Performance Analyses of Circuit Switched Networks

Submitted to
Department of Electronic Engineering
In Partial Fulfillment of the Requirements
for the Degree of Doctor of Philosophy

By

WANG Meiqian

September 2013
Abstract

Over the last quarter of a century, the Internet has evolved from being a packet-switched network that provides only data services, such as e-mail and file transfer, to a network that provides a wide range of multimedia services. Switching large application flows associated with multimedia services or high-energy physics experiments by using packet switching at the IP layer requires unacceptable energy consumption levels. Given the potential for orders of magnitude savings in energy per bit by using lower (optical) layers, circuit switching, which has been widely used in telephone networks, is expected to have a renewed and important role in future optical networks.

Both static and dynamic traffic will coexist in future optical networks. The classification of connections into long-lived and short-lived has been discussed in various papers. Static circuit-switched connections are likely to have priority over dynamic ones as static connections can be booked well in advance. Thus, the performance of static connections is not affected by the loading of dynamic circuit-switched connections. We consider a circuit-switched network with non-hierarchical alternate routing and trunk reservation that involves two types of connections modeled as long-lived and short-lived calls. Long-lived connections can be reserved well in advance, and short-lived connections are provided on demand. Therefore, we assume that long-lived connections have strict priority over the short-lived ones. We develop approximations for the estimation of blocking probability based on the quasi-stationary approach in two ways. One uses Erlang fixed-point approximation (EFPA), and the other uses overflow priority classification approximation (OPCA). We compare the results of the approximations with simulation results and discuss the accuracy of the approximations under different system parameters, such as the ratio of offered load, the number of links per trunk, the maximum allowable number of deflections, and trunk reservation. We also discuss the robustness of quasi-stationary approximation to the ratio of the mean holding times of long-lived and short-lived traffic.
streams as well as that of EFPA and OPCA to the shape of holding time distribution. Finally, we demonstrate that OPCA can be applied to a large network, such as Coronet.

For large circuit-switched networks, previous work has shown that EFPA achieves accurate results for networks that have a large number of channels (circuits) per link. However, a conventional application of EFPA for large networks is computationally prohibitive. In cases where the EFPA solution is unattainable, we propose, in Chapter 3, to use an asymptotic approximation, which we call A-EFPA, for the link blocking probability and demonstrate computation time savings of many orders of magnitude for blocking probability approximation in realistically sized networks with a large number of circuits per link. For NSFNet and Internet2, we demonstrate that blocking probability can be accurately calculated by using simulations, EFPA, and A-EFPA, each of which is used for a different range of parameter values. We also demonstrate the scalability of the approximations by applying them to the 100-node CORONET network.

Different rates, holding times, and bandwidth requirements are relevant to future circuit switching applications on the Internet. Bandwidth on demand (BoD) services are provided where fixed capacity is allocated for the service duration and then released by the user. Potential customers are cloud service providers, smaller operators, enterprises, research networks, and even retail customers. Accordingly, we can expect a scenario in which all these BoD service classes compete for the same pool of optical capacity available in the core network that needs to be allocated efficiently. Each class can be characterized according to the arrival rate of its burst, flow, or connection requests; its mean holding time; and its capacity requirement. A network operator that aims to provide such a wide range of BoD services needs a way to efficiently dimension its network to meet QoS requirements. A scalable and accurate method is needed to evaluate the blocking probability for each relevant scenario of network topology, parameter values, and traffic demand. To this end, we consider a circuit-switched multi-service multi-rate network with non-hierarchical deflection routing and trunk reservation in Chapter 4. Based on
the fundamental concept of OPCA, we develop two approximations for the estimation of
blocking probability, namely, OPCA and service-based OPCA. We also apply classical
EFPA for the estimation of blocking probability in the network and propose \( \max(\text{EFPA}, \text{service-based OPCA}) \) as an accurate and normally conservative evaluation. We compare
the approximations with simulation results and discuss the accuracy of the blocking prob-
abilities of the classes under different system parameter values, such as service rates,
bandwidth requirements, number of links per trunk, maximum allowable number of de-
flections, and trunk reservation. We also discuss the robustness of the approximations
to the shape of the holding time distribution and their performances under asymmetrical
cases. We also present that the approximations can be applied to large networks such as
Coronet. We demonstrate that based on testing over a wide range of parameter values,
\( \max(\text{EFPA}, \text{service-based OPCA}) \) gives a very accurate and conservative estimation of
network blocking probability in a multi-service multi-rate network.
Acknowledgements

My time as a PhD candidate has been the most valuable and impressive journey of my whole life. I have learned how to be a researcher and acquired precious experience during these years. I would like to express my deepest appreciation to Dr. Eric Wong for being a wonderful supervisor and making all this possible for me. His innovative ideas have inspired brilliant solutions and improvements, and have guided me through my exploration in the field. I also owe my deepest gratitude to my co-supervisor Professor Moshe Zukerman for his generous support and help and for everything that I have learned from him, which will definitely benefit me my whole life. I feel very lucky to have them as my supervisors and I would not have accomplished all this without them. I am also honored to work with Li Shuo, Zhang jianan and Jun Guo, as well as Vyacheslav Abramov whose idea gave rise to an asymptotic approximation for blocking probability approximation in realistically sized networks with large number of circuits per link.

I also would like to thank my parents and my husband for their unconditional love that encourages me during these days.
## Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abstract</td>
<td>i</td>
</tr>
<tr>
<td>Acknowledgements</td>
<td>iv</td>
</tr>
<tr>
<td>List of Tables</td>
<td>viii</td>
</tr>
<tr>
<td>List of Figures</td>
<td>ix</td>
</tr>
<tr>
<td><strong>1 Introduction</strong></td>
<td>1</td>
</tr>
<tr>
<td>1.1 Background</td>
<td>1</td>
</tr>
<tr>
<td>1.2 Thesis Outline</td>
<td>8</td>
</tr>
<tr>
<td>1.3 Contributions</td>
<td>10</td>
</tr>
<tr>
<td>1.4 Publications</td>
<td>10</td>
</tr>
<tr>
<td><strong>2 Circuit Switched Networks with long-lived and short-lived connections</strong></td>
<td>12</td>
</tr>
<tr>
<td>2.1 Introduction</td>
<td>12</td>
</tr>
<tr>
<td>2.2 The Model</td>
<td>14</td>
</tr>
<tr>
<td>2.3 Blocking Probability Approximations</td>
<td>19</td>
</tr>
<tr>
<td>2.3.1 EFPA</td>
<td>21</td>
</tr>
<tr>
<td>2.3.2 OPCA</td>
<td>25</td>
</tr>
<tr>
<td>2.4 Numerical Results</td>
<td>29</td>
</tr>
<tr>
<td>2.4.1 Default parameter setting</td>
<td>31</td>
</tr>
<tr>
<td>2.4.2 Blocking probability for the long-lived traffic</td>
<td>32</td>
</tr>
<tr>
<td>2.4.3 Blocking probability for the short-lived traffic</td>
<td>34</td>
</tr>
</tbody>
</table>
2.4.4 The effect of the ratio between the offered long-lived traffic and offered short-lived traffic ........................................ 37
2.4.5 The effect of the number of links on each trunk ..................... 39
2.4.6 The effect of maximum allowable number of alternate paths ...... 44
2.4.7 The effect of trunk reservation ........................................ 45
2.4.8 Robustness of the quasi-stationary approximation .................. 47
2.4.9 The effect of the shape of the holding time distribution .......... 48
2.4.10 Computational complexity of the algorithms ....................... 50
2.4.11 The Coronet ............................................................ 51
2.5 Summary ........................................................................ 52

3 Computation of blocking probability for large circuit switched networks 54
3.1 Introduction ...................................................................... 54
3.2 Model ............................................................................ 56
3.3 A-EFPA algorithm .......................................................... 57
3.4 Numerical Results ........................................................... 60
3.5 Summary ........................................................................ 65

4 Performance analysis of Circuit Switched Multi-service Multi-rate Networks with Deflection Routing 67
4.1 Introduction ...................................................................... 67
4.2 The Model ....................................................................... 77
4.3 Blocking Probability Approximations .................................... 80
  4.3.1 EFPA ......................................................................... 81
  4.3.2 OPCA ......................................................................... 83
  4.3.3 The trunk offered load in non-disjoint path cases ............... 88
  4.3.4 The max(EFPA, service-based OPCA) approximation .......... 90
4.4 Numerical Results ............................................................ 91
# List of Tables

1.1 EFPA Errors .......................................................... 6

2.1 Summary of Notation ................................................. 20

2.2 Computational complexity of the algorithms ................. 51

3.1 Comparison of the times used by EFPA and A-EFPA to calculate the blocking probabilities in NSFNet. .......................... 62

3.2 Effect of offered load on A-EFPA. ................................. 62

3.3 Comparison of the blocking probabilities calculated by EFPA and A-EFPA for NSFNet and Internet2. .......................... 63

3.4 Comparison of the times used by EFPA and A-EFPA to calculate the blocking probabilities in Internet2. .......................... 64

4.1 Network blocking probability for class 1 in 6-node fully mesh network with moment matching implemented in EFPA and service-based OPCA. 111

4.2 Network blocking probability for class 2 in 6-node fully mesh network with moment matching implemented in EFPA and service-based OPCA. 112
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Blocking probabilities for the long-lived traffic (single priority) versus $T$ in (a) a 6-node fully-meshed network and (b) NSF network. 33</td>
</tr>
<tr>
<td>2.2</td>
<td>NSF network topology with 13 nodes, connected by bi-directional trunks between them. 34</td>
</tr>
<tr>
<td>2.3</td>
<td>Blocking probabilities for the short-lived traffic in (a) a 6-node fully-meshed network and (b) NSF network. The ratio of the offered long-lived traffic to offered short-lived traffic is 1:1. 34</td>
</tr>
<tr>
<td>2.4</td>
<td>Blocking probabilities for the short-lived traffic in a 6-node fully-meshed network. The ratio of the offered long-lived traffic to offered short-lived traffic is (a) 1:2, (b) 1:5, (c) 2:1, and (d) 5:1. 37</td>
</tr>
<tr>
<td>2.5</td>
<td>Blocking probabilities for the short-lived traffic in NSF network. The ratio of the offered long-lived traffic to offered short-lived traffic is (a) 1:2, (b) 1:5, (c) 2:1, and (d) 5:1. 39</td>
</tr>
<tr>
<td>2.6</td>
<td>Blocking probabilities for (a) long-lived traffic (single priority) and (b) short-lived traffic in a 6-node fully-meshed network with 50 links each trunk. 40</td>
</tr>
<tr>
<td>2.7</td>
<td>Blocking probabilities for (a) long-lived traffic (single priority) and (b) short-lived traffic in a 6-node fully-meshed network versus number of links on each trunk. 41</td>
</tr>
<tr>
<td>Section</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td>2.8</td>
<td>Blocking probabilities for (a) long-lived traffic (single priority) and (b) short-lived traffic in NSF network with 50 links each trunk.</td>
</tr>
<tr>
<td>2.9</td>
<td>Blocking probabilities for (a) long-lived traffic (single priority) and (b) short-lived traffic in NSF network versus number of links per trunk.</td>
</tr>
<tr>
<td>2.10</td>
<td>Blocking probability for (a) long-lived traffic (single priority) and (b) short-lived traffic for a 6-node fully-meshed network</td>
</tr>
<tr>
<td>2.11</td>
<td>Blocking probability for short-lived traffic in a 6-node fully-meshed network with long-lived and short-lived traffic. The ratio of the offered long-lived traffic to offered short-lived traffic is (a) 1:1 and (b) 5:1.</td>
</tr>
<tr>
<td>2.12</td>
<td>The average blocking probabilities for the short-lived traffic with different values of $\mu_2$ for a 6-node fully-meshed network with 20 links each trunk and $\mu_1 = 1$.</td>
</tr>
<tr>
<td>2.13</td>
<td>The average blocking probabilities for (a) long-lived traffic (single priority), and (b) short-lived traffic, considering different service time distributions for a 6-node fully-meshed network with 20 links in each trunk.</td>
</tr>
<tr>
<td>2.14</td>
<td>The average blocking probabilities for (a) long-lived traffic (single priority), and (b) short-lived traffic, considering different service time distributions for the NSF network with 20 links in each trunk.</td>
</tr>
<tr>
<td>2.15</td>
<td>The Coronet topology.</td>
</tr>
<tr>
<td>2.16</td>
<td>The blocking probabilities obtained by OPCA for (a) long-lived traffic (single priority), and (b) short-lived traffic for the Coronet with 50 links per trunk.</td>
</tr>
<tr>
<td>3.1</td>
<td>The Internet2 topology.</td>
</tr>
<tr>
<td>3.2</td>
<td>Blocking probabilities in NSFNet and Internet2 evaluated by EFPA and A-EFPA.</td>
</tr>
<tr>
<td>3.3</td>
<td>Blocking probabilities in 13-node NSFNet evaluated by EFPA, A-EFPA and simulation with two classes of traffic.</td>
</tr>
</tbody>
</table>
3.4 Blocking probability in the Coronet evaluated by EFPA and A-EFPA.

4.1 Examples of non-disjoint paths.

4.2 Network blocking probabilities for (a) class 1 calls and (b) class 2 calls.

4.3 Network blocking probabilities for (a) class 1 calls and (b) class 2 calls.

4.4 Network blocking probabilities for (a) class 1 calls and (b) class 2 calls.

4.5 13-node NSFNet topology.

4.6 Network blocking probabilities for (a) class 1 calls and (b) class 2 calls in NSFNet. The offered load of class 2 is 0.05 Erlang.

4.7 Network blocking probabilities for (a) class 1 calls and (b) class 2 calls.

The offered load of class 2 is 1 Erlang. The service rate of class 1 and class 2 are 3 and 1, respectively.

4.8 Network blocking probabilities for both classes. The offered load for class 2 is 1 Erlang. The bandwidth requirements of class 1 are 1 for (a) and (b), 4 for (c) and (d). The bandwidth requirement of class 2 remains 5.

4.9 Network blocking probabilities for both classes. The offered load for class 1 is 6 Erlangs. The bandwidth requirements of class 2 are 3 for (a) and (b), 8 for (c) and (d). The bandwidth requirement of class 1 remains 2.

4.10 Network blocking probabilities for the classes in the default scenario with 100 channels each trunk. The offered load of Class 2 is 3 Erlangs.

4.11 Network blocking probabilities of both classes in the default scenario with offered load 5 Erlangs and 1 Erlang for class 1 and class 2, respectively.

4.12 Network blocking probabilities of both classes in the default scenario.

The threshold of class 2 \( T_2 \) remains 40 (80%).
4.13 Network blocking probabilities of both classes in the default scenario.
The threshold of class 1 $T_1$ remains 38 (76%).

4.14 Network blocking probabilities of both classes, considering different service time distributions in the default scenario. The offered load for class 2 is 1 Erlang.

4.15 Network blocking probabilities of both classes, while 20 OD pairs will only send the class 1 traffic and the rest 10 OD pairs will only transmit class 2 traffic in a 6-node fully-meshed network with 50 channels in each trunk. The offered load for class 2 is 1 Erlang.

4.16 Network blocking probabilities of both classes, while 104 OD pairs will only send the class 1 traffic and the rest 52 OD pairs will only transmit class 2 traffic in 13-node NSFNet with 50 channels in each trunk. The offered load for class 2 is 0.1 Erlang.

4.17 The Coronet topology.

4.18 Network blocking probabilities for (a) class 1 calls and (b) class 2 calls. The offered load of class 2 is 0.001 Erlang.

4.19 Network blocking probabilities for (a) class 1 calls, (b) class 2 calls and (c) class 3 calls. The offered load of class 1 and class 2 are 3 Erlangs and 1 Erlang, respectively.

4.20 Network blocking probabilities for (a) class 1 and (b) class 2 calls. The offered load of class 2 is 1 Erlang.

4.21 Network blocking probabilities for (a) class 1 and (b) class 2 calls. The offered load are 5.5 erlangs and 1 erlang of class 1 and class 2, respectively. The propagation delay is set 0.001 second on every trunk and the processing delay is 0.0001 second in each node.
4.22 Network blocking probabilities for (a) class 1 and (b) class 2 calls. The offered load are 5.5 erlangs and 1 erlang of class 1 and class 2, respectively. The propagation delay is set 0.001 second on every trunk and the processing delay is 0.0001 second in each node.

4.23 Network blocking probabilities for (a) class 1 and (b) class 2 calls in NSFNet. The offered load are 0.2 erlang and 0.05 erlang of class 1 and class 2, respectively.

4.24 Number of channels per trunk required to keep the biggest blocking probability of the three classes below 0.001.
Chapter 1

Introduction

1.1 Background

Over the last quarter of a century, the Internet has evolved from being a packet-switched network that provides only data services, such as e-mail and file transfer, to a network that provides a wide range of services. Nowadays, an increasing number of Internet users transmit extremely large flows of data. Internet user include cloud service providers (CSPs), such as Google, Facebook, and Yahoo!, which often replicate their content across multiple data centers that transmit massive amounts of data [1]. In fact, the total CSP inter-data center traffic was over 400 exabytes in 2011 and grew by over 30% yearly [2]. Internet users also include CERN’s Large Hadron Collider (LHC), which transmits several petabytes of data per year [3].

Switching such extremely large flows by using packet switching at the IP layer requires unacceptable energy consumption levels [4–6]. Given the potential for order of magnitude savings in energy per bit by using lower (optical) layers [7], circuit switching, which has been widely used in telephone networks, is expected to have a renewed and important role in future optical networks [8–15]. Many energy-consuming operations, such as buffering packets, processing individual packet headers, performing table
look-up, and counting packets, are avoided by using circuit switching. These advantages of circuit switching for wide bandwidth networks were identified nearly a quarter of a century ago \[16\] in terms of “simplicity” rather than energy consumption.

If the bit rate offered to a circuit-switched multimedia network is sufficiently high and traffic is well engineered, such a network can guarantee quality-of-service (QoS) to customers in a way that can even lead to efficient link utilization and low energy consumption. For example, a 100 GB burst transmitted from the LHC can be efficiently transmitted by setting up a circuit of one or more wavelengths that will be fully utilized during its holding time. Other benefits provided by circuit switching include overload control without congestion collapse, robustness due to fast recovery of circuit-switching equipments, and simple provision of synchronization.

Circuit-switched networks may encounter traffic overload situations when the offered traffic load exceeds the network capacity, and in such situations, new connections should be rejected (blocked or dropped). Such blocking or dropping will adversely affect QoS, which will be perceived by users. Therefore, the likelihood or probability of such events should not exceed a certain predetermined value. For over a century, operators have considered blocking probability as a key performance measure for circuit-switched network design and dimensioning. In today’s competitive environment, limiting blocking probability is clearly important.

To reduce blocking probability in circuit-switched networks, various dynamic or alternate routing approaches have been studied \[11, 17–30\], where new connections that cannot be admitted by their primary routes may overflow to other routes that may be more costly in terms of the use of network resources. To establish a connection, a control packet is sent to reserve the capacity on each trunk along the primary path toward the destination on the control plane. If one or more trunks along the primary path do not have enough capacity for the connection, the connection request is overflowed to its first alternative path, and a control packet is sent to reserve capacity on that path. A connection
can be permitted to overflow several times, which may be limited by network topology or a design parameter. When a light path is established by the control packet transmitted on the control plane, the payload of the connection is transmitted on the data plane. The use of increasingly costly alternate routes may adversely affect blocking probability. Therefore, the number of allowable overflow attempts by a particular call may be limited. Thus, a call is blocked after a finite number of alternate route attempts. The proportion of blocked traffic is defined as blocking probability. Accurate evaluation of blocking probability is important for traffic management and for network design and dimensioning, which ensures that sufficient resources are available to meet QoS requirements.

