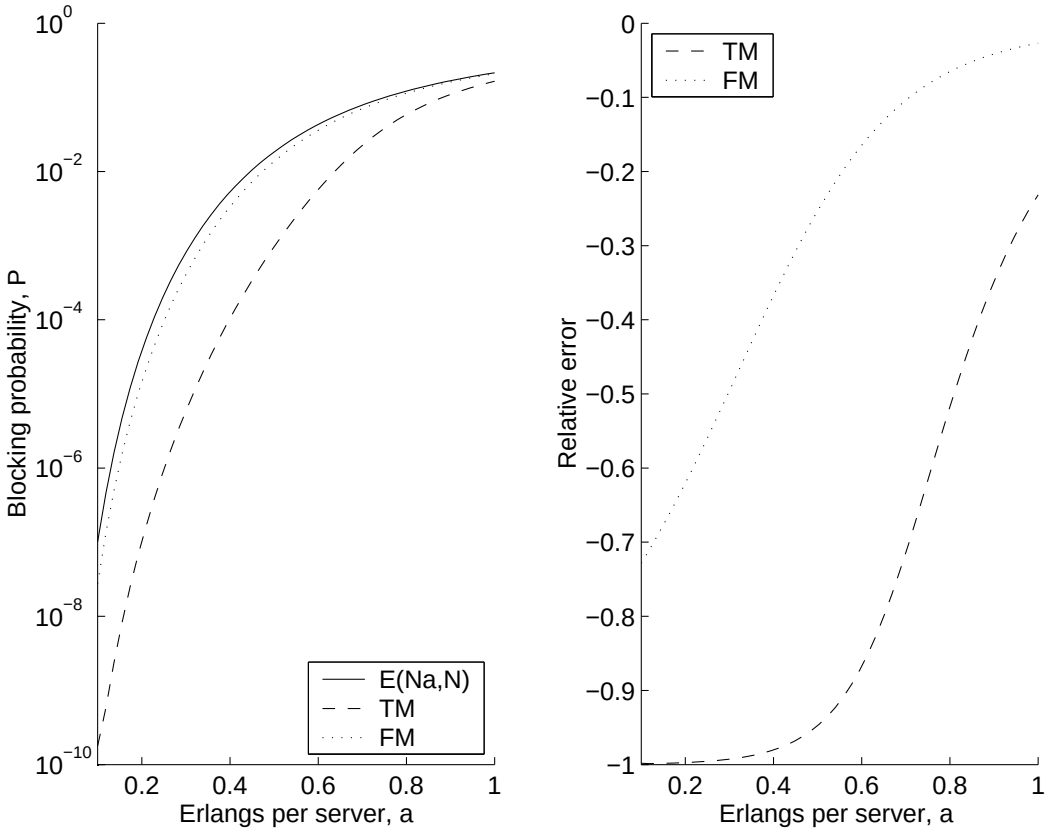


Fig. 3. Gauging the relative error in estimating blocking probability via the TM and FM estimate for $N = 10$

blocking probability via $\tilde{P}_{M_T}(a, N)$ and $\tilde{P}_{M_F}(a, N)$ relative to the benchmark provided by the Erlang B formula $P(a, N) = \mathbf{E}(Na, N)$ is plotted in Fig. 3.

Based on the outcome of this experiment, it is evident that the FM estimate is remarkably more accurate than the TM estimate. And since the FM estimate can be computed recursively, it seems that OPCA facilitates a better approximation both in terms of accuracy and numerical robustness.

In closing this section, it must be emphasized that OPCA is relevant to a broad range of overflow loss networks. Several examples demonstrating the versatility of OPCA shall be considered in Section 5 in the context of circuit-switched networks using alternative routing. Although, as it shall be seen, cases can be constructed for which OPCA yields a poorer estimate of blocking probability relative to EFPA.

Table 2
Example of notational convention

P_{M_T}	Exact blocking probability in the TM
P_{M_F}	Exact blocking probability in the FM
$\tilde{P}_{M_T}$	Estimate of blocking probability in the TM as per EFPA
$\tilde{P}_{M_F}$	Estimate of blocking probability in the FM as per OPCA

4 Rationale Behind the New Approximation

The aim of this section shall be to describe the rationale underpinning OPCA in an intuitive manner, and to provide support for this rationale with empirical results.

The subscripts M_T and M_F are used to set apart notation common to both models and a tilde is used to denote an estimate, as opposed to the exact value, that is derived via EFPA in the case of the TM and via OPCA in the case of the FM. Table 2 gives an example of this convention in terms of blocking probability. This convention shall be used to clarify *any* ambiguous notation.

The rationale behind OPCA is underpinned by a supposition and a corollary of Lemma 5.

Supposition 1 *The FM and TM are approximately equivalent in terms of blocking probability and server time congestion. That is,*

$$P_{M_T} \approx P_{M_F}. \tag{31}$$

Note that in writing (31), it is not implied that $\tilde{P}_{M_T} \approx \tilde{P}_{M_F}$.

Corollary 1 *The proportion of the total intensity offered to a server that is owing to the stream formed by* 0*-calls, is larger in the FM than the TM. In particular,*

$$\frac{\tilde{a}_{M_T}(0)}{\sum_{j=0}^{N-1} \tilde{a}_{M_T}(j)} \leq \frac{\tilde{a}_{M_F}(0)}{\sum_{j=0}^{N-1} \tilde{a}_{M_F}(j)}.$$

Proof: According to the proof of Lemma 5, $\nu_{M_F}(N-1) \leq \nu_{M_T}(N-1)$, $N \geq 1$, where the inequality is strict for $N = 1$. Since $\tilde{a}_{M_T}(0) = \tilde{a}_{M_F}(0) = a$, it suffices to show that $\sum_{j=0}^{N-1} \tilde{a}_{M_F}(j) \leq \sum_{j=0}^{N-1} \tilde{a}_{M_T}(j)$, which follows from the fact that $\sum_{j=0}^{N-1} \tilde{a}_{M_F}(j) = \nu_{M_F}(N-1) \leq \nu_{M_T}(N-1) = \sum_{j=0}^{N-1} \tilde{a}_{M_T}(j)$. ■

The proportion of the total intensity offered to a server that is owing to the stream formed by n-calls, $n = 0, \ldots, N-1$, in the TM and FM is plotted in Fig. 4 for $N = 10$ and $N = 20$ given that $a = 0.8$. In Fig. 4, a vertical

