Bounds of the Overflow Priority Classification for Blocking Probability Approximation in OBS Networks

Shuo Li, Meiqian Wang, Eric W. M. Wong, *Senior Member, IEEE,* Vyacheslav Abramov, Moshe Zukerman, *Fellow, IEEE*

Abstract – It has been demonstrated that the Overflow Priority Classification Approximation (OPCA) is an accurate method for blocking probability evaluation for various networks and systems including optical burst switched networks with deflection routing. OPCA is a hierarchical algorithm that requires fixed-point iterations in each layer of its hierarchy. This may imply a long running time. We prove here that the OPCA iterations alternately produce upper and lower bounds which consistently become closer to each other as more fixed-point iterations in each layer are used, and we demonstrate numerically that only a small number of iterations per layer are required for the bounds to be sufficiently close to each other. This behavior is demonstrated for various system parameters including offered load, number of channels per trunk, and maximum allowable number of deflections.

Index Terms: Performance analysis, deflection routing, optical burst switching (OBS), optical packet switching (OPS), non-hierarchical networks, loss networks, overflow priority classification approximation, lower and upper bounds

I. Introduction

Optical burst switching (OBS) [1]–[7] is an optical networking technique, where packets are aggregated into data bursts at the network edge and each burst is transmitted as one unit. It aims to achieve faster connection time than optical circuit switching (OCS) [8]–[11], and to avoid energy consuming processing of individual packets and excessive overhead due to guard-band provision between packets, as in optical packet switching (OPS) [6], [12]–[14].

An important consideration in OBS networks is burst contention that may lead to burst dumping and consequently loss of data [15]–[18]. Given that buffering data in the optical domain is difficult, especially for large bursts, one of the key contention resolution options is deflection routing [18]. As such, OBS deflection routing and its performance analysis has attracted significant attention [19]–[32]. However, most of the existing performance studies were either based on simulations, or were limited to a single OBS node. In [32], the blocking probability of an OBS network with deflection was evaluated using the Erlang Fixed Point Approximation (EFPA) [33]. Then, in [34], a recently developed Overflow Priority Classification Approximation (OPCA) [35] was used in combination with EFPA to obtain blocking probability approximation for this problem which, as demonstrated there, is significantly more accurate than the merely approximation using EFPA. Given the important role of OPCA in accurately evaluating blocking probability of telecommunications networks and systems as demonstrated in [34] for OBS networks, and in [35], [36] for other networks, the main focus of this paper is not on the accuracy and performance of OPCA; instead, it provides new information about the properties of the OPCA algorithm applied to bufferless OBS networks based on Just-Enough-Time (JET) signalling [37]. This new information has the potential for wider applicability as it can be used for further development of OPCA in other applications.

A weakness of both EFPA and the original OPCA applied to OBS networks is that they require a fixed-point solution, which may require a large number of iterations. Because of the fixed-point iterations, analytical results for the complexities of both EFPA and OPCA are unattainable. However, numerical studies presented in this paper indicate that OPCA consumes less time than EFPA, and that the advantage of OPCA increases with the capacity of the network (see Section V-B). For example, EFPA requires 3006, versus 397 seconds for OPCA, to evaluate blocking probability for an NSFNet topology with 10000 channels per trunk (such a number is not unreasonable [38]–[40], especially if sub-wavelength channels are considered [41]). The advantage of OPCA in running time is probably due to the fact that OPCA is based on a hierarchical structure with a finite number of layers, where at each layer, a separate set of fixed-point iterations are performed. Experience shows that this divide and rule approach tends to reduce the total number of iterations. Although accuracy is not the main topic of this paper, we do provide some new numerical results that complement the running time comparison and demonstrate that for the cases of OBS/JET considered, within a practical traffic loading range, i.e., so that acceptable grade of service (GoS) is met, OPCA is faster than and at least as accurate as EFPA.

We discover in this paper a new important property of the OPCA algorithm. In particular, we show that we can find upper and lower bounds for the blocking probability evaluated by OPCA that they draw near each other with increasing number of iterations. Specifically, in each iteration, the distance between them is never larger than in the previous iteration. The

Shuo Li, Meiqian Wang, Eric Wong and Moshe Zukerman are with the Department of Electronic Engineering, City University of Hong Kong, Hong Kong SAR.

Vyacheslav Abramov is with the Center for Advanced Internet Architectures, Swinburne University of Technology, John Street, P.O. Box 218, Hawthorn 3122, Australia.

effects of the design parameters (such as maximum allowable number of deflection and number of channels per trunk) on the behavior of the OPCA blocking probability bounds are also discussed. It is important to clarify that the *bounds* discussed in this paper are always *the bounds of the OPCA result* and not the bounds of the exact blocking probability result.

The remainder of the paper is structured as follows. The description of the model is given in Section II. In Section III, we recall basics of the OPCA method and in Section IV we provide bounds of the OPCA results. To support the results of Section IV, in Section V, numerical results for the OPCA bounds for the 13-node National Science Foundation network (NSFNET) are provided, as well as the effects of the number of channels per trunk and the maximum allowable number of deflections. In Section VI, we conclude the paper.

II. The Model

We consider an OBS network described by a graph $G(\mathcal{N},\mathcal{E})$, where $\mathcal{N}$ is a set of n nodes and $\mathcal{E}$ is the set of e arcs. The nodes are designated $1,2,\dots,n$, each of which is either an optical cross connect or an edge-router. The e arcs represent trunks, where trunk $i\in\mathcal{E}$ is composed of f_i fibers, each of which supports w_i wavelengths. We assume full wavelength conversion in this paper, so accordingly, trunk $i\in\mathcal{E}$ carries $C_i=f_iw_i$ unidirectional wavelength channels, which are called *channels*. Note that our model is also applicable for networks where OBS uses sub-wavelength channels [41] in which case the term channel represents a sub-wavelength channel. If all trunks have the same number of channels, then $C_j=C$ for all j. However, we note that the results presented in this paper are equally applicable to networks with no wavelength conversion which has f_j, instead of f_jw_j, channels on each intermediate trunk (excluding the first trunk) in a route.

Each unique pair of origin and destination nodes forms a directional source-destination (SD) pair, m. The set of all SD pairs in the network is denoted $\beta=\{1,2,\dots,N(N-1)\}$. Thus, $m=\{x,y\}\in\mathcal{N}$ represents traffic composed of bursts sent from node x to node y. These bursts arrive at node x according to a Poisson process with parameter ρ_m. For tractability, the burst lengths are assumed to be exponentially distributed with unit mean. The effect of this exponential assumption has been numerically studied in [34], and it has been demonstrated by comparison to scenarios involving heavy tailed bursts that the performance results are only to a small extent sensitive to the shape of the burst length distribution. It is very likely that for a directional SD pair $m\in\beta$, there are more than a single route between the source and the destination. We designate a route with the least number of hops as the primary path of the SD pair. Then all the other routes are ranked alternative paths.

It is convenient to maintain the set

$$\{\mathbf{U}_m(0),\mathbf{U}_{m,j_1}(1),\mathbf{U}_{m,j_2}(1),\dots,\mathbf{U}_{m,j_n}(T_m)\}$$

of the alternative routes for the directional SD pair $m\in\beta$. In this set, $\mathbf{U}_m(0)$ denotes the primary path, and $\mathbf{U}_{m,j}(d)$ denotes the alternative path with traffic deflected from trunk j, that including this deflection has already been deflected d times. T_m is the maximum number of available alternative paths for the directional SD pair m. Note that T_m is based on the network topology which limits the number of available alternative paths. For example, $T_m=0$ in the trivial example of a network of two nodes and two opposite-directional trunks that connect the two nodes.

In our model, the ranking of alternative paths is based on the number of hops and in case of equality of the number of hops, the rank is chosen randomly. However, in practice, various cost functions (e.g. geographic distance) can be also used for ranking. If capacity is available on all trunks of the primary path, then it will be used for the transmission of a burst from the source node x to the destination node y. However, if all the channels are occupied on at least one of the trunks of the primary route, then a burst will be deflected to the first trunk of the first alternative path of that blocked trunk. If there is a free channel on this trunk, then the burst is transmitted on it; otherwise, the burst is deflected to the first trunk of the second alternative path.

A given burst is permitted to be deflected at most D times. A burst is blocked, namely, dumped and cleared from the network, if it arrives at a given node where all output trunks are busy or while trying alternative trunks, the burst reaches the limit D of allowable number of deflections. Setting the limit D implies that a burst in the directional SD pair m can be deflected no more than

$$T(m)=\min\{T_m,D\}$$

times.

This paper does not consider the use of trunk reservation as in [34]. Obtaining OPCA bounds for OBS/JET network with trunk reservation is still an open problem.

III. Overflow Priority Classification Approximation for OBS Networks

A detailed description of OPCA is given in [34], [35]. To be self-contained, the paper repeats this definition using the earlier notation of [34].

For each SD pair m, let ρ_m be the offered traffic load. The term k-deflection burst is used to represent a burst that has been deflected k times ($k\in\{1,\dots,T(m)\}$). Original bursts are the bursts that have not been deflected. In other words, they are 0-deflection bursts. Let $a_j^k(m)$ be the k-deflection bursts' offered traffic load of SD pair m on trunk $j\in\mathcal{E}$, and let b_j^k denote the probability that a k-deflection burst is blocked on trunk j. If the first trunk of the primary route between SD pair m is i_1, then the offered load to this trunk is equated to the offered load of the SD pair, i.e. $a_{i_1}^0(m)=\rho_m$. By the *carried load* of the first trunk i_1, we mean the proportion of the offered load to trunk i_1 that is not blocked in trunk i_1. Then the offered load in the second trunk i_2 is equated to the carried load of trunk i_1, i.e.,

$$a_{i_2}^0(m)=a_{i_1}^0(m)(1-b_{i_1}^0)=\rho_m(1-b_{i_1}^0). \tag{1}$$

When the network is congested and the trunk of a given route is fully occupied, then the bursts that initially tried to use the original trunk are deflected onto alternative trunks and routes. Let trunk i be the busy trunk that transmits k-deflection bursts. Then there is a deflection on the present route being caused by this trunk i. The load offered to the first trunk l_1 of the first choice alternative route is related to the load offered to trunk i by

$$a_{l_1}^{k+1}(m) = a_i^k(m)b_i^k, \tag{2}$$

where k is the number of deflections prior to the latest deflection. Similarly, due to the deflection from l_1, the load offered to the first trunk l_2 of the second choice alternative route is

$$a_{l_2}^{k+2}(m) = a_{l_1}^{k+1}(m)b_{l_1}^{k+1} = a_i^k(m)b_i^k b_{l_1}^{k+1}. \tag{3}$$

Let a_j^k be the total offered load, of k-deflection bursts, on trunk j. The variables a_j^k and $a_j^k(m)$ for $k = 0, 1, \ldots, D$ are related by

$$a_j^k = \sum_{m \in \beta} a_j^k(m). \tag{4}$$

Assume

$$I(i, j, \mathbf{U}_{m,p}(k)) = \begin{cases} 1, & \text{if } i, j \in \mathcal{E} \text{ and trunk } i \text{ strictly preceds} \\ & \text{(not necessarily immediately) trunk } j \\ & \text{along } k \text{ deflection route } \mathbf{U}_{m,p}(k) \\ 0, & \text{otherwise.} \end{cases}$$

Equation (4) can also be written as

$$a_j^k = \sum_{m \in \beta, j \in \mathbf{U}_{m,p}(k)} \rho_{m,p}^k \prod_{i \in \mathcal{E}} (1 - I(i, j, \mathbf{U}_{m,p}(k))b_i^k), \tag{5}$$

where $\rho_{m,p}^k$ is the offered load from trunk p to the kth deflection route of SD pair m for $k > 1$, and

$$a_j^0 = \sum_{m \in \beta, j \in \mathbf{U}_m(0)} \rho_m \prod_{i \in \mathcal{E}} (1 - I(i, j, \mathbf{U}_m(0))b_i^0), \tag{6}$$

for the primary layer.