A circuit-switched network with alternate routing, under the basic assumptions of the Poisson arrival of calls for any origin destination (OD) pair and exponential call holding times, can be modeled as a Markovian overflow loss network. The transition rates between states of a trunk are obtained considering the arrival and departure rates of connections on their primary path and those that overflow to the trunk from other fully occupied trunks. Stationary occupancy distribution can, in principle, be obtained by a numerical solution of the steady-state equations of a multidimensional Markov process. Such models usually do not admit product-form solutions [31] and are not amenable to analysis that lead to a scalable solution of realistic size networks. Therefore, accurate, robust, and scalable approximations for such problems should be obtained.

Circuit-switched networks with alternate routing can be classified into two classes: hierarchical and non-hierarchical. In hierarchical networks, server groups (or trunks) are ranked into several tiers. New calls first attempt to access server groups from the lowest tier. If these calls are rejected, they overflow and attempt to access server groups from higher tiers. As a result of the one-way hierarchy, congestion in lower tiers causes congestion in higher tiers, but not vice versa. This hierarchical routing structure was common in early telephone networks. For hierarchical networks, blocking probability can be approximated accurately by using moment matching approaches [32,40] that rely
on the assumption that each tier can be treated independently, where the input traffic of each tier in the hierarchy follows an arrival process whose moments match those of the output process of the previous (lower) tier.

A more difficult problem is the accurate approximation of blocking probabilities in non-hierarchical networks where the overflow event of each server group may directly or indirectly affect the load of any other server group. In particular, the concept of mutual overflow \cite{21} refers to a situation where congestion exists in a specific server group and thus causes overflow to the other server groups and where this overflow loads up other server groups, which yield overflow back to the original server group. Clearly, in such non-hierarchical systems, load dependencies may be much stronger than in hierarchical systems that do not have mutual overflow, and modeling that involves independence assumptions is more likely to lead to significant errors. Despite wide applicability and importance and a century-long research effort, no robust and generic methodology is available for approximating the blocking probability of general non-hierarchical networks that captures their overflow-induced state dependencies in a scalable way. However, several publications have studied overflow and blocking probability in non-hierarchical loss networks by using moment matching for particular applications. For example, \cite{41} studied the performance of a grid computing network with a ring structure. Other methods for evaluating the blocking probability of stochastic networks that do not focus on the overflow phenomenon have been proposed, such as those in \cite{42,43}.

One simple and commonly used approach for the approximation of blocking probabilities in non-hierarchical networks is Erlang fixed-point approximation (EFPA). EFPA is based on the decoupling of a given system into independent server groups (subsystems), each of which is modeled by an M/M/K/K queuing system. Under this approach, the total traffic offered to any server group of K servers follows a Poisson process with a rate equal to the sum of the rates of all individual input streams and all overflow attempts. In this way, the overflow probability perceived by the aggregate stream as well as by each
of the individual input streams to a server group is approximated by the Erlang B formula. Let $E_K(A)$ denote the blocking probability of an $M/M/K/K$ system with $K$ servers and traffic load $A$, defined by $A = \lambda / \mu$ where $\lambda$ is the arrival rate and $\mu$ is the service rate per server. Let $\lambda_i$ be the original rate of the $i$-th stream. The overflow events of the $i$-th stream may form an overflow stream of rate $\lambda_i E_K(\sum_i \lambda_i / \mu)$ to a subsequent server group. This approach inherently gives rise to non-linear equations for the overflow probability at each server group that may be solved by a fixed-point solution. It is known that convergence and uniqueness of solutions of these fixed-point techniques are not always guaranteed [23, 44].

EFPA was first proposed by Cooper and Katz [33] for the analysis of circuit-switched networks and has remained a cornerstone of telecommunications networks and systems analysis to this day. Applications of EFPA are discussed in [17, 21, 23, 25, 30, 45–52] and references therein. Despite its popularity, EFPA is known to introduce two types of errors [53]:

1. **The Poisson error** — EFPA assumes that the traffic offered to any link follows a Poisson process whereas in fact the traffic offered by an overflow stream is known to have higher variance than a Poisson process [40], when traffic offered to a sequence of links on a path may be smoothed out when offered to one link due to blocking in another link. Note that in their original work, the authors of [33] considered an improvement of accuracy by matching both the mean and the variance. However, a common use of EFPA, henceforth referred to as the basic version of EFPA, is based on only matching the mean and the Poisson assumption.

2. **The independence error** — EFPA assumes that server groups are mutually independent, whereas they are in fact statistically dependent because a large number of busy servers in a server group may indicate a heavy traffic period and potential overflow to other server groups, so that other server groups are also likely to be heavily loaded at that time. Dependence between links also occur even if overflowed is not
allowed (i.e. in fixed routing networks); if for example there is high load on one link on a certain end-to-end path, it may be a result of high load on this path, in which case, other links on this path will also have high load.

Various attempts to address Poisson and independence errors include [33, 54] providing means to reduce the Poisson error by moment matching and [55] addressing the independence error by capturing the correlation between links. Traditionally, these methods are the two distinct ways to refine the basic version of EFPA [21]. Both Poisson and independence errors are related to two effects that are characteristics of circuit-switched networks, namely, an effect associated with overflows and an effect associated with the fact that a connection requires a multi-hop path to be established. Thus, EFPA errors caused by these two effects are called overflow errors and path errors. Overflow errors cause the underestimation of blocking probability (ignoring high variance of overflow traffic and dependence), whereas the path error overestimates blocking probability because it ignores the effect of traffic smoothing, and the positive correlation of link occupancy along the path increases the probability that calls will be admitted. These relationships between the errors and their effects are summarized in Table 1.1.

Table 1.1: EFPA Errors

<table>
<thead>
<tr>
<th></th>
<th>Overflow Error</th>
<th>Path Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>Poisson Error</td>
<td>underestimate</td>
<td>overestimate</td>
</tr>
<tr>
<td>Independence Error</td>
<td>underestimate</td>
<td>overestimate</td>
</tr>
</tbody>
</table>

In recent years, we have developed a new method called overflow priority classification approximation (OPCA) [53, 56, 58], which can either improve or complement EFPA to achieve more accurate blocking probability approximations. OPCA imposes a fictitious preemptive priority structure in a given real network model, which then produces a surrogate network model. The blocking probability for the surrogate model is derived by using EFPA, which is expected to yield a close but somewhat different blocking probability by
from that of the original overflow network model and, in many cases, a better approxima-
tion of it than the one obtained by directly applying EFPA to the original model.

The OPCA surrogate network model introduces a preemptive priority regime to calls
in the original overflow network model. For the purpose of approximation, a multi-level
traffic class (hierarchical) structure based on the number of overflows of each call is con-
sidered for each server group of the surrogate. This structure can be viewed as an ex-
tension of EFPA. Like EFPA, OPCA is based on the principle of decoupling a system
into multiple independent $M/M/K/K$ subsystems, making the independence and Poisson
assumptions.

OPCA does not impose any sort of hierarchy on servers. In the surrogate network
model, servers remain non-hierarchical, and one server cannot be distinguished from an-
other in terms of hierarchy. Instead, OPCA stratifies calls based on the number of times a
call has overflowed, or equivalently, on the priority of a call. Calls that have overflowed
zero times reside at the top of this virtual hierarchy because they can preempt all other
calls. Calls that have overflowed once compose the second tier of this hierarchy because
they can preempt all other calls, except calls that compose the top tier. The hierarchy
progresses as such according to the number of times a call has overflowed. The low-
est tier of the hierarchy comprises those calls that have overflowed to all servers in the
network. As such, OPCA imposes a virtual hierarchy (through traffic, not servers) on
a non-hierarchical network, that is, instead of the original non-hierarchical network that
suffers from mutual overflow, under OPCA, we now have a hierarchical structure that
avoids mutual overflow-related errors.

OPCA is only relevant to network with alternate routing (overflow network). If the
original network model is based on fixed routing, that is, no overflow is allowed, then
OPCA is reduced to EFPA because in the surrogate network model, all calls belong to
a single priority class, and blocking probability is obtained by EFPA. EFPA yields an
asymptotically exact solution for many channels per trunk in fixed routing models and
for many trunks and routes in diverse routing models [23]. Therefore, OPCA, which is equivalent to EFPA under these circumstances, also provides an asymptotically exact solution.

1.2 Thesis Outline

The subject of this thesis is the blocking probability analysis of circuit-switched networks. Approximation methods are proposed for accurate and robust evaluation of blocking probabilities in different scenarios. We compare the performance of these methods in various different scenarios and parameter ranges.

In Chapter 2, we consider a circuit-switched network with non-hierarchical alternate routing and trunk reservation involving two types of connections modeled as long-lived and short-lived calls. Long-lived connections can be reserved well in advance, and short-lived connections are provided on demand. Therefore, we assume that long-lived connections have strict priority over the short-lived ones. We develop approximations for the estimation of blocking probability in two ways based on the quasi-stationary approach. One uses EFPA, and the other uses OPCA. We compare the results of the approximations with simulation results and discuss the accuracy of the approximations under different system parameters, such as the ratio of offered load, the number of links per trunk, the maximum allowable number of deflections, and trunk reservation. We also discuss the robustness of quasi-stationary approximation to the ratio of the mean holding times of the long-lived and short-lived traffic streams as well as that of EFPA and OPCA to the shape of the holding time distribution. Finally, we demonstrate that OPCA can be applied to a large network, such as Coronet.

The ever-growing Internet and mounting evidence of the important role of circuit switching increase the need for an accurate and scalable means for the performance evaluation of large circuit-switched networks. EFPA achieves accurate results for such net-
works where the number of channels (circuits) per link is large. However, a conventional application of EFPA for large networks is computationally prohibitive. In cases where the EFPA solution is unattainable, we propose the use of asymptotic approximation, which we call A-EFPA, for link blocking probability in Chapter 3 and demonstrate computation time savings of many orders of magnitude for blocking probability approximation in realistically sized networks with many circuits per link. We demonstrate accurate calculations of blocking probability for NSFNet and Internet2 through simulations, namely, EFPA and A-EFPA, each of which is used for a different range of parameter values. We also present the results of the three methods in Coronet.

In Chapter 4, we consider a circuit-switched multi-service multi-rate network with non-hierarchical deflection routing and trunk reservation. Based on the fundamental concept of OPCA, we develop two approximations for the estimation of blocking probability, OPCA, and service-based OPCA. We also apply classical EFPA for the estimation of blocking probability in the network and propose max(EFPA, service-based OPCA) as an accurate and normally conservative evaluation. We compare the approximations with simulation results and discuss the accuracy of the blocking probabilities of the classes under different system parameter values, such as service rates, bandwidth requirements, number of links per trunk, maximum allowable number of deflections, and trunk reservation. We also discuss the robustness of the approximations to the shape of the holding time distribution and their performances under asymmetrical cases. We also present that the approximations can be applied to large networks, such as Coronet. We demonstrate that based on testing over a wide range of parameter values, max(EFPA, service-based OPCA) gives a very accurate and conservative estimation of network blocking probability in a multi-service multi-rate network.

Chapter 5 concludes the thesis and discusses future work.
1.3 Contributions

The contributions of the thesis are as follows.

1. Beyond the work of \cite{21, 23, 30, 53, 57} that considered single-priority circuit-switched networks, we have considered a circuit-switched network with long-lived and short-lived connections where the long-lived connections can preempt the short-lived ones. We used EFPA and OPCA combined with the quasi-stationary approximation to estimate the blocking probabilities and provided intuitive explanations for the performance of the approximation methods (Chapter \ref{chap:2}).

2. Conventional application of EFPA \cite{17, 21, 23, 25, 30, 45-52} for large networks is computationally prohibitive. We proposed an asymptotic approximation for the evaluation of network blocking probability in circuit switched networks for the cases where the EFPA solution is unattainable. We demonstrated savings of many orders of magnitudes in computation time for blocking probability approximation in realistically sized networks with large number of circuits per link (Chapter \ref{chap:3}).

3. We extended the work of \cite{53, 57} that studied single-class circuit-switched networks and considered multi-service multi-rate circuit-switched networks. Based on the fundamental concept of overflow priority classification approximation (OPCA), we developed two approximations for the estimation of the blocking probability, OPCA and service-based OPCA. We also applied the classical Erlang fixed-point approximation (EFPA) for the estimation of the blocking probability in the network and proposed max(EFPA, service-based OPCA) as an accurate and normally conservative evaluation (Chapter \ref{chap:4}).

1.4 Publications

My publications and submissions are listed as follows.


Chapter 2

Circuit Switched Networks with long-lived and short-lived connections

2.1 Introduction

Circuit switching has been widely used in telephony and is expected to have a renewed and important role in future optical networks [13, 14, 16, 59]. Circuit switching normally does not require buffering, which is very costly in the optical domain. If traffic on a circuit-switched network is sufficiently heavy and well managed, such a network can guarantee QoS to customers in a way that can lead to efficient trunk utilization and low consumption of energy per bit [60]. In the core Internet, where traffic is heavily multiplexed, high utilization is easy to achieve. Therefore, the role of circuit switching at the core is clearly important. Circuit switching can also lead to a green and efficient operation end-to-end for large bursts of data if the amount of data to be transmitted is known in advance.

Most published analyses of circuit-switched networks assume that holding times are independent and follow identical exponential distribution. In this chapter, we retain the assumption of independent holding times but assume that circuit-switched connections are classified into two types, long-lived (static) and short-lived (dynamic). The holding
times of long-lived connections are significantly longer than those of short-lived connections. This assumption is justified by the fact that variations in holding times in circuit switching-based optical networks can be very wide. Permanent or semi-permanent connections between major cities, which are used to serve many flows of multiple users over a long period of time, can be considered long-lived, where holding time can be in terms of hours. By contrast, on-demand short-lived connections between individual OD pairs of users in the order of seconds or less may be classified as dynamic.

The two types of traffic are expected to coexist in future optical networks. The classification of connections into long-lived and short-lived has been discussed in various papers (see e.g. [61], [62], [63]). Long-lived circuit-switched connections are likely to have priority over short-lived ones because they can be booked well in advance. Thus, their performance will not be affected by the loading of short-lived circuit-switched connections. The assumption of strict priority of long-lived circuit-switched connections also makes blocking probability tractable, and as we demonstrate here, the approximation method by using OPCA can also be accurate, robust, and computationally efficient. These assumptions justify the use of so-called quasi-stationary approximation [64–67] and are suitable in cases where changes in system states observed by one type of traffic because of changes in other traffic types are rare.

The problem of the blocking probability evaluation of circuit-switched networks with alternate routing and multiple priorities assigned to long-lived and short-lived calls has not been studied before. In this chapter, we apply both OPCA and EFPA to evaluate the blocking probability of the two traffic types in a circuit-switched network with alternate routing. We compare the results obtained against simulation benchmarks and explain their performance in various scenarios and parameter ranges. We then discuss insights into performance tradeoffs as well as design and dimensioning implications.

The remainder of the chapter is organized as follows. In Section 2.2 we describe our network model in detail and define notation and basic concepts. In Section 2.3 we
describe in detail the two approximations, EFPA and OPCA, as applied to our circuit-switched network model. In Section 2.4, we provide numerical results over a wide range of parameters for two network topologies (fully meshed and NSF) and discuss performance and design implications. We also discuss and illustrate the effects of the ratio between the offered traffic of the two priority classes, the number of links per trunk, the maximum allowable alternate paths, and trunk reservation as well as the accuracy of quasi-stationary approximation and the sensitivity of the shape of the holding time distribution. The chapter is concluded in Section 2.5.

2.2 The Model

We consider a circuit-switched network described by a graph \( G(N, E) \) where \( N \) is a set of \( n \) nodes and \( E \) is the set of \( e \) arcs. The \( e \) arcs correspond to trunks where trunk \( i \in E \) carries \( C(i) \) links. The \( N \) nodes are designated \( 1, 2, 3, \ldots, N \); each node has circuit switching capabilities. We assume that all the nodes have full wavelength conversion capabilities and can switch traffic from any link on one trunk to any other link on an adjacent trunk.

In the context of a core WDM network, a wavelength channel can be viewed as a link. In this case, trunk \( i \in E \) is composed of \( f(i) \) fibers, each of which supports \( w(i) \) wavelengths. Accordingly, trunk \( i \in E \) carries \( C(i) = f(i)w(i) \) wavelength channels called links. However, if the WDM network is further extended to the metropolitan or local areas, a link can have a sub-wavelength capacity [68]. The assumption that all nodes can switch traffic from any link on one trunk to any other link on another trunk in the WDM context implies that we assume that all switches have full wavelength conversion capabilities. In principle, our model can be extended to exclude this assumption, as we can, in our model, split every switch and trunk to multiple “sub-switches” and “sub-trunks”, each of which is dedicated to one color wavelength. We then increase the number of allowable alternate routes by a factor of the number of wavelengths. However, this increase implies signifi-
cantly increased computational complexity of our solutions as the graph that describes the network and the number of alternate routes significantly increases. Through simulations, Katib and Medhi [11] studied the tradeoff between alternate routing and the number of converters for single-priority networks. This chapter focuses on developing accurate approximations for blocking probability for networks with alternate routing and long-lived and short-lived traffic streams.

Let $\Gamma$ be a set of directional Source-Destination (SD) pairs. Every directional SD pair $m \in \Gamma$, is defined by its end-nodes. Thus, $m = \{s,d\} \in \Gamma$ represents the directional SD pair $s$ to $d$. We distinguish between the term SD pair which is an unordered set of the two endpoints: Source and Destination, and the directional SD pair that refers to the ordered set: Source-Destination.

The calls are classified according to their priority $p$ ($p = 1, 2$). long-lived calls ($p = 1$) have preemptive priority over short-lived calls ($p = 2$). For each directional SD pair $m \in \Gamma$, calls of priority $p$ arrive according to a Poisson process with an arrival rate of $\lambda(m,p)$. The holding times of calls are assumed to be exponentially distributed with a mean of $1/\mu(m,p)$. We assume that holding times are exponentially distributed for tractability. However, it is well known that in loss systems, blocking probability is highly insensitive to the shape of the holding time distribution and only dependent on the mean value of the holding times. This has been proven for the $M/G/K/K$ system. In Section 2.4, we demonstrate numerically that the blocking probability of our model is also insensitive to the shape of the holding times in Subsection 2.4.1. Let

$$\rho(m, p) = \frac{\lambda(m, p)}{\mu(m, p)}$$

be the offered traffic (measured in erlangs) for directional SD pair $m$. We set

$$\rho(p) = \sum_{m \in \Gamma} \rho(m, p).$$
A route between source $s$ and destination $d$ is the sequence of trunks associated with the corresponding arcs in the path between $s$ and $d$ in $G(N, E)$. A path between $s$ and $d$ comprises a sequence of trunks, one on each trunk on the route between $s$ and $d$.

It is very likely that for a directional SD pair $m \in \Gamma$, there are multiple routes between the source and the destination that do not share a common trunk. Such routes are often called *edge-disjoint paths* or *disjoint paths* [69–71]. Edge-disjoint alternate routing is often used to achieve load balancing in optical and other networks [72,73]. Although using disjoint path reduces blocking probability, non-disjoint paths are often used in practice. Our approximation methods are also applicable in the cases where paths are not disjoint. However, at this stage the strong dependency between trunks that may increase network blocking probability causes our approximation methods to underestimate the blocking probability. The development of algorithms to improve accuracy in the case of non-disjoint paths is still an open problem. In this chapter, we consider only disjoint paths in the numerical examples presented in Section 2.4.

For each $m \in \Gamma$, we designate a route with the least number of hops as the primary path $U(m, 0)$ of the directional SD pair $m$. If there are multiple routes with the least number of hops, for tractability, the choice is made randomly with equal probabilities. In practice, this choice may have the advantage of keeping the routing table unchanged. Considering a new topology from which the trunks of the primary path are excluded, the first alternative path for $m$ is chosen to minimize the number of hops in the new topology. Again, a tie is broken randomly. Therefore, the primary path and alternative paths for $m$ are edge-disjoint. Let $T_m$ be the maximum number of available alternative paths a directional SD pair $m$ can have based on the network topology.