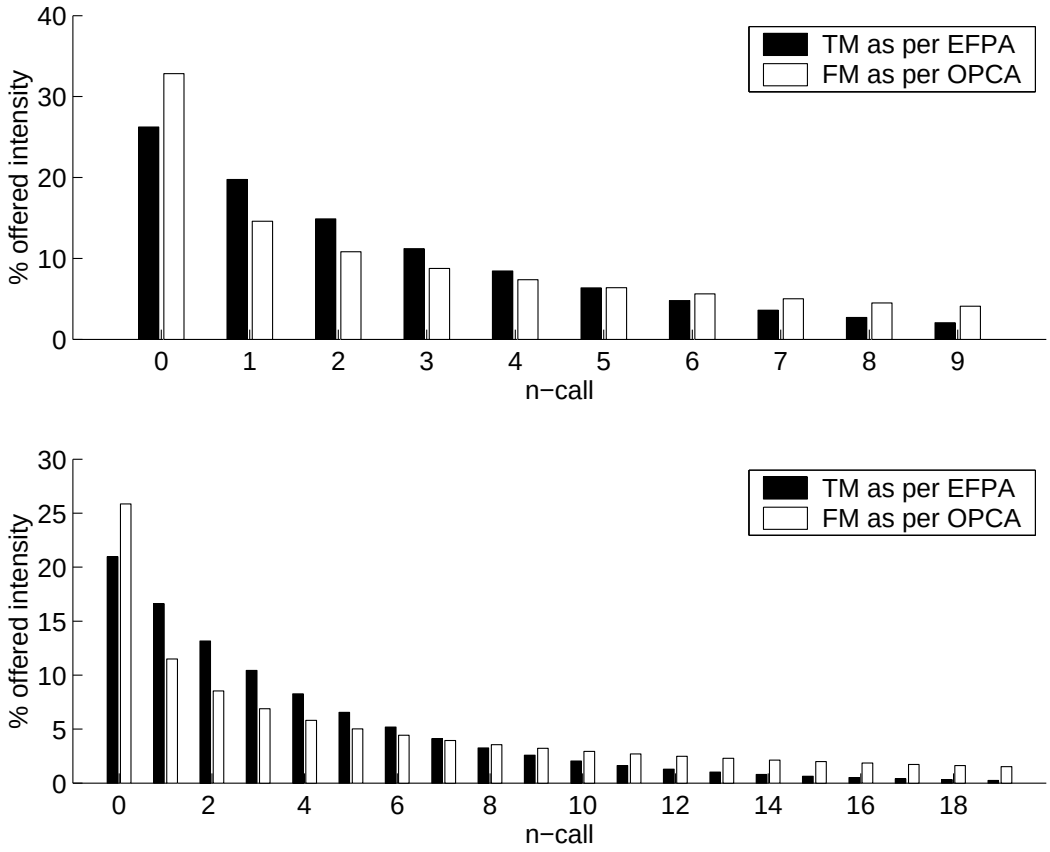


Fig. 4. Proportion of n-calls offered to a server for $N = 10$ (upper) and $N = 20$ (lower) given $a = 0.8$

bar associated with an n-call represents the percentage of the total intensity offered to a server that is owing to the stream formed by n-calls.

The theme of the next subsection shall be to provide justification of Supposition 1.

4.1 Justification of Supposition 1

A more precise restatement of (31) is

$$P_{M_F}(a, N) = P_{M_T}(a, N) + \epsilon_P, \tag{32}$$

where $\epsilon_P \geq 0$ are small constants and $\epsilon_P \to 0$ as $a \to 0$.

An intuitive interpretation of (32) is apparent by noting that it is possible for a call to be blocked and cleared in the FM even if there exists an idle server. For example, suppose server i is engaged by an $(N-1)$-call. (Recall that an $(N-1)$-call is a call that finds idle the last server listed in its random hunt.)

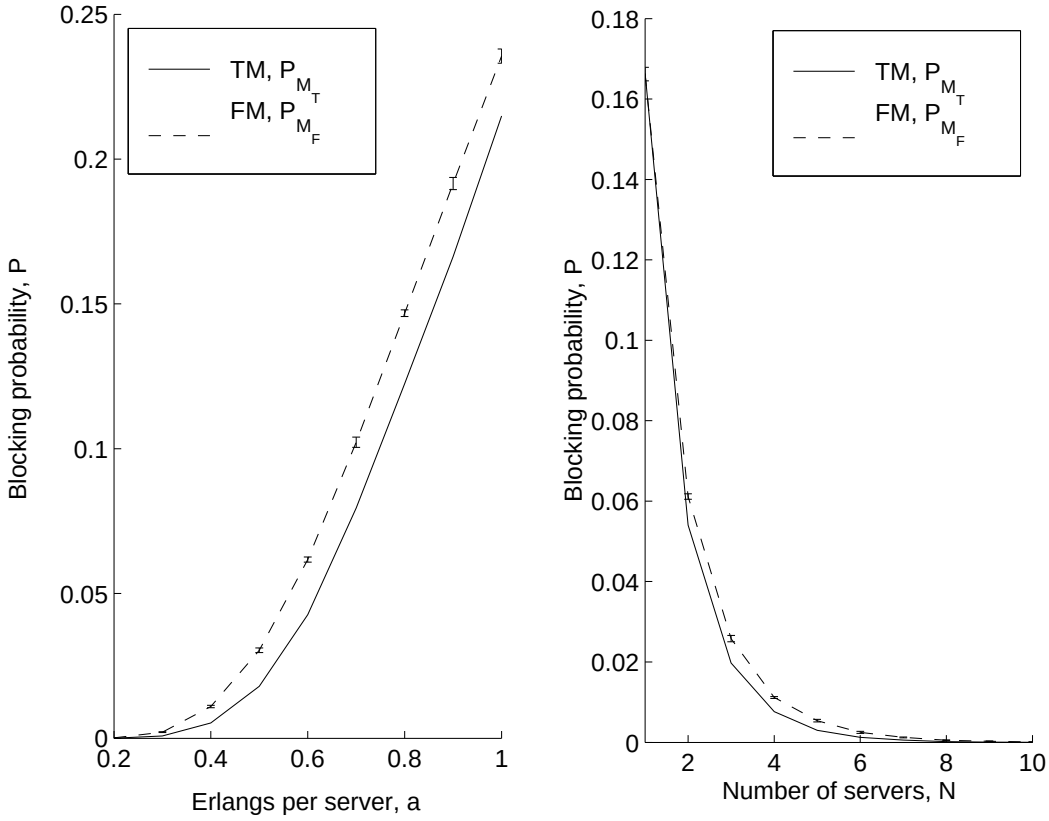


Fig. 5. Gauging the magnitude of ϵ_P

Then if an n-call, $n < N-1$, arrives at server i, this $(N-1)$-call is preempted, even if there exists an idle server, and is thus blocked. This event does not occur in the TM and is responsible for higher blocking probability in the FM.