In addition, let $\tilde{a}_j^k$ be the offered load of bursts that have been deflected up to k times, i.e.

$$\tilde{a}_j^k = \sum_{h=0}^{k} a_j^h. \tag{7}$$

The averaged blocking probability $\bar{b}_j^k$ on trunk $j \in \mathcal{E}$, for bursts with deflections up to k times, is equal to

$$\bar{b}_j^k = E\left(\tilde{a}_j^k, C_j\right), \tag{8}$$

where $E(x, C) = \frac{x^C/C!}{\sum_{n=0}^{C} x^n/n!}$ is the Erlang-B formula with offered load x and the number of channels per trunk is C. The blocking probability, for k-deflection bursts, $k \in \{0, \ldots, D\}$, on trunk j is estimated by

$$b_j^k = \begin{cases} \bar{b}_j^0, & k = 0 \\ \frac{\bar{b}_j^k \tilde{a}_j^k - \bar{b}_j^{k-1} \tilde{a}_j^{k-1}}{a_j^k}. & 1 \le k \le D \end{cases} \tag{9}$$

Note that the blocking probability of undeflected bursts is calculated by using the Erlang-B formula.

To obtain the OPCA blocking probability estimates, we start with the primary traffic, i.e., $k = 0$. Then, we solve the fixed-point equations described by (6-9), with the aid of the successive substitution method in order to obtain the values a_j^0 for $j \in \mathcal{E}$ and $\bar{b}_j^0 = b_j^0 = E\left(a_j^0, C_j\right)$. These calculations for the primary traffic and blocking probabilities are defined as layer 0 calculations.

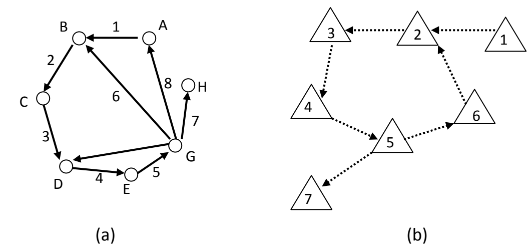


Fig. 1. Model for LBT: (a) a network model, (b) an example for direct trunks and LBT in the network shown in (a).

Next, having completed the layer 0 calculations to obtain the parameters related to the primary traffic ($k = 0$), we progress to compute the parameters associated with the first deflection traffic ($k = 1$). Similarly, we solve the fixed-point equations (5) and (7-9) (existence of a solution follows from Brouwer's theorem [42]; e.g. see [43] where Brouwer's theorem [42] was used to prove existence of a solution of the fixed-point equation used in that paper) to obtain the values a_j^1 for $j \in \mathcal{E}$, as well as $\bar{b}_j^1$ and b_j^1 using equations (8) and (9), respectively, for every $j \in \mathcal{E}$, where $\tilde{a}_j^1$ is given by (7).

Then, having completed the layer 0 and layer 1 calculations to obtain the parameters related to the primary and the first deflection traffic ($k = 0$ and $k = 1$), we compute the parameters associated with the second deflection traffic ($k = 2$), which we call layer 2 calculations.

The process of deriving the parameters for $k > 1$ repeats itself until we have all the parameter values for all $k \in \{1, \ldots, D\}$ layers.

IV. Bounds of trunk blocking probabilities of OPCA

In this section we derive the OPCA upper and lower bounds and some properties of the bounds.

Let $b_j^{k_s}$ denote the blocking probability obtained in the sth iteration for k-deflection burst (kth layer) on trunk $j \in \mathcal{E}$, and let $a_j^{k_s}$ be the offered traffic load obtained in the s iteration for k-deflection bursts on trunk $j \in \mathcal{E}$. Let $\{a_j^{k^*}, b_j^{k^*}\}$ be a set of pairs of the fixed-point solutions of equation (5-9). Let us denote by $a_j^{k^*_{min}}$ and $a_j^{k^*_{max}}$ the minimum and maximum values amongst the set of values $\{a_j^{k^*}\}$, respectively, and denote by $b_j^{k^*_{min}}$ and $b_j^{k^*_{max}}$ the minimum and maximum values amongst the set of values $\{b_j^{k^*}\}$, respectively. Denote the upper and lower bounds of the set $\{b_j^{k^*}\}$ by $b_j^{k_U}$ and $b_j^{k_L}$, respectively, and let the upper and lower bounds of the set $\{a_j^{k^*}\}$ be $a_j^{k_U}$ and $a_j^{k_L}$, respectively.

Consider now a network model presented in Fig. 1 part (a). There are three SD pairs: A to D, D to H and C to B with traffic in the network, where only primary paths are allowed. In Fig. 1, part (b), triangles are used to represent a trunk, and arrows from any trunk i to trunk j are used in the cases where traffic that passes through trunk i follows directly to trunk j. Trunks from 2 to 6 in that figure form a closed loop.

For trunks that are inside a closed loop, such as trunks 2-6 in Fig. 1, part (b), and for trunks that receive any traffic that passed through the trunks that are inside a closed loop, such as trunk 7 in the same figure, their offered load and trunk blocking probability are calculated on the basis of the successive substitution method leading finally to the fixed-point solution. For any layer k, we will define a set of trunks that include all such trunks.

Definition A set of trunks $\mathcal{L}_k$ is a *loop based trunk* (LBT) *set* in layer k, if the following hold:

(i) $\mathcal{L}_k$ has at least one closed loop of trunks in layer k.

(ii) If in layer k, trunk α receives traffic from any trunk in $\mathcal{L}_k$, then $\alpha \in \mathcal{L}_k$.

If a trunk belongs to an LBT set, the trunk is designated as an LBT.

For simplicity, in the example presented in Fig.1, we consider only one trunk (trunk 7) that is not included in the closed loop, but it is part of the loop tree. In general, we can have a large set of trunks (which may even include other loops) that receive traffic that from the LBT, and such a set of trunks is included in the LBT set.

All the trunks that are not LBTs, in layer k, are called direct trunks in layer k. There are the following three examples (non-exhaustive) for the direct trunks in layer k.

(i) A direct trunk where there is no traffic in a trunk in layer k and, so there is no offered load there. In this case, the trunk blocking probability in layer k is set to zero.

(ii) A set of trunks that form a tree structure (namely, traffic is directed towards the leaves of the trees) in layer k. In this case, all the bounds of the offered load and trunk blocking probabilities for all trunks can be calculated one by one directly without iterations in that layer, from top to bottom of the tree.

(iii) A set of trunks that feed traffic to a closed loop in layer k – such trunks are not in the loop and they do not receive traffic from the loop. For instance, it is trunk 1 in Fig. 1, part (b). In this case, the bounds of offered load and trunk blocking probability are calculated one by one directly without iterations as well. These trunks can also form one or more tree structure.

Calculation of the bounds for the trunk blocking probabilities starts from layer 0. After $s \geq 2$ iterations, for trunk $j \in \mathcal{E}$, we obtain the upper bound $b_j^{0_U}$ and the lower bound $b_j^{0_L}$ for the trunk blocking probability, and the upper bound $a_j^{0_U}$ and the lower bound $a_j^{0_L}$ for the trunk offered load. We prove that $b_j^{0_U} = b_j^{0_L}$ and $a_j^{0_U} = a_j^{0_L}$ for the direct trunks. For the LBTs, when s increases, $b_j^{0_U}$ and $b_j^{0_L}$ become closer to each other and the set of fixed-point solutions for trunk blocking probability $b_j^{0^*}$ are always between $b_j^{0_U}$ and $b_j^{0_L}$; $a_j^{0_U}$ and $a_j^{0_L}$ also become closer to each other and the set of fixed-point solutions for trunk offered load $a_j^{0^*}$ are always between $a_j^{0_U}$ and $a_j^{0_L}$. Then the upper and lower bounds for the traffic load offered from SD pair m to trunk j is calculated based on the bounds of the trunk blocking probabilities. After that, we calculate the upper bound $a_q^{0_U}(m)b_q^{0_U}$ and the lower bound $a_q^{0_L}(m)b_q^{0_L}$ of the overflowed traffic to layer 1 from SD pair m on the congested trunk q in layer 0, for $m \in \beta, q \in \mathbf{U}_m(0)$.

After obtaining the bounds for the offered load and trunk blocking probabilities for all trunks, including direct trunks and LBTs in layer 0, we then perform the calculations for layer 1. After $s \geq 2$ iterations, for trunk $j \in \mathcal{E}$, we obtain the upper bound $b_j^{1_U}$ and lower bound $b_j^{1_L}$ for the trunk blocking probability, and the upper bound $a_j^{1_U}$ and the lower bound $a_j^{1_L}$ for the trunk offered load. For direct trunks, we obtain either $a_{1j}^{1_U} = a_j^{1^*_{max}} = a_j^{1^*_{min}} = a_j^{1_L}$ and $b_j^{1_U} = b_j^{1^*_{max}} = b_j^{1^*_{min}} = b_j^{1_L}$, or $a_j^{1_U} > a_j^{1^*_{max}} \geq a_j^{1^*_{min}} > a_j^{1_L}$ and $b_j^{1_U} > b_j^{1^*_{max}} \geq b_j^{1^*_{min}} > b_j^{1_L}$. For LBTs, the bounds have the same properties as those for LBTs in layer 1. Then we calculate the bounds of the overflowed traffic to layer 2.

The procedure repeats itself until the bounds of the offered load and blocking probabilities in each trunk for all layers are found. In each layer, we first calculate the bounds for the offered load and trunk blocking probabilities for the direct trunks, and then, we iteratively calculate the bounds for the LBTs. Then, based on the bounds of the trunk offered load and blocking probabilities, we obtain the bounds of the network blocking probabilities estimated by OPCA, and prove that the fixed-point solutions are always between these upper and lower bounds.

The following subsections provide the equations used for obtaining the bounds for the blocking probability and the offered load estimated by OPCA in each trunk, for all the layers, and the OPCA network blocking probability, as well as the proof of the behavior of the bounds.

A. Bounds of trunk blocking probabilities for direct trunks

If the primary layer (layer 0) is a layer with direct trunks, then its upper and lower bounds for the offered load and trunk blocking probabilities are calculated from top to bottom of the tree by the following equations:

$$a_j^{0_U} = \sum_{m \in \beta, j \in \mathbf{U}_m(0)} \rho_m \prod_{i \in \mathcal{E}} (1 - I(i,j,\mathbf{U}_m(0)) b_i^{0_L}), \tag{10}$$

$$a_j^{0_L} = \sum_{m \in \beta, j \in \mathbf{U}_m(0)} \rho_m \prod_{i \in \mathcal{E}} (1 - I(i,j,\mathbf{U}_m(0)) b_i^{0_U}), \tag{11}$$

$$b_j^{0_L} = E\left(a_j^{0_L}, C_j\right), \tag{12}$$

and

$$b_j^{0_U} = E\left(a_j^{0_U}, C_j\right). \tag{13}$$

The calculations start from trunk j on the top of the tree. The offered load of this trunk does not depend on those in other trunks, so its bounds of offered load are calculated directly using the equations (10-11). We obtain $a_j^{0_U} = a_j^{0^*_{max}} = a_j^{0^*_{min}} = a_j^{0_L}$. This means the uniqueness of the fixed-point solution for this trunk j. Then, substituting the bounds for the offered load into equations (12-13), we obtain the bounds of the trunk blocking probability of this trunk, $b_j^{0_U} = b_j^{0^*_U} = b_j^{0^*_L} = b_j^{0_L}$. After that, we pass to the next trunk that bellow the top trunk in the tree. The bounds of the offered load of this trunk depend only on those of the top trunk, so substituting the bounds of the trunk blocking probability of the top trunk into equations (10-11), we obtain the bounds of the offered load for this second top trunk. Then substituting these bounds into equations (12-13), we obtain the bounds of the trunk blocking probability of this second top trunk. Repeating the steps from top to bottom in the tree, we obtain all the bounds of the offered load and trunk blocking probability one by one. They are $a_j^{0_U} = a_j^{0^*_{max}} = a_j^{0^*_{min}} = a_j^{0_L}$ and $b_j^{0_U} = b_j^{0^*_{max}} = b_j^{0^*_{min}} = b_j^{0_L}$.