Furthermore, a maximum number $D$ of overflow attempts to alternate paths are set for calls from all directional SD pairs in $\Gamma$. Setting the limit $D$ implies that a call of the
directional SD pair $m$ can use only

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

alternative paths. Therefore, before a call is blocked, the procedure continues until all available and allowable $T(m)$ routes are attempted.

It is convenient to maintain the entire set

$$\{U(m,0), U(m,1), \ldots, U(m,T(m))\}$$

of alternative routes for the directional SD pair $m \in \Gamma$ where $U(m,0)$ is the primary path and $U(m,d)$ is the $d$th alternate path. This option allows for cases where $D$ does not limit the number of usable alternative paths.

In our model, the ranking of alternative paths is based on the number of hops and in the case of equality in the number of hops, the rank is chosen randomly. Based on our ranking, if $d(u) > d(v)$, then the number of hops of $U(m,d(u))$ is equal to or higher than the number of hops in $U(m,d(v))$. However, in practice, other cost functions (e.g. geographic distance) can also be used for the ranking.

If a request for a call arrives at source node $s$ to the destination node $d$, and capacity is available on all trunks of the primary path $U(\{s,d\},0)$, then this primary path is used for the transmission of this call.

An arriving call of any type can use any free link on any trunk. When a long-lived call arrives, it can obtain a path on the primary path if no trunk of its primary path has all the links used by long-lived calls. In this case, if the call finds that all the links in any trunk on its primary path are busy, it will preempt a randomly chosen short-lived call. The preempted short-lived call then releases its resource to the long-lived call and overflows to its next alternate path. If the arriving high priority long-lived call finds that all the links are occupied by higher priority calls on at least one of the trunks of the primary path, it
will attempt a route on the first alternate path. In such a case, the long-lived call is said to have *overflowed* from its primary path and to attempt its first alternate path. The same procedure is repeated until the long-lived call exhausts all its \( D \) alternate path attempts. If the call still cannot obtain a path, it will be blocked and cleared from the network.

When a short-lived call arrives, it can obtain a path on the primary path if no trunk of its primary path has all the links used by either long-lived or short-lived calls. Otherwise the call overflows to its first alternate path. The same procedure is repeated. If the call can not obtain a path during its \( D \) alternate path attempts, the call is blocked and cleared from the network.

Call reattempts can affect blocking probability. However, normally our intention is to dimension the network so that the blocking probability is maintained below a certain small value. In this case, the proportion of call reattempts in the total arriving calls is small and their effect on blocking probability is negligible.

Considering the stability of the network, and the fact that fewer resources are used by a call that uses its primary path, priority is given to such calls. To facilitate such priority, a certain number of unoccupied links are reserved for calls that attempt their primary path. In particular, if the number of links on trunk \( j \) which is being used to transmit data is no less than a given reservation threshold \( RT(j, p) \), the overflowed calls of priority \( p \) are not allowed to use that trunk. In the chapter, we use trunk reservation to reserve a certain number of links to primary path connections to avoid large number of overflowed connections in the network that may cause instability. In all the numerical examples that are presented in the chapter (excluding the case of the large Coronet network discussed in Subsection 2.4.11), both EFPA and OPCA converge to a unique solution.
2.3 Blocking Probability Approximations

In this section, we describe the approximations we use for blocking probability evaluation for the long-lived and short-lived streams. We use the term 0-call for a call transmitted on its primary path, and the term $d$-call for a call transmitted on its $d$th alternate path, for $d = 1, 2, \ldots, T(m)$. Accordingly, the term $(d, m, p)$-call refers to a $d$-call of priority $p$, $p = 1, 2$ and directional SD pair $m$, in which for long-lived traffic $p = 1$ and for short-lived traffic $p = 2$. Assume that the arrivals of the $(d, m, p)$-calls follow a Poisson process with rate $a(d, m, p)$ and the arrivals of the $(d, m, p)$-calls at trunk $j \in U(m, d)$ also follow a Poisson process with rate $a(d, m, p, j)$. If $j$ is the first trunk on the path of the $(d, m, p)$-calls, then $a(d, m, p, j) = a(d, m, p)$. Let $b(d, j, p)$ be the blocking probability for priority $p$ $d$-calls on trunk $j \in E$.

The $(d, m, p)$-calls occur only when $(d - 1, m, p)$-calls are blocked. Therefore, we have

$$a(d, m, p) = a(d - 1, m, p)(1 - \prod_{j \in U(m, d - 1)} (1 - b(d - 1, j, p)))$$ \hspace{1cm} (2.1)

and $a(0, m, p) = \rho(m, p)$. For a particular trunk along the path $j \in U(m, d)$, we have

$$a(d, m, p, j) = a(d, m, p) \frac{\prod_{i \in U(m, d)} (1 - b(d, i, p))}{1 - b(d, j, p)}$$ \hspace{1cm} (2.2)

for $d = 0, 1, \ldots, T(m)$. For $d > T(m)$ or $j \notin U(m, d)$, $a(d, m, p, j) = 0$.

Let $a(d, j, p)$ be the total offered load of priority $p$ $d$-calls, on trunk $j$. The variables $a(d, j, p)$ and $a(d, m, p, j)$ are related by

$$a(d, j, p) = \sum_{m \in \Gamma} a(d, m, p, j).$$ \hspace{1cm} (2.3)
Also, let \( \bar{a}(d, j, p) \) be the total offered load of priority \( p \) calls that include 0-calls, 1-calls, 2-calls \( \ldots \) \( d \)-calls, on trunk \( j \). The variables \( \bar{a}(d, j, p) \) and \( a(d, j, p) \) are related by

\[
\bar{a}(d, j, p) = \sum_{i=0}^{d} a(i, j, p).
\]

(2.4)

Since long-lived traffic has preemptive priority over short-lived traffic, the blocking probability of higher priority long-lived traffic can be evaluated as if it were alone in the network. In other words, it is sufficient to consider a network with a single class of traffic and EFPA and OPCA can be directly applied to it for the estimation of the blocking probability of long-lived traffic. For lower priority short-lived traffic, the available capacity is the leftover of long-lived carried traffic and therefore, the blocking probability of short-lived calls in a trunk is dependent on the number of links occupied by long-lived calls in the trunk. To evaluate the blocking probability of short-lived traffic, we use quasi-stationary approximation in both EFPA and OPCA and calculate the conditional blocking probability of short-lived traffic for each state of long-lived traffic occupancy and then compute the weighted average of these probabilities using the stationary distribution of the long-lived traffic link occupancy.

Table 2.1: Summary of Notation

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>( d )-call</td>
<td>a call transmitted on its ( d )th alternate path</td>
</tr>
<tr>
<td>((d, m, p))-call</td>
<td>( d )-call of priority ( p ) (( p = 1, 2 )) and directional SD pair ( m )</td>
</tr>
<tr>
<td>( p(m, p) )</td>
<td>offered load of SD pair ( m ) and priority ( p )</td>
</tr>
<tr>
<td>( a(d, m, p) )</td>
<td>offered load of ((d, m, p))-calls</td>
</tr>
<tr>
<td>( a(d, m, p, j) )</td>
<td>offered load of ((d, m, p))-calls on trunk ( j )</td>
</tr>
<tr>
<td>( a(d, j, p) )</td>
<td>total offered load of priority ( p ) ( d )-calls on trunk ( j )</td>
</tr>
<tr>
<td>( \bar{a}(d, j, p) )</td>
<td>total offered load of priority ( p ) up to ( d )-calls on trunk ( j )</td>
</tr>
<tr>
<td>( b(d, j, p) )</td>
<td>blocking probability for priority ( p ) ( d )-calls on trunk ( j )</td>
</tr>
<tr>
<td>( B(m, p) )</td>
<td>blocking probabilities for priority ( p ) traffic from SD pair ( m )</td>
</tr>
<tr>
<td>( B(p) )</td>
<td>network blocking probability for priority ( p ) traffic</td>
</tr>
</tbody>
</table>
2.3.1 EFPA

To estimate the network blocking probability for the long-lived traffic by EFPA, we begin by setting initial randomly chosen Uniform\([0,1)\) values to trunk blocking probabilities. The probability that a call is admitted on a trunk is 1 minus the trunk blocking probability. With these initial values, we treat the trunks as if they were independent. The probability that a call is admitted to the primary path is the product of the probability that a call is admitted to all the individual trunks along the path. The primary path blocking probability is 1 minus the probability that a call is admitted to the path. The carried traffic on the primary path for a particular SD pair is the unblocked proportion of the offered traffic, which is also the carried traffic on each trunk along the path attributed to this SD pair. Accordingly, the offered load of the SD pair to a trunk is obtained by the carried traffic attributed to that SD pair divided by 1 minus the trunk blocking probability.

The traffic overflowed to the first alternative path is the traffic offered to the primary path, multiplied by the primary path blocking probability. Similarly, traffic offered to other alternative paths can be obtained. Having obtained the traffic offered to every SD pair that uses the trunk, the total traffic offered to the trunk is their sum. Then, we calculate the steady-state probabilities of each trunk in the network and update the trunk blocking probabilities with the ones obtained based on the state probabilities. All these steps compose one iteration of the fixed-point equations. The iterations continue until the difference of the trunk blocking probabilities of two successive iterations is less than a given value. Having obtained the trunk blocking probabilities, we can obtain network blocking probability for the long-lived traffic.

Let \(q(j, 1, i)\) be the steady-state probability of having \(i\) links busy with long-lived traffic in trunk \(j\). We then evaluate the trunk state probability \(q(j, 1, i)\), for each \(j\) and \(i \in \{1, \ldots, C(j)\}\) by
\[ q(j, 1, i) = \left( a(0, j, 1) + 1 \{ RT(j, 1) > i - 1 \} \sum_{n=1}^{D} a(n, j, 1) \right) \times q(j, 1, i - 1) / i, \tag{2.5} \]

where \( 1 \{ \} \) is the indicator function and \( q(j, 1) \) is set such that \( \sum_{i=0}^{C(j)} q(j, 1, i) = 1 \) is satisfied. The blocking probability, for the long-lived traffic with \( d \) overflows, on trunk \( j \) is estimated by

\[
b(d, j, 1) = \begin{cases} 
q(j, 1, C(j)) & d = 0, \\
\sum_{i=RT(j,1)}^{C(j)} q(j, 1, i) & d \geq 1.
\end{cases} \tag{2.6}
\]

To evaluate the blocking probability for the lower priority short-lived traffic, we use quasi-stationary approximation (in the sense of \([64–67]\)). Such an approximation is often used when the variations in the system state observed by one type of traffic are very rare. In our case, as the holding times of long-lived calls are far longer than those of short-lived calls, short-lived calls only rarely observe changes in their service rate during their holding time and can approximately reach steady-state while the number of long-lived calls remains unchanged. Under such conditions, accurate approximation for the blocking probability for the short-lived calls can be obtained by computing the short-lived traffic blocking probability for each state of the long-lived traffic trunk occupancy and then by computing the weighted average of these probabilities with the stationary distribution of the long-lived traffic trunk occupancy. A more detailed description of this approximation for our case follows.

To obtain the network blocking probability approximation for the short-lived traffic by EFPA, the procedure is similar to that for the long-lived traffic. The difference is that the steady-state probabilities and blocking probabilities of the short-lived traffic are conditional on the number of links occupied by the long-lived calls on that trunk.

Let \( b(d, j, k, 2) \) be the blocking probabilities for the short-lived calls with \( d \) overflows for each trunk \( j \) when there are \( k \in \{0, \ldots, C(j)\} \) links free in trunk \( j \).
First, set: \( b(d,j,0,2) = 1 \), for each \( d \), and also for \( d = 1,2,\ldots,D \), when \( k \leq C(j) - RT(j,2) \), set \( b(d,j,k,2) = 1 \).

Next, to evaluate other \( b(d,j,k,p) \) values, we evaluate the trunk state probability \( q(j,k,2,i) \) for each trunk \( j \) and each state \( i \in \{1,\ldots,k\} \) using

\[
q(j,k,2,i) = \left( a(0,j,2) + 1 \{ R(j,k) > i - 1 \} \sum_{n=1}^{D} a(n,j,2) \right) \times q(j,k,2,i-1)/i, \tag{2.7}
\]

where \( R(j,k) = RT(j,2) + k - C(j) \) and \( q(j,k,2,0) \) is set such that \( \sum_{i=0}^{k} q(j,k,2,i) = 1 \) is satisfied.

Then we obtain

\[
b(d,j,k,2) = \begin{cases} 
1 & k = 0 \text{ or } R(j,k) \leq 0, \\
q(j,k,2,k) & d = 0 \text{ and } R(j,k) > 0, \\
\sum_{i=R(j,k)}^{k} q(j,k,2,i) & d \geq 1 \text{ and } R(j,k) > 0.
\end{cases} \tag{2.8}
\]

The blocking probability, for the short-lived traffic with \( d \) overflows, on trunk \( j \) is estimated by

\[
b(d,j,2) = \sum_{i=0}^{C(j)} q(j,1,i) \times b(d,j,C(j) - i,2). \tag{2.9}
\]

Equations (2.1) – (2.9) form a set of fixed-point equations that can be solved by successive substitutions.

Having obtained the results of the fixed-point equations, we calculate the blocking probabilities for the long-lived traffic \((p = 1)\) and short-lived traffic \((p = 2)\) from SD pair \( m \) by

\[
B(m,p) = 1 - \sum_{d=0}^{D} a(d,m,j,0)(1 - b(d,j,p))(1 - b(d,j,p)))/p(m,p), \tag{2.10}
\]

where \( j \) is the last trunk in the route for the calls of SD pair \( m \) that overflow \( d \) times. Let \( B(p) \) be the network blocking probability for long-lived traffic \((p = 1)\) and short-lived
traffic \((p = 2)\), which is the average of blocking probabilities of all SD pairs, weighted by their offered load.

\[
B(p) = \sum_{m \in \Gamma} B(m, p) \times \rho(m, p) / \sum_{m \in \Gamma} \rho(m, p).
\] (2.11)

Algorithm 1 is used to obtain the network blocking probability \(B(1)\) for the long-lived traffic.

**Algorithm 1** Compute \(B(1)\) and \(q(j, 1, i)\) by EFPA

**Require:** \(\rho(m, 1)\) for \(m \in \Gamma\)

**Initial:** \(b(d, j, 1) \leftarrow 0, \hat{b}(d, j, 1) \leftarrow 1\) for \(j \in \mathcal{E}, d \in \{0, \ldots, D\}\)

while \(\sum_{d \in \{0, \ldots, D\}} \sum_{j \in \mathcal{E}} |b(d, j, 1) - \hat{b}(d, j, 1)| > \text{error}\) do

for \(j \in \mathcal{E}, d \in \{0, \ldots, D\}, m \in \Gamma\) do

\(\hat{b}(d, j, 1) \leftarrow b(d, j, 1)\)

compute \(a(d, m, 1)\) in Eq. (2.1)

compute \(a(d, m, j, 1)\) in Eq. (2.2)

compute \(a(d, j, 1)\) in Eq. (2.3)

for \(i \in \{1, \ldots, C(j)\}\) do

compute \(q(j, 1, i)\) in Eq. (2.5)

end for

compute \(b(d, j, 1)\) in Eq. (2.6)

end for

end while

for \(m \in \Gamma\) do

compute \(B(m, 1)\) in Eq. (2.10)

end for

compute \(B(1)\) in Eq. (2.11)

The relative error is a parameter set to measure the difference of the substitution results and the iteration stops when

\[
\sum_{j \in \mathcal{E}} |b(d, j, 1) - \hat{b}(d, j, 1)| < \text{error}.
\] (2.12)

In this chapter, we set

\[
\text{error} = 10^{-8}.
\] (2.13)
Algorithm 2 describes the computation of the network blocking probability for the short-lived traffic which is dependent on the state probability $q(j, 1, i)$ computed by Algorithm 1.

**Algorithm 2** Compute $B(2)$ by EFPA

Require: $\rho(m, 1), \rho(m, 2)$ for $m \in \Gamma$, $q(j, 1, i)$ for $j \in \mathcal{E}$ and $i \in \{1, \ldots, C(j)\}$

initial: $\hat{b}(d, j, 2) \leftarrow 1$ for $j \in \mathcal{E}, d \in \{0, \ldots, D\}$

while $\sum_{d \in \{0, \ldots, D\}} \sum_{j \in \mathcal{E}} |b(d, j, 2) - \hat{b}(d, j, 2)| > \text{error}$ do

for $j \in \mathcal{E}, d \in \{0, \ldots, D\}, m \in \Gamma$ do

\[ \hat{b}(d, j, 2) \leftarrow b(d, j, 2) \]

compute $a(d, m, 2)$ in Eq. (2.1)

compute $a(d, m, j, 2)$ in Eq. (2.2)

compute $a(d, j, 2)$ in Eq. (2.3)

for $k \in \{1, \ldots, C(j)\}, i \in \{1, \ldots, k\}$ do

compute $q(j, k, 2, i)$ in Eq. (2.7)

compute $b(d, j, k, 2)$ in Eq. (2.8)

end for

compute $b(d, j, 2)$ in Eq. (2.9)

end for

end while

for $m \in \Gamma$ do

compute $B(m, 2)$ in Eq. (2.10)

end for

compute $B(2)$ in Eq. (2.11)

---

2.3.2 OPCA

OPCA imposes a fictitious preemptive priority structure to a given model of a real network to yield a surrogate network model and applies an EFPA-like algorithm to estimate the blocking probability for the model. To generate the surrogate network model, calls are classified according to how many times they have overflowed. Preemptive priority is assigned to those that have overflowed fewer times and those that have overflowed more are preempted and overflowed to other paths, carrying “information” about busy paths that they have already tried. In this way, the surrogate system operates as a hierarchical network where the traffic is strictly prioritized and layered according to how many times
it overflows. Note that the preemptive priority in the surrogate model is artificially introduced to the original model, in which there is no priority exists among calls [53, 56–58].

The purpose of introducing preemptive priority to the real network model is for a better estimation of the network blocking probability. In many cases, applying EFPA to the surrogate model obtains more accurate estimation of the network blocking probability than applying EFPA to the original model directly. The reasons are as following.

1. OPCA avoids the adverse effects of mutual overflow on the accuracy of the approximation.

2. OPCA increases the proportion of the 0-calls in the system and reduces the overflowed traffic. Since 0-calls do not violate the Poisson and independence assumptions, increasing the proportion can reduce the Poisson and independence errors.

3. Having more primary traffic reduces also the independence errors.

As mentioned in Section 1.3, in this chapter, we extend the work of [21, 23, 30, 53, 57], which studied single-priority circuit-switched networks and consider a circuit-switched network with long-lived and short-lived connections where the long-lived connections can preempt the short-lived ones. The procedure of the blocking probability calculation by OPCA is similar to that of EFPA and the difference between the two is due to the preemptive priority of the surrogate model of OPCA. In OPCA, we first solve the fixed-point equations considering only traffic offered to the primary path that has not overflowed yet, and obtain the network blocking probability for it. We then calculate the total traffic comprises the traffic that has not overflowed and the traffic that has overflowed once. We again solve the fixed-point equations and calculate the blocked portion of the total traffic. By subtracting the traffic that was blocked once in the primary path from the total blocked traffic, we obtain the traffic that is blocked twice, which gives the blocking probability for the traffic that has overflowed once. The blocking probabilities of traffic that has overflowed more than once are obtained recursively in a similar way.
In the following we provide detailed information on how to apply OPCA to the present problem of approximating the blocking probability of circuit switched networks for the long-lived and short-lived traffic.