To gauge the magnitude of ϵ_P, an experiment was conducted in which blocking probability was computed via simulation in the FM, while the Erlang B benchmark was used to compute blocking probability in the TM. The magnitude of ϵ_P is shown in Fig. 5 for the cases in which: the intensity offered to each of ten servers is varied over the range $[0.2, 1]$ (left); and, the number of servers, which are each offered an intensity of 0.2, is varied over the range $1, \ldots, 10$ (right). The confidence intervals shown are commensurate to two standard deviations and have been estimated using the method of batch means.

This experiment gives credence to (31).

4.2 The Rationale

The rationale behind OPCA is founded on the idea of increasing the proportion of the total intensity offered to a server that is owing to the stream formed by 0-calls, as per Corollary 1, and to counterbalance this increase, decreasing

the proportion of the total intensity offered to a server that is owing to the streams formed by n-calls, $n > 0$. As it shall be argued, this 're-proportioning' of the total intensity offered to a server is effective in combating independence error 1 and the Poisson error.

4.2.1 Combatting Independence Error 1

Let i_1 and i_2, $i_1 \neq i_2$, denote two servers in the FM. Recall that independence error 1 arises from treating the random variables X_{i_1} and X_{i_2} as if they were independent. The dependence between the random variables X_{i_1} and X_{i_2} is decreased in the FM relative to the TM because the combined stream offered to server i_1 comprises a larger proportion of 0-calls, which are by definition independent of the random variable X_{i_2}; and vice-versa, the combined stream offered to server i_2 comprises a larger portion of 0-calls, which are by definition independent of the random variable X_{i_1}. Hence, by increasing the proportion of the total intensity offered to a server owing to the stream formed by 0-calls, the magnitude of independence error 1 is reduced.

4.2.2 Combatting the Poisson Error

The peakedness of the combined stream offered to a server is reduced in the FM relative to the TM because it comprises a larger proportion of 0-calls, which by definition form a Poisson stream. Hence, the magnitude of the error in treating the combined stream offered to a server as if it were a Poisson stream is reduced.

Remark 1 *It is important to remark that for some n^*, the proportion of the combined stream offered to a server owing to the streams formed by $(n^*, \ldots, N-1)$-calls is slightly larger in the FM relative to the TM. This effect is clearly exhibited in Fig. 4, and is undesirable because the streams formed by $(n^*, \ldots, N-1)$-calls are of the greatest peakedness. Hence, characterizing them as Poisson streams admits more Poisson error than characterizing the streams formed by $(1, \ldots, n^*-1)$-calls as Poisson streams. However, the proportion of $(n^*, \ldots, N-1)$-calls is significantly dwarfed by the proportion of $(0, \ldots, n^* - 1)$-calls, and thus it is reasonable to suggest that this undesirable effect is negligible.*

To surmise, the success of OPCA lies in its ability to reduce the magnitude of independence error 1 and the Poisson error by increasing the proportion of the total intensity offered to a server owing to the stream formed by 0-calls.

According to Supposition 1, the TM and FM are approximately equivalent in terms of blocking probability. However, the magnitude of the error in estimating blocking probability via EFPA is less in the FM relative to the TM (because the magnitude of independence error 1 and the Poisson error is re-

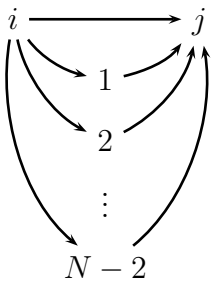


Fig. 6. Switching office pair (i, j) of a fully-meshed circuit-switched network using alternative routing

duced in the FM). Hence, it is desirable to estimate blocking probability in the distributed-server network via OPCA.

5 Extension to Circuit-Switched Networks using Alternative Routing

To this point, OPCA has only been considered in terms of a pedagogical example. This section shall exhibit the versatility of OPCA by using it to estimate blocking probability in a variety of circuit-switched networks using alternative routing. There is some sentiment in considering circuit-switched networks for it was circuit-switched networks that provided the impetus to conceive EFPA.

5.1 A Symmetric Fully-Meshed Circuit-Switched Network

Adopted is the usual model of a circuit-switched network that has been presented in the papers of Kelly [14,15] and Whitt [22]. The network comprises N switching offices. Each pair of switching offices is interconnected via a trunk group comprising K cooperative servers. Therefore, there exists a one-hop route as well as $N-2$ two-hop alternative routes between each pair of switching offices, as shown in Fig. 6. Calls arrive at each switching office pair according to independent and time-homogeneous Poisson processes of intensity $a \geq 0$. A call foremost seeks to engage the one-hop route between the pair of switching offices at which it arrives. A call that finds all K trunks on this one-hop route busy overflows to one of the $N-2$ two-hop alternative routes with equal probability and without delay. A call continues to overflow as such until either: it encounters a two-hop alternative route possessing an idle trunk on *both* of its constituent links, in which case the call engages both of these idle trunks for its entire holding time; or, it has sought to engage all $N-2$ two-hop alternative routes, in which case it is blocked and cleared. According to the TM and FM convention, this model serves as the TM.

Instead of repeating parts of Subsection 2.2, several shortcuts shall be taken to avoid pedantically laboring through the derivation of EFPA for a second time. Let b be the probability that all K servers are busy on an arbitrary trunk group. It suffices to consider an arbitrary trunk group as a consequence of symmetry. It can be verified that

$$b = \mathbf{E}\left(a + 2ab(1-b)\sum_{j=0}^{N-3}\left(1-(1-b)^2\right)^j, K\right) \tag{33}$$

and that call blocking probability is estimated by

$$\tilde{P}_{M_T} = b\Big(1-(1-b)^2\Big)^{N-2}. \tag{34}$$

Equation (33) shall be justified on a term-by-term basis. The factor of two multiplying the summation manifests itself after enumerating all stochastic permutations in which a call can be offered to an arbitrary trunk group. The term $\left(1-(1-b)^2\right)^j$ is the probability that a call overflows from j two-hop alternative routes, while the term $1-b$ is the factor by which intensity must be reduced to ensure that the intensities carried by both links of a two-hop alternative route are equal. For example, suppose a two-hop alternative route is offered a Poisson stream of intensity a. The portion of a that is offered to each of the two links constituting this two-hop alternative route is calculated as $a(1-b)$ to ensure that the intensities carried by both links are equal and given by $a(1-b)^2$.