For the same primary layer, the lower and upper bounds of the offered load for SD pair m are calculated by the formulae

$$a_j^{0_L}(m) = \rho_m \prod_{i \in \mathcal{E}} (1 - I(i, j, \mathbf{U}_m(0)) b_i^{0_U}). \tag{14}$$

and

$$a_j^{0_U}(m) = \rho_m \prod_{i \in \mathcal{E}} (1 - I(i, j, \mathbf{U}_m(0)) b_i^{0_L}). \tag{15}$$

Note that the notation $a_j^{0_L}(m)$ and $a_j^{0_U}(m)$ given by (14) and (15) is not the same as the perviously defined notation $a_j^{0_L}$ and, respectively, the value $a_j^{0_L}$ is the lower bound of the total offered load to trunk j in layer 0, while $a_j^{0_L}(m)$ is the lower bound of the offered load to trunk j by SD pair m. The difference between the notation $a_j^{0_U}$ and $a_j^{0_U}(m)$ is explained similarly.

For layer k, we assume that the offered load of bursts deflected less than k times is $\tilde{a}_j^{k-1}$. If the layer having direct trunks is not the primary layer (that is, it is layer $k > 0$), then the overflowed traffic from SD pair m in the congested trunk q in layer $k-1$ forms the traffic to the paths in layer k. Then, the bounds of the offered load of the direct trunks in layer k are calculated by the equations

$$a_j^{k_U} = \sum_{m \in \beta, q \in \mathcal{E}, j \in \mathbf{U}_{m,q}(k)} \{ a_q^{k-1_U}(m) b_q^{k-1_U} \times \prod_{i \in \mathcal{E}} (1 - I(i, j, \mathbf{U}_{m,q}(k)) b_i^{k_L}) \}, \tag{16}$$

and

$$a_j^{k_L} = \sum_{m \in \beta, q \in \mathcal{E}, j \in \mathbf{U}_{m,q}(k)} \{ a_q^{k-1_L}(m) b_q^{k-1_L} \times \prod_{i \in \mathcal{E}} (1 - I(i, j, \mathbf{U}_{m,q}(k)) b_i^{k_U}) \}. \tag{17}$$

The equation for the trunk blocking probability is

$$b_j^k = \frac{E(\tilde{a}_j^{k-1} + a_j^k, C_j)(\tilde{a}_j^{k-1} + a_j^k) - E(\tilde{a}_j^{k-1}, C_j)(\tilde{a}_j^{k-1})}{a_j^k}. \tag{18}$$

The left hand side of equation (18) increases when $\tilde{a}_j^{k-1}$ or a_j^k increases (see Appendix for the proof), and hence,

$$\begin{aligned}
&\frac{E(\tilde{a}_j^{k-1_L} + a_j^{k_L}, C_j)(\tilde{a}_j^{k-1_L} + a_j^{k_L}) - E(\tilde{a}_j^{k-1_L}, C_j)(\tilde{a}_j^{k-1_L})}{a_j^{k_L}} \\
&\leq \frac{E(\tilde{a}_j^{k-1_U} + a_j^{k_L}, C_j)(\tilde{a}_j^{k-1_U} + a_j^{k_L}) - E(\tilde{a}_j^{k-1_U}, C_j)(\tilde{a}_j^{k-1_U})}{a_j^{k_L}} \\
&\leq \frac{E(\tilde{a}_j^{k-1_U} + a_j^{k_U}, C_j)(\tilde{a}_j^{k-1_U} + a_j^{k_U}) - E(\tilde{a}_j^{k-1_U}, C_j)(\tilde{a}_j^{k-1_U})}{a_j^{k_U}},
\end{aligned}$$

and

$$\begin{aligned}
&\frac{E(\tilde{a}_j^{k-1_L} + a_j^{k_L}, C_j)(\tilde{a}_j^{k-1_L} + a_j^{k_L}) - E(\tilde{a}_j^{k-1_L}, C_j)(\tilde{a}_j^{k-1_L})}{a_j^{k_L}} \\
&\leq \frac{E(\tilde{a}_j^{k-1_L} + a_j^{k_U}, C_j)(\tilde{a}_j^{k-1_L} + a_j^{k_U}) - E(\tilde{a}_j^{k-1_L}, C_j)(\tilde{a}_j^{k-1_L})}{a_j^{k_U}} \\
&\leq \frac{E(\tilde{a}_j^{k-1_U} + a_j^{k_U}, C_j)(\tilde{a}_j^{k-1_U} + a_j^{k_U}) - E(\tilde{a}_j^{k-1_U}, C_j)(\tilde{a}_j^{k-1_U})}{a_j^{k_U}}.
\end{aligned}$$

Thus, the upper and lower bounds of the trunk blocking probabilities of direct trunks that not in the primary layer are calculated by the formulae:

$$b_j^{k_U} = \frac{E(\tilde{a}_j^{k-1_U} + a_j^{k_U}, C_j)(\tilde{a}_j^{k-1_U} + a_j^{k_U}) - E(\tilde{a}_j^{k-1_U}, C_j)(\tilde{a}_j^{k-1_U})}{a_j^{k_U}}, \tag{19}$$

and

$$b_j^{k_L} = \frac{E(\tilde{a}_j^{k-1_L} + a_j^{k_L}, C_j)(\tilde{a}_j^{k-1_L} + a_j^{k_L}) - E(\tilde{a}_j^{k-1_L}, C_j)(\tilde{a}_j^{k-1_L})}{a_j^{k_L}}. \tag{20}$$

The upper and lower bounds of the offered load to each direct trunk in SD pair m in layer k are determined by the formulae:

$$a_j^{k_L}(m) = \sum_{q \in \mathcal{E}, j \in \mathbf{U}_{m,q}(k)} a_q^{k-1_L}(m) b_q^{k-1_L} \times \prod_{i \in \mathcal{E}} (1 - I(i, j, \mathbf{U}_{m,q}(k)) b_i^{k_U}), \tag{21}$$

and

$$a_j^{k_U}(m) = \sum_{q \in \mathcal{E}, j \in \mathbf{U}_{m,q}(k)} a_q^{k-1_U}(m) b_q^{k-1_U} \times \prod_{i \in \mathcal{E}} (1 - I(i, j, \mathbf{U}_{m,q}(k)) b_i^{k_L}). \tag{22}$$

For the kth layer, we first determine the bounds of the offered load in trunk j_1, which is on the top of the tree. The bounds of the offered load depend only on the overflowed traffic in the $k-1$ layer. From equations (16-17), we obtain $a_{j_1}^{k_U} = a_{j_1}^{k^*_{max}} = a_{j_1}^{k^*_{min}} = a_{j_1}^{k_L}$ provided that $a_q^{k-1_U}(m) = a_q^{k-1_L}(m)$ is satisfied for all overflowed traffic into top trunk j; in the opposite case, if $a_q^{k-1_U}(m) = a_q^{k-1_L}(m)$ is not satisfied at least for one of available overflowed traffic, then we obtain $a_{j_1}^{k_U} > a_{j_1}^{k^*_{max}} \geq a_{j_1}^{k^*_{min}} > a_{j_1}^{k_L}$. (The equation $a_q^{k-1_U}(m) = a_q^{k-1_L}(m)$ is satisfied if for the upper layer we obtain exact value of the offered load). Substituting these bounds of the offered load into equations (19-20), we obtain the bounds for the trunk blocking probability in the top trunk. If $a_q^{k-1_U}(m) = a_q^{k-1_L}(m)$ for all overflowed traffic into the top trunk, then $b_{j_1}^{k_U} = b_{j_1}^{k^*_{max}} = b_{j_1}^{k^*_{min}} = b_{j_1}^{k_L}$, and otherwise, we obtain $b_{j_1}^{k_U} > b_{j_1}^{k^*_{max}} \geq b_{j_1}^{k^*_{min}} > b_{j_1}^{k_L}$. In the next step, we calculate the bounds for the offered load and blocking probability for each of the trunks that are the second from the top. The calculation for each one these trunks is identical to the others and depends only on the trunk blocking probability in the top trunk j_1 and overflowed traffic from the

$k-1$ layer. Therefore, without loss of generality, let j_2 denote any of these trunks. We obtain $a_{j_2}^{k_U} = a_{j_2}^{k^*_{max}} = a_{j_2}^{k^*_{min}} = a_{j_2}^{k_L}$ and $b_{j_2}^{k_U} = b_{j_2}^{k^*max} = b_{j_2}^{k^*_{min}} = b_{j_2}^{k_L}$ provided that $a_q^{k-1_U}(m) = a_q^{k-1_L}(m)$ is satisfied for all overflowed traffic into trunk j_1 and trunk j_2. Otherwise, we have $a_{j_2}^{k_U} > a_{j_2}^{k^*_{max}} \geq a_{j_2}^{k^*_{min}} > a_{j_2}^{k_L}$ and $b_{j_2}^{k_U} > b_{j_2}^{k^*_{max}} \geq b_{j_2}^{k^*_{min}} > b_{j_2}^{k_L}$, respectively. Repeating these steps recurrently to the other direct trunks in the kth layer from top to bottom of the tree, we obtain all bounds for the offered load and trunk blocking probabilities.

B. Bounds of trunk blocking probabilities for LBTs

In Proposition 1 below and in all other statements following then, $b_j^{k_1}$ denotes the initial (setup) value of the blocking probability for the kth layer on trunk j. *Proposition 1*: Assume that $b_j^{k_1} = 0$ $(k = 0,1,\ldots,D)$, $j \in \mathcal{L}$ with a nonzero offered load and trunk j is a LBT. Then, for any positive integer z, we have the following inequalities for lower and upper bounds of $a_j^{k^*}$ and $b_j^{k^*}$:

$$a_j^{k_{2z-1}} > a_j^{k_{2z+1}} > a_j^{k^*_{max}},$$

$$a_j^{k_{2z-2}} < a_j^{k_{2z}} < a_j^{k^*_{min}},$$

$$b_j^{k_{2z-2}} > b_j^{k_{2z}} > b_j^{k^*_{max}},$$

and

$$b_j^{k_{2z-1}} < b_j^{k_{2z+1}} < b_j^{k^*_{min}}.$$

Proof: The computation starts from layer 0 of the primary path bursts. First we calculate lower and upper bounds of the offered load and trunk blocking probabilities for the direct trunks. Then, setting $b_j^{0_1} = 0$, we provide calculations for other trunks, where j belongs to the LBT set, and 0_1 denotes the first iteration in the primary path.

For the offered load on trunk j, we have:

$$a_j^{0_1} = \sum_{m \in \beta, j \in \mathbf{U}_m(0)} \rho_m \prod_{i \in \mathcal{L}} (1 - I(i,j,\mathbf{U}_m(0))(1 - H(0,i))b_i^{0_1}) \times (1 - I(i,j,\mathbf{U}_m(0))H(0,i)b_i^{0_L}), \tag{23}$$

where

$$I(i,j,\mathbf{U}_m(0)) = \begin{cases} 1, & \text{if } i,j \in \mathcal{L} \text{ and trunk } i \text{ strictly preceds} \\ & \text{(not necessarily immediately) trunk } j \\ & \text{along primary route of SD pair } m \\ 0, & \text{otherwise,} \end{cases}$$

and

$$H(k,i) = \begin{cases} 1, & \text{if trunk } i \text{ is a LBT in layer k} \\ 0, & \text{if trunk } i \text{ is a direct trunk in layer k.} \end{cases}$$

Hence, the blocking probability $b_j^{0_2}$ is obtained by the Erlang-B formula,

$$b_j^{0_2} = E\left(a_j^{0_1}, C_j\right). \tag{24}$$

According to assumption, $b_j^{0_1} = 0$ and $a_j^{0_1} > 0$ are true for all trunks j belonging to the LBT set in layer 0. Hence, comparing $b_j^{0_1}$ with $b_j^{0_2}$ yields $b_j^{0_2} > b_j^{0_1}$ for all j belonging to the LBT set in layer 0.

By the successive substitution method, we are to replace $b_j^{0_1}$ in equation (23) by $b_j^{0_2}$, and then the load is updated by the formula:

$$a_j^{0_2} = \sum_{m \in \beta, j \in \mathbf{U}_m(0)} \rho_m \prod_{i \in \mathcal{L}} (1 - I(i,j,\mathbf{U}_m(0))(1 - H(0,i))b_i^{0_2}) \times (1 - I(i,j,\mathbf{U}_m(0))H(0,i)b_i^{0_U}), \tag{25}$$

Taking into account $b_j^{0_2} > b_j^{0_1}$ for all j belonging to the LBT set in layer 0 and $b_i^{0_U} = b_i^{0_L}$ when $H(0,i) = 1$, by comparing relationships (23) and (25), we arrive at the inequality $a_j^{0_2}(m) < a_j^{0_1}(m)$ for all j belonging to the LBT set in layer 0.