We begin by evaluating the trunk state probability \( t(d, j, p, i) \) of long-lived traffic \( (p = 1) \) for each trunk \( j \), for \( d \) deflections and each state \( i \in \{1, \ldots, C(j)\} \) using

\[
t(d, j, 1, i) = \left( a(0, j, 1) + 1 \{RT(j, 1) > i - 1\} \sum_{n=1}^{d} a(n, j, 1) \right) \times t(d, j, 1, i - 1)/i, \quad (2.14)
\]

where \( t(d, j, 1, 0) \) is set such that \( \sum_{i=0}^{C(j)} t(d, j, 1, i) = 1 \) is satisfied.

The average blocking probability \( \bar{b}(d, j, 1) \) on trunk \( j \), for the long-lived calls with up to and including \( d \) overflows, is estimated by

\[
\bar{b}(d, j, 1) = \sum_{n=1}^{d} \left( a(n, j, 1) \sum_{i=RT(j, 1)}^{C(j)} t(d, j, 1, i) \right) / \bar{a}(d, j, 1) + \frac{a(0, j, 1)t(d, j, 1, C(j))}{\bar{a}(d, j, 1)} \quad (2.15)
\]

and \( \bar{b}(0, j, 1) \) is calculated by the Erlang-B formula, i.e. \( \bar{b}(0, j, 1) = E(a(0, j, 1), C(j)) \).

The blocking probability for the long-lived traffic, for \( d \)-overflows calls, on trunk \( j \) is estimated by

\[
b(d, j, 1) = \begin{cases} \bar{b}(0, j, 1) & d = 0, \\ \frac{\bar{b}(d, j, 1)\bar{a}(d, j, 1) - \bar{b}(d-1, j, 1)\bar{a}(d-1, j, 1)}{\bar{a}(d, j, 1)} & 1 \leq d \leq D. \end{cases} \quad (2.16)
\]

Note that the blocking probability for the unoverflowed calls is calculated by the Erlang-B formula. With the trunk blocking probability \( b(d, j, 1) \), the network blocking probability can be computed according to equations (2.10) and (2.11). Algorithm 3 is used to compute the network blocking probability for the long-lived traffic.

After calculating all the blocking probabilities for different layers for the long-lived traffic, we move on to calculate the blocking probabilities for the short-lived traffic.
Let \( h(j, i) \) for \( i \in \{1, \ldots, C(j)\} \) and each trunk \( j \) be the probability that there are \( i \) number of long-lived calls in trunk \( j \). So that

\[
h(j, i) = \left( a(0, j, 1) + 1 \{RT(j, 1) > i - 1\} \sum_{n=1}^{D} a(n, j, 1) \right) \times h(j, i - 1)/i \quad (2.17)
\]

where \( h(j, 0) \) is set such that \( \sum_{i=0}^{C(j)} h(j, i) = 1 \) is satisfied.

Let \( \overline{b}(d, j, k, 2) \) be the blocking probabilities for the short-lived calls with \( d \) overflows for each trunk \( j \) when there are \( k \in \{0, \ldots, C(j)\} \) links free in the trunk \( j \). For \( d = 0, 1, \ldots, D \), \( \overline{b}(d, j, 0, 2) = 1 \). For \( d = 1, 2, \ldots, D \), when \( R(j, k) = k - C(j) + RT(j, 2) \leq 0 \), \( \overline{b}(d, j, k, 2) = 1 \). To evaluate other \( \overline{b}(d, j, k, p) \), we evaluate the trunk state probability \( t(j, k, d, 2, i) \) for each trunk \( j \), each state \( i \in \{0, 1, \ldots, k\} \) and \( d \) using

\[
t(d, j, k, 2, i) = \left( a(0, j, 2) + 1 \{R(j, k) > i - 1\} \sum_{n=1}^{d} a(n, j, 2) \right) \times t(d, j, k, 2, i - 1)/i,
\]

where \( t(d, j, k, 2, 0) \) is set such that \( \sum_{i=0}^{k} t(d, j, k, 2, i) = 1 \) is satisfied, and \( t(0, j, k, 2, i) \) is estimated by using the Erlang-B formula, i.e. \( t(0, j, k, 2, k) = E(a(0, j, 2), k) \).

Then we obtain

\[
\overline{b}(d, j, k, 2) = \begin{cases} 
    t(0, j, k, 2, k) & d = 0, \\
    \sum_{i=R(j,k)}^{k} t(d, j, i, 2, i) & d \geq 1 \text{ and } R(j,k) > 0, \\
    1 & \text{otherwise.}
\end{cases}
\]

(2.19)
The averaged blocking probability, for the short-lived traffic with $d$ overflows, on trunk $j$ is estimated by

$$
\bar{b}(d, j, 2) = \sum_{i=0}^{C(j)} h(j, i) \times \bar{b}(d, j, C(j) - i, 2).
$$

(2.20)

The blocking probability for the short lived $d$-calls, on trunk $j$ is estimated by

$$
b(d, j, 2) =
\begin{cases} 
\bar{b}(0, j, 2) & d = 0, \\
\frac{\bar{b}(d, j, 2)\tilde{a}(d, j, 2) - \bar{b}(d-1, j, 2)\tilde{a}(d-1, j, 2)}{a(d, j, 2)} & 1 \leq d \leq D.
\end{cases}
$$

(2.21)

Then the network blocking probability can be computed by equations (2.10) and (2.11). Algorithm 4 is used to compute the network blocking probability $B(2)$ for the short-lived traffic.

### 2.4 Numerical Results

We begin this section by comparing the performance of OPCA with that of EFPA in approximating the blocking probability for the long-lived and short-lived traffic using simulations. The comparison is performed for a 6-node fully-meshed network and the 13-node National Science Foundation (NSF) network.

We then extend our comparison, to consider a range of scenarios and parameter values for each scenario. In all cases considered, we also provide intuitive explanation of the discrepancies between the two approximations and the simulation results for the blocking probability as it varies according to the various effects. In particular, we consider traffic effects such as offered load, and the difference between the two types of traffic, in terms of the offered load and mean holding times. The latter is especially important to observe the accuracy and sensitivity of quasi-stationary approximation.
Algorithm 3 Compute $B(1)$ and $h(j,i)$ by OPCA

Require: $\rho(m,1)$ for $m \in \Gamma$

for $d \in \{0, \ldots, D\}$ do
  initial: $b(d,j,1) \leftarrow 0, \hat{b}(d,j,1) \leftarrow 1$ for $j \in \mathcal{E}$
  while $\sum_{j \in \mathcal{E}} |b(d,j,1) - \hat{b}(d,j,1)| > \text{error}$ do
    for $j \in \mathcal{E}, m \in \Gamma$ do
      $\hat{b}(d,j,1) \leftarrow b(d,j,1)$
      compute $a(d,m,1)$ in Eq. (2.1)
      compute $a(d,m,j,1)$ in Eq. (2.2)
      compute $a(d,j,1)$ in Eq. (2.3)
      compute $\bar{a}(d,j,1)$ in Eq. (2.4)
      for $i \in \{1, \ldots, C(j)\}$ do
        compute $t(d,j,1,i)$ in Eq. (2.14)
      end for
      compute $\bar{b}(d,j,1)$ in Eq. (2.15)
      compute $b(d,j,1)$ in Eq. (2.16)
    end for
  end while
end for

for $m \in \Gamma$ do
  compute $B(m,1)$ in Eq. (2.10)
end for

compute $B(1)$ in Eq. (2.11)

for $j \in \mathcal{E}, i \in \{1, \ldots, C(j)\}$ do
  compute $h(j,i)$ in Eq. (2.17)
end for

Finally, we consider design factors such as the number of links per trunk, the maximum allowable number of alternate paths, and the effect of trunk reservation. We also discuss the robustness of the approximations to the shape of the holding time distribution.

In all scenarios considered, the arrival process of calls for each directional SD pair and each type of traffic follows a Poisson process and the total traffic offered to each directional SD pair is equal to $T$. The variable $T$ is our measure of traffic load for all cases in both topologies. The shortest path is set to be the primary route for each SD pair, and the alternate routes are pre-assigned ordered by their length. For those routes with the same lengths, the order is chosen randomly and remains unchanged afterwards.
Algorithm 4 Compute $B(2)$ by OPCA

Require: $\rho(m,1), \rho(m,2)$ for $m \in \Gamma$, $h(j,i)$ for $j \in E$ and $i \in \{1,\ldots,C(j)\}$

for $d \in \{0,\ldots,D\}$ do
  initial: $b(d,j,2) \leftarrow 0$, $\hat{b}(d,j,2) \leftarrow 1$ for $j \in E$
  while $\sum_{j \in E} |b(d,j,2) - \hat{b}(d,j,2)| > \text{error}$ do
    for $j \in E$, $m \in \Gamma$ do
      $\hat{b}(d,j,2) \leftarrow b(d,j,2)$
      compute $a(d,m,2)$ in Eq. (2.1)
      compute $a(d,m,j,2)$ in Eq. (2.2)
      compute $a(d,j,2)$ in Eq. (2.3)
      compute $\tilde{a}(d,j,2)$ in Eq. (2.4)
      for $k \in \{1,\ldots,C(j)\}$, $i \in \{1,\ldots,k\}$ do
        compute $t(d,j,k,2,i)$ in Eq. (2.18)
        compute $\bar{b}(d,j,k,2)$ in Eq. (2.19)
      end for
    end for
    compute $\tilde{b}(d,j,2)$ in Eq. (2.20)
    compute $b(d,j,2)$ in Eq. (2.21)
  end while
end for
for $m \in \Gamma$ do
  compute $B(m,2)$ in Eq. (2.10)
end for
compute $B(2)$ in Eq. (2.11)

2.4.1 Default parameter setting

In many experiments we repeatedly use the same set of parameters with possibly small variations. It is convenient to present them once in this section and through the section only point out the deviations from this default set. This set of parameters is henceforth referred to as Default parameter setting and is described as follows.

For both the 6-node fully meshed network and the NSF network, both long-lived and short-lived calls arrive according to a Poisson process and the ratio of offered long-lived traffic to that of short-lived traffic is 1 : 1. The holding times of both long-lived and short-lived calls are exponentially distributed and the mean holding time of long-lived traffic is 200 times higher than that of the short-lived traffic. The total number of links per trunk is 20. The threshold of long-lived traffic is 16 (80%) and the threshold of the short-lived
traffic is 18 (90%). The maximum allowable number of alternate paths is 4 for both long-lived and short-lived calls in the 6-node fully meshed network and 2 in the NSF network, respectively.

We have chosen to present the results, for each network, for the long-lived and short-lived traffic streams in two separate figures because they are different by several orders of magnitude, and if presented in the same figure as a function of traffic load, for reasonable traffic loads that give acceptable blocking probability to short-lived traffic, in many cases, the blocking probability for the long-lived traffic will be too low for accurate evaluation by simulation.

2.4.2 Blocking probability for the long-lived traffic

We evaluate here the blocking probability for the long-lived traffic by EFPA, OPCA and simulations. Since long-lived traffic has preemptive priority over short-lived traffic, the blocking probability for the long-lived traffic can be evaluated as if it were alone in the network. In other words, it is sufficient to consider a network with a single class of traffic. We note that the case of a single class of traffic was considered in [57]. Here we add examples, intuitive explanations and interpretation of the results.

At first, we consider a 6-node fully meshed network model. In such a network, there are in total 15 different SD pairs (or equivalently, 30 directional SD pairs). As the offered traffic for each directional SD pair is $T$, so the total offered traffic in the network is $30T$.

In Fig. 2.1 (a), we present results for the blocking probabilities obtained by OPCA, EFPA and simulations for long-lived traffic (single priority). We observe in the figure that EFPA and OPCA tend to underestimate blocking probability when the offered load is low. This is due to the fact that in a fully meshed network with low traffic load, and therefore fewer overflows, long paths are very rare. Accordingly, overflow error dominates path errors causing the underestimation of blocking probability.
Figure 2.1: Blocking probabilities for the long-lived traffic (single priority) versus $T$ in (a) a 6-node fully meshed network and (b) NSF network.

Since the surrogate model of OPCA gives preemptive priority to new calls, and therefore allocates more resources to the primary path traffic, this surrogate model has less overflow traffic than the original model approximated by EFPA, leading to less overflow error, less Poisson error, and therefore less underestimation of blocking probability for the long-lived traffic (single priority). Recall that OPCA is based on approximating the blocking probability of the surrogate model treating each link independently and assuming it is loaded by Poisson arrivals.

Furthermore, we observe that as the traffic load increases, the underestimation for both EFPA and OPCA of the blocking probability is reduced. This is consistent with the fact that in high load overflow probability increases, leading to overflow path length growth, and therefore path error increase. As observed, the path error in the cases of high traffic load may cancel out the overflow error to improve the approximation.

Next, we consider the NSF network with 13 nodes and 16 bidirectional trunks. As the number of SD pairs is $(13 \times 12/2) = 78$, the total offered traffic in this case is $156T$. The topology of the NSF network is shown in Fig. 2.2. The results for the blocking probabilities for the long-lived traffic as a function of $T$ is shown in Fig. 2.1(b). We obtain results that have similar behavior to those obtained for the 6-node fully meshed network case in the sense that OPCA outperforms EFPA. But since the number of alternate paths
in the NSF network is much less than that in 6-node fully meshed network, in the NSF network, OPCA outperforms EFPA less significantly than in the case of the 6-node fully meshed network.

2.4.3 Blocking probability for the short-lived traffic

Figures 2.3 (a) and (b) present the blocking probabilities obtained by OPCA, EFPA and simulations for short-lived traffic, as a function of $T$, in the same 6-node fully meshed and NSF networks described above. However, now we consider the two classes of traffic using the networks so $T$ is now the total offered traffic of both long-lived and short-lived
calls for each directional SD pair. The parameters are set as in the Default parameter setting in 2.4.1.

As expected, comparing Figs. 2.3 (a) and (b) with their long-lived traffic (single priority) counterparts Figs. 2.1 (a) and (b), respectively, we observe significantly lower blocking probability (by several orders of magnitude) for the long-lived traffic than for the short-lived traffic. Specifically, we observe the blocking probability in Fig. 2.3 (a) for a load of $T = 12$ versus the equivalent load of $T = 6$ in Fig. 2.1 (a). Also, for the NSF network, we compare the blocking probability in Fig. 2.3 (b) for load of $T = 0.5$ to that of Fig. 2.1 (b) for load of $T = 0.25$.

Furthermore, we observe that OPCA provides better approximations for both long-lived traffic and short-lived traffic in both networks than EFPA.

If the threshold of long-lived traffic is higher than or equal to that of short-lived traffic, since long-lived calls can preempt short-lived calls and not vice versa, the set of states in which long-lived calls are blocked is a strict subset of the set of states in which short-lived calls are blocked, and since both processes arrive in accordance with Poisson processes, blocking probability for the long-lived traffic must be lower than that of short-lived traffic even if the offered long-lived traffic is much larger than that of short-lived traffic. In general, alternate routing further benefits long-lived traffic, because longer alternate routes of long-lived traffic, that use more network resources per bit than primary path traffic, have a more detrimental effect on short-lived traffic than on long-lived traffic.

To provide some protection to short-lived traffic, we set the threshold of short-lived traffic higher than that of long-lived traffic so that if the offered short-lived traffic is much smaller than that of the long-lived traffic, the short-lived traffic may have a smaller blocking probability than long-lived traffic. Notice that this threshold difference does not affect the “right” of the long-lived traffic to behave as if it is alone in the system according to its own threshold limitation on overflowed calls. This limitation also provides some pro-
tection to short-lived traffic from overflowed long-lived traffic by using long alternative paths inefficiently.

However, in the present case, the offered short-lived traffic is the same as that of long-lived traffic, thus the blocking probability for the short-lived traffic is larger than that for the long-lived traffic.

The preemptive property of long-lived traffic affects the approximations of short-lived traffic in two distinct ways:

1. The capacity available to short-lived traffic is the leftover of long-lived carried traffic and short-lived traffic can be preempted by long-lived calls. Therefore, short-lived calls may be forced to take longer alternate routes, so the proportion of the overflow traffic in the total short-lived traffic is higher than for long-lived traffic, leading to higher overflow error and path error. For low traffic, since the overflow error is dominant, this higher proportion of overflow traffic in the total short-lived traffic will cause further underestimation of the blocking probability for the short-lived traffic for both EFPA and OPCA. On the other hand, in a high loading scenario, since the path error is dominant in high loading, this higher proportion causes further overestimation of the blocking probability for the short-lived traffic for both EFPA and OPCA.

2. Long-lived carried traffic, which can be viewed as long-lived background traffic for the short-lived traffic, exhibits dependencies among various trunks. More specifically, the congestion of long-lived traffic on one trunk is likely to cause congestion of long-lived traffic on other trunks. This congestion dependence of long-lived traffic (background traffic) on trunks in turn causes dependence in capacity limitation for short-lived traffic among different links which leads to congestion dependence of short-lived traffic. However, in the quasi-stationary approach, long-lived carried traffic are assumed to be independent. This assumption introduces another kind of dependence error, which does not occur for long-lived traffic, that causes a fur-
ther underestimation of the blocking probability for the short-lived traffic for both OPCA and EFPA. This effect causes further underestimation for low traffic for both EFPA and OPCA. As the total offered load increases and this effect tends to underestimate the blocking probability obtained by EFPA and OPCA, which cancels out the overestimation due to the first effect and leads to accurate predictions by both EFPA and OPCA.

2.4.4 The effect of the ratio between the offered long-lived traffic and offered short-lived traffic

Figure 2.4: Blocking probabilities for the short-lived traffic in a 6-node fully-meshed network. The ratio of the offered long-lived traffic to offered short-lived traffic is (a) 1:2, (b) 1:5, (c) 2:1, and (d) 5:1.
Figs. 2.3 (a) and 2.4 (a)-(d) show the blocking probabilities for the short-lived traffic when the ratios of the offered long-lived traffic to offered short-lived traffic are 1:1, 1:2, 1:5, 2:1 and 5:1, respectively while all the other parameters are kept the same as in the Default parameter setting in 2.4.1. As long-lived traffic is not affected by short-lived traffic, we only consider here the accuracy of OPCA and EFPA for short-lived traffic.

Several observations are made from Figs. 2.3 (a) and 2.4 (a)-(d):

1. OPCA is generally more accurate than EFPA.
2. The accuracy of OPCA in predicting blocking probability for the short-lived traffic decreases when the proportion of long-lived traffic increases.
3. EFPA accuracy is not significantly affected by the ratio of the two traffic types.
4. Both EFPA and OPCA accuracy increases with increased traffic load.

Observations 1 and 4 are consistent with what we have observed in the case when the proportion was 1:1, and the explanations above are applicable. To explain observations 2 and 3, recall that EFPA under light load suffers from the dependency error. This fact is invariant to the proportion between the offered long-lived traffic and offered short-lived traffic. However, OPCA is able to reduce the dependency error of short-lived traffic (notice that the quasi-stationary approximation assumes independence between the long-lived and short-lived traffic streams for both OPCA and EFPA). Therefore, if the short-lived traffic is reduced, the ability of OPCA to neutralize the dependence effects is also reduced.

Since OPCA has better performance in the cases of low short-lived loading and EFPA performs better in the cases of high loading, max\{OPCA,EFPA\} is the best approximation. In [58], we obtained a similar conclusion that max\{OPCA,EFPA\} is the best approximation for an optical burst switched network.

For the NSF network with 13 nodes and 16 trunks, the results for the blocking probabilities for the short-lived traffic are shown in Figs. 2.3 (b), 2.5 (a)-(d) when the ratios
of the offered long-lived traffic to offered short-lived traffic are 1:1, 1:2, 1:5, 2:1 and 5:1, respectively while all the other parameters are kept the same as in the Default parameter setting in 2.4.1. We obtain results that have similar behavior to those obtained for the 6-node fully meshed network case in the sense that OPCA slightly outperforms EFPA in the cases of low loading.

