Available online at www.sciencedirect.com
ScienceDirect
Computer Networks 51 (2007) 2958–2975
Computer Networks
www.elsevier.com/locate/comnet

A new method for approximating blocking probability in overflow loss networks

Eric W.M. Wong [a], Andrew Zalesky [b,*], Zvi Rosberg [c], Moshe Zukerman [b]

[a] *Department of Electronic Engineering, City University of Hong Kong, Hong Kong SAR, China*
[b] *ARC Special Research Centre for Ultra-Broadband Information Networks, Department of Electrical and Electronic Engineering, The University of Melbourne, Melbourne, Vic. 3010, Australia*
[c] *CSIRO ICT Centre, PO BOX 76, Epping NSW 1710, Sydney, Australia*

Received 1 November 2005; received in revised form 15 May 2006; accepted 19 December 2006
Available online 3 January 2007

Responsible Editor: N.B. Shroff

Abstract

In this paper, we present a new approximation for estimating blocking probability in overflow loss networks and systems. Given a system for which an estimate of blocking probability is sought, we first construct a second system to act as a surrogate for the original system. Estimating blocking probability in the second system with Erlang's fixed point approximation (EFPA) provides a better estimate for blocking probability in the original system than if we were to use the conventional approach of directly using EFPA in the original system. We present a combination of numerical and theoretical results that indicate our new approximation offers a better estimate than EFPA for a certain pure overflow loss network. Moreover, we demonstrate the accuracy of our new approximation for circuit-switched networks using alternative routing. We argue that the success of our new approximation is due to its ability to utilize congestion information imbedded in overflow traffic, whereas the conventional approach fails to utilize such information.

Keywords: Erlang's fixed-point approximation; Reduced-load approximation; Overflow loss network; Blocking probability; Overflow priority classification

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 [2], optical networks [29], and multiprocessor systems with one redundant processor that can be used to alleviate congestion on active processors [9]. Roughly 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.

* Corresponding author. Tel.: +61 3 83443821.
E-mail address: a.zalesky@ee.unimelb.edu.au (A. Zalesky).

1389-1286/$ - see front matter
doi:10.1016/j.comnet.2006.12.007

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

Approximations therefore play an crucial role in estimating blocking probability in overflow loss networks. The simplest yet crudest approach to estimating blocking probability in an overflow loss network proceeds via a one-moment approximation in which stream i is characterized solely in terms of its offered intensity m_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 blocking probability perceived by the combined stream as well as each marginal stream i comprising the combined stream is estimated by $\mathbf{E}\left(\sum_i m_i, N\right)$, where

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

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 $m_i\mathbf{E}\left(\sum_i m_i, N\right)$ to a subsequent server group.

Usually referred to as *Erlang's fixed-point approximation* (EFPA) in its most general form, this approximation was proposed in [4] in 1964 for the analysis of circuit-switched networks and has remained a cornerstone of network performance evaluation even to this day. The basic idea of EFPA is to decompose the overflow loss system into a number of server-group subsystems and treat each subsystem as if it were an independent Erlang B sub-system. See [1,3,8,13,12,17–19,22–26] 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 there may be statistical dependence. 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. The first attempt to combat the Poisson error was made in [4], using Wilkinson's equivalent random method [21]. A similar approach was later used in [11,15]. Combatting the independence error was first considered in [10]. See the extended version of this paper [28] for a comprehensive survey of strengthened formulations of EFPA.

In this paper, we present a new approximation for estimating blocking probability in overflow loss networks, which is fundamentally different from EFPA and its strengthened formulations. Given a system for which an estimate of blocking probability is sought, we first construct a second system to act as a surrogate for the original system. Estimating blocking probability in the second system with EFPA provides a better estimate for blocking probability in the original system than if we were to use the conventional approach of directly using EFPA in the original system.

The new constructed system 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, $n < m$.

A simple overflow loss network model will be defined in the next section, which facilitates the presentation of our approximation. This simple model

[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.

is fundamental in the sense that it retains overflow effects but excludes other effects such as reduced load and the destabilizing effect of alternative routing in circuit-switched networks. It is therefore the simplest and the most suitable example to expose weaknesses of EFPA and to demonstrate our new approximation. Moreover, its pure and fundamental nature makes it more amenable to analysis and more suited for understanding the overflow traffic behavior and the blocking probability performance of the various approaches.

In Section 3, the new approximation will be introduced, its supporting intuition will be discussed and we will present some results that lead us to conjecture that our approximation yields a more accurate estimate of blocking probability than EFPA. Section 4 will demonstrate the versatility of our approximation by considering its extension to circuit-switched networks using alternative routing. Numerical results will 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 [6]. The network comprises N cooperative and identical servers. Calls (i.e., requests to download video streams) initiated by users are offered to each server according to an independent time-homogenous 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. According to this definition, an N-call is a call that is blocked and cleared. In summary, 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 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.

2.1. 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 $\mathbf{X}=(X_1,\ldots,X_N)\in\{0,1\}^N$ and

$$b_i=\mathbb{P}(X_i=1). \tag{2}$$

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

$$\mathbb{P}(\mathbf{X}=\mathbf{x})=\prod_{i=1}^{N}\mathbb{P}(X_i=x_i),\quad \mathbf{x}\in\{0,1\}^N. \tag{3}$$

All N servers are statistically identical in that $b_i=b_j$ for all $i,j=1,\ldots,N$. This is because the random hunt ensures the intensity of n-calls offered to server i is the same as the intensity of n-calls offered to server j. We therefore suppress the subscript i in b_i and refer to an arbitrary server.

It can be verified that n-calls arriving at a server offer an intensity of

$$\begin{aligned} a(n)&=\sum_{i_1,\ldots,i_n\neq i}a(b_{i_1},\ldots,b_{i_n})\frac{(N-1)!}{(N-n-1)!}\\ &=ab^n,\quad n=0,\ldots,N-1, \end{aligned} \tag{4}$$

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 $i_1,\ldots,i_n\in\{1,\ldots,N\}-\{i\}$. To explain (4), we note that 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, 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 $a(1)=ab(N-1)/(N-1)$.

According to EFPA,

$$b=\mathbf{E}\left(\sum_{n=0}^{N-1}a(n),1\right)=\frac{\sum_{n=0}^{N-1}a(n)}{1+\sum_{n=0}^{N-1}a(n)}. \tag{5}$$

The Poisson error is a result of treating each of the marginal streams offered to a server as if they were a Poisson stream when in fact it is only the stream corresponding to 0-calls that is a Poisson stream. In admitting the Poisson error, we have that the combined stream offered to each server is a Poisson stream that offers an intensity given by the sum of intensities of each of the marginal streams, as shown in (5).

Substituting (4) into (5) gives the fixed-point equation

$$b = \frac{a\sum_{n=0}^{N-1} b^n}{1 + a\sum_{n=0}^{N-1} b^n} = a - ab^N \tag{6}$$

in which a and N are given and b is to be determined.

It follows that the EFPA estimate of blocking probability is given by

$$P = \mathbb{P}(c = N\text{-call}) = b^N. \tag{7}$$

2.2. 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. For a survey of these approaches see the extended version [28]. In this paper, we consider the three strengthened formulations that are listed in Table 1.