Repeating the steps above, for the value $b_j^{0_3}$, we obtain

$$b_j^{0_3} = E\left(a_j^{0_2}, C_j\right). \tag{26}$$

Since $E(x,C)$ is an increasing function in x, we have $0 = b_j^{0_1} < b_j^{0_3} < b_j^{0_2}$.

In the third iteration, we have the formula

$$a_j^{0_3} = \sum_{m \in \beta, j \in \mathbf{U}_m(0)} \rho_m \prod_{i \in \mathcal{L}} (1 - I(i,j,\mathbf{U}_m(0))(1 - H(0,i))b_i^{0_3}) \times (1 - I(i,j,\mathbf{U}_m(0))H(0,i)b_i^{0_L}), \tag{27}$$

and then we arrive at

$$b_j^{0_4} = E\left(a_j^{0_3}, C_j\right). \tag{28}$$

Since $0 = b_j^{0_1} < b_j^{0_3} < b_j^{0_2}$, we have the property $a_j^{0_2} < a_j^{0_3} < a_j^{0_1}$ and $0 = b_j^{0_1} < b_j^{0_3} < b_j^{0_4} < b_j^{0_2}$ for all $j \in \mathcal{L}$.

Since $a_j^{0_4} = \sum_{m \in \beta, j \in \mathbf{U}_m(0)} \rho_m \prod_{i \in \mathcal{L}} (1 - I(i,j,\mathbf{U}_m(0))b_i^{0_4})$, we obtain $a_j^{0_2} < a_j^{0_4} < a_j^{0_3} < a_j^{0_1}$ for all j belonging to the LBT set in layer 0.

In a similar way, we define the recurrent relations for $s > 1$:

$$b_j^{0_s} = E\left(a_j^{0_{s-1}}, C_j\right), \tag{29}$$

$$a_j^{0_s} = \sum_{m \in \beta, j \in \mathbf{U}_m(0)} \rho_m \prod_{i \in \mathcal{L}} (1 - I(i,j,\mathbf{U}_m(0))(1 - H(0,i))b_i^{0_s}) \times (1 - I(i,j,\mathbf{U}_m(0))(1 - H(0,i))b_i^{0_L}). \tag{30}$$

Notice that $b_i^{0_L} = b_i^{0_U}$ when $H(0,i) = 0$.

Repeating the same procedure, we obtain all the necessary relations for $a_j^{0_s}$ and $b_j^{0_s}$ for given integer values s. We also have the inequalities

$$a_j^{0_1} > a_j^{0_3} > a_j^{0_5} > \cdots > \lim_{z \to \infty} a_j^{0_{2z-1}} = a_j^{0^*_{max}}, \tag{31}$$

$$a_j^{0_2} < a_j^{0_4} < a_j^{0_6} < \cdots < \lim_{z \to \infty} a_j^{0_{2z}} = a_j^{0^*_{min}}, \tag{32}$$

Let S be the number of iterations. We will write $S = 2z$ in the case of the even number of iterations and $S = 2z+1$ in the case when the number of iterations is odd.

In the case of $S = 2z$, the upper bound of the set of values $\{a_j^{0_s}, s = 1,2,\ldots,2z\}$ is $a_j^{0_U} = a_j^{0_{2z-1}}$ and the lower bound is $a_j^{0_L} = a_j^{0_{2z}}$. In the case of $S = 2z+1$, the upper bound of the set of these values is $a_j^{0_U} = a_j^{0_{2z+1}}$ and the lower bound is $a_j^{0_L} = a_j^{0_{2z}}$.

For the set of the values $\{b_j^{0_s}, s = 1,2,\ldots,S\}$, the bounds are defined similarly.

$$b_j^{0_2} > b_j^{0_4} > b_j^{0_6} > \cdots > \lim_{z\to\infty} b_j^{0_{2z}} = b_j^{0^*_{max}}, \qquad (33)$$

$$b_j^{0_1} < b_j^{0_3} < b_j^{0_5} < \cdots < \lim_{z\to\infty} b_j^{0_{2z-1}} = b_j^{0^*_{min}}. \qquad (34)$$

In the case of $S = 2z$, the upper and the lower bounds of the set of values $\{b_j^{0_s}, s = 1,2,\ldots,2z\}$ are $b_j^{0_U} = b_j^{0_{2z}}$ and $b_j^{0_L} = b_j^{0_{2z-1}}$, respectively. In the case of $S = 2z+1$, the upper and the lower bounds of the set of these values are $b_j^{0_U} = b_j^{0_{2z}}$ and $b_j^{0_L} = b_j^{0_{2z+1}}$, respectively.

Inequalities (31-34) follow by induction.

Indeed, we earlier proved (31-34) for $z = 1$. Hence, assuming that in the case $z = i$ the inequalities

$$a_j^{0_{2i-1}} > a_j^{0_{2i+1}} > a_j^{0_{2i+2}} > a_j^{0_{2i}}, \qquad (35)$$

are satisfied, we are to prove that the inequalities

$$a_j^{0_{2i+1}} > a_j^{0_{2i+3}} > a_j^{0_{2i+4}} > a_j^{0_{2i+2}} \qquad (36)$$

are true as well (case $z = i+1$). At the next step, on the basis of the inequalities (35), we prove

$$b_j^{0_{2i}} > b_j^{0_{2i+2}} > b_j^{0_{2i+3}} > b_j^{0_{2i+1}}. \qquad (37)$$

Indeed, since the function $E(x, C_j)$ is increasing in x, and hence, (37) follows from (35) by direct substitution of the values into (29).

Now on the basis of (37) we prove (36). The function $F(\mathbf{x}) = \prod_{n=1}^{i-1}(1 - x_n)$ is a decreasing function in vector $\mathbf{x} = (x_1, x_2, \ldots, x_n)$. That is, for any two vectors $\mathbf{x}^{(1)} = (x_1^{(1)}, x_2^{(1)}, \ldots, x_n^{(1)})$ and $\mathbf{x}^{(2)} = (x_1^{(2)}, x_2^{(2)}, \ldots, x_n^{(2)})$ satisfying the componentwise inequalities $x_i^{(1)} \le x_i^{(2)}$, $i = 1,2,\ldots,n$, we have $F(\mathbf{x}^{(1)}) \ge F(\mathbf{x}^{(2)})$. The strong inequality $F(\mathbf{x}^{(1)}) > F(\mathbf{x}^{(2)})$ holds if in addition $x_i^{(1)} < x_i^{(2)}$ is satisfied at least for one of indices i.

Hence, substituting the values of (37) into (30) we arrive at the inequality

$$a_j^{0_{2i+1}} > a_j^{0_{2i+3}} > a_j^{0_{2i+2}} > a_j^{0_{2i}}. \qquad (38)$$

From (38), it is easy to obtain the desired inequality (36). To this end, we first substitute (38) into (29). This yields

$$b_j^{0_{2i+2}} > b_j^{0_{2i+4}} > b_j^{0_{2i+3}} > b_j^{0_{2i+1}}. \qquad (39)$$

Finally, substituting (39) into (30) once again, we arrive at (36).

For layer 0, the lower and upper bounds of the offered load to each trunk of SD pair m are also calculated by the equations (14-15).

For layer 1, the bounds of overflowed traffic from SD pair m causing by the congestion in trunk q in layer 0, is calculated by the equation

$$a_j^1 = \sum_{m\in\beta, q\in\mathcal{E}, j\in\mathbf{U}_{m,q}(1)} a_q^0(m) b_q^0 \prod_{i\in\mathcal{E}} (1 - I(i,j,\mathbf{U}_{m,q}(1)) \times (1 - H(1,i)) b_i^1)(1 - I(i,j,\mathbf{U}_{m,q}(1)) H(1,i) b_i^1). \qquad (40)$$

On the basis of the lower and upper values for each $a_i^0(m)$ and b_i for direct trunks i, we have the following transformed formulae of (40)

$$a_j^{1_U} = \sum_{m\in\beta, q\in\mathcal{E}, j\in\mathbf{U}_{m,q}(1)} a_q^{0_U}(m) b_q^{0_U} \prod_{i\in\mathcal{E}} (1 - I(i,j,\mathbf{U}_{m,q}(1)) \times (1 - H(1,i)) b_i^1)(1 - I(i,j,\mathbf{U}_{m,q}(1)) H(1,i) b_i^L), \qquad (41)$$

and

$$a_j^{1_L} = \sum_{m\in\beta, q\in\mathcal{E}, j\in\mathbf{U}_{m,q}(1)} a_q^{0_L}(m) b_q^{0_L} \prod_{i\in\mathcal{E}} (1 - I(i,j,\mathbf{U}_{m,q}(1)) \times (1 - H(1,i)) b_i^1)(1 - I(i,j,\mathbf{U}_{m,q}(1)) H(1,i) b_i^U). \qquad (42)$$

Since a_j^1 increases when the term $a_q^0(m) b_q^0$ increases, then substituting the same value of b_j^1 into formulae (41) and (42), we obtain the inequality $a_j^{1_U} > a_j^{1_L}$.

Setting $b_j^{1_1} = 0$, for upper and lower bounds of a_j^1 and b_j^1, we have the following relations

$$a_j^{1_{2z-1}} = \sum_{m\in\beta, q\in\mathcal{E}, j\in\mathbf{U}_{m,q}(1)} a_q^{0_U}(m) b_q^{0_U} \prod_{i\in\mathcal{E}} \{1 - I(i,j,\mathbf{U}_{m,q}(1)) \times (1 - H(1,i)) b_i^{1_{2z-1}}\}\{1 - I(i,j,\mathbf{U}_{m,q}(1)) H(1,i) b_i^{1_L}\}, \qquad (43)$$

$$a_j^{1_{2z}} = \sum_{m\in\beta, q\in\mathcal{E}, j\in\mathbf{U}_{m,q}(1)} a_q^{0_L}(m) b_q^{0_L} \prod_{i\in\mathcal{E}} \{1 - I(i,j,\mathbf{U}_{m,q}(1)) \times (1 - H(1,i)) b_i^{1_{2z}}\}\{1 - I(i,j,\mathbf{U}_{m,q}(1)) H(1,i) b_i^{1_U}\}, \qquad (44)$$

and

$$b_j^{1_{2z}} = \frac{E(a_j^{0_U} + a_j^{1_{2z-1}}, C_j)(a_j^{0_U} + a_j^{1_{2z-1}}) - E(a_j^{0_U}, C_j)(a_j^{0_U})}{a_j^{1_{2z-1}}},$$

$$b_j^{1_{2z+1}} = \frac{E(a_j^{0_L} + a_j^{1_{2z}}, C_j)(a_j^{0_L} + a_j^{1_{2z}}) - E(a_j^{0_L}, C_j)(a_j^{0_L})}{a_j^{1_{2z}}}.$$

Then we have the inequalities

$$a_j^{1_1} > a_j^{1_3} > a_j^{1_5} > \cdots > \lim_{z\to\infty} a_j^{1_{2z-1}} \ge a_j^{1^*_{max}}, \qquad (45)$$

$$a_j^{1_2} < a_j^{1_4} < a_j^{1_6} < \cdots < \lim_{z\to\infty} a_j^{1_{2z}} \le a_j^{1^*_{min}}, \qquad (46)$$

$$b_j^{1_2} > b_j^{1_4} > b_j^{1_6} > \cdots > \lim_{z\to\infty} b_j^{1_{2z}} \ge b_j^{1^*_{max}}, \qquad (47)$$

$$b_j^{1_1} < b_j^{1_3} < b_j^{1_5} < \cdots < \lim_{z\to\infty} b_j^{1_{2z-1}} \le b_j^{1^*_{min}}. \qquad (48)$$

In the case $S = 2z$, the upper and lower bounds of a_j^1 are $a_j^{1_U} = a_j^{1_{2z-1}}$ and $a_j^{1_L} = a_j^{1_{2z}}$, respectively, and the upper and lower bounds of a_j^1 are $a_j^{1_U} = a_j^{1_{2z-1}}$ and $a_j^{1_L} = a_j^{1_{2z}}$, respectively. In the case $S = 2z+1$, the upper bound of a_j^1 is changed to $a_j^{1_U} = a_j^{1_{2z+1}}$, and the lower bound of b_j^1 is changed to $b_j^{1_L} = b_j^{1_{2z+1}}$.