Equation (34) states that a call is blocked in the event that it overflows from its one-hop route, which occurs with probability b, and then overflows from each of its $N-2$ two-hop alternative routes, which occurs with probability $\left(1-(1-b)^2\right)^{N-2}$.

It is difficult to ascertain properties regarding existence and uniqueness of solution for (33). Of further concern is that it cannot be said if the sequence $\{b_i\}_{i=0}^{\infty}$ generated according to the usual fixed-point mapping $b_{i+1} = \mathbf{E}\Big(a + 2ab_i(1-b_i)\sum_{j=0}^{N-3}\left(1-(1-b_i)^2\right)^j, K\Big)$ converges.

Multiple fixed-points are not at all uncommon for circuit-switched networks using alternative routing and correspond to different equilibria that may exist in steady-state between which the network fluctuates. Fluctuating between multiple equilibria is considered an unstable mode of behavior and is usually combatted via barring alternatively routed calls from engaging an idle trunk on any trunk group that already contains more than a certain number of busy trunks, which is a mechanism referred to as *trunk reservation* [18]. Trunk reservation was not considered in this paper since it was found experimentally that

unstable modes of behavior did not arise for the moderately sized networks that were considered.

The TM, and the FM to which it gives rise, are defined in a completely analogous manner. In particular, an n-call is given strict preemptive priority over an m-call, $n < m$, given that both calls compete for a common trunk group. The definition of a n-call must be adjusted to a call that overflows from n routes before engaging the $(n+1)$th route.

Let $b(n)$ be the blocking probability perceived by an n-call, $n = 0, \ldots, N-2$, seeking to engage an arbitrary trunk group. It can be verified that for the FM,

$$b(n) = \begin{cases} \mathbf{E}(a, K), & n = 0, \\ \frac{B_n \mathbf{E}(B_n, K) - B_{n-1} \mathbf{E}(B_{n-1}, K)}{B_n - B_{n-1}}, & n > 0, \end{cases} \tag{35}$$

where $B_n = \sum_{j=0}^{n} a_j$ and

$$a_n = 2ab(0)\Big(1 - b(n)\Big) \prod_{j=1}^{n-1} \Big(1 - (1 - b(j))^2\Big), \quad n > 0, \tag{36}$$

is the total intensity offered by n-calls to an arbitrary trunk group. Hence, $a_0 = a$. Call blocking probability is then estimated as

$$\tilde{P}_{M_F} = b(0) \prod_{j=1}^{N-2} \Big(1 - (1 - b(j))^2\Big). \tag{37}$$

Equation (35) follows the same justification provided for (17).

The term $1 - b(n)$ in (36) precludes the use of a recursion to compute the blocking probabilities $b(1), \ldots, b(N-2)$, and thus an appropriate fixed-point mapping must be used. We considered approximating (36) such that $a_n = 2ab(0)\Big(1-b(n-1)\Big) \prod_{j=1}^{n-1} \Big(1-(1-b(j))^2\Big)$ to facilitate computation of $b(1), \ldots, b(N-2)$ recursively; however, the accuracy of this approach was poor.

An experiment was conducted in which blocking probability was estimated in a network comprising four switching offices with ten trunks per trunk group. The error in estimating blocking probability via EFPA and OPCA was gauged against a simulation and is plotted in Fig. 7.

Based on the outcome of this experiment, although EFPA yields a better estimate of blocking probabilities that are greater than about 0.02, OPCA is preferred for the range of blocking probabilities that are considered most relevant to engineering approximations.

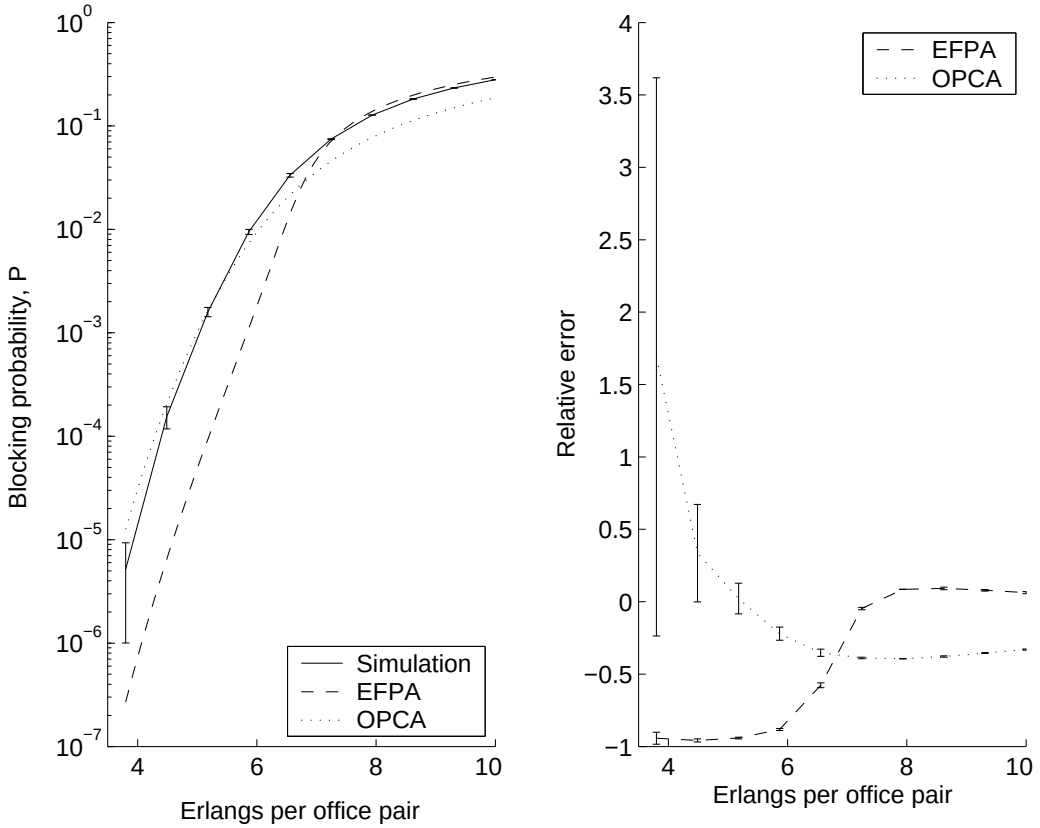


Fig. 7. Estimating blocking probability in a fully-meshed circuit-switched network using alternative routing, $N = 4$, $K = 10$

5.2 Other Circuit-Switched Networks

The error in estimating blocking probability via OPCA shall be gauged for three other general circuit-switched networks and compared to EFPA as well as a simulation. To conclude, a somewhat artificial example shall be constructed in which OPCA yields a significantly poorer estimate of blocking probability than EFPA. This example serves as a warning against using OPCA carelessly.