Our purpose is 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 10 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.

Table 1
Formulations of EFPA

EFPA	One-moment formulation
EFPA 2M	Two-moment formulation (Hayward's method) [7]
EFPA BPP	EFPA strengthened via Bernoulli–Poisson–Pascal (BPP) approximation [5]
EFPA IPP	EFPA strengthened via Interrupted Poisson Process (IPP) approximation [14]

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.

Fig. 1 indicates that EFPA may yield an estimate of blocking probability that is too inaccurate for engineering purposes. 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

The distributed-server model developed in the previous section will be called the true model (TM) in this section for reasons that will become apparent soon. The purpose of this section is to introduce our new approximation and discuss the intuition that we believe underpins its success.

In short, the new approximation is based on transforming the TM to a new model that we call the fictitious model (FM). Given an overflow network for which an estimate of blocking probability is required, we consider estimating blocking probability in the FM using EFPA. This estimate is usually more accurate for the TM than if we directly use EFPA to estimate blocking probability in the TM.

In this section, we will consider our new approximation in the context of the distributed-server network. In other words, the TM is equal to the model of the distributed-server network developed in the previous section. We first address the question of how to construct the FM from the TM.

The FM is constructed by imposing preemptive priorities in the TM. The preemptive priorities are such that 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.

An n-call that arrives at a server engaged by an m-call, $n < m$, is given the right to preempt the

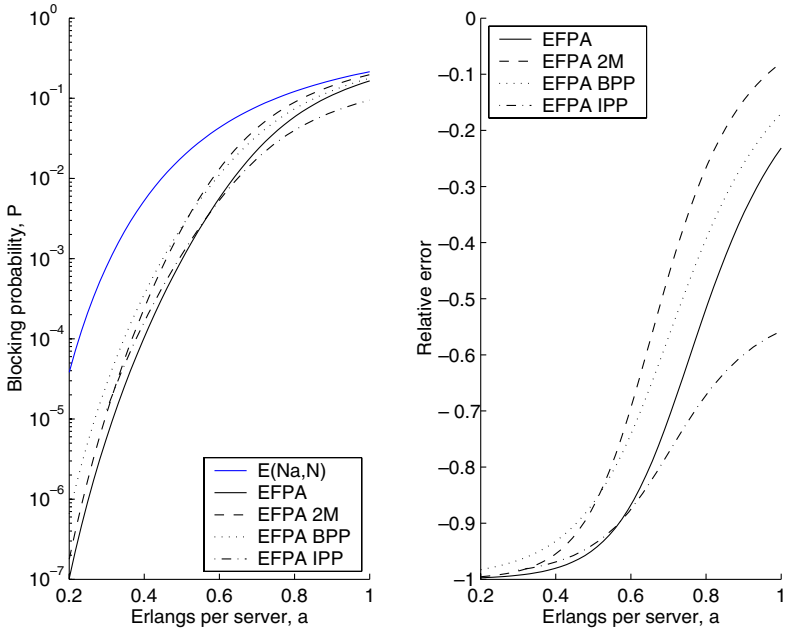

Fig. 1. Gauging the relative error in estimating blocking probability via EFPA and its strengthened formulations for $N = 10$.

m-call and seize the server for itself. The preempted m-call must then seek to engage a server that it has not yet visited. 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, we call this preemptive priority regime *overflow priority classification* (OPC) [27]. A depiction of the TM and FM convention is shown in Fig. 2. A description of our new approximation is as follows:

> *Given an instance of the TM, impose on it the OPC preemptive priority regime to yield the corresponding FM. Estimate blocking probability in the FM using EFPA.*

The two-step process of constructing the FM from the TM and applying EFPA to the FM is called the *OPC approximation* (OPCA). OPCA is in contrast to EFPA proper in which EFPA or one of its strengthened formulations is applied directly to the TM.

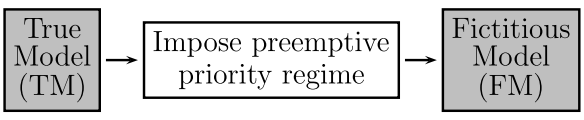

Fig. 2. A conceptual depiction of the TM and FM convention.

We therefore have a TM and FM estimate of blocking probability, which we denote as $\widetilde{P}_{M_T}$ and $\widetilde{P}_{M_F}$, respectively. The TM estimate of blocking probability $\widetilde{P}_{M_T}$ is calculated as given by (7), while the FM estimate will be derived soon. We have used the subscripts M_T and M_F to set apart notation common to both models. Furthermore, a tilde is used to denote an estimate of blocking probability as opposed to its true value. Table 2 gives an example of this convention.

We continue by discussing the intuition we believe underlies our new approximation. Extensive numerical testing and the explanations that follow provide support for the following sequence of inequalities:

$$\widetilde{P}_{M_T} \leqslant \widetilde{P}_{M_F} \leqslant P_{M_T} \leqslant P_{M_F}. \tag{8}$$

Table 2
Example of notational convention

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

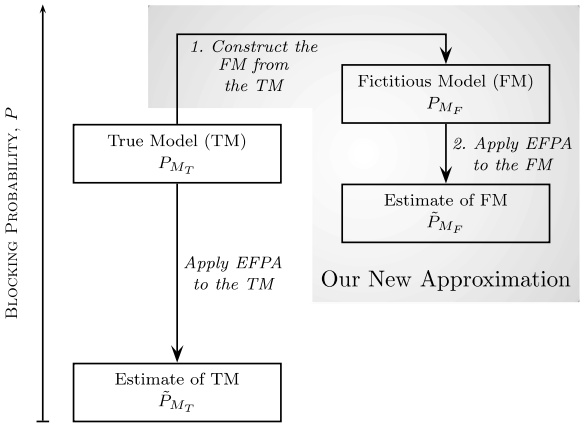

Fig. 3. Conceptual depiction of our new approximation.

Our goal is to estimate P_{M_T}. Since $\widetilde{P}_{M_F}$ lies between $\widetilde{P}_{M_T}$ and P_{M_T}, obviously $\widetilde{P}_{M_F}$ is a more accurate estimate of P_{M_T} compared to $\widetilde{P}_{M_T}$.

We will prove the first inequality in (8) in Proposition 2, we will provide numerical evidence supporting the second, though a proof is not given, and we provide intuition supporting the last inequality. In short, we claim that $\widetilde{P}_{M_F}$ and P_{M_F} are close to each other making $\widetilde{P}_{M_F}$ a good approximation for P_{M_T}, which is sandwiched between them. This is illustrated in Fig. 3, where all three are close together and $\widetilde{P}_{M_T}$ is somewhat lower.

In Fig. 3, the starting point is the TM, for which we seek to estimate blocking probability. Our new approximation is contained within the shaded region in Fig. 3. The first step of our new approximation involves constructing the FM from the TM by imposing the OPC priority regime on the TM. The second step involves applying EFPA to the FM.

The remainder of this section is organized as follows. We present our intuitive discussions in Sections 3.1 and 3.2, and then present some rigorous results in Sections 3.3 and 3.4.

3.1. Intuition supporting $P_{M_T} \approx P_{M_F}$

Consider a particular server engaged with an n-call, $n > 0$, and suppose a new call arrives at this server while the n-call is in service. The new call is considered a 0-call at the instant it arrives. What happens next depends on whether we are in the FM or the TM.