The proof of equations (45-48) is similar to that of equations (31-34).

The lower and upper bounds of the offered load to each trunk of SD pair m in layer 1 are determined by the equations

$$a_j^{1_L}(m) = \sum_{q\in\mathcal{E},\, j\in\mathbf{U}_{m,q}(1)} a_q^{0_L}(m) b_q^{0_L} \prod_{i\in\mathcal{E}} (1 - I(i,j,\mathbf{U}_{m,q}(1)) b_i^{1_U}), \tag{49}$$

and

$$a_j^{1_U}(m) = \sum_{q\in\mathcal{E},\, j\in\mathbf{U}_{m,q}(1)} a_q^{0_U}(m) b_q^{0_U} \prod_{i\in\mathcal{E}} (1 - I(i,j,\mathbf{U}_{m,q}(1)) b_i^{1_L}). \tag{50}$$

The upper and lower bounds of $\tilde{a}_j^{k-1}$ are $\tilde{a}_j^{(k-1)_U} = \sum_{i=0}^{k-1} a_j^{i_U}$ and $\tilde{a}_j^{(k-1)_L} = \sum_{i=0}^{k-1} a_j^{i_L}$.

The values of $a_j^{k_s}$ and $b_j^{k_s}$ are calculated by the formulae:

$$\begin{aligned} a_j^{k_{2z-1}} = & \sum_{m\in\beta,q\in\mathcal{E},j\in\mathbf{U}_{m,q}(k)} a_q^{(k-1)_U}(m) b_q^{(k-1)_U} \\ & \times \prod_{i\in\mathcal{E}} \{1 - I(i,j,\mathbf{U}_{m,q}(k))(1-H(k,i)) b_i^{k_{2z-1}}\} \\ & \times \{1 - I(i,j,\mathbf{U}_{m,q}(k)) H(k,i) b_i^{k_L}\}, \end{aligned} \tag{51}$$

$$\begin{aligned} a_j^{k_{2z}} = & \sum_{m\in\beta,q\in\mathcal{E},j\in\mathbf{U}_{m,q}(k)} a_q^{(k-1)_L}(m) b_q^{(k-1)_L} \\ & \times \prod_{i\in\mathcal{E}} \{1 - I(i,j,\mathbf{U}_{m,q}(k))(1-H(k,i)) b_i^{k_{2z}}\} \\ & \times \{1 - I(i,j,\mathbf{U}_{m,q}(k)) H(k,i) b_i^{k_U}\}, \end{aligned} \tag{52}$$

$$\begin{aligned} b_j^{k_{2z}} = & \frac{E(\tilde{a}_j^{(k-1)_U} + a_j^{k_{2z-1}}, C_j)(\tilde{a}_j^{(k-1)_U} + a_j^{k_{2z-1}})}{a_j^{k_{2z-1}}} \\ & - \frac{E(\tilde{a}_j^{(k-1)_U}, C_j)(\tilde{a}_j^{(k-1)_U})}{a_j^{k_{2z-1}}}, \end{aligned} \tag{53}$$

$$\begin{aligned} b_j^{k_{2z+1}} = & \frac{E(\tilde{a}_j^{(k-1)_L} + a_j^{k_{2z}}, C_j)(\tilde{a}_j^{(k-1)_L} + a_j^{k_{2z}})}{a_j^{k_{2z}}} \\ & - \frac{E(\tilde{a}_j^{(k-1)_L}, C_j)(\tilde{a}_j^{(k-1)_L})}{a_j^{k_{2z}}}. \end{aligned} \tag{54}$$

Then we obtain the inequalities

$$a_j^{k_1} > a_j^{k_3} > a_j^{k_5} > \cdots > \lim_{z\to\infty} a_j^{k_{2z-1}} \geq a_j^{k^*_{max}}, \tag{55}$$

$$a_j^{k_2} < a_j^{k_4} < a_j^{k_6} < \cdots < \lim_{z\to\infty} a_j^{k_{2z}} \leq a_j^{k^*_{min}}, \tag{56}$$

$$b_j^{k_2} > b_j^{k_4} > b_j^{k_6} > \cdots > \lim_{z\to\infty} b_j^{k_{2z}} \geq b_j^{k^*_{max}}, \tag{57}$$

$$b_j^{k_1} < b_j^{k_3} < b_j^{k_5} < \cdots < \lim_{z\to\infty} b_j^{k_{2z-1}} \leq b_j^{k^*_{min}}. \tag{58}$$

In the case $S = 2z$, the upper and lower bounds of a_j^k are $a_j^{k_U} = a_j^{k_{2z-1}}$ and, respectively, $a_j^{k_L} = a_j^{k_{2z}}$, and the upper and lower bounds of b_j^k are $b_j^{k_U} = b_j^{k_{2z}}$ and $b_j^{k_L} = b_j^{k_{2z-1}}$, respectively. In the case $S = 2z+1$, the upper bound of a_j^k is changed to $a_j^{k_U} = a_j^{k_{2z+1}}$, and the lower bound of b_j^k is changed to $b_j^{k_L} = b_j^{k_{2z+1}}$.

The upper and lower bounds of the offered load to each trunk of SD pair m in layer k are determined by the formulae:

$$\begin{aligned} a_j^{k_L}(m) = & \sum_{q\in\mathcal{E},\, j\in\mathbf{U}_{m,q}(k)} \{a_q^{k-1_L}(m) b_q^{k-1_L} \\ & \prod_{i\in\mathcal{E}} (1 - I(i,j,\mathbf{U}_{m,q}(k)) b_i^{k_U})\}, \end{aligned} \tag{59}$$

and

$$\begin{aligned} a_j^{k_U}(m) = & \sum_{q\in\mathcal{E},\, j\in\mathbf{U}_{m,q}(k)} \{a_q^{k-1_U}(m) b_q^{k-1_U} \\ & \prod_{i\in\mathcal{E}} (1 - I(i,j,\mathbf{U}_{m,q}(k)) b_i^{k_L})\}. \end{aligned} \tag{60}$$

Corollary 1: Consider two OPCA runs. Let the number of iterations in layer i in the first run be I_i, and let the number of iterations in layer i in the second run be I'_i. Let S_j^i and S'^i_j be the distance between the bounds for the blocking probability for trunk j in layer i, in the first and second OPCA runs, respectively. If $I'_i > I_i$ for one layer and the number of iterations in other layers are the same and there are LBTs in the layer, then, $S_j^k > S'^k_j$ for the trunks with the following situations:

(1) if trunk j is a loop trunk, then for all $k \geq i$, $S_j^k > S'^k_j$;

(2) if in a layer $k' > i$, a trunk j that receives traffic, passes through or overflows from a trunk j_1 in the upper layer satisfies the inequality $S_{j_1}^{k'-1} > S'^{k'-1}_{j_1}$, then for all $k' \leq k \leq D$, $S_j^k > S'^k_j$.

Proof: According to the construction, for layers $k < i$, the bounds of the trunk blocking probability for $j \in \mathcal{E}$ are the same in two OPCA runs.

Since there are LBTs, then for layer i by inequalities (57-58), for LBTs we obtain inequalities $S_j^{i_z} < S_j^{i_{z-1}} < \cdots < S_j^{i_1}$, in which $S_j^{i_z} = | b_j^{i_{z+1}} - b_j^{i_z} |$. Thus, if $I'_i > I_i$, then $S_j^i > S'^i_j$.

As well, if $I'_i > I_i$, then by relations (21-22) and inequalities (55-58), we obtain $a'^{i_U}_j(m) < a_j^{i_U}(m)$ and $a'^{i_L}_j(m) > a_j^{i_L}(m)$, $b'^{i_U}_j < b_j^{i_U}$ and $b'^{i_L}_j > b_j^{i_L}$ for $j \in \mathcal{E}$ and $m \in \beta$.

For layer $i+1$, let us first consider a trunk j for which $S_j^i > S'^i_j$ in layer i is satisfied. Then we have $a'^{i_U}_j < a_j^{i_U}$ and $a'^{i_L}_j > a_j^{i_L}$. Substituting these inequalities into equations (19-22), we obtain the inequalities $b'^{i+1_U}_j < b_j^{i+1_U}$, $b'^{i+1_L}_j > b_j^{i+1_L}$, $a'^{i+1_U}_j(m) < a_j^{i+1_U}(m)$ and $a'^{i+1_L}_j(m) > a_j^{i+1_L}(m)$. Thus, if trunk j is a loop trunk in layer i, then we have $S_j^{i+1} > S'^{i+1}_j$.

Consider now a direct trunk j receiving traffic that overflowed from trunks j_1 for which $S_{j_1}^i > S'^i_{j_1}$ in layer i or passed through a trunk j_2 in layer $i+1$ for which $S_{j_2}^{i+1} > S'^{i+1}_{j_2}$. Substituting the inequalities $a'^{i_U}_{j_1}(m) < a_{j_1}^{i_U}(m)$ and $a'^{i_L}_{j_1}(m) > a_{j_1}^{i_L}(m)$, $b'^{i_U}_{j_1} < b_{j_1}^{i_U}$ $b'^{i_L}_{j_1} > b_{j_1}^{i_L}$ or $b'^{i+1_U}_{j_2} < b_{j_2}^{i+1_U}$ and $b'^{i+1_L}_{j_2} > b_{j_2}^{i+1_L}$ into equations (17) and (16), we obtain $a'^{i+1_U}_j < a_j^{i+1_U}$ and $a'^{i+1_L}_j > a_j^{i+1_L}$. Then, substituting the inequalities into equations (19-22), we obtain the inequalities $b'^{i+1_U}_j < b_j^{i+1_U}$, $b'^{i+1_L}_j > b_j^{i+1_L}$, $a'^{i+1_U}_j(m) < a_j^{i+1_U}(m)$ and $a'^{i+1_L}_j(m) > a_j^{i+1_L}(m)$. Thus, the inequality $S_j^{i+1} > S'^{i+1}_j$ is satisfied if trunk j is a direct trunk receiving traffic that overflowed from LBTs in layer i.

Let us consider now LBTs receiving traffic that overflowed from LBTs in layer i or passing through a direct trunk for which the inequality $S_j^{i+1} > S'^{i+1}_j$ is satisfied in layer $i+1$. We start from the setup value $b_j^{i+1_1} = 0$. According to relation (51) for layer $i+1$, we obtain the inequality $a_j^{i+1_1} > a_j^{\prime i+1_1}$. Substituting this inequality into (53), we in turn obtain $b_j^{i+1_2} > b_j^{\prime i+1_2}$. Repeating the procedure, we again substitute the inequality obtained for (52) and now obtain $a_j^{i+1_1} > a_j^{\prime i+1_1} > a_j^{\prime i+1_2} > a_j^{i+1_2}$. Further substitution of the inequality obtained for (54) yields $b_j^{i+1_2} > b_j^{\prime i+1_2} > b_j^{\prime i+1_1} > b_j^{i+1_1}$. After a number of repetitions of these steps for any integer z, we finally arrive at the following inequalities,

$$a_j^{i+1_{2z-1}} > a_j^{\prime i+1_{2z-1}} > a_j^{\prime i+1_{2z}} > a_j^{i+1_{2z}},$$

and

$$b_j^{i+1_{2z}} > b_j^{\prime i+1_{2z}} > b_j^{\prime i+1_{2z-1}} > b_j^{i+1_{2z-1}}.$$

By the same method in layer i, we obtain $S_j^{i+1} > S_j^{\prime i+1}$. As well, we obtain $a_j^{i+1_U}(m) > a_j^{\prime i+1_U}(m) > a_j^{\prime i+1_L}(m) > a_j^{i+1_L}(m)$ and $b_j^{i+1_U} > b_j^{\prime i+1_U} > b_j^{\prime i+1_L} > b_j^{i+1_L}$.

Repeating the same steps as in layer $i+1$, we obtain the same solution for layer $k > i+1$.