The topologies of the three circuit-switched networks to be considered are shown in Fig. 11, where each double-arrowed line represents a trunk group comprising K trunks. Routing is implemented in a sequential manner in all three networks as follows. For each switching office pair, the maximum number of alternative routes that are disjoint in terms of trunk groups are enumerated via appropriate use of the maximum network flow algorithm and stored in a routing table. (It is insisted that each alternative route for an office pair is disjoint in terms of trunk groups to avoid the additional complexity of dealing with the conditional blocking probabilities that would require consideration if this was not the case. The procedure suggested by Chan [3] can be used in both OPCA and EFPA to cope with alternative routes that are not disjoint in

Table 3
Guide to Empirical Results

Network	*Topology*	*Blocking Probability*
Eight Node Ring	Fig. 11(a)	Fig. 8
Nine Node Wheel	Fig. 11(b)	Fig. 9
NSF (T1)	Fig. 11(c)	Fig. 10

this sense.) The routing table is then ordered such that the shortest hop route is listed first and the longest hop route is listed last. Ties between routes of equal hop length are resolved by flipping a coin.

Calls arrive at each office pair according to independent Poisson processes of intensity $a \geq 0$ and sequentially traverse (without delay) the sorted routing table for an idle route such that the shortest hop route is sought foremost. (An idle route is a route that contains at least one idle trunk on each of its trunk groups at the time of a call arrival.) A call is blocked and cleared if it cannot engage a route for its entire service period.

By observing the network topologies presented in Fig. 11, it is apparent that the blocking probability perceived by a call may vary according to which switching office pair it is assigned, even though the ring and wheel topology are perfectly symmetric in a topological sense.

OPCA is defined in a completely analogous manner as in earlier sections. Although more laborious, deriving OPCA for the case of general circuit-switched networks follows the same principles used in the preceding subsection. The main difference is that each trunk group as well as each switching office pair must be treated separately because of asymmetry considerations.

Of interest is gauging the error in estimating blocking probability in each of the three circuit-switched networks via OPCA and EFPA. To this end, the error in estimating blocking probability was computed for all three circuit-switched networks for the case $K = 10$. A guide to these empirical results is shown in Table 3.

The intensity offered to each switching office pair was varied over a range that resulted in blocking probabilities that spanned the range $[10^{-5}, 10^{-1}]$. The set of fixed-point equations inherent to OPCA and EFPA were solved by iterating as described earlier.

Based on these empirical results, it evident that OPCA provides a more accurate estimate of blocking probability for all three circuit-switched networks relative to EFPA (assuming $K = 10$ and the considered routing strategy is in place). Since minimal additional computational effort is required to cal-

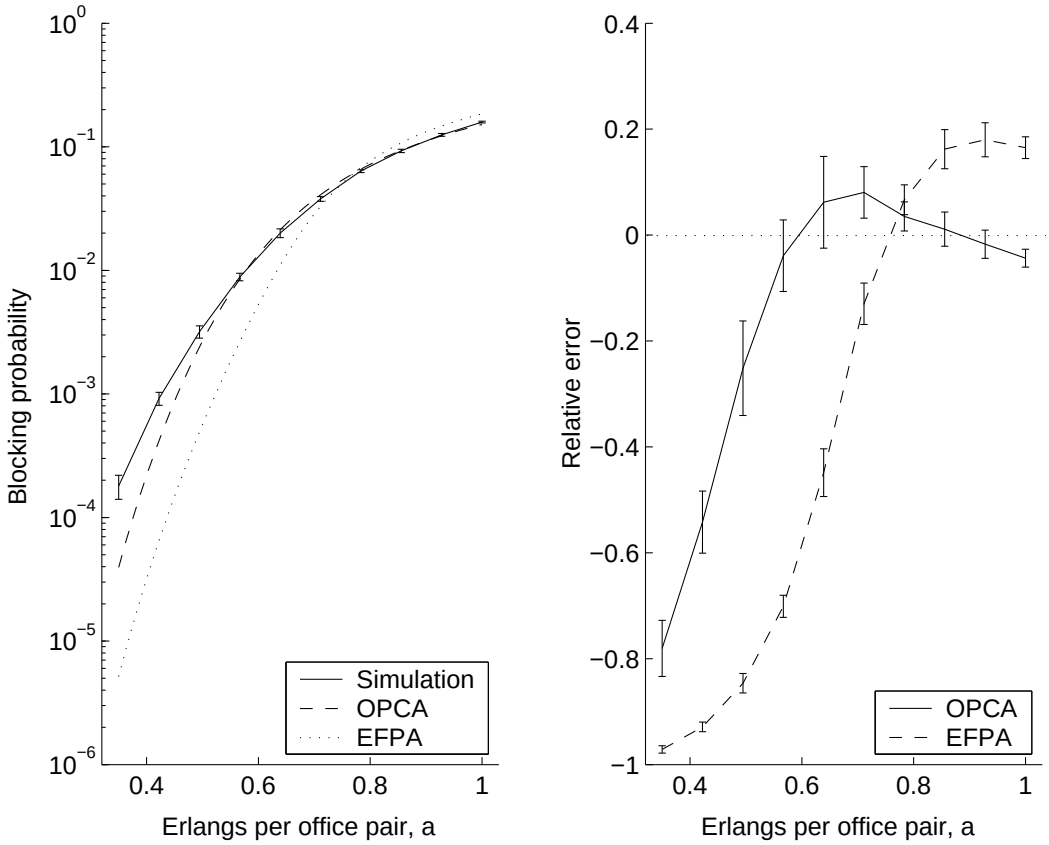


Fig. 8. Eight node ring network

culate an estimate via OPCA relative to EFPA, it seems that OPCA is a preferred approximation. The additional computational effort in calculating an estimate via OPCA is a consequence of the need to calculate the intensity offered and blocking probability perceived for *each* of $(0, 1 \ldots)$-calls offered to a trunk group, whereas EFPA only requires calculation of these two statistics for the *single* combined stream formed by pooling together the marginal streams formed by $(0, 1, \ldots)$-calls.