In the FM, the new call preempts the n-call causing it to overflow to an alternative server at which it has not visited before. There is exactly $N - n - 1$ such servers. But we can view the preemption of the n-call in a different way. In particular, the n-call's remaining service time at the time the new call arrives is equal to the service time of the new call. This is because service periods are independent, identical and exponentially distributed. Therefore, instead of preempting the n-call, we can force the new call to overflow to any of the $N - n - 1$ servers that the n-call has not visited. From the point of view of the blocking probability P_{M_F}, there is no difference between preempting the n-call and forcing the new call to overflow.

In this way, the new call is instantly transformed into an n-call, even though it has just arrived and has not overflowed from any server. But what is the purpose of instantly transforming a new call to an n-call? The purpose is that it limits the number of servers that a new call can visit. In particular, a new call that arrives at a server engaged with an n-call, $n > 0$, perceives a *limited availability system* comprising only $N - n - 1$ servers.

In contrast, in the TM, a new call perceives a *full availability system* comprising N servers, irrespective of whether or not it arrives at a server engaged with an n-call, $n > 0$. In particular, in the TM, a new call that arrives at a server already engaged with a call must overflow from all N servers before it is blocked.

Because a new call in the FM perceives a limited availability system, while a new call in the TM perceives a full availability system, it is apparent that $P_{M_T} \leqslant P_{M_F}$.

To show that this inequality is tight, we revisit our example in which a new call arrives at a server engaged with an n-call, $n > 0$. Although in the FM the new call sees a limited availability system comprising $N - n - 1$ servers, we argue that there would be little benefit in the new call visiting the other n servers. In particular, it is likely these other n servers are still engaged with calls because we know our n-call visited each of these n servers not too long ago and found each of them engaged with a call. There is a relatively small probability that one of these n servers becomes idle in the period beginning from when our n-call visited them and found them busy, and ending at the arrival time of the new call. The duration of this period is less than our n-call's service time. Therefore, we argue that $P_{M_T} \approx P_{M_F}$. We numerically verify this claim in Section 3.4.

3.2. Intuition supporting $|P_{M_F} - \widetilde{P}_{M_F}| \leqslant |P_{M_T} - \widetilde{P}_{M_T}|$

OPCA increases the proportion of the total intensity offered to a server that is owing to the stream formed by 0-calls. We will prove this result in Corollary 1, which is presented in Section 3.4. To counterbalance this increase, the proportion of the total intensity offered to a server that is owing to the streams formed by n-calls, $n > 0$, is decreased. This 're-proportioning' of the total intensity offered to a server is effective in combating the independence error and the Poisson error.

We discussed in Section 3.1 that in the FM, a new call arriving at a server engaged with an n-call, $n > 0$, perceives a limited availability system. Therefore, the maximum number of servers from which the new call can overflow is less than if it were in the TM, which is a full availability system. This is the reason why the proportion of the total intensity offered to a server that is owing to the streams formed by n-calls, $n > 0$, is smaller in the FM.

As we discuss next, the benefit of reducing the proportion of calls that have overflowed and increasing the proportion of calls that have not is to reduce the magnitude of the independence error and the Poisson error.

3.2.1. Combatting the independence error

Let i_1 and i_2, $i_1 \neq i_2$, denote two servers in the FM. Independence error 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 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 the independence error is reduced.

3.2.2. Combatting the Poisson error

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

We conclude this subsection by arguing that OPCA has the ability to utilize congestion information imbedded in a call that has overflowed. To demonstrate, suppose a new call arrives at a server engaged with an $(N-1)$-call. The presence of an $(N-1)$-call indicates the likelihood of a highly congested system. Therefore, it is likely that the new call is blocked. But what actually happens to the new call in the FM and TM?

In the FM, we can instantly transform the new call to an $(N-1)$-call at the instant it arrives. We discussed why this is possible in Section 3.1. Therefore, the new call is blocked without ever overflowing from a server. We argue that there may not have been much benefit in allowing the new call to overflow in the hope of finding an idle server. This is because not too long before the arrival of the new call, the original $(N-1)$-call visited all the $N-1$ servers and found each of them engaged with a call. There is a relatively small probability that one of these $N-1$ servers became idle in the period beginning from when our $(N-1)$-call visited them and found them busy, and ending at the arrival time of the new call. It is as if our original $(N-1)$-call tells the new call: "Don't even bother trying to find an idle server because I've just visited each of them and found each of them to be engaged." The new call accepts this advice and leaves the system without overflowing. From an approximation perspective, this is a desirable feature because it reduces the number of calls that overflow.

In contrast, in the TM, the new call must visit all $N-1$ servers before it is blocked, which is a likely outcome given that the new call arrives to a server engaged with an $(N-1)$-call. Unlike in the FM, the presence of the $(N-1)$-call conveys no information to the new call.

3.3. Analysis of the fictitious model

Let $X_i = n$ if server i is busy with a $(0,1,2,\ldots,n)$-call and $X_i = -1$ if server i is idle.

Let $\mathbf{X} = (X_1,\ldots,X_N) \in \{-1,0,\ldots,N-1\}^N$ and rewrite (2) such that

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

As before, the random variables $X_1,\ldots,X_N$ are treated as if they were independent and thus (3) holds except the state-space must be enlarged to $\mathbf{x} \in \{-1,0,\ldots,N-1\}^N$. Owing to the same rationale described in Section 2.1, all N servers are

statistically equivalent and thus it can be written that $b(n) = b_i(n)$.

Parallel to the reasoning leading to (4), n-calls arriving at a server offer an intensity of

$$a(n) = \begin{cases} a, & n = 0, \\ a(b(0)\cdots b(n-1)), & n = 1,\ldots,N-1. \end{cases} \tag{9}$$

The stream formed by n-calls, $n > 0$, arriving at a server is characterized as if it were a Poisson stream of intensity $a(n)$. 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

$$b(0) = \mathbf{E}(a(0), 1). \tag{10}$$

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

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

And in general,

$$b(n) = \frac{a(n+1)}{a(n)} = \frac{\mathbf{E}\left(\sum_{i=0}^{n} a(i), 1\right)\sum_{i=0}^{n} a(i) - \sum_{i=1}^{n} a(i)}{a(n)} \tag{11}$$

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 $\mathrm{O}(N)$. This recursion is more desirable than solving the fixed-point equation given by (6), and then dealing with concerns regarding the existence and uniqueness of a fixed-point as well as convergence of iteration.