![Blocking probabilities for short-lived traffic in NSF network.](image)

Figure 2.5: Blocking probabilities for the short-lived traffic in NSF network. The ratio of the offered long-lived traffic to offered short-lived traffic is (a) 1:2, (b) 1:5, (c) 2:1, and (d) 5:1.

### 2.4.5 The effect of the number of links on each trunk

To examine the effect of the number of links (wavelength channels) on each trunk on blocking probability and on the accuracy of EFPA and OPCA, we increase now the number of links on each trunk to 50 in the 6-node fully meshed network we consider above.
In particular, we consider a scenario where the thresholds for long-lived traffic and short-lived traffic are 40 (80%) and 45 (90%), respectively while all the other parameters are kept the same as in the *Default parameter setting* in 2.4.1.

Figure 2.6: Blocking probabilities for (a) long-lived traffic (single priority) and (b) short-lived traffic in a 6-node fully-meshed network with 50 links each trunk.

In Fig. 2.6 (a) we provide the results obtained for the blocking probabilities for the long-lived traffic. We can observe that the accuracy of EFPA improves compared with that of the case of 20 links per trunk shown in Fig. 2.1 (a). The improvement in accuracy is achieved because of the following reasons.

1. When the number of links on each trunk increases, the variance of the overflow traffic decreases, leading to a lower Poisson error.

2. The increase in the number of links on each trunk also reduces the proportion of overflowed traffic and therefore reduces the overflow error, which also increase the accuracy of EFPA.

We also observed that OPCA, in general, is superior to EFPA, so it is sandwiched between EFPA and the simulation results.

Fig. 2.6 (b) shows the blocking probability for the short-lived traffic. We observe again that the accuracy of EFPA is improved comparing to the case of 20 links per trunk.
shown in Fig. 2.3 (a) and OPCA is generally sandwiched between EFPA and the simulation results. The reasons for the improvement in EFPA results are the same as those discussed in the case of the long-lived traffic blocking probability evaluation.

Notice also that for the blocking probability evaluation for both long-lived and short-lived traffic, OPCA still outperforms EFPA in the case of 50 links per trunk. This together with the improved accuracy as the number of links on each trunk increases from 20 to 50, provide some evidence that OPCA can be accurate as the network capacity scales upwards and performs even better than for networks with lower capacity.

Figure 2.7: Blocking probabilities for (a) long-lived traffic (single priority) and (b) short-lived traffic in a 6-node fully-meshed network versus number of links on each trunk.

We further increase the number of links on each trunk in 6-node fully meshed network and in Fig. 2.7 (a), we provide the results obtained by EFPA and OPCA for the blocking probabilities for the long-lived traffic versus number of links on each trunk. The threshold of long-lived traffic is kept at 80% and the maximum allowable alternate paths is kept at 4. Let $C$ be the number of links on each trunk. The offered long-lived traffic (Erlangs) is $0.4C$ for each directional SD pair, so the total traffic per SD pair in both directions is $0.8C$. As increasing the traffic and the number of links on each trunk at the same rate, will decrease the proportion of overflowed traffic, thus in large capacity networks, only negligible traffic is overflowed, so that almost all end-to-end paths in our fully meshed networks are single link (the network approximately turns into a fixed-routing network), this will mean that
approximately all links will be 80% utilized. In fixed-routing large capacity networks, EFPA is accurate based on Kelly [23]. Since there is only negligible overflow in large capacity networks, as mentioned in the Introduction, we have the condition under which model is based on fixed routing, i.e. no overflow is allowed, OPCA is reduced to EFPA. This is consistent with the results presented in Fig. 2.7(a).

Fig. 2.7(b) shows the blocking probability for the short-lived traffic versus the number of links per trunk. The thresholds of long-lived traffic and short-lived traffic are set as before at 80% and 90%, respectively while all the other parameters are kept the same as in the Default parameter setting in 2.4.1. The offered traffic load for are both long-lived and short-lived calls for each directional SD pair are \(0.2C\). Although, in general, we observe similar results for short-lived traffic blocking probability to those obtained for long-lived traffic, we notice a slight difference between the EFPA and OPCA results for the cases of 50 and 100 links per trunk, which indicates that some overflows at these level occur which lead to conditions where EFPA underestimate the blocking probability slightly more than OPCA.

![Figure 2.8: Blocking probabilities for (a) long-lived traffic (single priority) and (b) short-lived traffic in NSF network with 50 links each trunk.](image-url)

We also increase the number of links per trunk for the NSF network to 50. In particular, we consider a scenario where the thresholds for long-lived traffic and short-lived
traffic are 40 (80%) and 45 (90%), respectively while all the other parameters are kept the same as in the Default parameter setting in 2.4.1 In Fig. 2.8 (a) and (b) we provide the results obtained for the blocking probabilities for the long-lived traffic and the short-lived traffic, respectively. The trends and behavior of the results presented for the case of NSF network are consistent with the results provided for the 6-node fully meshed network case.

![Diagram](image)

Figure 2.9: Blocking probabilities for (a) long-lived traffic (single priority) and (b) short-lived traffic in NSF network versus number of links per trunk.

We also increase the number of links on each trunk in the NSF network, and in Fig. 2.9 (a), we again provide the results obtained by EFPA and OPCA for the blocking probabilities for the long-lived traffic versus number of links per trunk. These results are based on having the offered long-lived traffic (Erlangs) to be 0.02C for each directional SD pair, the threshold and the maximum allowable alternate paths are kept the same as in the Default parameter setting in 2.4.1.

Then we consider the NSF network with long-lived and short-lived traffic where the thresholds of long-lived traffic and short-lived traffic are set as before at 80% and 90%, respectively while all the other parameters are kept the same as in the Default parameter setting in 2.4.1. Both the offered long-lived traffic and offered short-lived traffic for each directional SD pair are 0.01C. In Fig. 2.9 (b) we present the blocking probabilities for the short-lived traffic versus number of links per trunk obtained by EFPA and OPCA.
Figure 2.10: blocking probability for (a) long-lived traffic (single priority) and (b) short-lived traffic for a 6-node fully-meshed network

Compared with Figs. 2.9 (a), 2.7 (a), and Figs. 2.9 (b), 2.7 (b), the results for NSF network are generally consistent with those for the 6-node fully meshed network. The results based on OPCA and EFPA are almost identical. The small discrepancy observed for short-lived traffic in the case of fully meshed network does not exist in the present case because under NSF the allowable number of alternate routes is smaller so this scenario is closer to fixed-routing network than the fully-meshed alternate-routing network, in which case we already know that if large capacity is available on trunks both OPCA and EFPA are very accurate.

2.4.6 The effect of maximum allowable number of alternate paths

Here we examine how the blocking probability is affected by the maximum allowable number of alternate paths. The maximum allowable number of alternate paths $D$ limits how many times traffic can overflow. Traffic that has already overflowed $D$ times is not allowed to overflow again and will be blocked and cleared from the network. For single class networks with light traffic, increasing $D$ appropriately means more opportunities to overflow and benefits the system by reducing the blocking probability. However, when the offered load in the network is high, increasing $D$ may not reduce the blocking probability.
because of the inefficiency associated with having the average number of links used per call unnecessarily high.

Fig. 2.10(a) demonstrates the effect of maximum allowable number of alternate paths on the blocking probability for the long-lived traffic obtained by simulation, EFPA and OPCA. The offered long-lived traffic is 6.3 Erlangs and the threshold is 16 (80%). Fig. 2.10(b) demonstrates the effect of maximum allowable number of alternate paths on the blocking probability for the short-lived traffic obtained by simulation, EFPA and OPCA. We focus on the 6-node fully meshed network as the smaller number of alternative paths allowable by the NSFNet topology. The offered traffic load for both long-lived and short-lived traffic are 3.6 Erlangs. We change the maximum allowable number of alternate paths, while keeping all the other parameters the same as in the Default parameter setting in 2.4.1.

We observe that there is a clear benefit, in the present example, of 6-node fully meshed network, to increase the maximum number of overflow to at least 2. After that, the rate of decrease in the blocking probability for both long-lived and short-lived traffic slow down as \( D \) increases, due to the inefficiency of the long alternate paths.

### 2.4.7 The effect of trunk reservation

In general, thresholds are applied to reserve channels for the primary path traffic and prevent the network from being crowded by the overflow traffic. Also, the threshold of long-lived traffic can protect the short-lived traffic from being preempted by long lived overflow traffic which requires longer paths to establish a call, uses more resources, and may cause congestion. Threshold of long-lived traffic should be chosen carefully because a small threshold for long-lived traffic prevents many overflow calls from entering the network, leading to large blocking probability for the long-lived traffic. By contrast, a large threshold for long-lived traffic invites too much overflow traffic, thereby congesting the network and preemptioning short-lived calls.
We again consider a 6-node fully meshed networks and the offered long-lived traffic and offered short-lived calls are both 3.6 Erlangs. We change the threshold of long-lived traffic, while keeping all the other parameters the same as in the Default parameter setting in 2.4.1.

For this case, Fig. 2.11 (a) illustrates the effect of threshold for long-lived traffic on the blocking probability for the short-lived traffic.

We can observe that when the both offered long-lived traffic and offered short-lived traffic are equal, varying threshold of long-lived traffic does not affect the blocking probability for the short-lived traffic. This is because the blocking probability for the long-lived traffic is very small and therefore the change of the carried load of long-lived traffic (background traffic of short-lived traffic) caused by the change of threshold for long-lived traffic is also very small, so its effect on blocking probability for the short-lived traffic is negligible.

Fig. 2.11 (b) shows the effect of changing threshold of long-lived traffic on the blocking probability for the short-lived traffic when the ratio of the offered long-lived traffic and offered short-lived traffic is 5:1. The offered loads are set to be 6 and 1.2, for long-lived and short-lived traffic, respectively, and all the other parameters are the same as in the
Default parameter setting in 2.4.1. We can observe that when long-lived traffic is much higher than short-lived traffic, changing threshold of long-lived traffic will increase the blocking probability for the short-lived traffic, due to the fact that the blocking probability for the long-lived traffic is in a range that increasing threshold of long-lived traffic will considerably reduce the blocking probability and increase the carried load of long-lived traffic (which is the background traffic of short-lived traffic).

2.4.8 Robustness of the quasi-stationary approximation

As stated in Section 2.3, when the holding times of long-lived calls are far longer than those of the short-lived calls, we can use the quasi-stationary approximation to obtain accurate results. We use a 6-node fully-meshed network with 20 links each trunk to illustrate by how much the holding time of long-lived traffic should be longer than that of short-lived traffic, for the quasi-stationary approximation to be accurate. The result is shown in Fig. 2.12. In the scenario we consider, both the offered long-lived traffic and offered short-lived traffic are again 3.6 Erlangs, and the mean holding time of long-lived calls \((1/\mu_1)\) is 1.
As expected, when the holding times of short-lived calls ($1/\mu_2$) are larger than or close to those of long-lived calls, the short-lived calls cannot reach steady-state while the number of long-lived calls remains unchanged, so the quasi-stationary approximation is inaccurate. However, when the holding times of short-lived calls are significantly shorter than those of long-lived calls (e.g., by more than two orders of magnitude, namely $\mu_2 > 100$), the blocking probability for the short-lived traffic becomes invariant to further increase in the ratio $\mu_2/\mu_1$, indicating that short-lived traffic may approximately reach steady-state while its background state (due to long-lived traffic activities) remains approximately unchanged, so the quasi-stationary approximation can lead to accurate results. The errors shown in Fig. 2.12 in this condition are mainly due to overflow error and path error discussed above. From the figure, we can observe that our approximation methods work well when the average holding times of short-lived calls is less than 5% of the average holding times of long-lived calls.

2.4.9 The effect of the shape of the holding time distribution

Figure 2.13: The average blocking probabilities for (a) long-lived traffic (single priority), and (b) short-lived traffic, considering different service time distributions for a 6-node fully-meshed network with 20 links in each trunk.
The results presented above are based on the assumption that the holding times of both the long-lived and short-lived traffic are exponentially distributed. It is therefore important to examine the robustness of the approximations to the shape of the holding time distribution. To this end, we compare the results obtained under the exponential assumptions versus results obtained under heavy-tailed holding time distribution, where we maintain the same mean for the two alternatives.

In particular, we consider our heavy-tailed holding times, to follow a Pareto distribution. In the complementary distribution function (CDF) presented in Subsection H-1 in [58], \( \delta \) (seconds) is the scale parameter and minimum holding time and \( \gamma \) is the shape parameter.

In our simulation we set \( \delta = 0.5 \) for long-lived traffic and \( \delta = 0.0025 \) for short-lived traffic and \( \gamma = 2 \) for both types of traffic. All the other parameters are kept the same as in the Default parameter setting in 2.4.1.

Fig. 2.13 (a) shows the simulation blocking probability for the long-lived traffic (single priority) for 6-node fully meshed network with holding time exponential distributed and Pareto distributed. The two curves are very close to each other and their confidence interval are overlapped, which shows that blocking probability for the long-lived traffic (single priority) is insensitive to the holding time distribution.

Fig. 2.13 (b) shows the simulation blocking probability for the short-lived traffic for 6-node fully meshed network of three cases:

1. exponential exponential – The holding times of both long-lived and short-lived calls are exponentially distributed

2. exponential Pareto – The holding time of long-lived traffic is exponentially distributed while that of the short-lived traffic is Pareto distributed

3. Pareto Pareto – The holding times of both long-lived and short-lived calls are Pareto distributed.
The closeness of the simulation results of the three cases illustrates that the blocking probability for the short-lived traffic for 6-node fully meshed network is insensitive to the holding time distribution shape.

![Figure 2.14](image)

**Figure 2.14**: The average blocking probabilities for (a) long-lived traffic (single priority), and (b) short-lived traffic, considering different service time distributions for the NSF network with 20 links in each trunk.

We have produced equivalent simulation results for the case of the NSF network topology. The results are presented in Fig. 2.14(a) and (b). We observe similar behavior as in the case of fully meshed networks that provide further evidence that the assumption of exponential holding time distribution is reasonable and that the blocking probability is not very sensitive to the shape on the holding time distribution.

These results are consistent with equivalent results obtained in [58] for OBS networks.

### 2.4.10 Computational complexity of the algorithms

It is difficult to provide general analytical results for computational complexity of OPCA and EFPA because both require fixed-point iterations to converge. Nevertheless, we provide numerical examples that illustrate the time and memory complexity of the algorithms.
Table 2.2: Computational complexity of the algorithms

<table>
<thead>
<tr>
<th>Trunk capacity</th>
<th>Running time of EFPA (seconds)</th>
<th>Running time of OPCA (seconds)</th>
<th>Memory of EFPA (bytes)</th>
<th>Memory of OPCA (bytes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>C = 20</td>
<td>0.173369</td>
<td>0.413720</td>
<td>66840</td>
<td>89040</td>
</tr>
<tr>
<td>C = 100</td>
<td>0.431417</td>
<td>3.034318</td>
<td>95640</td>
<td>137040</td>
</tr>
<tr>
<td>C = 1000</td>
<td>278.515913</td>
<td>1065.012123</td>
<td>419640</td>
<td>677040</td>
</tr>
</tbody>
</table>

In Table III, we provide information on running times and memory usage of both EFPA and OPCA required for the network blocking probability computation in a six-node fully meshed network for three cases representing different trunk capacity values.

We have observed from these numerical examples, that for a small network, OPCA requires more computation time and more memory than EFPA, but the overall computing resources are manageable. As we demonstrate in the next subsection OPCA is also applicable to the large Coronet network while the equivalent EFPA results are unattainable.

2.4.11 The Coronet

![Figure 2.15: The Coronet topology.](image)
We demonstrate here that OPCA is applicable to large scale networks such as the Coronet, shown in Fig. 4.17, while simulation results are computationally prohibitive for such large scale networks. With all of the 9900 SD pairs in the network and one alternative path for each SD pair, the network blocking probabilities for long-lived and short-lived traffic can be obtained by OPCA within reasonable running time, as shown in Fig. 2.16. The number of links per trunk is 50 and the thresholds for long-lived traffic and short-lived traffic are 40 (80%) and 45 (90%). The maximum allowable number of alternate paths is 1. All the other parameters are kept the same as in the Default parameter setting in subsection 2.4.1. The running times used to calculate the network blocking probabilities in the Coronet is about 121.734439 seconds by OPCA, obtained using MATLAB 7.6.0 executed on a desktop PC with IntelR CoreTM 2 Quad @ 3 GHz CPU, 4 GHz RAM and 32-bit operating system.

![Figure 2.16: The blocking probabilities obtained by OPCA for (a) long-lived traffic (single priority), and (b) short-lived traffic for the Coronet with 50 links per trunk.](image)

2.5 Summary

In this chapter, we have considered a circuit-switched network with long-lived and short-lived connections where the long-lived connections can preempt the short-lived ones. We
use EFPA and OPCA combined with the quasi-stationary approximation to estimate the blocking probabilities. The results demonstrate that in most cases, OPCA can estimate the blocking probabilities reasonably well, and generally, better than EFPA. As long-lived connections provide background traffic for short-lived ones, the ratio of their offered load also affects the accuracy of the approximations. Reduction of offered long-lived traffic together with increase of offered short-lived traffic improves the accuracy of OPCA, whereas that of EFPA is not improved significantly. However, when the number of links on each trunk increases, the performance of EFPA is improved. Allowing more alternate path traffic, either by increasing the maximum allowable alternate paths or the long-lived traffic threshold, is beneficial under light traffic. However, when the network is fully occupied, it is important to restrict alternate path traffic. We have observed that the quasi-stationary approximation requires that the mean holding time of long-lived connections is at least 20 times longer than that of short-lived connections. Nevertheless, this is not a very restrictive requirement if long-lived connections represent static connections and short-lived connections represent dynamic ones. We have also demonstrated that approximating blocking probability based on the exponential holding time assumption is not very sensitive to the shape of the holding time distribution, and is fairly accurate also for heavy-tailed holding time distributions. We have illustrated by numerical examples that, for small network, OPCA requires more computation time and more memory than EFPA, but the overall computing resources are manageable. For a large scale network such as the Coronet, we have demonstrated that OPCA is also applicable, whereas the equivalent EFPA results are unattainable.
Chapter 3

Computation of blocking probability for large circuit switched networks

3.1 Introduction

For several decades, numerous publications predicted and promoted an increasing role of circuit switching (CS) in the evolving ultra-broadband multi-service Internet. As early as 1988, O’Reilly [16] identified the advantages of CS over packet switching, including easier provision of grade of service, simpler transport (which implies energy saving) and better suitability for multimedia services. CS attributes have given rise to dynamic CS-based network technologies and designs such as optical flow switching [59], and networks tailored for transmitting large bursts from the Large Hadron Collider [3] and for inter-data center traffic [1].

Considering the renewed CS importance, and the exponential growth of the Internet, there is a need for scalable and accurate means to evaluate blocking probability in large CS networks. The Erlang Fixed Point Approximation (EFPA) has been the method of choice for this purpose since 1964 [33]. EFPA decouples the network into independent server groups (links), and considers the traffic on each link to follow a Poisson process.
independent traffic on other links. At each iteration of the EFPA algorithm, the blocking probability on each link is computed, where its offered traffic comprises all the end-to-end traffic streams that use that link minus the blocked traffic (which initially can be set to zero). End-to-end blocking probabilities on all links are then recalculated, and the process repeats itself until convergence is achieved.

The Poisson and independence assumptions introduce errors. However, Kelly [23] showed that for a CS network with fixed routing, EFPA yields an asymptotically exact solution for a large number of channels per trunk. Today’s DWDM networks with nearly hundreds of wavelengths per optical fiber and hundreds of optical fibers per cable yield tens of thousands of wavelength channels per cable. If a wavelength is further subdivided into hundreds of TDM channels, millions of channels per cable is a realistic scenario. Furthermore, the analysis by Kelly [23] applies to multiple classes of calls each requiring a different number of channels and its performance result is insensitive to the shape of the holding time distribution.