C. Bounds for network blocking probability of OPCA

Proposition 2: Let $B_U(m)$ and $B_L(m)$ denote the upper and, respectively, the lower bounds of the network blocking probability for SD pair m. We have:

$$B^U(m) = 1 - \frac{\rho_m \prod_{i \in \mathbf{U}_m(0)} (1 - b_i^{0_U})}{\rho_m} - \frac{\sum_{q \in \mathcal{E}} \sum_{k=1}^{T}(m) \rho^L_{\mathbf{U}_{m,q}(k)} \prod_{p \in \mathbf{U}_{m,q}(k)} (1 - b_p^{k_U})}{\rho_m}, \tag{61}$$

where

$$\rho^L_{\mathbf{U}_{m,q}(k)} = \begin{cases} a_q^{(k-1)_L}(m) b_q^{(k-1)_L}, & \text{if path } \mathbf{U}_{m,q}(h) \text{ exists,} \\ 0, & \text{otherwise,} \end{cases}$$

and

$$B^L(m) = 1 - \frac{\rho_m \prod_{i \in \mathbf{U}_m(0)} (1 - b_i^{0_L})}{\rho_m} - \frac{\sum_{q \in \mathcal{E}} \sum_{h=1}^{T}(m) \rho^U_{\mathbf{U}_{m,q}(k)} \prod_{p \in \mathbf{U}_{m,q}(k)} (1 - b_p^{k_L})}{\rho_m}, \tag{62}$$

where

$$\rho^U_{\mathbf{U}_{m,q}(k)} = \begin{cases} a_q^{(k-1)_U}(m) b_q^{(k-1)_U}, & \text{if path } \mathbf{U}_{m,q}(k) \text{ exists,} \\ 0, & \text{otherwise.} \end{cases}$$

Proof: In order to calculate the blocking probability $B(m)$ for SD pair m, it is required first to calculate the received load from every path by the destination node. Then, we calculate the probability that a message will be served and the blocking probability

$$B(m) = 1 - \frac{\rho_m \prod_{i \in \mathbf{U}_m(0)} (1 - b_i^0)}{\rho_m} - \frac{\sum_{q \in \mathcal{E}} \sum_{k=1}^{T}(m) \rho_{\mathbf{U}_{m,q}(h)} \prod_{p \in \mathbf{U}_{m,q}(k)} (1 - b_p^k)}{\rho_m}, \tag{63}$$

where $\rho_{\mathbf{U}_{m,q}(k)}$ is the offered load to the path $\mathbf{U}_{m,q}(k)$, and it is calculated by the formula

$$\rho_{\mathbf{U}_{m,q}(k)} = \begin{cases} a_q^{k-1}(m) b_q^{k-1}, & \text{if path } \mathbf{U}_{m,q}(k) \text{ exists} \\ 0, & \text{otherwise.} \end{cases}$$

After calculation the lower and upper bounds for a_j^k and b_j^k ($k = 0, 1, \ldots, T(m)$), we derive equations (61-63). For more iterations, the sequence $B^U(m)$ is not increasing. If there is at least one layer having LBTs, then for more iterations the sequence $b_i^{h_U}$ is decreasing and $\rho^L_{\mathbf{U}_{m,q}(k)}$ is a decreasing sequence in $m \in \beta$ and $k = 0, 1, \ldots, T(m)$ as well. Hence, $B^U(m)$ is a decreasing sequence. Unlike $B^U(m)$, $B^L(m)$ is not an increasing sequence. When there is at least one layer having LBTs, the sequence $B^L(m)$ is decreasing for more iterations of the OPCA algorithm.

Substituting $a_j^{k_U}(m) \geq a_j^{k^*_{max}}(m) \geq a_j^{k^*_{min}}(m) \geq a_j^{k_L}(m)$ and $b_j^{k_U} \geq b_j^{k^*_{max}} \geq b_j^{k^*_{min}} \geq b_j^{k_L}$ into equations (63-63), we arrive at the inequality $B_m^U \geq B^*_{max}(m) \geq B^*_{min}(m) \geq B_m^L$, where $B^*_{max}(m)$ and $B^*_{min}(m)$ denote the maximum and minimum value in the set $\{B^*(m)\}$ of the fixed-point solutions for $B(m)$.

Proposition 3: Let $\{B^*\}$ be the set of the network blocking probabilities obtained by the fixed-point solutions for trunk blocking probabilities in trunks $j \in \mathcal{E}$. Let B^*_{min} and B^*_{max} denote the minimum and maximum values amongst the set of values $\{B^*\}$. Let B_U and B_L denote the upper and lower bounds of the network blocking probabilities. We have

$$B_U \geq B^*_{max} \geq B^*_{min} \geq B_L.$$

B_U and B_L are calculated by the equations

$$B_U = \frac{\sum_{m \in \beta} B^U(m)}{\#SD}, \tag{64}$$

$$B_L = \frac{\sum_{m \in \beta} B^L(m)}{\#SD}, \tag{65}$$

where $\#SD$ is the number of SD pairs in the network.

Proof: Substituting $B_m^U \geq B^*_{max}(m) \geq B^*_{min}(m) \geq B_m^L$ into equations (64-65), we obtain the bounds for the network blocking probability

$$B_U \geq B^*_{max} \geq B^*_{min} \geq B_L$$

Corollary 2: Consider an OPCA run. Let M be the distance between the bounds of the network blocking probability. Increasing the number of iterations in one or more layers among which at least one of those layers has LBTs and obtaining the new distance between the bounds of the network blocking probability M', then $M' < M$.

Proof: Corollary 1 shows that when the number of iterations increases in one layer having trunk loops, then the distance between the upper and lower bounds of the trunk blocking probability for some trunks decreases. If there at least one trunk j, in which the distance between the upper and lower bounds of the trunk blocking probability decreases in layer i, then for SD pair m having a traffic passing through trunk j in layer i we have the inequalities $B^U(m) > B^{\prime U}(m)$ and $B^L(m) < B^{\prime L}(m)$. Then, we arrive at $B_U > B'_U$ and $B_L < B'_L$. Hence, for $M = B_U - B_L$ and $M' = B'_U - B'_L$, we arrive at $M' < M$.

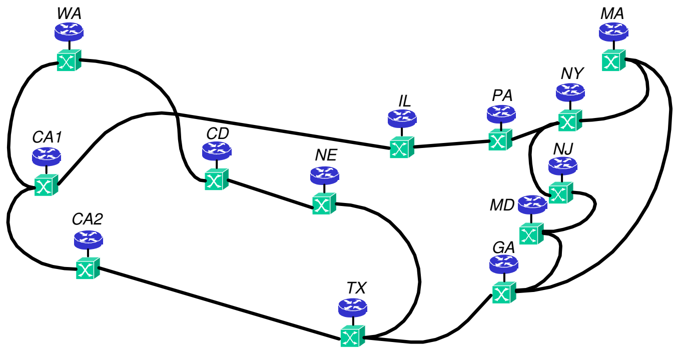


Fig. 2. NSF network topology in which each solid line represents two uni-directional trunks in opposing directions.

If the trunks in all layers are direct trunks, then the solution obtained by the OPCA is not a fixed-point solution, and it is obtained in only a finite number of steps. This number of steps is bounded by $J \times (D+1)$ where J is the total number of trunks, and D is the maximum number of allowable deflections in the network. If there are LBTs at least in one layer, then the bounds of the OPCA fixed-point solutions always become closer when the number of iterations in those layers are increased.

Thus, the OPCA either finds its solution in a finite number of steps, or the bounds of its fixed-point solutions always become closer with increasing number of iterations.

V. Numerical Results

In this section we provide numerical results for OBS/JET without trunk reservation over a 13-node National Science Foundation network (NSFNET) in order to illustrate the behavior of the bounds of OPCA algorithm. In particular, we focus on illustrating how the bounds become closer to each other with the increase of the number of iterations. The numerical results in this section demonstrate that the bounds become closer to each other even after a small number of iterations (per layer). We will illustrate here the behavior of the OPCA blocking probability bounds for the aforementioned network considering a wide range of parameters and design factors, such as: the number of channels per trunk and the maximum allowable number of deflections. We will also compare the running time and accuracy of the EFPA and OPCA algorithms.

In all scenarios considered, the arrival process of calls for each directional SD pair follows a Poisson process. The shortest path is set to be the primary route for each SD pair, and the alternative routes are pre-assigned ordered by their length. For those routes with the same lengths, the order is chosen randomly and remains unchanged afterwards. All the results in this section are obtained using MATLAB software executed on a labtop with Intel® Core™ i7-3520M CPU @ 2.96 GHz 2.96 GHz, 8 GHz RAM and 64-bit operating system.

A. Network topology and SD pairs

We now consider the NSF network with 13 nodes and 30 directional trunks. The topology of the NSF network is shown in Fig. 2. We randomly select a set of 12 SD pairs shown in Table. I.

TABLE I
Ingress and Egress SD pairs

Ingress	WA	CA1	CA1	CA2	TX	GA
Egress	MD	IL	MA	MA	CD	MA
Ingress	MD	IL	MA	MA	CD	MA
Egress	WA	CA1	CA1	CA2	TX	GA

B. Comparison of the computational time of EFPA and OPCA

TABLE II
Comparison of the times used by EFPA and OPCA to calculate the blocking probabilities in NSFNet.

Calculation task	Running time of EFPA in seconds	Running time of OPCA in seconds
Blocking probability of the whole network and $C=50$	0.271	0.197
Blocking probability of the whole network and $C=2000$	64.45	12.91
Blocking probability of the whole network and $C=10000$	3006	397
Blocking probability of the whole network and $C=20000$	13665	1232

Table II provides the running times used to calculate the blocking probability in NSFNet for different C values by EFPA and OPCA. The offered load to each SD pair is $0.5C$. In each iteration, we only consider 4 significant digits of the fixed-point solutions for the trunks with blocking probability larger than 10^{-50}, whereas trunk blocking probabilities lower than 10^{-50} are set equal to 0. From Table II, we observe that EFPA consumes much more time than OPCA for all three different C values and when the C value increases, the computational time of EFPA grows faster than that of OPCA. To gain some insight into the reason why EFPA consumes much more time than OPCA, we count the total number of iterations required by EFPA, and by each layer of OPCA, with $C=10000$. The results are shown in Table III. We observe that EFPA requires 78 iterations to converge, but the first layer of OPCA only requires 6 iterations and the other layers requires even fewer iterations. Layer 3 for the OPCA algorithm consumes only 0.0024 seconds because there is no overflowed traffic from layer 2; therefore the offered load to each trunk in layer 3 is 0, in which case, all the layer 3 trunk blocking probabilities equals to 0 without the need to run the Erlang-B formula.

C. Accuracy of OPCA and EFPA for the NSF Network

Having demonstrated that OPCA converges faster than EFPA, it is important to evaluate the accuracy of the two algorithms to see whether the longer running time enables EFPA to provide more accurate results than OPCA. Note also that the presented results here is for an OBS network without trunk reservation, so these results also complement [34] that apply to networks with trunk reservation.

Results for the comparison of the accuracy of OPCA and EFPA for the case $C=50$ are presented in Fig. 3. The results

TABLE III
COMPARISON OF THE TIMES USED BY EFPA AND OPCA IN EACH LAYER TO CALCULATE THE BLOCKING PROBABILITIES IN NSFNET WITH 10000 CHANNELS PER TRUNK.

Algorithm	Layer number	Number of iterations	Total running time in seconds
EFPA	only 1 layer	78	3006
OPCA	layer 0	6	177.9
	layer 1	5	119.7
	layer 2	4	99.7
	layer 3	1	0.0024

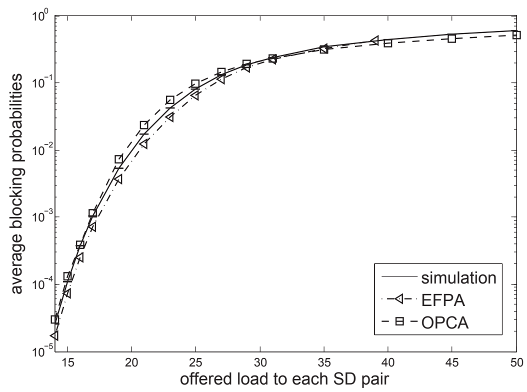


Fig. 3. Blocking probabilities in the NSF network.

are limited to the case $C = 50$ because simulations for large C values are computationally prohibitive. The results are based on a comparison of the two approximations with simulation results for the case of the NSF Network example of Fig. 2 with the 12 SD pairs shown in Table I setting the maximum allowable number of deflections to 3. Error bars for 95% confidence intervals based on Student's t-distribution are provided for all the simulation results although in many cases the intervals are too small to be clearly visible. We observe that OPCA generally slightly overestimates the blocking probability for this example while EFPA underestimates it when the traffic is low; however, OPCA turns to underestimate the blocking probability when the traffic is high. Notice that when the offered load is within 40–50, the results of EFPA are missing. This is because we can not achieve a convergence in cases the offered load to each SD pair draws near the number of channels per trunk which is equal to 50. It is known that evaluating blocking probability by EFPA for OBS may fail to converge in certain instances for unprotected deflection routing as shown in [32].