Proposition 1. *Given $a > 0$, the blocking probability perceived by an n-call can be computed in $\mathrm{O}(n)$ via the recursion*

$$A_n = \begin{cases} a, & n = 0, \\ A_{n-1} + a - \frac{A_{n-1}}{1+A_{n-1}}, & n > 0, \end{cases} \tag{12}$$

and then

$$b(n) = \frac{A_{n+1} - A_n}{A_n - A_{n-1}}, \quad n > 0. \tag{13}$$

Proof. The proof is presented in Appendix A. □

Analogous to (7), the OPCA estimate of blocking probability is given by

$$\begin{aligned} \widetilde{P}_{M_\mathrm{F}} &= \mathbb{P}(c = N\text{-call}) \\ &= b(0)b(1)b(2)\cdots b(N-1). \end{aligned} \tag{14}$$

3.4. Some rigorous results

We claim that for all $a \geqslant 0$ and $N \in \mathbb{N}$,

$$\widetilde{P}_{M_\mathrm{T}}(a,N) \leqslant \widetilde{P}_{M_\mathrm{F}}(a,N) \leqslant P_{M_\mathrm{T}}(a,N). \tag{15}$$

We prove the first inequality in (15) and we demonstrate the second one numerically. We have performed extensive numerical tests over a wide range of parameters and could not find a case where the second inequality in (15) does not hold. As such, we refer to the second inequality in (15) as the *OPC Conjecture*. From the numerical experiments we have conducted, we believe the difficulty in proving the second inequality is because it is quite tight, as demonstrated in Fig. 4, which is a very desirable property.

Our task is to prove $\widetilde{P}_{M_\mathrm{T}}(a,N) \leqslant \widetilde{P}_{M_\mathrm{F}}(a,N)$. Let

$$v_M(n) = \sum_{i=0}^{n} a_M(i), \quad M \in \{M_\mathrm{T}, M_\mathrm{F}\}, \tag{16}$$

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

To prove $\widetilde{P}_{M_\mathrm{T}}(a,N) \leqslant \widetilde{P}_{M_\mathrm{F}}(a,N)$, we first express $\widetilde{P}_{M_\mathrm{T}}(a,N)$ and $\widetilde{P}_{M_\mathrm{F}}(a,N)$ in terms of the common function given in Lemma 1, which then allows for the main result stated in Proposition 2.

Lemma 1. *For $M \in \{M_T, M_F\}$,*

$$\widetilde{P}_M(a,N) = 1 - \frac{v_M(N-1)(1 - \mathbf{E}(v_M(N-1),1))}{a}. \tag{17}$$

Proof. The proof is presented in Appendix B. □

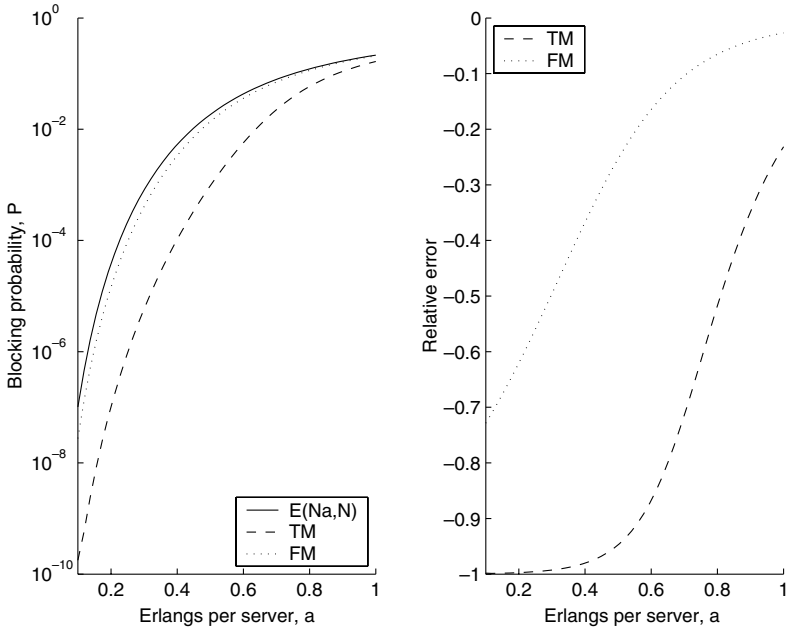

Fig. 4. Gauging the relative error in estimating blocking probability via the TM and FM estimate for $N = 10$.

Proposition 2. *For all* $a \geqslant 0$ *and* $N \in \mathbb{N}$,

$\widetilde{P}_{M_T}(a,N) \leqslant \widetilde{P}_{M_F}(a,N)$.

Proof. The proof is presented in Appendix C. □

Finally, we prove the following corollary of Proposition 2, which is pertinent to the discussion we gave in Section 3.2.

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*,

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

Proof. According to the proof of Proposition 2, $v_{M_F}(N-1) \leqslant v_{M_T}(N-1)$, $N \geqslant 1$, 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 $\sum_{j=0}^{N-1}\tilde{a}_{M_F}(j) \leqslant \sum_{j=0}^{N-1}\tilde{a}_{M_T}(j)$, which follows from the fact that $\sum_{j=0}^{N-1}\tilde{a}_{M_F}(j) = v_{M_F}(N-1) \leqslant v_{M_T}(N-1) = \sum_{j=0}^{N-1}\tilde{a}_{M_T}(j)$. □

4. Circuit-switched networks using alternative routing

This section will demonstrate the versatility of OPCA by using it to estimate blocking probability in a variety of circuit-switched networks using alternative routing.

4.1. A symmetric fully meshed circuit-switched network

Adopted is the usual model of a circuit-switched network that has been used in [12,13,20]. 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. 5. Calls arrive at each switching office pair according to independent and time-homogenous Poisson processes of intensity a. 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-

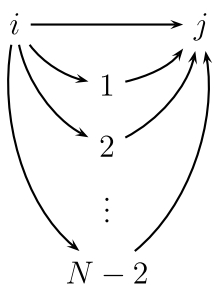

Fig. 5. Switching office pair (i,j) of a fully meshed circuit-switched network using alternative routing.

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.

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 applying EFPA to the TM gives

$$b = \mathbf{E}\left(a + 2ab(1-b)\sum_{j=0}^{N-3}(1-(1-b)^2)^j, K\right) \quad (18)$$

and that call blocking probability is estimated by

$$\widetilde{P}_{M_T} = b(1-(1-b)^2)^{N-2}. \quad (19)$$

Eq. (18) will be justified on a term-by-term basis. The factor of two multiplying the summation arises after enumerating all permutations in which a call can be offered to an arbitrary trunk group. The term $(1-(1-b)^2)^j$ 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 $a(1-b)^2$.

Eq. (19) 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 $(1-(1-b)^2)^{N-2}$.

It is difficult to ascertain properties regarding existence and uniqueness of solution for (18). 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 $b_{i+1} = \mathbf{E}\left(a + 2ab_i(1-b_i)\sum_{j=0}^{N-3}(1-(1-b_i)^2)^j, K\right)$ converges.

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,

$$b(n) = \begin{cases} \mathbf{E}(a, K), & n = 0, \\ \frac{B_n \mathbf{E}(B_n, K) - B_{n-1}\mathbf{E}(B_{n-1}, K)}{B_n - B_{n-1}}, & n > 0, \end{cases} \quad (20)$$

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

$$a_n = 2ab(0)(1-b(n))\prod_{j=1}^{n-1}(1-(1-b(j))^2), \quad n > 0 \quad (21)$$

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

$$\widetilde{P}_{M_F} = b(0)\prod_{j=1}^{N-2}(1-(1-b(j))^2). \quad (22)$$

Eq. (20) follows the same justification provided for (11).

The term $1-b(n)$ in (21) 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. In particular, we use the following iterative procedure specified in Algorithm 1. This iterative algorithm has been used extensively in the context of EFPA [12,13,20]. In Algorithm 1, $b_k(n)$ denotes the value of $b(n)$ at iteration k.