Given that the results of [23] are applicable to today’s networks, it is important to apply them in a way that the blocking probability is computable for realistic size networks. Using the conventional EFPA which repeatedly computes the link blocking probability by applying a recursive procedure to the Erlang-B formula, is computationally infeasible for such networks with many channels per cable. In this chapter, we consider an implementation of EFPA, which is called Asymptotic EFPA (A-EFPA). A-EFPA computates the link blocking probability using the asymptotic result for a large number of channels per cable which is essentially based on a fluid approximation. A-EFPA can achieve comparable accuracy to EFPA when the number of channels per cable is large, but it saves computing time by many orders of magnitude. Thus, it can be used in the range where EFPA cannot. We demonstrate for NSFNet and Internet2 networks, that given today’s computing power, simulations can be used in the range where EFPA is inaccurate, and now A-EFPA is shown to be accurate in the range where EFPA is computationally prohibitive. Accord-
ingly, for both networks, very accurate blocking probability results are achievable for the full range of practical parameter values.

Henceforth, we use the terminology of [23]. In particular, we use the term circuit to describe the basic channel used for a connection (e.g. a wavelength channel) and the term link to refer to a trunk where many circuits are bundled together to provide transport between nodes.

3.2 Model

We consider a general CS network. The links between nodes are numbered by \( j = 1, 2, \ldots, J \), and \( C_j \) is the number of circuits in link \( j \). The subscript is omitted if all links have the same number of circuits (then \( C = C_j \) for all \( j \)). For convenience, the notation \( J = \{1, 2, \ldots, J\} \) is used in the sequel. A call on route \( r \) uses \( A_{j,r} \) circuits from link \( j \). Let \( \mathcal{R} \) be a set of possible routes. The calls that request route \( r \) are assumed to arrive as an independent Poisson process of rate \( \nu_r \). A call that requests route \( r \) is blocked and lost if at least in one of the links \( j, j \in J \), in route \( r \), fewer than \( A_{j,r} \) circuits are free. Otherwise, the call is admitted and simultaneously holds \( A_{j,r} \) circuits from all links \( j, j \in J \), for the holding period of the call. Without loss of generality the call holding times are assumed to be independently and identically distributed with the unit mean. According to the analysis and asymptotic conditions of [23], the blocking probability at the limit is only a function of the offered load and the number of circuits per link. Therefore, the arrival rate can be adjusted to compensate for non-unity holding times, and a call that requires 10 circuits is equivalent to 10 calls that require one circuit, if the total offered load for each OD pair is unchanged.
3.3 A-EFPA algorithm

As in [23], we consider a sequence of networks indexed by $N$ where the number of circuits per link and the offered traffic per OD pair grows linearly in $N$. For positive $x$ and integer $C$, denote $E(x, C) = \frac{x^C}{\sum_{i=0}^{C} x^i/i!}$. It is proved in [23] that the system of equations

$$x_j = E \left( \frac{1}{1-x_j} \sum_{r \in R} A_{j,r} \nu_r \prod_{i=1}^{J} (1-x_i)^{A_{i,r}, C_j} \right), \quad j \in J \quad (3.1)$$

has a unique solution in $(0,1)^J$. In Eq. (3.1), the blocking probability $x_j$ of link $j$ is obtained by Erlang-B formula based on this link capacity $C_j$ and the total offered load to that link (i.e. excluding traffic blocked on other links) of the OD pairs that use this link. EFPA solves (3.1) by fixed-point iterations. For (3.1), the following limit theorem for a series of CS networks is proved in [23].

**Theorem 3.3.1** (Kelly [23]) Assume that the parameters $\nu_r(N)$ and $C_j(N)$, $(r \in R, j \in J)$, increase to infinity such that

$$\lim_{N \to \infty} \frac{\nu_r(N)}{N} = \lambda_r, \quad \lim_{N \to \infty} \frac{C_j(N)}{N} = c_j.$$

Then,

$$\lim_{N \to \infty} \prod_{j=1}^{J} [1-x_j(N)]^{A_{j,r}} = \prod_{j=1}^{J} (1-B_j)^{A_{j,r}},$$

where $x(N) = \{x_1(N), \ldots, x_J(N)\}$ is a series of the vector-valued solutions of (3.1) indexed by $N$, and $B_j$, $j \in J$, is the asymptotic blocking probability on link $j$.

According to Theorem 3.3.1 for large circuit-switched networks the blocking probabilities $B_j$ are supposed to be approximated by (3.1), in which the parameters $\nu_r(N)$ and $C_j(N)$ are large. The further asymptotic analysis enables us to substantially simplify the calculation. As $N \to \infty$, for the function $E(x(N), C(N))$, in which $x(N) \approx x^*N$ and $C(N) \approx c^*N$, where $x^*$ is a positive real value and $c^*$ is a positive integer value, we have...
the following asymptotic expansion:

\[
E(x(N),C(N)) = \begin{cases} 
1 - \frac{c^*}{x} + o(1), & \text{if } c^* \leq x^*, \\
o(1), & \text{otherwise.}
\end{cases}
\]  

(3.2)

Henceforth, for two sequences \(x_n\) and \(y_n\) increasing to infinity \(x_n \asymp y_n\) means \(\lim_{n \to \infty} \frac{x_n}{y_n} = 1\).

The difficulty of using asymptotic expansion (3.2) directly is that the expression

\[
\frac{1}{1-x_j} \sum_{r \in R} A_{j,r} \nu_r \prod_{i=1}^{J} (1-x_i)^{A_{i,r}}
\]

contains unknowns \(x_j, j = 1, 2, \ldots, J\), and, hence, we do not know whether or not the inequality

\[
\frac{1}{1-x_j(N)} \sum_{r \in R} A_{j,r} \nu_r(N) \prod_{i=1}^{J} (1-x_i(N))^{A_{i,r}} \geq C_j(N)
\]

is satisfied. Accordingly, if we ultimately use the term

\[
1 - C_j(N) \left[ \frac{1}{1-x_j(N)} \sum_{r \in R} A_{j,r} \nu_r(N) \prod_{i=1}^{J} (1-x_i(N))^{A_{i,r}} \right]^{-1} \tag{3.3}
\]

which is the asymptotic expansion of (3.2) applied to the right-hand side of (3.1) under the asymptotic conditions of Theorem 3.3.1 then we may arrive at biased values for the estimates of blocking probabilities (in the cases where terms (3.3) are negative). We use the term *biased* to refer to values that are not the right blocking probabilities (which cannot be negative).

**Example** Consider a simple CS network that consists of two links \(a\) and \(b\) with capacities \(C_a\) and \(C_b\), respectively, and assume that calls are of two types: \(ab\)-calls which use a circuit in both links \(a\) and \(b\) and \(b\)-calls which use a circuit in link \(b\) only. Denote by \(\nu_{ab}\) the input rate of \(ab\)-calls and by \(\nu_b\) the input rate of \(b\)-calls. For a series of networks
assume \( C_a(N) = C_b(N) = cN, \nu_{ab}(N) = \nu_b(N) = \lambda N \) and \( c < \lambda \). These yield the following system of equations.

\[
x_a = 1 - \frac{c}{\lambda(1 - x_b)}, \tag{3.4}
\]

\[
x_b = 1 - \frac{c}{\lambda(1 - x_a) + \lambda}. \tag{3.5}
\]

It is readily seen from this system of equations that \( x_a < 0 \). Indeed, assume that \( x_a > 0 \). Then, it follows from (3.4) that \( x_b < 1 - \frac{c}{\lambda} \). From (3.5) we obtain the opposite inequality: \( x_b > 1 - \frac{c}{\lambda} \). This contradiction leads to the conclusion \( x_a \leq 0 \). Now substituting \( x_a = 0 \) into (3.4) we obtain \( x_b = 1 - \frac{c}{\lambda} \). The same substitution into (3.5) yields \( x_b = 1 - \frac{c}{2\lambda} \).

Hence, \( x_a \) cannot be equal to zero, and we arrive at \( x_a < 0 \). This solution gives biased values for blocking probabilities. Negative \( x_a \) value has to be set to zero, and \( x_b \) has to be recalculated accordingly.

A-EFPA is based on decomposing the set \( \mathcal{J} \) into three non-intersected subsets: \( \mathcal{J} = \mathcal{J}^- \cup \mathcal{J}^0 \cup \mathcal{J}^+ \) as described below.

1. Check the inequalities

\[
\sum_{r \in \mathcal{R}_j} A_{j,r} \lambda_r > c_j \tag{3.6}
\]

for all \( j \in \mathcal{J} \). Then, the subset \( \mathcal{J}^0 \) characterizes the set of values \( j \) for which (3.6) is not satisfied. Set \( x_j = 0 \) for all \( j \in \mathcal{J}^0 \). By separating the subset of links \( \mathcal{J}^0 \), we reduce the dimension of the original system of equations. The new system of equations is as follows:

\[
x_j = E \left( \frac{1}{1 - x_j} \sum_{r \in \mathcal{R}_j} A_{j,r} \nu_r \prod_{i=1}^{j} (1-x_i)^{A_{i,r}} C_j \right), \quad j \in \mathcal{J} \setminus \mathcal{J}^0. \tag{3.7}
\]
2. Solve the system of equations, in which the right-hand side of (3.7) is replaced by the main term in the asymptotic expansion:

\[ x_j = 1 - c_j \left[ \frac{1}{1 - x_j} \sum_{r \in R} A_{j,r} \lambda_r \prod_{i=1}^{J} (1 - x_i)^{A_{i,r}} \right]^{-1}, \quad j \in J \setminus J^0. \]

Let \( J^- \) be the set of values \( J \setminus J^0 \) for which \( x_j \leq 0 \). Set \( x_j = 0 \) for all \( j \in J^- \). Then, after eliminating the set of equations in which \( j \in J^- \), we obtain the new system of equations:

\[ x_j = 1 - c_j \left[ \frac{1}{1 - x_j} \sum_{r \in R} A_{j,r} \lambda_r \prod_{i=1}^{J} (1 - x_i)^{A_{i,r}} \right]^{-1}, \quad j \in J^+ = J \setminus (J^0 \cup J^-). \]  

(3.8)

3. Solve the system of equations (3.8). This is the final step, where we obtain only a positive solution. That positive solution together with the solutions that are set 0 in steps 1 and 2 give a full set of the asymptotic solution of the system of equations.

A-EFPA is essentially based on a reduction to a fluid model. If the traffic offered to a link is higher than its capacity, the difference between the offered traffic and capacity is the blocked traffic, and the ratio of the blocked traffic to the offered traffic is the blocking probability. Otherwise, the blocking probability is zero.

### 3.4 Numerical Results

In this section, we compare the blocking probability and running time of EFPA and A-EFPA for NSFNet (13 nodes and 16 bidirectional links) and Internet2 (52 nodes and 60 bidirectional links) networks – see Fig. 2.2 and 3.1. For both networks, we choose all possible OD pairs with the shortest path routing where a tie is broken randomly. Simulation results are also provided whenever they are achievable within a reasonable time.
All the results are obtained by using MATLAB software executed on a desktop PC with Intel® Core™ 2 Quad @ 3 GHz CPU, 4 GHz RAM and 32-bit operating system.

For NSFNet, there are 156 OD pairs. For each OD pair, the offered load is set to be 0.025\(C\), so that the total load offered to some of the links is larger than the capacity (note that in the NSFNet example, some links serve almost 100 OD pairs). Fig. 3.2 illustrates that only when the capacity is sufficiently large, the A-EFPA results are very close to those of EFPA. For \(C \geq 20,000\), the relative discrepancy is less than 0.2\%. Simulations confirm that the accuracy of EFPA increases with an increasing number of circuits per link. Confidence intervals are too small to be seen in the plots, but for the presented results, the radius of the 95\% confidence interval based on Student’s t-distribution is less than 1.5\% of the mean.

Table 3.1 provides the running time used to calculate the blocking probability in NSFNet for different \(C\) values. Observe that for \(C = 20000\), A-EFPA saves 99.9999\% of the time used by the EFPA, and achieves similar accuracy.

It is intuitively clear that if the offered load is sufficiently small, the offered load on all links will be less than their capacities, so the link blocking probabilities calculated by A-EFPA are all 0, thus the network blocking probability is also 0. In such a scenario, according to the theory, as \(C \to \infty\), the exact blocking probability and the blocking probability predicted by EFPA, both will also approach zero. However, for fixed \(C\), some error is caused by using A-EFPA. In Table 3.2, we illustrate the effect of traffic load on the accuracy of A-EFPA for NSFNet when \(C = 20,000\). In the first row, we illustrate the
Table 3.1: Comparison of the times used by EFPA and A-EFPA to calculate the blocking probabilities in NSFNet.

<table>
<thead>
<tr>
<th>Calculation task</th>
<th>Running time of EFPA in seconds</th>
<th>Running time of A-EFPA in seconds</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blocking probabilities in one link and $C = 20000$</td>
<td>14.94</td>
<td>0.000001</td>
</tr>
<tr>
<td>Blocking probability of the whole network and $C = 2000$</td>
<td>5.91</td>
<td>0.068</td>
</tr>
<tr>
<td>Blocking probability of the whole network and $C = 20000$</td>
<td>1800.63</td>
<td>0.069</td>
</tr>
</tbody>
</table>

case of zero blocking predicted by A-EFPA where although the relative error is 100%, the absolute error is small and will be further reduced as $C$ increases. We then observe the improved accuracy of A-EFPA as the traffic increases.

Table 3.2: Effect of offered load on A-EFPA.

<table>
<thead>
<tr>
<th>Offered load</th>
<th>Blocking probability of EFPA</th>
<th>Blocking probability of A-EFPA</th>
<th>Difference</th>
<th>Relative error</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0205</td>
<td>$9.9 \times 10^{-10}$</td>
<td>$0$</td>
<td>$9.9 \times 10^{-10}$</td>
<td>100%</td>
</tr>
<tr>
<td>0.022</td>
<td>0.0103034</td>
<td>0.0090968</td>
<td>0.000206</td>
<td>3.8%</td>
</tr>
<tr>
<td>0.025</td>
<td>0.0449448</td>
<td>0.0448718</td>
<td>0.000073</td>
<td>0.16%</td>
</tr>
<tr>
<td>0.035</td>
<td>0.1786133</td>
<td>0.1785714</td>
<td>0.000041</td>
<td>0.023%</td>
</tr>
</tbody>
</table>

For Internet2, for each OD pair, the offered load is set to be 0.12% of $C$, so that the total load offered to some links is larger than their capacities (note that in our Internet2 example some links serve around a thousand OD pairs). The blocking probabilities for Internet2 evaluated by EFPA and A-EFPA are shown in Fig. 3.2b. As for NSFNet, the results obtained by A-EFPA are very close to those of EFPA when the capacity is large. When the number of circuits per link reaches $C = 50000$, the relative discrepancy (vs. EFPA) of the blocking probability calculated by A-EFPA is 1.0%. However, comparing the time used by the two algorithms shown in Table 3.4, we observe that A-EFPA can save 99.9999% of the time. The trends and behaviors of the results presented for Internet2 are
consistent with the results provided for NSFNet. The limitation of simulations motivates using EFPA or A-EFPA which are asymptotically exact as the number of circuits per link increases. Confidence intervals are again unnoticeable in the plots, but the radius of the 95% confidence interval based on Student’s t-distribution is always less than 1.5% of the mean result. EFPA overestimates the blocking probability obtained by simulation because under fixed routing, traffic is smoothed out along an end-to-end path and thus gives lower loss than estimated by EFPA. This overestimation is more prominent in Internet2 than in NSFNet because the average path in Internet2 is longer.

Table 3.3: Comparison of the blocking probabilities calculated by EFPA and A-EFPA for NSFNet and Internet2.

<table>
<thead>
<tr>
<th></th>
<th>EFPA</th>
<th>A-EFPA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blocking probability</td>
<td>0.04494</td>
<td>0.04487</td>
</tr>
<tr>
<td>for NSFNet with C = 20000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Blocking probability</td>
<td>0.01417</td>
<td>0.01383</td>
</tr>
<tr>
<td>for Internet2 with C = 20000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Blocking probability</td>
<td>0.01397</td>
<td>0.01383</td>
</tr>
<tr>
<td>for Internet2 with C = 50000</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Blocking probability results for NSFNet with multirate traffic are presented in Fig. 3.3. Each OD pair has two classes of calls. A Class 1 call requires a single channel end to
Table 3.4: Comparison of the times used by EFPA and A-EFPA to calculate the blocking probabilities in Internet2.

<table>
<thead>
<tr>
<th>Calculation task</th>
<th>Running time of EFPA in seconds</th>
<th>Running time of A-EFPA in seconds</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blocking probability of the whole network and $C = 2000$</td>
<td>66.68</td>
<td>0.72</td>
</tr>
<tr>
<td>Blocking probability of the whole network and $C = 50000$</td>
<td>62501.81</td>
<td>1.27</td>
</tr>
</tbody>
</table>

Figure 3.3: Blocking probabilities in 13-node NSFNet evaluated by EFPA, A-EFPA and simulation with two classes of traffic.

end and a Class 2 call requires five channels end to end. The offered load to each class in each OD pair is $0.015C$. The presented results exhibit similar trends and behavior to the results presented for NSFNet with a single class of calls.

We observe that for Class 1 calls, EFPA overestimates the blocking probability as in the single class case because of the smoothing effect along the path. However, for Class 2 traffic EFPA underestimates the blocking probability because it assumes Poisson arrivals of individual circuit demands and ignores the larger variance introduced by the 5-circuit batch demands. These dominate the path smoothing effect, so the overall variance is higher than that assumed by EFPA.
Figure 3.4: Blocking probability in the Coronet evaluated by EFPA and A-EFPA.

Fig. 3.4 shows the results when EFPA and A-EFPA are applied to the large scale Coronet, as shown in Fig. 4.17. We also observe that in Coronet, when the capacity is sufficiently large, the A-EFPA results are very close to those of EFPA.

In all the results presented here, we observe that the simulation curve meets (or becomes very close to) the EFPA curve, significantly before the EFPA and A-EFPA curves are very close. These results indicate that given today’s computing power, we can use simulations to reach the point in terms of the number of circuits per link, where EFPA is accurate, then EFPA can provide accurate results until A-EFPA becomes accurate, which in turn can guarantee accuracy for far larger values of circuits per link.

3.5 Summary

We consider A-EFPA and EFPA implementation for CS networks with fixed routing based on asymptotic link blocking probability derivation. From the numerical results obtained using NSFNet, Internet2 networks, we observe that when link capacity is large, A-EFPA results are very close to those of EFPA, but A-EFPA saves approximately 99.9999% of
the computing time. We observe the consistent behaviors of A-EFPA and EFPA when they are applied to the Coronet. We have demonstrated very accurate calculations of the blocking probability through simulations, EFPA and A-EFPA. These three complementary methods are used for different ranges of parameter values.
Chapter 4

Performance analysis of Circuit Switched Multi-service Multi-rate Networks with Deflection Routing

4.1 Introduction

Over the last quarter of a century, the Internet has evolved from a packet switched network that provides only data services, such as email and file transfer, to a network that provides a wide range of services. Nowadays, there is an increasing number of Internet users that transmit extremely large flows of data. Several examples follow. The users of the Internet include cloud service providers (CSP) such as Google, Facebook and Yahoo! that often replicate their content across multiple data centers transmitting massive amount data [1]. In fact, the total CSPs inter-datacenter traffic was over 400 exabyte during the year 2011 and growing at over 30% yearly growth rate [2]. The Internet users also include the CERNs Large Hadron Collider (LHC) that transmits several Petabytes of data per year [3].