In the extended version of this paper [26], empirical results are presented suggesting that the rationale described in Section 4 also holds for general circuit-switched networks. In particular, the proportion of n-calls offered to an average trunk group was plotted for a low intensity regime as well as a high intensity regime for all three circuit-switched networks. These plots have been omitted due to space limitations.

It is fitting to end this section by constructing and example in which OPCA yields a poorer estimate of blocking probability than EFPA. To construct this example, the model of the distributed-server network shall be revisited. In particular, reconsider the model of the distributed-server network, but suppose it is only those calls that arrive at one particular server that are permitted to overflow in the usual manner prescribed by the random hunt. These calls

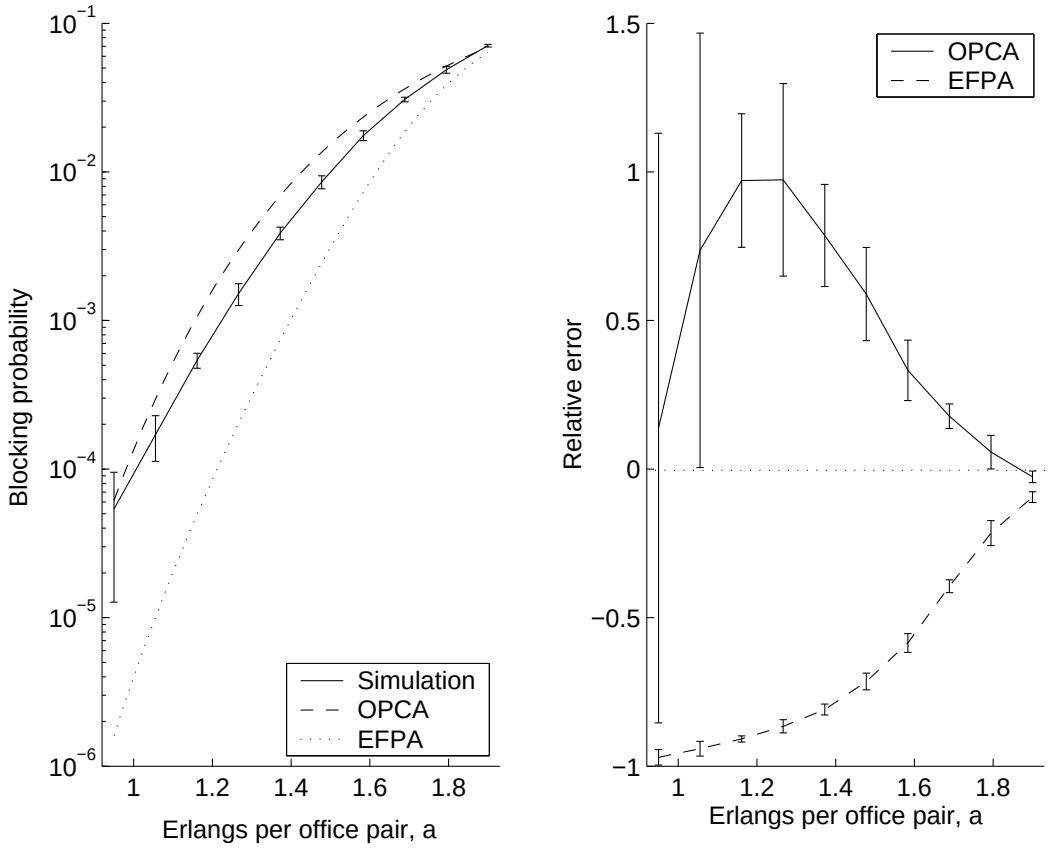


Fig. 9. Nine node wheel network

are referred to as *premium calls* and arrive according to a Poisson process of intensity $a^* \geq 0$ to this one particular server. Calls arriving at all other servers are barred from overflowing and thus either: engage the first server at which they arrive, in the case that this server is idle; or, are blocked and cleared, in the case that this server is busy. These calls are referred to as *standard calls* and arrive at all these other servers according to independent Poisson processes of intensity $a \geq 0$.

The blocking probability perceived by premium calls and standard calls as well as the average perceived blocking probability was estimated for a network comprising four servers (of which one of these four servers is offered only premium calls) via OPCA and EFPA. In this experiment, $a = 0.5$ and a^* was varied within the range $[0.3, 1.8]$. A simulation was also implemented to gauge errors. The results are plotted in Fig. 12.

Upon observing Fig. 12, it is clear that EFPA provides a better estimate of the blocking probability perceived by premium calls and standard calls. An interesting point is that the estimate of blocking probability perceived by standard calls is independent of a^* for OPCA, which is not the case in practice. This is because for the FM of this network, standard calls are oblivious to the existence of premium calls since a standard call is always given the