It is observed from Fig. 3 that in the practical loading range, EFPA does not performs better than OPCA. EFPA is only more accurate than OPCA when the offered load is within 35–40 where the blocking probability is above 3×10^{-1} which is way above what is considered an acceptable grade of service. Notice that these results are consistent with the results in [34] (for cases with trunk reservations) where we observed that EFPA is more accurate than OPCA for high load. Thus we observed in the present example that in the practical traffic loading range, normally used for dimensioning purposes, the longer running time does not enable EFPA to provide more accurate results than OPCA.

D. OPCA Bounds behaviour

In Fig. 4 we present results for the bounds of the OPCA result iterations with a function of the number of iterations in each layer for different offered loads to each directional SD pair in the NSF network. The maximum number of deflections is set to 3 and each trunk has 50 channels. We observe in part (a) of the figure that the lower and upper bounds of the OPCA result for overall network blocking probability become closer to each other, and when 6 iterations are made in each layer, the distance between the lower and upper bounds is less than 10^{-5}% of the lower bound value. Similar results are observed in all four cases of different offered load alternatives presented in Fig. 4. Notice however that as the offered load increases the rate that the bounds become closer to each other is somewhat reduced. Still, in the cases presented in parts (a) and (b), the distance between the bounds is less than 10^{-5}% of the lower bound value when six iterations per layer have been completed. However, when the offered load is 30 for each SD pair, which is the case presented in part (c) of Fig. 4, seven iterations per layer are required to achieve distance between the bounds to be 10^{-5}% of the lower bound value and when the offered load increases to 55 (presented in part (d) of Fig. 4), eight iterations per layer are required to achieve the same accuracy. This is due to the fact that in the NSF network, there are LBTs in each layer. For the LBTs, when the offered load increases in all layers, the first upper bound of trunk blocking probability obtained by the first iteration also increases. This fact can be observed by equations (24) and (53) which show that the trunk blocking probability increases in offered load. Since the first lower bound is 0 for all LBTs, and the first upper bound increases as the traffic increases, the distance between the first upper and lower bounds of trunk blocking probability obtained by the first iteration is larger when the offered load is larger. These general assertions are consistent with observations in [34]. We also observe in the figure, that as the offered load to the network increases, which implies more primary and deflected bursts in the network, it apparently makes it somewhat more difficult for the bounds to become closer to each other.

1) The effect of the number of channels per trunk: Fig. 5 shows the bounds of the OPCA blocking probability results for the NSF network considering four scenarios where in each scenario there are different channels per trunk. In all the scenarios, the offered load to each directional SD pair is $0.4C$, where C is the number of channels per trunk and the maximum number of deflection is set to 3. We observe from the figure that when the number of channels per trunk increases, the lower and upper bounds become closer to each other faster. In particular, when there are 20 channels per trunk, six iterations per layer are required to achieve a distance between the lower and upper bounds to be around 10^{-5}% of the lower bound value, but when there are 100 channels per trunk, in five iterations per layer we achieve a much lower distance between the bounds, namely, 10^{-8}% approx. of the lower bound value. The results are related to the fact that with larger number

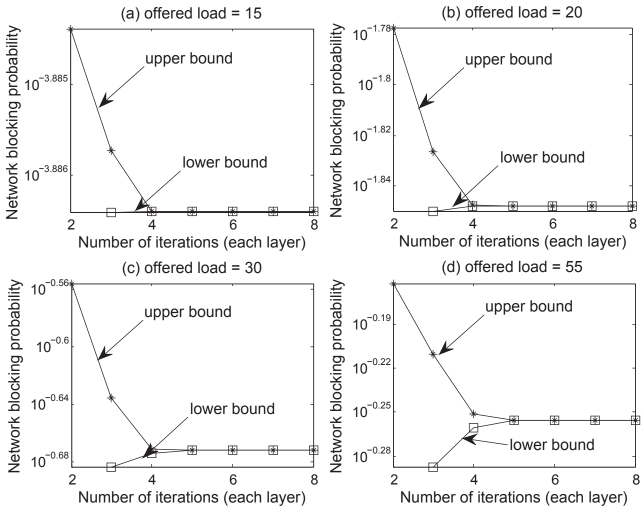


Fig. 4. Bounds of OPCA blocking probabilities in NSF network with different offered load to each directional SD pair.

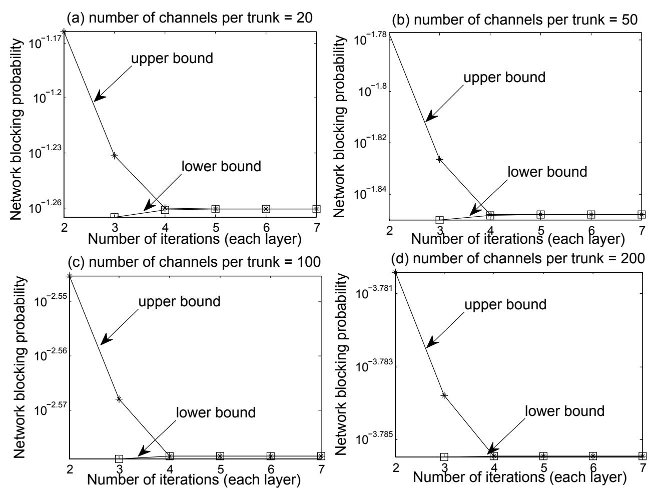


Fig. 5. Bounds of OPCA blocking probabilities in NSF network with different number of channels per trunk (C) in which the offered load to each directional SD pair is $0.4C$

of channels per trunk, the variance of the number of busy channels is lower which implies less deflected bursts in the networks, as we have already observed before, this makes it easier for the bounds to become closer to each other.

2) Effect of the maximum allowable number of deflections: Fig. 6 shows the bounds of the OPCA blocking probability results for the NSF network considering four scenarios where in each scenario we set a different value for the maximum allowable number of deflections (D). In all scenarios, the offered load to each directional SD pair is 20 Erlangs and each trunk has 50 channels. We observe from the figure that when D increases, the lower and upper bounds become closer to each other slightly slower since the overflowed traffic increases with D increasing. However, this effect does not seem to be significant when $D \geq 2$ because the traffic in layers k (for $k \geq 2$) is very small and its effect to the end-to-end blocking probability is negligible. In particular, when $D = 0$ and six iterations per layer are made, the distance between the lower

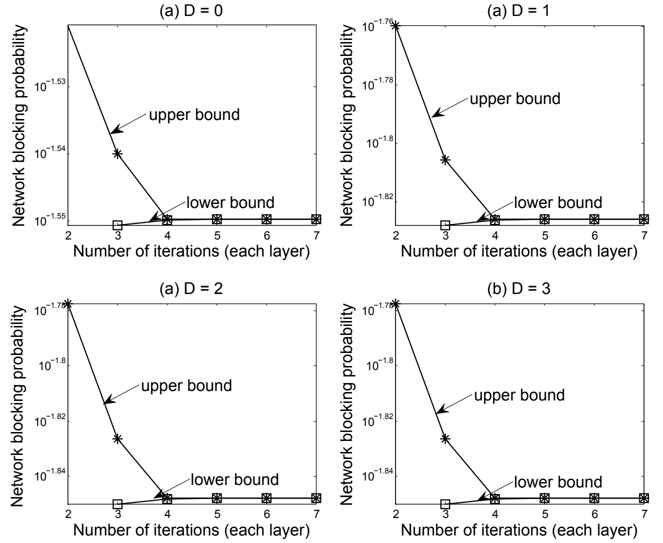


Fig. 6. Bounds of OPCA blocking probabilities in the NSF network with different maximum allowable number of deflections (D) in which the offered load to each directional SD pair is 20 Erlangs

and upper bounds is around $10^{-8}\%$ of the lower bound value, but when $D = 2$ and $D = 3$, in six iterations per layer the distance between the lower and upper bounds is only around $10^{-6}\%$ of the lower bound value.

VI. Conclusions

In this paper, we have presented the bounds of the OPC for blocking probability approximation in OBS Networks with deflection routing. The bounds obtained by the iterations of the OPCA algorithm consistently become closer to each other, and after a small number of steps yield a satisfactory blocking probability approximation. We have observed that the speed of the bounds moving closer decreases when the proportion of the overflowed traffic increases, due to the growth of the offered load or the maximum allowable number of deflections, as well as the reduction of the number of channels per trunk. We have also demonstrated in the case of NSFNet with 50 channels per trunk that OPCA is faster and at least as accurate as EFPA.

VII. Appendix

Let $F(x) = xP_k(x)$, where $P_k(x) = \frac{x^k/k!}{\sum_{i=0}^{k} x^i/i!}$ for positive integer k and $x \geq 0$. The meaning of $P_k(x)$ is the loss probability in the $M/G/k/k$ queueing system with the offered traffic load x. $P_k(x)$ is an increasing function in x.

The challenge is to prove that

$$Q(x,a) = \frac{F(x+a) - F(x)}{a} \tag{66}$$

is an increasing function in a.

We can write function (66) as

$$Q(x,a) = \frac{F(x+a) - F(x)}{a} = \frac{1}{a}\int_{x}^{x+a} F'(y)\mathrm{d}y. \tag{67}$$

The function $F(x) = xP_k(x)$ is a convex function (for direct proof e.g. see [44], [45]). Using the first mean value theorem for integration, we have

$$Q(x,a) = F'(x+\theta a),$$

where $0 < \theta < 1$.

Using the fact that $F(x)$ is a convex function, we have the following properties,

1) $Q(x,a)$ increases when x increases.
2) Let $a_1 < a_2$. Then, because $F'(x)$ is an increasing function, we have the inequality $a_1\theta_1 \le a_2\theta_2$, where θ_1 and θ_2 are such the constants belonging to the interval (0,1) for which $Q(x,a_1) = F'(x+\theta_1 a_1)$ and $Q(x,a_2) = F'(x+\theta_2 a_2)$ (as shown in Fig. 7).

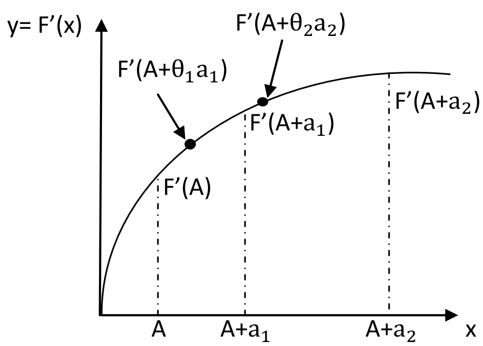


Fig. 7. Positions of $Q(x,a_1)$ and $Q(x,a_2)$

Acknowledgements

The work on this paper was done when Vyacheslav Abramov was with the Department of Electronic Engineering, City University of Hong Kong.

References

[1] C. Qiao and M. Yoo, "Optical burst switching (OBS) - a new paradigm for an optical Internet," *Journal of High Speed Networks*, vol. 8, no. 1, pp. 69–84, Jan. 1999.

[2] S. Verma, H. Chaskar, and R. Ravikanth, "Optical burst switching: A viable solution for terabit IP backbone," *IEEE Network*, vol. 14, pp. 48–53, Nov./Dec. 2000.

[3] M. Yoo, C. Qiao, and S. Dixit, "Optical burst switching for service differentiation in the next-generation optical Internet," *IEEE Commun. Mag.*, vol. 39, no. 2, pp. 98–104, Feb. 2001.

[4] T. Battestilli and H. Perros, "An introduction to optical burst switching," *IEEE Commun. Mag.*, vol. 41, no. 8, pp. S10–S15, Aug. 2003.

[5] Y. Chen, C. Qiao, and X. Yu, "Optical burst switching: a new area in optical networking research," *IEEE Network*, vol. 18, no. 3, pp. 16–23, May/Jun. 2004.