Algorithm 1

Calculate $b(1), \ldots, b(N-2)$
Require N, ϵ, a // Number of trunks, error criterion and offered load
$b_1(n)$, $b_0(n) \sim Uniform(0,1)$ $\forall n = 1, \ldots, N-2$ // Initialization
$k = 1$
while $\exists \ |b_k(n) - b_{k-1}(n)| > \epsilon$ for any $n = 1, \ldots, N-2$ **do**
  for $n = 1, \ldots, N-2$ **do**
    $a_n = 2ab_k(0)(1-b_k(n))\prod_{j=1}^{n-1}(1-(1-b_k(j))^2)$
    $B_n = \sum_{j=0}^{n} a_j$
    $b_{k+1}(n) = \begin{cases} \mathbf{E}(a, K), & n = 0 \\ \frac{B_n \mathbf{E}(B_n, K) - B_{n-1}\mathbf{E}(B_{n-1}, K)}{B_n - B_{n-1}}, & n > 0 \end{cases}$
  end for
  $k = k + 1$
end while
$\widetilde{P}_{M_F} = b_k(0)\prod_{j=1}^{N-2}(1-(1-b_k(j))^2)$ // Return

Although convergence of Algorithm 1 is not a certainty, divergence is rare in practice and can often be overcome by periodically re-initializing with a convex combination of the most recent iterations.

OPCA can be generalized to the case of circuit-switched networks protected with trunk reservation. With trunk reservation in place, an n-call can be preempted from a two-hop alternative route by an m-call, $m < n$, in the usual preemptive priority regime defined by OPC. However, an n-call, where $n > 0$, can *also* be barred from engaging a two-hop alternative route possessing an idle trunk on both of its constituent links if the total number of busy trunks that are engaged with $(0,\ldots,n)$-calls on either of the links is greater than or equal to a predefined threshold. Let that threshold be denoted by M. If the total number of busy trunks that are engaged with $(0,\ldots,n)$-calls on either of the links exceeds or equals M, the n-call must seek another alternative route; or, if it has sought to engage all two-hop alternative routes, it is blocked and cleared. See [16] for some rules of thumb governing the choice of M.

To generalize OPCA to the case of circuit-switched networks with trunkp reservation, (20) is replaced with

$$b(n) = \begin{cases} \mathbf{E}(a,K), & n = 0, \\ \dfrac{a\mathbf{Q}(n) + \mathbf{R}(n)\sum_{j=1}^{n} a_j - \sum_{j=0}^{n-1} a_j b(j)}{a_n}, & n > 0, \end{cases} \tag{23}$$

where $\mathbf{Q}(n)$ and $\mathbf{R}(n)$ are functions of the steady-state probabilities of a one-dimensional birth-and-death process characterizing a trunk group. In particular, $\mathbf{Q}(n) = \pi_K(n)$ and $\mathbf{R}(n) = \pi_M(n) + \pi_{M+1}(n) + \cdots + \pi_K(n)$, where for a given n, the steady-state probabilities $\{\pi_j(n)\}_{j=0}^{K}$ are computed via the recursion

$$\pi_j(n) = \begin{cases} \dfrac{(a_0 + a_1 + \cdots + a_n)^j \pi_0(n)}{j!}, & j = 1,\ldots,M, \\ \dfrac{a_0^{j-M}(a_0 + a_1 + \cdots + a_n)^M \pi_0(n)}{j!}, & j = M+1,\ldots,K. \end{cases}$$

For each n, the normalization constant $\pi_0(n)$ is determined by solving $\sum_{j=0}^{K} \pi_j(n) = 1$. Note that $a\mathbf{Q}(n) + \mathbf{R}(n)\sum_{j=1}^{n-1} a_j$ divided by B_n is the average steady-state blocking probability perceived by $(0,\ldots,n)$-calls. Eq. (21) does not require any modification for the case of trunk reservation. Trunk reservation will not be considered in the remainder of this paper.

An experiment was conducted in which blocking probability was estimated in a network comprising four switching offices with 10 trunks per trunk group. The error in estimating blocking probability via EFPA and OPCA was gauged against a simulation and is plotted in Fig. 6.

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.

4.2. Other circuit-switched networks

The error in estimating blocking probability via OPCA will be gauged for three other general circuit-switched networks and compared to EFPA as well as a simulation. To conclude, a somewhat artificial example will be constructed in which OPCA yields a 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. 10, where each double-arrowed line represents two trunk groups aligned in opposing directions, each 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 with respect to trunk groups are enumerated and stored in a routing table. The routing table is then ordered such that the shortest hop route is listed first and the longest hop route is listed last.

Calls arrive at each office pair according to independent Poisson processes of intensity a and sequentially traverse (without delay) the sorted routing table for an idle route. 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.

We gauged the error in estimating blocking probability via OPCA and EFPA for all three circuit-switched networks for the case $K = 10$. A guide to the numerical results is shown in Table 3. The intensity offered to each switching office pair was varied over a range that resulted in blocking

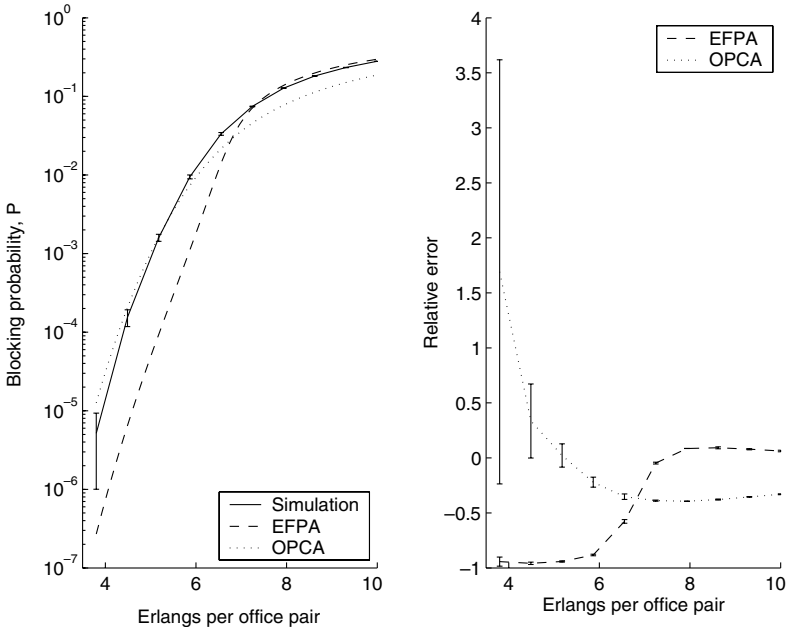

Fig. 6. Estimating blocking probability in a fully meshed circuit-switched network using alternative routing, $N = 4$, $K = 10$.