Switching such extremely large flows using packet switching at the IP layer requires unacceptable levels of energy consumption [4–6]. Given the potential for orders of mag-
nitude savings in energy per bit using lower (optical) layers [7, 74], it is envisaged that circuit switching, which has been widely used in telephone networks, will have a renewed and important role in future optical networks [8–15]. In packet switching, many energy consuming operations such as buffering packets, processing individual packet headers, performing table look-up, and counting the number of packets at the destination node [75] are avoided by circuit switching [4]. Note that these advantages of circuit-switching for wide bandwidth networks were pointed out nearly a quarter of a century ago [16] in terms of simplicity rather than energy consumption.

If the bit-rate offered to a circuit-switched multimedia network is sufficiently high and the traffic is well engineered, such a network can guarantee quality-of-service (QoS) to customers in a way that can even lead to efficient transmission resource utilization and low energy consumption. For example, a 100 GByte burst transmitted from the Large Hadron Collider (LHC) can be efficiently transmitted by setting up a circuit of one or more wavelengths which will be fully utilized during its holding time. Other benefits provided by circuit switching include overload control without congestion collapse, robustness brought by the fast recovering of circuit switching equipments, as well as relatively simple provision of synchronization.

Different rates, different holding times and different bandwidth requirements are relevant to future circuit switching applications in the Internet. Two classes of circuit switched capacity allocation are envisioned [62]. The first is a commonly used class of circuit-switched long-lived connections which are based on setting up a lightpath that provides permanent or semi-permanent connections between two metropolitan edge routers that normally serve many flows that come and go and use that lightpath. Such long-lived connections are used by Internet service providers, or by corporations as leased lines to create private networks.

The second is characterized by dynamic resource allocation for relatively short-lived circuit-switched connections that serve dynamic demands and provide lightpath connec-
tions whenever and wherever they are needed. They can be provided end-to-end or edge-
to-edge. These short-lived dynamic connections require quick connection set-up, but
when the capacity resource is allocated, it can be efficiently utilized especially if the
amount of data to be transported is known in advance [1,3]. In this way, bandwidth on
demand services [1,76,77] are provided where fixed capacity is allocated for the service
duration and then released by the user. Both the allocated capacity and the service dura-
tion are based on customer requirements. The capacity is available to the user exclusively
for the required duration of the service, whether or not it is fully used in accordance with
circuit switching principles.

According to the 2009 book [78], an Internet model where short-lived dynamic con-
nections are provided “is a futuristic model since lightpaths today are relatively long lived,
but it is quite possible that lightpath will be provided on demand by some operators in the
future.” Such possibility is justified by market pull and technology push described in the
following. One recent technology push is the proposed Globally Reconfigurable Intel-
ligent Photonic Network (GRIPhoN) [1,79] that aims to provide bandwidth on demand
(BoD) service for inter-data center communication in the core network. It is motivated
by market pull created by CSPs. Having a dedicated network for such inter-data center
communication in the core network incurs a major cost component of the total cost of
cloud computing [80] especially if they use it only occasionally. It makes economic sense
for an operator to provide BoD service to a multiplicity of CSPs that share the network so
that it can be more efficiently utilized, so that CSPs can enjoy cost saving.

Furthermore, not only CSPs can benefit from BoD. The potential customers can be
smaller operators, enterprises, research networks, and even retail customers. The vision of
Internet2 Dynamic Circuit Network (DCN) [10,12,81–84] aims to provide BoD services
[1,76,77] where fixed capacity is allocated for the service duration and then released
by the user. Both the allocated capacity and the service duration are based on customer
requirements. The capacity is available to the user exclusively for the required duration of
the service, whether or not it is fully used in accordance with circuit switching principles. In many cases, the BoD is provided for a large known burst of data, so the circuits are nearly fully utilized once the circuit path is established. Example of the latter is BoD service based on circuit switching provided to the above mentioned LHC burst of data generated by high energy physics experiments. In particular, such a burst uses the LHC Open Network Environment (LHCONE) which is part of Internet2 as an access network and is transported through the LHC Optical Private Network (LHCOPN) which serves as the core network [3].

Accordingly, we can expect a scenario where all these BoD service classes compete for the same pool of optical capacity available in the core network that needs to be efficiently allocated. Each of these classes can be characterized according to the arrival rate of its burst, flow or connection requests, its mean holding time and capacity requirement. For example, certain very large dynamic flows generated by the Large Hadron Collider (LHC) [3] will require more capacity and/or longer holding time than smaller retail customers flows, but far less capacity than an inter-data center flow transmitted by one of the large CSPs.

A network operator that aims to provide such a wide range of BoD services needs means to efficiently dimension the network to meet QoS requirements. To this end, there is a need for a scalable and accurate method to evaluate performance for each relevant scenario of network topology, parameter values and traffic demand. One important measure of performance is the blocking probability. Since the end-to-end blocking probability is an important QoS measure perceived by users, having accurate blocking probability approximation will enable network designers to dimension the network resources so that the blocking probabilities for each class will not exceed the required value. Various approaches for alternate routing have been studied aiming to reduce blocking probability in circuit switched networks [11, 21, 24, 29]. Alternative routing also helps distribute load among the trunks improving load balancing and provides protection in case of trunk fail-
In a circuit switched network with deflection routing, new calls which cannot be admitted by their primary routes can overflow to other alternative routes, which are usually longer than the primary routes. The inefficiency of alternative paths may in turn increase blocking probability. To prevent the use of very long alternative paths, there needs to be a limit to the maximum number of times that a call can overflow. If we assume Poisson call arrivals for any origin destination (OD) pair and exponential call holding times, a circuit-switched network with alternate routing can be modeled as a Markovian overflow loss network and the stationary occupancy distribution can, in principle, be obtained by solving its steady-state equations. Such models usually do not admit product-form solutions and are not amenable to analysis that leads to a scalable solution for realistic size networks.

In this chapter we consider a model of a circuit switched multi-service network with deflection routing for which we provide accurate methods for evaluation of blocking probability. The chapter is also applicable to versions of MPLS where sufficient capacity for LSP is reserved in advance to avoid significant need for buffering in the network. The labels enable the establishment of end-to-end circuits for transmissions of packets, which is fundamentally similar to CS. The term multi-service network (or system) refers to a network where there are multiple classes of circuit requests between each origin destination (OD) pair. The classes are characterized by different arrival rates, holding times and capacity requirements. Equivalent terms which are often used instead of multi-service in the context of multi-service networks are multiclass and heterogeneous. In [86], we consider two classes of demands with different holding times and give priority to the one with longer holding time. In this chapter, the multi-classes also have different capacity requirements and fair opportunity to compete for the pool of resources.

In multi-service network models it is often assumed that the arrivals of circuit requests for each class follow a Poisson process. We will also make this assumption for tractability. It is accurate for busy hour periods when the number of sources is large and
the sources are independent. We acknowledge that certain pre-arranged scheduling with advance reservation may reduce blocking probability, so in such cases our model provides conservative results. In other cases, where the sources are random and dependent, the Poisson assumption may underestimate the blocking probability. We also consider non-hierarchical deflection routing for cases where the least cost route is not available. When we use the term deflection routing, we also mean alternate routing which has been often used in the context of circuit switched telephone networks. The focus on busy hour is important for network dimensioning to provide sufficient resources to meet the demand during this critical period.

The prevalence of multi-service systems and networks in modern telecommunications and the history and future potential of circuit switching give rise to extensive research on modeling, analyses and performance evaluation of circuit switched multi-service systems and networks. For a single trunk with multiple channels with different classes of demand, under the complete sharing policy, the steady-state probabilities can be obtained numerically. A recursive algorithm based on a product form solution is provided by [87] and [88]. Other improvements and generalizations are described in [54], [89] and [90].

For a multi-service loss network with fixed routing, the steady-state distribution has an explicit product-form solution [91–93]. However, obtaining the state probabilities requires computation of a normalization constant which is difficult for realistic size networks with e.g., over 100 wavelengths per trunk. Such an approach is only applicable to network problems of low dimensionality, or to networks of a special topology, e.g., tree networks.

Owing to the difficulties of obtaining exact solutions for realistically large networks, approximations have been developed and used. One approximation known as the Erlang Fixed point Approximation (EFPA) first proposed in 1964 [33] for the analysis of circuit-switched networks and has remained a cornerstone of telecommunications networks and systems analysis to this day. Kelly [23] have shown that EFPA leads to exact result for
the blocking probability for a multiservice network based on fixed routing, under the asymptotic conditions where trunk capacities are arbitrarily large relative to the capacity required by the most demanding traffic class. However, for our network model based on non-hierarchical deflection routing, Kelly’s results do not apply, and EFPA is only an approximation even under Kelly’s asymptotic conditions.

Other approximations for certain special cases have been proposed in [94], which consider only one OD pair and several alternate paths, and in [95] which considered fixed routing in a multi-stage network.

Methods involving moment matching have been used to estimate blocking probability in circuit-switched networks with deflection routing [32–40, 54]. However, they were all confined to relatively simple cases involving only a single service and where the trunks have a hierarchical structure (hierarchical networks).

Non-hierarchical deflection routing is more flexible and efficient than hierarchical routing due to the fact that it can accommodate a sudden strong increase of offered traffic in different OD pairs, which may happen at different times during the day [96–99]. The drawback of non-hierarchical deflection routing is that it may cause instability and collapse of throughput with heavy or overload conditions which can be prevented by control schemes such as trunk reservation [98]. Overall, non-hierarchical deflection routing is considered an advantage and it was shown to be capable of reducing about 10% cost compared to its hierarchical counterpart [96].

However, for non-hierarchical circuit-switched networks, no robust and generic methodology is available for the approximation of blocking probability (even for cases involving only a single service) that captures networks’ overflow-induced state dependencies in a scalable way, except for special cases [41]. The difficulty in obtaining accurate blocking probability is due to the effect of mutual overflows.

One simple and commonly used approach for approximation of blocking probability in non-hierarchical networks is the above mentioned EFPA. In [23], EFPA assumes that
the arrivals to each trunk follow a Poisson process and the trunks are independent of each other. See [21, 23, 25, 30, 45-47, 50-52] and references therein for applications of EFPA. However, the accuracy of EFPA is not always satisfactory due to errors introduced by the assumptions of EFPA. EFPA assumes that the call arrivals to each trunk follow a Poisson process while, in fact, overflow traffic is more peaky than Poisson and the traffic offered to a sequence of trunks on a path is actually smoother. EFPA also assumes the trunks are independent while there can be strong dependence among their traffic loads.

There have been various attempts to refine the basic version of EFPA by addressing the errors introduced by its assumptions. In [33], the authors proposed the original idea of EFPA together with a moment matching method to reduce the error introduced by non-Poisson overflow traffic and considered the path dependency when calculating the blocking probability for a single service three-node hierarchical network. In [54], a recursive scheme and the equivalent random theory are combined to estimate the blocking probability and the variance of the overflowed streams. However, the analysis of [54] is limited to a single trunk with single service and no application to realistic size network is provided. In [42] and [55], approaches to capture the dependency between trunks along a path in fixed routing networks with multiservice demands are proposed. However, they are limited only to fixed routing networks and do not consider overflow effects. In [100], a method to compute the correlation coefficients between trunks along a path has been proposed and also, it cannot be used in multiservice alternative routing networks.

Another way to categories the errors of EFPA is to classify them into: overflow error and path error [53], which are caused by the effect of overflow and the fact that a path contains a sequence of trunks, respectively. While overflow errors always cause underestimation of blocking probability (ignoring high variance of overflow traffic and dependence), the path error overestimates blocking probability because it ignores the effect of traffic smoothing, and the positive correlation of trunk occupancy along the path that increases the probability to admit calls.
Another recently developed and proven to be more accurate in various circumstances method is the *Overflow Priority Classification Approximation* (OPCA) [53, 56-58]. OPCA is an approximation applicable to overflow loss systems and networks. The idea of OPCA is to impose a fictitious preemptive priority structure in the given network model that yields a surrogate network model. In the OPCA surrogate network model, preemptive priority is given to calls according to the number of times they have overflowed (seniority). Calls that have overflowed fewer times (*junior calls*) have preemptive priorities over calls that overflowed more times (*senior calls*). Then to derive the blocking probability for the surrogate model using EFPA which is expected to yield a close but somewhat different blocking probability to that of the original overflow network model and, in many cases, a better approximation to it than the one obtained by directly applying EFPA to the original model. The reason lies in the fact that in the surrogate network model, more admission opportunities are provided to the junior calls and therefore the proportion of calls that are being transmitted in its primary path increases. Since these calls do not violate the Poisson assumption, increasing their proportion can reduce the overflow error. Furthermore, by imposing preemptive priority to the surrogate model, OPCA manages to capture the dependence between trunks caused by overflows, while all the existing approximations that aim to capture the dependence between trunks only address the dependence between trunks along the same path. To the best of our knowledge, OPCA is the first method that captures the overflow dependence. Moreover, since OPCA uses an EFPA-like algorithm for its surrogate model, namely decoupling the system into independent sub-systems with Poisson arrival, OPCA is applicable to all the scenarios where EFPA is applied to, and all the enhancements of EFPA can also be implemented in OPCA to further improve the approximation.

In our circuit-switched networks with non-hierarchical deflection routing we use trunk reservation to prevent low resource utilization due to large proportion of overflowed traffic, as in [58]. In our trunk reservation policy, a certain number of channels per trunk
are reserved for primary path calls. In this way, primary path calls obtain advantage over alternative path calls in order to reduce long and inefficient routes that may be used by alternative calls. This is different from trunk reservation in e.g., [101], where channels are reserved for certain class of traffic. Trunk reservation can also mitigate the instability caused by a large number of overflowed connections in a network. Although proof of convergence to a unique solution does not exist, it is known from experience that normally circuit switched networks with alternate routing and trunk reservation and its related EFPA solutions do converge to a unique solution. In all the numerical examples that are presented in this chapter, all the algorithms of EFPA, OPCA and service-based OPCA have used trunk reservation and they all have converged to a unique solution.

In this chapter for a multi-service network, two versions of OPCA are considered. The first is a straightforward application of OPCA, where in the surrogate all calls have preemptive priority over more senior calls. For this OPCA version, it is appropriate to use the name OPCA, so we simply call it OPCA. According to the second approach, called service-based OPCA, in the surrogate network, calls have preemptive priority only over more senior calls belonging to the same class.

We compare between the results obtained by the various approximations against simulation benchmarks and explain their performance in various different scenarios and parameter value ranges. Then we discuss the insight gained into performance tradeoffs as well as design implications.

In Chapter 2, we considered two classes with the same bandwidth requirement and assumed that one class has strict priority over the other. To evaluate the blocking probability of a circuit switched network with deflection routing and these two classes of demands, we only need to consider single service traffic for the higher priority while the resource for the lower priority traffic is the leftover of the higher priority and we introduce quasi-stationary approximation to evaluate the capacity left for lower priority. Unlike Chapter 2, we consider here a general number of multiservice demands with different capacity
requirements and fair opportunity to compete for the pool of resources. We develop new
algorithms to capture the effect of mutual overflow among the classes and also discuss
reduction of the overflow error by moment matching and the relaxation of the disjointed-
ness assumption. To the best of our knowledge, it is the first time that the performance
of multiservice demands in non-hierarchical circuit switched networks with deflection
routing is studied.

The remainder of the chapter is organized as follows. In Section 4.2 we provide a
detailed description of our network model and define notation and basic concepts. Next,
in Section 4.3 we describe in detail the approximations OPCA, service-based OPCA and
EFPA as applied to our multi-service circuit switched network model. Then, in Section
4.4 we provide numerical results over a wide range of parameter values and discuss per-
formance and design implications. We also discuss and illustrate there effects of services
rates, bandwidth requirements, the number of channels per trunk, the maximum allowable
number of alternate paths and trunk reservation, as well as the sensitivity of the shape of
the holding time distribution. We also apply them in asymmetrical networks and the
Coronet and discuss their performance. Finally, the chapter is concluded in Section 4.5.

4.2 The Model

We consider a circuit-switched network described by a graph \( G(N, E) \) where \( N \) is a set of
\( n \) nodes and \( E \) is a set of \( e \) arcs that connect the nodes. The \( e \) arcs correspond to trunks
where trunk \( i \in E \) carries \( C_i \) channels. The \( N \) nodes are designated 1, 2, 3, \ldots, \( N \), each of
them has circuit switching capabilities.

In the context of a hybrid TDM/WDM network, a wavelength channel is divided into
multiple fixed length time slots to increase utilization. These time slots, multiplexed
on the wavelength, can be viewed as channels. In this case, trunk \( i \in E \) is composed
of \( f_i \) fibers, each of which supports \( w_i \) wavelengths, composed of \( h_i \) TDM time slots.
Accordingly, trunk $i \in E$ carries $C_i = f_i w_i h_i$ channels. We assume that all the nodes have full wavelength conversion capabilities and can switch traffic from any channel on one trunk to any other channel on an adjacent trunk. Note that our model and algorithms are also applicable to cases with no or partial wavelength conversion. The number of channels on trunk $i$ is $C_i = f_i h_i$ with no wavelength conversion and $f_i h_i < C_i < f_i w_i h_i$ with partial wavelength conversion depending on the conversion range [102].

Let $\Gamma$ be a set of directional Origin-Destination (OD) pairs. Every directional OD pair $m \in \Gamma$, is defined by its end-nodes. Thus, $m = \{i, j\} \in \Gamma$ represents the directional OD pair $i$ to $j$. We will distinguish between the term OD pair which is an unordered set of the two endpoints: Origin and Destination, and the directional OD pair that refers to the ordered set: Origin-Destination.

The number of different service classes of calls offered to the network is $P$. For each directional OD pair $m \in \Gamma$, calls of class $p$, $p = 1, 2, \ldots, P$, arrive according to a Poisson process with arrival rate $\lambda_{m,p}$. The number of channels that a class $p$ requires is $v_p$, also referred to as the bandwidth requirement of class $p$. The holding times of calls are assumed exponentially distributed with mean $1/\mu_{m,p}$. Let

$$\rho_{m,p} = \frac{\lambda_{m,p}}{\mu_{m,p}}$$

be the offered traffic (measured in erlangs) for directional OD pair $m$. We set

$$\rho_p = \sum_{m \in \Gamma} \rho_{m,p}.$$  

A route between origin $i$ and destination $j$ is the sequence of trunks associated with the corresponding arcs in the path between $i$ and $j$ in $G(N, E)$.

It is very likely that for a directional OD pair $m \in \Gamma$, there are multiple routes between the origin and the destination that do not share a common trunk. Such routes are often
called edge-disjoint paths or disjoint paths\cite{69,71,103}. Edge-disjoint deflection routing is often used to achieve load balancing in optical and other networks\cite{72,73,104,106}.

For each $m \in \Gamma$, we designate a route with the least number of hops as the primary path $U_m(0)$ of the directional OD pair $m$. If there are multiple routes with the least number of hops, the choice is made randomly with equal probabilities. Then considering a new topology, where the trunks of the primary path are excluded, the first alternative path for $m$ is chosen to minimize the number of hops in the new topology. Again, a tie is broken randomly. Therefore, all the primary path and alternative paths for $m$ are edge-disjoint.

Let $R_m$ be the maximum number of available alternative paths a directional OD pair $m$ can have based on the network topology.

Furthermore, a maximal number $D$ of overflow attempts to alternate paths are set for all directional OD pairs in $\Gamma$. Setting the limit $D$ implies that a connection in the directional OD pair $m$ can only use

$$R(m) = \min\{R_m, D\}$$

alternative paths. Therefore, before a connection is blocked, the procedure continues until all available and allowable $R(m)$ routes are attempted.

It is convenient to maintain the entire set

$$\{U_m(0), U_m(1), \ldots, U_m(R_m)\}$$

of alternative routes for the directional OD pair $m \in \Gamma$ in which $U_m(0)$ is the primary path and $U_m(d)$ is the $d$th alternate path. This allows for cases where $D$ do not limit the number of usable alternative path.