[6] L. Xu, H. Perros, and G. Rouskas, "Techniques for optical packet switching and optical burst switching," *IEEE Commun. Mag.*, vol. 39, no. 1, pp. 136–142, Jan. 2001.

[7] K. Sriram, D. Griffith, S. Lee, and N. Golmie, "Optical burst swithcing: Benifits and challenges," in *Proc. 1st Int. Workshop on Optical Burst Switching (WOBS), Cojunction With OptiComm, Dallas, TX*, Oct. 2003, pp. 1–12.

[8] A. Ghafoor, M. Guizani, and S. Sheikh, "Architecture of an all-optical circuit-switched multistage interconnection network," *IEEE J. Sel. Areas Commun.*, vol. 8, no. 8, pp. 1595–1607, Oct. 1990.

[9] W.-Y. Yuen, A. Chen, A.-S. Wong, and C.-T. Lea, "Buffer-efficient routing strategies for optical circuit switching networks," in *Proc. BroadNets 2004*, Oct. 2004, pp. 328–330.

[10] K. Barker, A. Benner, R. Hoare, A. Hoisie, A. Jones, D. Kerbyson, D. Li, R. Melhem, R. Rajamony, E. Schenfeld, S. Shao, C. Stunkel, and P. Walker, "On the feasibility of optical circuit switching for high performance computing systems," in *Proc, ACM/IEEE SC 2005*, Nov. 2005.

[11] A. Munir, S. Tanwir, and S. Zaidi, "Requests provisioning algorithms for dynamic optical circuit switched (DOCS) networks: A survey," in *Proc. IEEE INMIC 2009*, Dec. 2009.

[12] G. N. Rouskas and L. Xu, "Optical packet switching," in *Emerging Optical Network Technologies*, K. M. Sivalingam and S. Subramaniam, Eds. Springer, 2005, pp. 111–127, 10.1007/0-387-22584-6_5.

[13] D. Blumenthal, "Optical packet switching," in *Proc. IEEE LEOS 2004*, vol. 2, Nov. 2004, pp. 910–912.

[14] C. Guillemot *et al.*, "Transparent optical packet switching: The european acts keops project approach," *J. Lightwave Technol.*, vol. 16, no. 12, p. 2117, Dec. 1998.

[15] A. Detti, V. Eramo, and M. Listanti, "Performance evaluation of a new technique for IP support in a WDM optical network: optical composite burst switching (OCBS)," *J. Lightwave Technol.*, vol. 20, no. 2, pp. 154–165, Feb. 2002.

[16] Z. Rosberg, H. L. Vu, M. Zukerman, and J. White, "Performance analyses of optical burst-switching networks," *IEEE J. Sel. Areas Commun.*, vol. 21, no. 7, pp. 1187–1197, Sep. 2003.

[17] Z. Rosberg, A. Zalesky, H. L. Vu, and M. Zukerman, "Analysis of OBS networks with limited wavelength conversion," *IEEE/ACM Trans. Networking*, vol. 14, no. 5, pp. 1118–1127, Oct. 2006.

[18] A. Zalesky, H. Vu, Z. Rosberg, E. Wong, and M. Zukerman, "OBS contention resolution performance," *Performance Evaluation*, vol. 64, no. 4, pp. 357–373, 2007.

[19] S. Bregni, A. Caruso, and A. Pattavina, "Buffering-deflection tradeoffs in optical burst switching," *Photonic Network Communications*, vol. 20, no. 2, pp. 193–200, Aug. 2010.

[20] S.-H. Ngo, X. Jian, and S. Horiguchi, "Hybrid deflection and retransmission routing schemes for OBS networks," in *Proc. High Performance Switching and Routing 2006*, 2006.

[21] S.-H. Ngo and X. Jiang, "Burst loss probability model for hybrid deflection and retransmission routing in OBS networks," in *Proc. IEEE RIVF 2010*, Nov. 2010.

[22] M. Levesque, H. Elbiaze, and W. Aly, "Adaptive threshold-based decision for efficient hybrid deflection and retransmission scheme in OBS networks," in *Proc. ONDM 2009*, Feb. 2009.

[23] T. Ozugur, F. Farahmand, and D. Verchere, "Single-anchored soft bandwidth allocation system with deflection routing for optical burst switching," in *Proc. High Performance Switching and Routing, 2002*, 2002, pp. 257–261.

[24] W. Dong, M. Fu, Z. Le, and X. Sun, "AIMD control for deflection routing in OBS networks," in *Proc. ACP 2009*, Nov. 2009.

[25] Y. Mori, T. Abe, H. Pan, Y. Zhu, Y.-B. Choi, and H. Okada, "Effective flow-rate control for the deflection routing based optical burst switching networks," in *Proc. APCC '06*, Aug. 2006.

[26] A. Belbekkouche, A. Hafid, and M. Gendreau, "A reinforcement learning-based deflection routing scheme for buffer-less OBS networks," in *Proc. IEEE GLOBECOM 2008*, Nov. 2008.

[27] A. Al Amin, M. Takenaka, T. Tanemura, K. Shimizu, R. Inohara, K. Nishimura, M. Usami, Y. Takita, Y. Kai, H. Onaka, H. Uetsuka, and Y. Nakano, "Demonstration of deflection routing with layer-2 evaluation at 40 Gb/s in a three-node optical burst switching testbed," *IEEE Photonics Technology Letters*, vol. 20, no. 3, pp. 178–180, Feb. 2008.

[28] Y. Chen, H. Wu, D. Xu, and C. Qiao, "Performance analysis of optical burst switched node with deflection routing," in *Proc. ICC '03*, vol. 2, May 2003, pp. 1355–1359.

[29] C.-F. Hsu, T.-L. Liu, and N.-F. Huang, "Performance analysis of deflection routing in optical burst-switched networks," in *Proc. IEEE INFOCOM 2002*, vol. 1, 2002, pp. 66–73.

[30] S. Yao, B. Mukherjee, S. Yoo, and S. Dixit, "A unified study of contention-resolution schemes in optical packet-switched networks," *J. Lightwave Technol.*, vol. 21, no. 3, pp. 672–683, Mar. 2003.

[31] S. Lee, K. Sriram, H. Kim, and J. Song, "Contention-based limited deflection routing protocol in optical burst-switched networks," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 8, pp. 1596–1611, Aug. 2005.

[32] A. Zalesky, H. L.Vu, Z. Rosberg, E. W. M. Wong, and M. Zukerman, "Stabilizing deflection routing in optical burst switched networks," *IEEE J. Sel. Areas Commun.*, vol. 25, no. 6, pp. 3–19, Aug. 2007.

[33] R. B. Cooper and S. S. Katz, "Analysis of alternate routing networks with account taken of the nonrandomness of overflow traffic," Bell Telephone Lab, Tech. Rep., 1964, memo.

[34] E. W. M. Wong, J. Baliga, M. Zukerman, A. Zalesky, and G. Raskutti, "A new method for blocking probability evaluation in OBS/OPS networks with deflection routing," *IEEE/OSA J. Lightwave Technol.*, vol. 27, no. 23, pp. 5335–5347, Dec. 2009.

[35] E. W. M. Wong, A. Zalesky, Z. Rosberg, and M. Zukerman, "A new method for approximating blocking probability in overflow loss networks," *Computer Networks*, vol. 51, no. 11, pp. 2958–2975, Aug. 2007.

[36] G. Raskutti, A. Zalesky, E. W. M. Wong, and M. Zukerman, "Enhanced blocking probability evaluation method for circuit-switched trunk reservation networks," *IEEE Commun. Lett.*, vol. 11, no. 6, pp. 543–545, Jun. 2007.

[37] C. Qiao, "Labeled optical burst switching for IP-over-WDM integration," *IEEE Commun. Mag.*, vol. 38, no. 9, pp. 104–114, Sep. 2000.

[38] D. C. Americas. Optical fiber cable catalog. [Online]. Available: http://communications.draka.com/sites/usa/Media/US%20resources/Catalogs/DrakaCatalog.pdf

[39] T. Kuri, H. Toda, and K. Kitayama, "Dense wavelength-division multiplexing millimeter-wave-band radio-on-fiber signal transmission with photonic downconversion," *Journal of lightwave technology*, vol. 21, no. 6, pp. 1510–1517, 2003.

[40] M. Zukerman, "Back to the future," *IEEE Commun. Mag.*, vol. 47, no. 11, pp. 36–38, 2009.

[41] H. Beyranvand and J. Salehi, "Efficient optical resource allocation and QoS differentiation in optical burst switching networks utilizing hybrid WDM/OCDM," *J. Lightwave Technol.*, vol. 30, no. 15, pp. 2427–2441, Aug. 2012.

[42] D. Gale, "The game of Hex and the brouwer fixed-point theorem," *The American Mathematical Monthly*, vol. 86, no. 10, pp. 818–827, 1979.

[43] F. P. Kelly, "Blocking probabilities in large circuit-switched networks," *Adv. in Appl. Probab.*, vol. 18, no. 2, pp. 473–505, Jun. 1986.

[44] K. P. Krishnan, "The convexity of loss rate in an Erlang loss system and sojourn in an erlang delay system with respect to arrival and service rates," *IEEE Trans. Commun.*, vol. 38, no. 9, pp. 1314–1316, Sep. 1990.

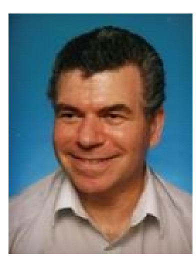
[45] E. Altman, R. El-Azouzi, and V. Abramov, "Non-cooperative routing in loss networks," *Performance Evaluation*, vol. 49, no. 1-4, pp. 257–272, 2002.

Shuo Li received the B.Sc. degree in electronic and communication engineering from City University of Hong Kong, Hong Kong, in 2009. She is currently working toward the Ph.D. degree in the department of Electronic Engineering at City University of Hong Kong. Her research interests are performance evaluation of telecommunication networks.

Meiqian Wang received her Bachelor degree and Master degree in Department of Electronic Engineering at Harbin institute of Technology in 2006 and 2009, respectively. She is now working towards the PhD degree at City University of Hong Kong. Her research interest lies in performance evaluation in circuit switching and burst switching networks.

Eric W. M. Wong (S'87–M'90–SM'00) 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 of Hong Kong, where he is now an Associate Professor with the Department of Electronic Engineering. His research interests include the analysis and design of telecommunications networks, optical switching and video-on-demand systems. His most notable research work involved the first accurate and workable model for state-dependent dynamic routing based on the Erlang fixed-point approximation presented in Infocom '90. Since 1991, the model has been used by AT&T to design and dimension its telephone network that uses real-time network routing.

Vyacheslav Abramov is graduated from Tajik State University (former USSR) in 1977. During 1977-1992 he worked at the Research Institute of Economics under the State Planning Committee of Tajikistan. In 1992 Abramov repatriated to Israel, where he worked in different institutions and software companies. Abramov received PhD in Mathematics from Tel Aviv University in 2005. During 2005-2011 he worked at the School of Mathematical Sciences in Monash University (Australia). During 2011-2012 Abramov worked at the Department of Electronic Engineering in City University of Hong Kong as a senior research fellow. Since August 2012 he has been working at Center of Advanced Internet Architectures (CAIA) in Swinburne University of Technology. Abramov is the author of two books and about 40 research papers in the areas of applied probability, statistics, queueing theory and financial engineering.



Moshe Zukerman (M'87–SM'91–F'07) received the B.Sc. degree in industrial engineering and management and the M.Sc. degree in operations research from the Technion – Israel Institute of Technology, Haifa, Israel, and the Ph.D. degree in engineering from University of California, Los Angeles, in 1985. He was an independent Consultant with the IRI Corporation and a Postdoctoral Fellow with the University of California, Los Angeles, in 1985–1986. In 1986–1997, he was with the Telstra Research Laboratories (TRL), first as a Research Engineer and, in 1988–1997, as a Project Leader. He also taught and supervised graduate students at Monash University in 1990–2001. During 1997C-2008, he was with The University of Melbourne, Victoria, Australia. In 2008 he joined City University of Hong Kong as a Chair Professor of Information Engineering, and a group leader. He has served on various editorial boards such as Computer Networks, IEEE Communications Magazine, IEEE Journal of Selected Ares in Communications, IEEE Transactions on Networking and the International Journal of Communication Systems.