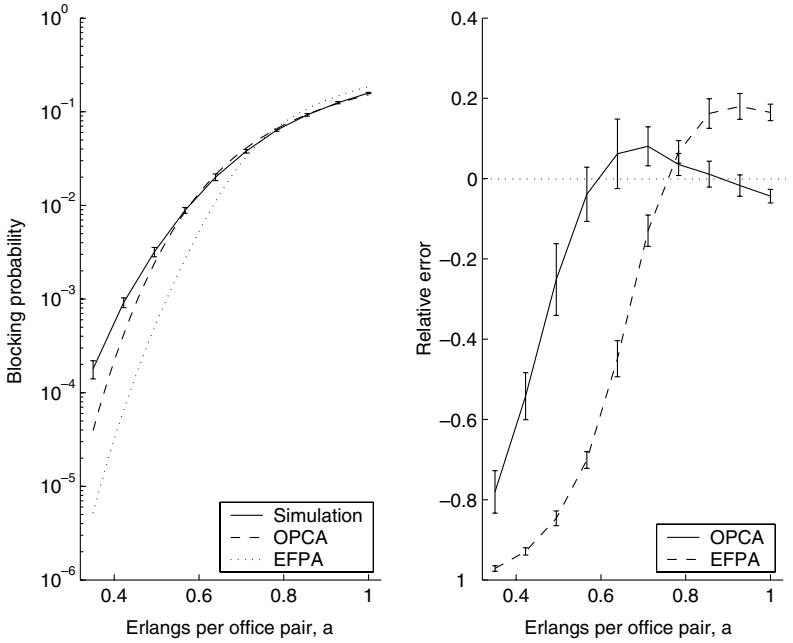

Fig. 7. Eight node ring network.

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 numerical results, it evident that OPCA provides a more accurate estimate of blocking probability for all three circuit-switched networks, assuming $K = 10$ and the considered routing strategy is in place. Since minimal additional computational effort is required to calculate an estimate via OPCA relative to EFPA, it seems that OPCA is the preferred approximation. The additional computational effort in calculating an estimate via OPCA is a consequence of the need

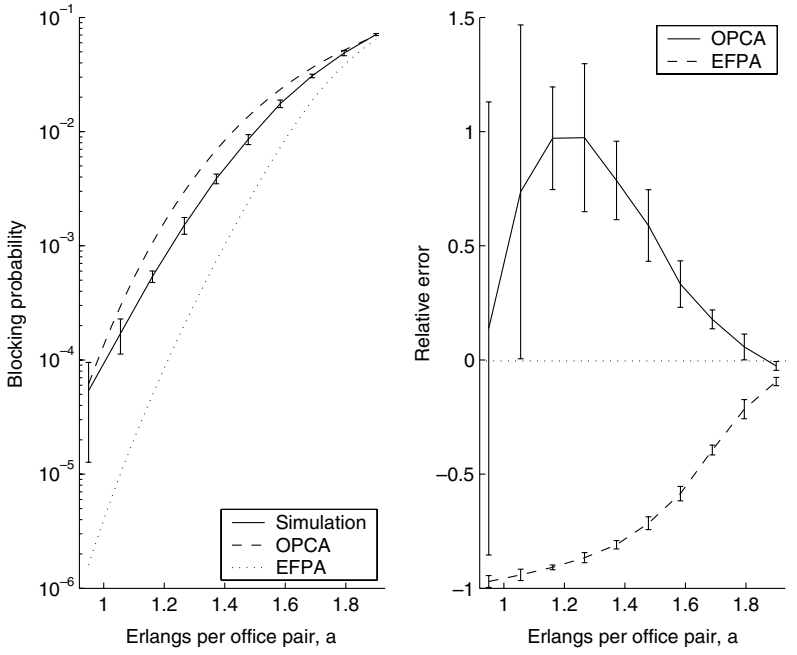

Fig. 8. Nine node wheel network.

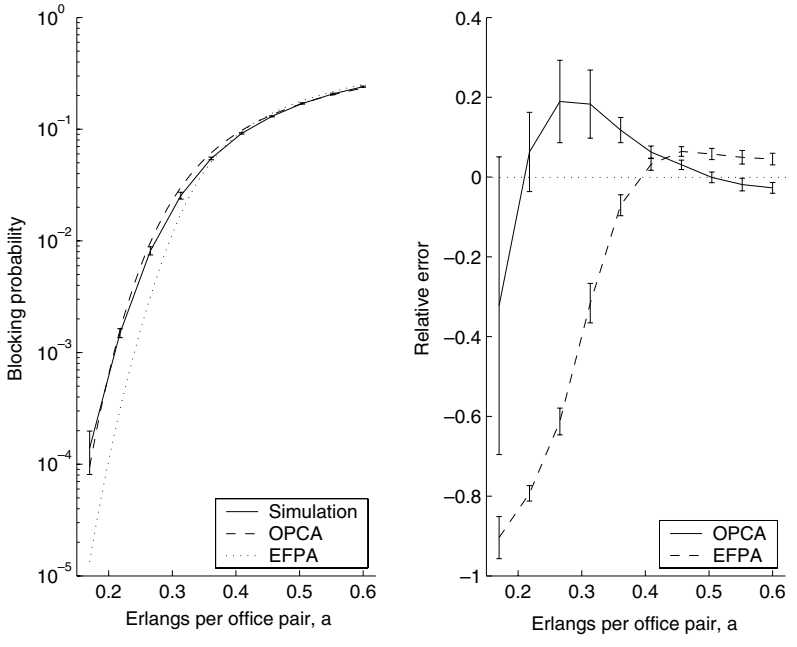

Fig. 9. NSF network.

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 parameters for the *single* combined stream formed by pooling together the marginal streams formed by $(0, 1, \ldots)$-calls.

To end this section, we construct an artificial example in which OPCA yields a poorer estimate of blocking probability than EFPA. 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

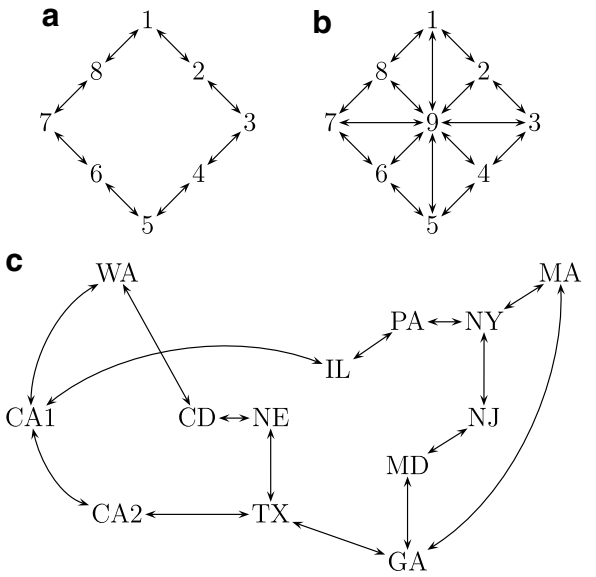

Fig. 10. Network topologies. (a) Eight node ring; (b) nine node wheel; (c) NSF (Version T1).

Table 3
Guide to numerical results

Network	Topology	Blocking probability
Eight node ring	Fig. 10a	Fig. 7
Nine node wheel	Fig. 10b	Fig. 8
NSF (T1)	Fig. 10c	Fig. 9

overflow in the usual manner prescribed by the random hunt. These calls are referred to as *premium calls* and arrive according to a Poisson process of intensity a^* 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.

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. 11.