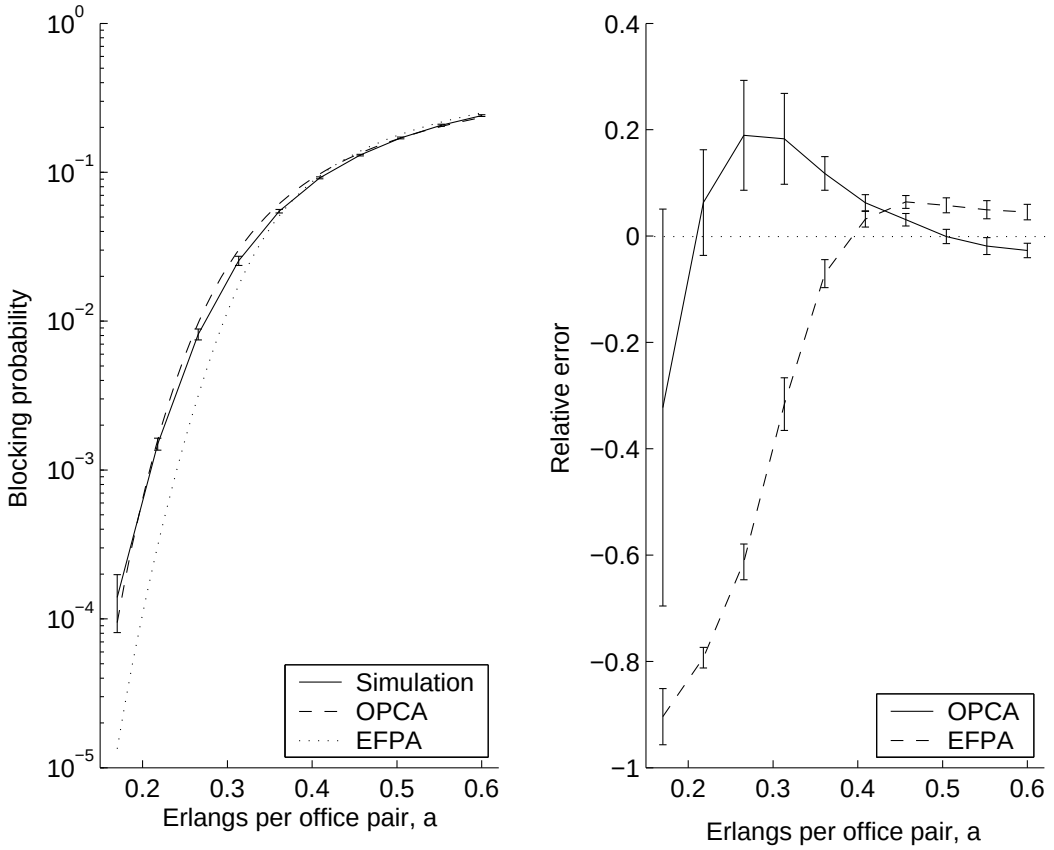


Fig. 10. NSF network

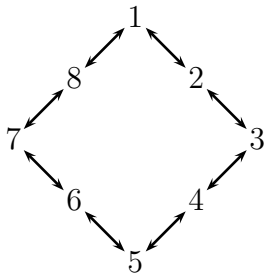
right to preempt a premium call in the FM. Hence, the result is that OPCA overestimates the blocking probability perceived by premium calls and underestimates the blocking probability perceived by standard calls, especially for high intensities.

Since the blocking probability perceived by a standard call is independent of a^* in the FM (but clearly increases with a^* in the TM), Supposition 1 does not hold for this network. This example serves as a warning against carelessly deeming OPCA to be a universally superior estimate of blocking probability.

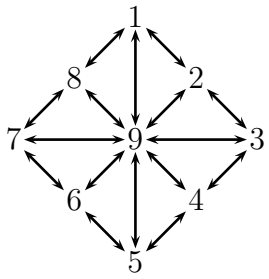
In qualitative terms, OPCA generally performs poorly for cases in which either:

1) Supposition 1 looses its validity in the sense that ϵ_P in (32) become large.
2) The analogue of Corollary 1 does not hold.
3) The undesirable effect specified in Remark 1 is not negligible.

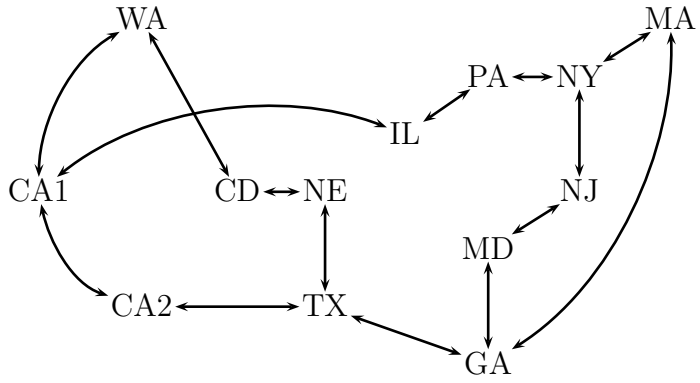
A disadvantage of OPCA is that it is difficult to ascertain a priori if any of the above three properties are manifested in a particular overflow loss network. Hence, it is difficult to make a recommendation on whether or not OPCA should be used in favor of EFPA for a given case. However, as a rule of thumb,



(a) Eight node ring



(b) Nine node wheel



(c) NSF (Version T1)

Fig. 11. Network topologies

OPCA usually performs well for networks of high symmetry and connectivity in which there are many overflow streams offered to each trunk group.

6 Conclusion

This paper introduced a new approximation referred to as OPCA for estimating blocking probability in overflow loss networks. OPCA was shown to outperform its complementary approach in EFPA for the case of a distributed-server network as well as several cases of circuit-switched networks using alternative routing. A rationale that was supported by empirical evidence suggested that the success of OPCA lies in its ability to combat the Poisson error as well as the independence error, which are two errors inherent to EPFA that especially manifest themselves in *overflow* loss networks. On the downside, it was stressed that networks may be encountered in which OPCA yields a poorer estimate of blocking probability than EFPA, and it is difficult to unequivocally ascertain a priori if a given case is best suited to OPCA or EFPA.