Upon observing Fig. 11, it is clear that EFPA provides a better estimate of the blocking probability perceived by premium calls and standard calls. An

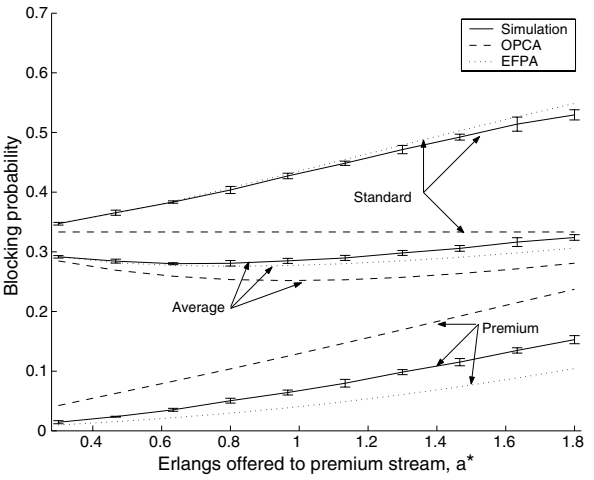

Fig. 11. An example in which OPCA performs poorly.

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 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), the inequality $P_{M_T} \leqslant P_{M_F}$ is not tight for this network. This examples serves as a warning against deeming OPCA to be a universally superior estimate of blocking probability. However, as a rule of thumb, OPCA usually performs well, and better than EFPA for networks of high symmetry and connectivity in which each traffic source is allowed to overflow many times before it is blocked.

5. Conclusion

This paper introduced a new approximation referred to as OPCA for estimating blocking probability in overflow loss networks. Unlike other methods that have improved EFPA through the enumeration of higher moments, OPCA is based on the most basic form of EFPA and hence remains simple and efficient but, through a system transformation, implicitly utilizes the congestion information

imbedded in the overflow traffic to achieve higher accuracy. OPCA was shown to outperform the conventional EFPA approach for the case of a distributed-server network as well as several cases of circuit-switched networks using alternative routing. Our rationale contends 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.

Acknowledgements

The authors are grateful of the two anonymous reviewers for their valuable comments and suggestions that have undoubtedly improved the presentation of the paper. This work was supported by the Australian Research Council (ARC).

Appendix A. Proof of Proposition 1

Let

$$A_n \triangleq \sum_{i=0}^{n} a(i),$$

where $a(i)$ is given by (9). Then according to (9), for $n > 0$ and $a > 0$, it follows that

$$b(n) = \frac{A_{n+1} - A_n}{A_n - A_{n-1}}.$$

Using (11), this can be rewritten as

$$\begin{aligned}\frac{A_{n+1} - A_n}{A_n - A_{n-1}} &= \frac{\mathbf{E}\left(\sum_{i=0}^{n} a(i), 1\right)\sum_{i=0}^{n} a(i) - \sum_{i=1}^{n} a(i)}{a(n)} \\ &= \frac{1}{a(n)}\left(\frac{A_n^2}{(1+A_n)} + a - A_n\right) \\ &= \frac{1}{(A_n - A_{n-1})}\left(a - \frac{A_n}{1+A_n}\right)\end{aligned}$$

Hence, resulting in the required recursion

$$A_{n+1} = A_n + a - \frac{A_n}{1+A_n}, \quad n \geqslant 0,$$

where $A_0 = a$. □

Appendix B. Proof of Lemma 1

Proof. For $M = M_T$, according to (5) and (7),

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

Using (6) in (24) gives

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

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

For $M = M_F$, according to (11),

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

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

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

where $v_{M_F}(-1) = 0$. Substituting (26) into (16) gives rise to a telescoping sum that results in the recursion

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

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

$$\begin{aligned}\widetilde{P}_{M_F} &= \frac{a + \Upsilon(v_{M_F}(N-1)) - v_{M_F}(N-1)}{a} \\ &= 1 - \frac{v_{M_F}(N-1)(1 - \mathbf{E}(v_{M_F}(N-1), 1))}{a}.\end{aligned}$$ □

Appendix C. Proof of Proposition 2

Proof. A simple rearrangement of Lemma 1 gives

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

Hence, it suffices to show that $v_{M_F}(N-1) \leqslant v_{M_T}(N-1)$. Induction will be used to show

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

According to (4),

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

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

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

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

$$v_{M_F}(1) = a + a\mathbf{E}(a,1) \leqslant a + a\mathbf{E}(v_{M_T}(N-1),1) \leqslant v_{M_T}(N-1).$$

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

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

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

Appendix D. Summary of notation

See Table 4.

Table 4
Summary of notation

n-call	A call that has overflowed from n servers or trunk groups
$(0,\ldots,n)$-call	Used to reference either a 0-call or a 1-call, ..., or an n-call
$a(0)$ or a	Exogenous load offered to a server or source and destination pair
$a(n)$	Load offered to a server or trunk group by calls that have overflowed n times
$b_i(n)$	Steady-state blocking probability perceived for a call that after overflowing n times seeks to engage server or trunk group i
$b(n)$	Used as a shorthand for $b_i(n)$ if all servers or trunk groups are statistically equivalent
b	Used only in the context of applying EFPA to the TM; denotes the probability that a server or trunk group is fully occupied
$\mathbf{E}(a,N)$	Blocking probability in an M/M/N/N queue offered intensity a
K	Number of trunks per trunk group
M	Trunk reservation threshold for circuit-switched networks; $M < K$
M_T and M_F	True model and fictitious model, respectively
P and $\widetilde{P}$	Exact and estimated system blocking probability, respectively, perceived by a call; see Table 2

References

[1] G.R. Ash, B.D. Huang, An analytical model for adaptive routing networks, IEEE Transaction on Communications 41 (11) (1993) 1748–1759.

[2] P. Chevalier, N. Tabordon, Overflow analysis and cross-trained servers, International Journal of Production Economics 85 (2003) 4760.

[3] S.P. Chung, A. Kashper, K.W. Ross, Computing approximate blocking probabilities for large loss networks with state-dependent routing, IEEE/ACM Transactions on Networking 1 (1) (1993) 105–115.

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

[5] L.E.N. Delbrouck, The use of Kosten's systems in provisioning of alternate trunk groups carrying heterogeneous traffic, IEEE Transactions of Communications COM-31 (6) (1983) 741–749.

[6] D. Deloddere, W. Verbiest, H. Verhille, Interactive video on demand, IEEE Communications Magazine 32 (5) (1994) 82–88.

[7] A.A. Fredricks, Congestion in blocking systems—a simple approximation technique, The Bell System Technical Journal 59 (6) (1980) 805–827.

[8] A. Girard, Routing and Dimensioning in Circuit-Switched Networks, Addison-Wesley, 1990.

[9] R. Guerin, L.Y.-C. Lien, Overflow analysis for finite waiting room systems, IEEE Transactions on Communications 38(Sept.) (1990) 1569–1577.

[10] J.M. Holtzman, Analysis of dependence effects in telephone trunking networks, The Bell System Technical Journal 50 (8) (1971) 2647–2662.

[11] S. Katz, Trunk engineering of non-hierarchical networks, International Teletraffic Congress 6 (1971) 142.1–142.8.

[12] F.P. Kelly, Blocking probabilities in large circuit-switched networks, Advances in Applied Probability 18 (1986) 473–505.

[13] F.P. Kelly, Loss networks, The Annals of Applied Probability 1 (3) (1991) 319–378.

[14] A. Kuczura, The interrupted Poisson process as an overflow process, The Bell System Technical Journal 52 (3) (1973) 437–448.

[15] A. Kuczura, D. Bajaj, A method of moments for the analysis of a switched communication network's performance, IEEE Transactions on Communications COM-25 (2) (1977) 185–193.

[16] R.S. Krupp, Stabilization of alternate routing networks, in: Proceedings of IEEE ICC 1982, Philadelphia, USA, June 1982.

[17] D. Mitra, Asymptotic analysis and computational methods for a class of simple circuit-switched networks with blocking, Advances in Applied Probability 19 (1987) 219–239.

[18] D. Mitra, J.A. Morrison, K.G. Ramakrishnan, ATM network design and optimization: a multirate loss network framework, IEEE/ACM Transactions on Networking 4 (4) (1996) 531–543.

[19] Z. Rosberg, H.L. Vu, M. Zukerman, J. White, Performance analyses of optical burst switched networks, IEEE Journal on Selected Areas in Communications 21 (Sept.) (2003) 1187–1197.

[20] W. Whitt, Blocking when service is required from several facilities simultaneously, AT&T Technical Journal 64 (1985) 1807–1856.
[21] R.I. Wilkinson, Theories of toll traffic engineering in the U.S.A., Bell System Technical Journal 35 (2) (1956) 421–514.
[22] I. Widjaja, Performance analysis of burst admission control protocols, IEE Proceeding on Communications 142 (Feb.) (1995) 7–14.
[23] 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 3 (June) (1990) 934–937.
[24] E.W.M. Wong, T.S. Yum, K.M. Chan, Analysis of the M and M^2 routings in circuit-switched networks, European Transactions on Telecommunications 6 (5) (1995) 613–619. An earlier version appeared in Proc. IEEE GLOBECOM'92, Orlando, FL, vol. 3, pp. 1487–1492, December 1992.
[25] E.W.M. Wong, T.S. Yum, K.M. Chan, A taxonomy of rerouting in circuited-switched networks, IEEE Communications Magazine 37 (Nov.) (1999) 116–122.
[26] E.W.M. Wong, K.M. Chan, T.S. Yum, Analysis of rerouting in circuit-switched networks, IEEE/ACM Transactions on Networking 8 (3) (2000) 419–427.
[27] E.W.M. Wong, M.Y.M. Chiu, Z. Rosberg, M. Zukerman, S. Chan, A. Zalesky, A novel method for modeling and analysis of distributed video on demand systems, in: Proc. ICC 2005, Seoul, Korea, May 2005.
[28] E.W.M. Wong, A. Zalesky, Z. Rosberg, M. Zukerman, A novel analysis of overflow loss networks (Extended Version), Internal Technical Report. Available from: <http://www.ee.cityu.edu.hk/~ewong/opc_extended.pdf>.
[29] 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, in: Proceedings of INFOCOM 2004, Hong Kong, China, March 2004, vol. 3, pp. 1864–1871.

Eric W.M. Wong received the B.Sc. and M.Phil. degrees in Electronic Engineering from the Chinese University of Hong Kong, Hong Kong, in 1988 and 1990, respectively, and the Ph.D. degree in Electrical and Computer Engineering from the University of Massachusetts, Amherst, in 1994. In 1994, he joined the City University (CityU) of Hong Kong, where he is now an Associate Professor in the Department of Electronic Engineering and a Member of the Optoelectronics Research Centre. His research interests are in the analysis and design of telecommunication networks, optical burst switching, and video-on-demand. His most notable research work involved the first workable model on state dependent dynamic routing. Since 1991, the model has been used by AT&T to design and dimension its telephone network using real-time network routing.

Andrew Zalesky received the B.E. degree in Electrical Engineering in 2002 and the B.Sc. degree in Applied Mathematics in 2003, both from the University of Melbourne, Australia. He received the Ph.D. degree in electrical engineering from the same institution in 2006. His research interests are in operations research. He is particularly interested in stochastic performance modeling of telecommunications networking.

Zvi Rosberg received the B.Sc., M.A. and Ph.D. degrees from the Hebrew University of Jerusalem. During his graduated studies he was a senior system analyst in the Central Computing Bureau of the Israeli government, where he was one of the chief designers of a new on-line Israeli population registration system. After graduation he held a research fellowship at the Center of Operation Research and Econometric (C.O.R.E.), Belgium and a visiting assistant professorship at the department of Business Administration, University of Illinois. At 1980 he joined the Computer Science department, Technion, Israel where he was until 1990. From 1990 to 1999 he was with the Haifa Research Laboratory, Science and Technology, IBM Israel, holding a position of a Program Manager of Communication Networks. From 2000 to 2001 he was with Radware Ltd., holding the chief scientist position. During the year of 2002 he visited the ARC Special Research Centre for Ultra-Broadband Information Networks (CUBIN), University of Melbourne. Currently he is with the Department of Communication Systems Engineering, Ben Gurion University, Beer-Sheva.

Since 1980 he held summer research positions and a two year visiting position in IBM Thomas J. Watson Research Center, Yorktown Heights. He also had summer research positions in the Department of Electrical and Computer Engineering, University of Massachusetts, Amherst, Department of Electrical Engineering and Computer Science, University of California, Berkeley, the Radio Communication Systems, Royal Institute of Technology (KTH) in Stockholm, the ARC Special Research Center for Ultra-Broadband Information Networks (CUBIN), University of Melbourne, and the Department of EEE, City University, Hong Kong. Presently, he is serving on the editorial board of the Wireless Networks (WINET) and the International Journal of Communication Systems. His research interest, where he has published numerous papers, include: Narrowband and spread spectrum wireless communication, Radio resource allocation and planning in cellular networks, Scheduling in wireless networks, Optical and ultra high speed networks, Control in queueing networks, Analysis of algorithms in communication and computing systems and Internet technologies.

Moshe Zukerman received his B.Sc. in Industrial Engineering and Management and his M.Sc. in Operations Research from Technion – Israel Institute of Technology and a Ph.D. degree in Electrical Engineering from the University of California Los Angeles in 1985. He was an independent consultant with IRI Corporation and a post-doctoral fellow at UCLA during 1985–1986. During 1986–1997 he served in Telstra Research Laboratories (TRL), first as a research engineer and, during 1988–1997, as a project leader. He is the recipient of the Telstra Research Laboratories Outstanding Achievement Award in 1990.

In 1997 he joined The University of Melbourne where he is now a professor responsible for promoting and expanding telecommunications research and teaching in the Electrical and Electronic Engineering Department. He has also taught and supervised graduate students at Monash University during 1990–2001. He served on the editorial board of the Australian Telecommunications Research Journal, Computer Networks, and the IEEE Communications Magazine. He also served as a Guest Editor of IEEE JSAC for two issues: Presently, he is serving on the editorial board of the IEEE/ACM Transactions on Networking and the International Journal of Communication Systems. He has over 200 publications in scientific journals and conference proceedings. He co-authored two award winning conference papers.