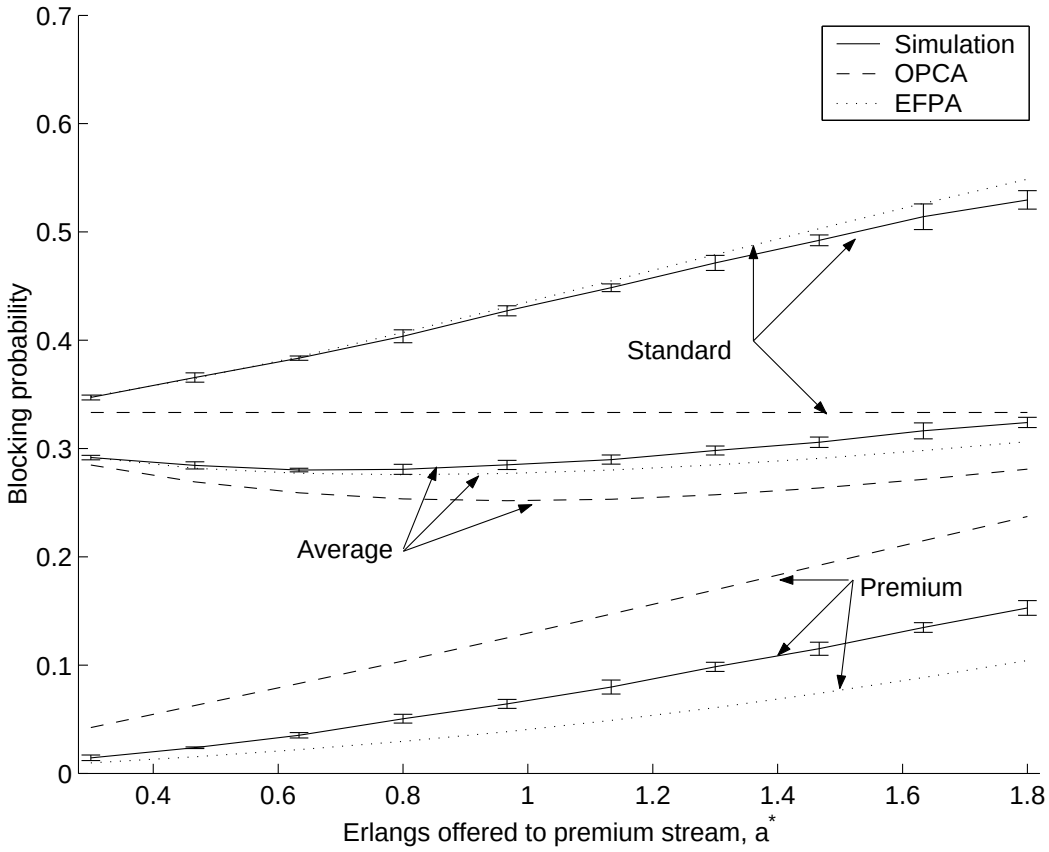


Fig. 12. An example in which OPCA performs poorly

References

[1] G. R. Ash and B. D. Huang, 'An Analytical Model for Adaptive Routing Networks', *IEEE Transaction on Communiations*, vol. 41 no. 11, Nov. 1993, pp. 1748-1759.

[2] R. B. Cooper and S. Katz, 'Analysis of alternate routing networks with account taken of nonrandomness of overflow traffic', Technical Report, Bell Telephone Lab. Memo., 1964.

[3] W. S. Chan, 'Recursive Algorithms for Computing End-to-End Blocking in a Network with Arbitrary Routing Plan', *IEEE Transactions on Communications*, February, 1980, no. 2, pp. 153-164.

[4] P. Chevalier and N. Tabordon, 'Overflow analysis and cross-trained servers', *International Journal of Production Economics*, vol. 85, 2003, pp. 4760.

[5] S. P. Chung, A. Kashper and K. W. Ross, 'Computing Approximate Blocking Probabilities for Large Loss Networks with State-Dependent Routing', *IEEE/ACM Transactions on Networking*, vol. 1, no. 1, February 1993, pp. 105-115.

[6] L. E. N. Delbrouck, 'The use of Kosten's Systems in Provisioning of

Alternate Trunk Groups Carrying Hetrogeneous Traffic', *IEEE Transactions of Communications*, vol. COM-31, no. 6, June 1983, pp. 741-749.

[7] D. Deloddere, W. Verbiest and H. Verhille, 'Interactive Video on Demand', *IEEE Comunications Magazine*, vol. 32, no. 5, pp. 82-88, May 1994.

[8] A. A. Fredricks, 'Congestion in blocking systems—a simple approximation technique', *The Bell System Technical Journal*, vol. 59, no. 6, pp. 805-827, July-Aug. 1980.

[9] A. Girard, *Routing and Dimensioning in Circiuit-Switched Networks*, Addison-Wesley, 1990.

[10] R. Guerin and L. Y.-C. Lien, 'Overflow Analysis for Finite Waiting Room Systems', *IEEE Transactions on Communications*, vol. 38, Sept. 1990, pp. 1569–1577.

[11] J. M. Holtzman, 'Analysis of Dependence Effects in Telephone Trunking Networks', *The Bell System Technical Journal*, vol. 50, no. 8, pp. 2647-2662, Oct. 1971.

[12] D. L. Jagerman, 'Methods in Traffic Calculations', *AT&T Bell Laboratories Technical Journal*, vol. 63, no. 7, pp. 1283-1301, Sept. 1984.

[13] S. Katz, 'Trunk Engineering of Non-Hierarchial Networks', *International Teletraffic Congress*, vol. 6, pp. 142.1-142.8, 1971.

[14] F. P. Kelly, 'Blocking probabilities in large circuit-switched networks', *Advances in Applied Probability*, vol. 18, pp. 473-505, 1986.

[15] ———, 'Loss networks', *The Annals of Applied Probability*, vol. 1, no. 3, pp. 319-378, August 1991.

[16] A. Kuczura, 'The Interrupted Poisson Process as an Overflow Process', *The Bell System Technical Journal*, vol. 52, no. 3, pp. 437-448, March 1973.

[17] A. Kuczura and D. Bajaj, 'A Method of Moments for the Analysis of a Switched Communication Network's Performance', *IEEE Trans. on Communications*, vol. COM-25, no. 2, pp. 185-193, Feb. 1977.

[18] R. S. Krupp, 'Stabilization of Alternate Routing Networks', *Proceedings of IEEE ICC 1982*, Philadelphia, USA, June 1982.

[19] D. Mitra, 'Asymptotic analysis and computational methods for a class of simple circuit-switched networks with blocking', *Advances in Applied Probability*, vol. 19, pp. 219-239, 1987.

[20] D. Mitra, J. A. Morrison and K. G. Ramakrishnan, 'ATM Network Design and Optimization: A Multirate Loss Network Framework', *IEEE/ACM Transactions on Networking*, vol. 4, no. 4, Aug. 1996, pp. 531-543.

[21] Z. Rosberg, H. L. Vu, M. Zukerman and J. White, "Performance Analyses of Optical Burst Switched Networks,' *IEEE Journal on Selected Areas in Communications*, vol. 21, Sept. 2003, pp. 1187–1197.

[22] W. Whitt, 'Blocking when Service is Required from Several Facilities Simulataneously', *AT&T Technical Journal*, vol. 64, pp. 1807-1856, 1985.

[23] R. I. Wilkinson, 'Theories of Toll Traffic Engineering in the U.S.A.', *Bell System Technical Journal*, vol. 35, no. 2, pp. 421-514, March 1956.

[24] I. Widjaja, 'Performance Analysis of Burst Admission Control Protocols', *IEE Proceeding on Communnications*, vol. 142, Feb. 1995, pp. 7–14.

[25] E. W. M. Wong, T.-S. Yum, 'Maximum Free Circuit Routing in Circuit-Switched Networks', *INFOCOM '90, Ninth Annual Joint Conference of the IEEE Computer and Communication Societies, Proceedings*, June 1990, vol. 3, pp. 934-937.

[26] E. W. M. Wong, A. Zalesky, Z. Rosberg, M. Zukerman, 'A Novel Analysis of Overflow Loss Networks (Extended Version)', Internal Technical Report, available at:

http://www.ee.cityu.edu.hk/∼ewong/opc_extended.pdf

[27] A. Zalesky, H. L. Vu, Z. Rosberg, E. W. M.Wong, M. Zukerman, 'Modelling and Performance Evaluation of Optical Burst Switched Networks with Deflection Routing and Wavelength Reservation', *Proceedings of INFOCOM 2004*, Hong Kong, China, March 2004, vol. 3, pp. 1864 - 1871.