In our model, the ranking of alternative paths is based on the number of hops and in case of equality in the number of hops, the rank is chosen randomly. Based on our ranking, if $d_i > d_j$ then the number of hops of $U_m(d_i)$ is equal to or higher than the number of hops
in $U_m(d_j)$. However, in practice, other cost functions (e.g., geographic distance) can be also used for the ranking.

If a request for a call arrives at original node $i$ to the destination node $j$, and capacity is available on all trunks of the primary path $U_{\{i,j\}}(0)$, then this primary path will be used for the transmission of this call.

An arriving call of any class can use any free channel on any trunk. When a class $p$ call of OD pair $m$ arrivals, it can establish a connection if all trunks of its primary path have no less than $v_p$ free channels. Otherwise it will overflow to its first alternate path. Then, the procedure repeats itself. If a newly arriving call is not able to obtain a lightpath in its $R(m)$ alternate path attempts, the call is blocked and cleared of the network. Let $\beta_p$ is the set of OD pairs that are transmitting class $p$ calls, then a class $p$ call can overflow in the network at most $\max R(m), m \in \beta_p$ times, which is defined as the maximum allowable number of overflow of class $p$ connections, referred to as $D_p$.

Considering the stability of the network, and recognizing that less resources are used by a call that uses its primary path, priority is given to such calls. To facilitate such priority, a certain number of unoccupied channels are reserved for calls attempting their primary path. In particular, if the number of channels occupied on trunk $j$ is greater than or equal to a given reservation threshold $T_p$, the overflowed calls of class $p$ are not allowed to use that trunk.

### 4.3 Blocking Probability Approximations

In this section, we describe the approximations we use for blocking probability evaluation of the multiservice model. We use the term 0-call for a call transmitted on its primary path, and the term $d$-call for a call transmitted on its $d$th alternate path, for $d = 1, 2, \ldots, \max R(m)$. Accordingly, the term $(d, m, p)$-call refers to a $d$-call of class $p$ from the original node towards the destination of the directional OD pair $m$ with offered
load \(a(d,m,p)\). Assume that the arrivals of the \((d,m,p)\)-calls at trunk \(j \in U_m(d)\) follow a Poisson process with offered load \(a(d,m,p,j)\). Let \(b_{j,p}(d)\) be the blocking probability of any class \(p\) \(d\)-call offered to trunk \(j \in \mathcal{E}\).

The \((d,m,p)\)-calls occur only when \((d-1,m,p)\)-calls are blocked for directional OD pair \(m\) and for \(d = 1,2,\ldots,R(m)\). Therefore, we have

\[
    a(d,m) = a(d-1,m)(1 - \prod_{j \in U_m(d-1)} (1 - b_{j,p}(d-1))) \quad (4.1)
\]

and \(a(0,m,p) = \rho_{m,p}\). For a particular trunk along the path \(j \in U_m(d)\), we have

\[
    a(d,m,p,j) = a(d,m,p) \frac{\prod_{i \in U_m(d)} (1 - b_{i,p}(d))}{1 - b_{j,p}(d)} \quad (4.2)
\]

for \(d = 0,1,\ldots,R(m)\). For \(d > R(m)\) or \(j \notin U_m(d)\), \(a(d,m,p,j) = 0\).

Let \(a(d,j,p)\) be the total offered load of class \(p\) \(d\)-calls, on trunk \(j\). The variables \(a(d,j,p)\) and \(a(d,m,p,j)\) are related by

\[
    a(d,j,p) = \sum_{m \in \Gamma} a(d,m,p,j). \quad (4.3)
\]

Also, let \(\tilde{a}(d,j,p)\) be the total offered load of class \(p\) calls that include 0-calls, 1-calls, 2-calls \(\ldots\) \(d\)-calls, on trunk \(j\). The variables \(\tilde{a}(d,j,p)\) and \(a(d,j,p)\) are related by

\[
    \tilde{a}(d,j,p) = \sum_{i=0}^{d} a(i,j,p). \quad (4.4)
\]

### 4.3.1 EFPA

Let \(q_j(i)\) be the steady-state probability of having \(i\) channels busy in trunk \(j\). For a single trunk loaded by multiservice traffic, where each class of calls follow a Poisson process and the attributes of all the calls are independent of each other, the steady-state probabilities
have a product form solution that can be readily obtained by a recursive algorithm \[107\].

We evaluate the trunk state probability \( q_j(i), j \in E \) and \( i \in \{1, \ldots, C_j\} \) by

\[
q_j(i) = \frac{1}{i} \sum_{r=1}^{P} \left( a(0, j, r) + 1 \{ T_r > i - v_r \} \sum_{n=1}^{D_r} a(n, j, r) \right)
\times v_r \times q_j(i - v_r), \tag{4.5}
\]

where \( 1 \{ \} \) is the indicator function and \( q_j(0) \) is set such that \( \sum_{i=0}^{C_j} q_j(i) = 1 \) is satisfied.

The blocking probability, for class \( p \) traffic with \( d \) overflows, on trunk \( j \) is estimated by

\[
b_{j,p}(d) = \begin{cases} 
\sum_{i=c_j-v_p+1}^{C_j} q_j(i) & d = 0, \\
\sum_{i=T_p}^{C_j} q_j(i) & d \geq 1.
\end{cases} \tag{4.6}
\]

Equations (4.1) – (4.6) form a set of fixed-point equations which can be solved by successive substitutions.

Having obtained the results of the fixed-point equations, we calculate the blocking probabilities for class \( p \) traffic from OD pair \( m \) by

\[
B_{m,p} = 1 - \sum_{d=0}^{R(m)} a(d, m, p, j) (1 - b_{j,p}(d)) / \rho_{m,p}, \tag{4.7}
\]

where \( j \) is the last trunk in the route for the calls of OD pair \( m \) that overflow \( d \) times.

Let \( B_p \) be the network blocking probability for class \( p \) traffic, which is the average of blocking probabilities of all OD pairs, weighted by their offered load.

\[
B_p = \sum_{m \in \Gamma} B_{m,p} \times \rho_{m,p} / \sum_{m \in \Gamma} \rho_{m,p}. \tag{4.8}
\]

Algorithm 5 is used to obtain the network blocking probability \( B_p, p = 1, 2, \ldots, P \) by EFPA.
Algorithm 5 Compute $B_p$ for $p = 1, 2, \ldots, P$ by EFPA

Require: $\rho_{m,p}$ for $m \in \Gamma, p = 1, 2, \ldots, P$

initial: $b_{j,p}(d) \leftarrow 0, \hat{b}_{j,p}(d) \leftarrow 1$ for $j \in \mathcal{E}, p = 1, 2, \ldots, P, d \in \{0, \ldots, D_p\}$

while $\sum_{r=1}^P \sum_{d \in \{0, \ldots, D_r\}} \sum_{j \in \mathcal{E}} |b_{j,r}(d) - \hat{b}_{j,r}(d)| > 1e-8$ do

for $j \in \mathcal{E}, p = 1, 2, \ldots, P, d \in \{0, \ldots, D_p\}, m \in \Gamma$ do

$\hat{b}_{j,p}(d) \leftarrow b_{j,p}(d)$

compute $a(d, m, p)$ in Eq. (4.1)

compute $a(d, m, p, j)$ in Eq. (4.2)

compute $a(d, j, p)$ in Eq. (4.3)

end for

for $j \in \mathcal{E}, d \in \{0, \ldots, D_p\}$ do

compute $q_j(i)$ in Eq. (4.5) for $i \in \{1, \ldots, C_j\}$

compute $b_{j,p}(d)$ in Eq. (4.6)

end for

end while

compute $B_{m,p}$ in Eq. (4.7) for $m \in \Gamma, p = 1, 2, \ldots, P$

compute $B_p$ in Eq. (4.8) for $p = 1, 2, \ldots, P$

4.3.2 OPCA

Although in our multiservice network model, no service class traffic has priority over another service class traffic, OPCA works by using a hierarchical surrogate second system in which junior calls have preemptive priority over senior calls and estimating the blocking probability in the second system by applying an EFPA-like algorithm.

For multiservice networks, the preemptive priority of junior calls in OPCA can be operated over more senior calls belonging to any class, referred to as OPCA, or over more senior calls belonging to the same class, referred to as service-based OPCA.

OPCA in multiservice networks

In the following we provide detailed information on how to apply OPCA to the present problem of approximating blocking probability of circuit switched networks with different classes of calls.
We begin by evaluating the trunk state probability \( t_{d,j}(i) \) for each trunk \( j \in E \), for \( d \in \{0, \ldots, \text{max}R(m)\} \) deflections and each state \( i \in \{1, \ldots, C_j\} \) using

\[
t_{d,j}(i) = \sum_{r=1}^{C_j} (a(0,j,r) + 1 \{T_r > i - v_r\}) \sum_{n=1}^{d} a(n,j,r) \times v_r \times t_{d,j}(i - v_r),
\]

where \( t_{d,j}(0) \) is set such that \( \sum_{i=0}^{C_j} t_{d,j}(i) = 1 \) is satisfied \cite{107}.

The average blocking probability \( \bar{b}_{j,p}(d) \) on trunk \( j \in E \), for class \( p \) calls with up to and including \( d \) overflows, is estimated by

\[
\bar{b}_{j,p}(d) = \sum_{i=T_p}^{C_j} t_{d,j}(i).
\]

The average blocking probability \( \bar{b}_{j,p}(0) \) for class \( p \) primary calls is estimated by

\[
\bar{b}_{j,p}(0) = \sum_{i=C_j-v_p+1}^{C_j} t_{0,j}(i).
\]

The term *average blocking probability* is referred to blocking probability in a non-priority system where the junior and senior calls have the equal opportunities. Equivalently, the term *average blocked traffic* is the blocked traffic in a non-priority system which is obtained by multiplying offered load.

The actual blocking probability of class \( p \) calls, for \( d \)-calls \( (d \geq 1) \) on trunk \( j \) is estimated by

\[
b_{j,p}(d) = \bar{b}_{j,p}(d) + \sum_{r=1}^{P} \left( \sum_{n=0}^{d-1} a(n,j,r) \times (\bar{b}_{j,r}(n) - b_{j,r}(n)) \right) \times (1 - \bar{b}_{j,p}(d))
\]

\[
\sum_{r=1}^{P} a(d,j,r) \times (1 - \bar{b}_{j,r}(d))
\]

(4.12)
where the first term \( \tilde{b}_{j,p}(d) \) is the average blocking probability of \( d \)-calls \((d \geq 1)\) of class \( p \) on trunk \( j \). Note that in the surrogate model of OPCA, the junior calls have preemptive priority over senior calls belonging to any class. To consider this preemptive priority, the term \( \sum_{r=1}^{P} (\sum_{n=0}^{d-1} a(n, j, r) \times (\tilde{b}_{j,r}(n) - b_{j,r}(n))) \) is the difference between the average blocked traffic of junior calls and its actual blocked traffic (in the preemptive priority system of the OPCA surrogate). This term becomes the total blocked traffic of \( d \)-calls of all the service classes. Then we estimate the part of blocked \( d \)-call traffic of class \( p \) according to its proportion of the total \( d \)-call carried traffic, which is \( a(d, j, p)(1 - \tilde{b}_{j,p}(d))/\sum_{r=1}^{P} a(d, j, r) \times (1 - \tilde{b}_{j,r}(d)) \). Dividing the latter by the offered load \( a(d, j, r) \) yields the difference between the blocking probability of \( d \)-calls of class \( p \) and the average blocking probability which is the second term in (4.12).

For class \( p \) calls that are transmitted on their primary paths \((d = 0)\),

\[
\tilde{b}_{j,p}(0) = \tilde{b}_{j,p}(0). \tag{4.13}
\]

Algorithm 6 is used to obtain the network blocking probability \( B_p, p = 1, 2, \ldots, P \) by OPCA.

**Service-based OPCA**

For multiservice networks, results of OPCA for the surrogate model may be biased relative to the real model in some classes. It is more difficult for high-bandwidth required calls to enter the network than low-bandwidth required ones and therefore they require more overflows to establish a connection. The priorities operated by OPCA, however, worsen the acceptance of these high-bandwidth required calls which have overflowed many times and can be preempted by the junior calls of the low-bandwidth required class. This effect additionally brought by the priority of OPCA will increase the blocking probability of the high-bandwidth required traffic to a large extent when the difference of the bandwidth
Algorithm 6 Compute $B_p$ for $p = 1, 2, \ldots, P$ by OPCA

Require: $\rho_{m,p}$ for $m \in \Gamma$, $p = 1, 2, \ldots, P$

initial: $b_{j,p}(d) \leftarrow 0$, $\hat{b}_{j,p}(d) \leftarrow 1$ for $j \in \mathcal{E}$, $p = 1, 2, \ldots, P$, $d \in \{0, \ldots, D_p\}$

for $d \in \{0, \ldots, \max R(m)\}$ do
  while $\sum_{p=1}^{P} \sum_{j \in \mathcal{E}} |b_{j,r}(d) - \hat{b}_{j,r}(d)| > 1e-8$ do
    for $j \in \mathcal{E}$, $p = 1, 2, \ldots, P$, $m \in \Gamma$ do
      $\hat{b}_{j,p}(d) \leftarrow b_{j,p}(d)$
      compute $a(d,m,p)$ in Eq. (4.1)
      compute $a(d,m,p,j)$ in Eq. (4.2)
      compute $a(d,j,p)$ in Eq. (4.3)
    end for
    for $j \in \mathcal{E}$ do
      if $d == 0$ then
        compute $t_{0,j}(i)$ in Eq. (4.9) for $i \in \{1, \ldots, C_j\}$
        compute $\bar{b}_{j,p}(0)$ in Eq. (4.11)
        compute $b_{j,p}(0)$ in Eq. (4.13)
      else
        compute $t_{d,j}(i)$ in Eq. (4.9) for $i \in \{1, \ldots, C_j\}$
        compute $\bar{b}_{j,p}(d)$ in Eq. (4.10)
        compute $b_{j,p}(d)$ in Eq. (4.12)
      end if
    end for
  end while
end for
compute $B_{m,p}$ in Eq. (4.7) for $m \in \Gamma$, $p = 1, 2, \ldots, P$
compute $B_p$ in Eq. (4.8) for $p = 1, 2, \ldots, P$

requirement of the classes is large or the offered load of the low-bandwidth required class is much more than the high-bandwidth required class. Taking this into consideration, we operate this kind of priority within but not across the classes, which means the junior calls have priority over the senior calls of the same class but not the ones of different classes.

We remind the reader that the prioritization introduced in the surrogate system of service-based OPCA is also artificially introduced to obtain a more accurate approximation and it is not a feature of the real network.

In the following we provide detailed information on how to apply service-based OPCA to the present problem of approximating blocking probability of circuit switched networks with different classes of calls.
We begin by evaluating the trunk state probability \( t_{d,j,p}(i) \) of class \( p \) for each trunk \( j \in E \), for \( d \in \{0, \ldots, D_p\} \) deflections and each state \( i \in \{1, \ldots, C_j\} \) using

\[
t_{d,j,p}(i) = \frac{v_r}{i} \sum_{r \in \{1, \ldots, P\}, r \neq p} a(0, j, r) \times t_{d,j,p}(i - v_r) + \frac{v_r}{i} \sum_{r \in \{1, \ldots, P\}, r \neq p} \{1 \{T_r > i - v_r\} \sum_{n=1}^{D_r} a(n, j, r) \} \times t_{d,j,p}(i - v_r) + \frac{v_p}{i} (a(0, j, p) + 1 \{T_p > i - v_p\} \sum_{n=1}^{d} a(n, j, p) \} \times t_{d,j,p}(i - v_p),
\] (4.14)

where \( t_{d,j,p}(0) \) is set such that \( \sum_{i=0}^{C_j} t_{d,j,p}(i) = 1 \) is satisfied [107].

The average blocking probability \( \bar{b}_{j,p}(d) \) on trunk \( j \in E \), for class \( p \) calls with up to and including \( d \in \{0, \ldots, D_p\} \) overflows, is estimated by

\[
\bar{b}_{j,p}(d) = \frac{\sum_{n=1}^{d} (a(n, j, p) \sum_{i=T_p}^{C_j} t_{d,j,p}(i))}{\bar{a}(d, j, p)} + \frac{a(0, j, p) \sum_{i=C_j-v_p+1}^{C_j} t_{d,j,p}(i)}{\bar{a}(d, j, p)}. \tag{4.15}
\]

The blocking probability of class \( p \) calls, for \( d \)-overflows calls, \( d \in \{0, \ldots, D_p\} \), on trunk \( j \) is estimated by

\[
b_{j,p}(d) = \begin{cases} \bar{b}_{j,p}(0) & d = 0, \\ \frac{\bar{b}_{j,p}(d) \bar{a}(d, j, p) - \bar{b}_{j,p}(d-1) \bar{a}(d-1, j, p)}{\bar{a}(d, j, p)} & 1 \leq d \leq D_p. \end{cases} \tag{4.16}
\]

Algorithm 7 is used to obtain the network blocking probability \( B_p, p = 1, 2, \ldots, P \) by service-based OPCA.
Algorithm 7 Compute $B_p$ for $p = 1, 2, \ldots, P$ by service-based OPCA

Require: $\rho_{m, p}$ for $m \in \Gamma$, $p = 1, 2, \ldots, P$

initial: $b_{j, p}(d) \leftarrow 0$, $\hat{b}_{j, p}(d) \leftarrow 1$ for $j \in E$, $p = 1, 2, \ldots, P$, $d \in \{0, \ldots, D_p\}$

while $\sum_{r=1}^P \sum_{d \in \{0, \ldots, D_r\}} \sum_{j \in E} |b_{j, r}(d) - \hat{b}_{j, r}(d)| > 1e-8$ do

for $j \in E$, $p = 1, 2, \ldots, P$, $m \in \Gamma$, $d \in \{0, \ldots, D_r\}$ do

\[\hat{b}_{j, p}(d) \leftarrow b_{j, p}(d)\]

compute $a(d, m, p)$ in Eq. (4.1)

compute $a(d, m, p, j)$ in Eq. (4.2)

compute $a(d, j, p)$ in Eq. (4.3)

compute $\bar{a}(d, j, p)$ in Eq. (4.4)

end for

for $j \in E$, $d \in \{0, \ldots, D_p\}$, $p = 1, 2, \ldots, P$ do

compute $t_{d, j, p}(i)$ in Eq. (4.14) for $i \in \{1, \ldots, C_j\}$

compute $\bar{b}_{j, p}(d)$ in Eq. (4.15)

compute $b_{j, p}(d)$ in Eq. (4.16)

end for

end while

compute $B_{m, p}$ in Eq. (4.7) for $m \in \Gamma$, $p = 1, 2, \ldots, P$

compute $B_p$ in Eq. (4.8) for $p = 1, 2, \ldots, P$

4.3.3 The trunk offered load in non-disjoint path cases

We now consider cases where the primary and the alternative paths of each OD pair are not necessarily disjoint, i.e., they may contain some common trunks. To calculate the blocking probability in such cases, notice that all we need is to calculate the traffic offered of each trunk. After that, the remaining procedure can be completed exactly as in the previous cases with disjoint paths for all the algorithms: EFPA, OPCA and service-based OPCA.

In the case of disjoint paths, the offered load to an alternative path is simply a function of the probability of one or more trunks in the primary path being congested. However, in the case of non-disjoint path, a blocked trunk in one path immediately implied that all alternative paths using this trunk are blocked, so the offered load contribution to an alternative path from another path that shares a blocked common trunk with it is zero. Therefore, derivation of the total offered traffic to an alternative path, from a previous path requires conditioning that all the trunks common to both paths are not blocked. This
creates a large number of overflow events that need to be considered when deriving the offered traffic to each alternative path. This introduces a significant complexity in writing different equations for the offered load of each trunk. This complexity significantly increases with the size of the network and D. There is no fundamental difficulty in evaluating blocking probability for these cases but the difficulty is caused by the complexity due to the large number of cases that must be considered.

This is illustrated in the following example.

![Diagram of two paths and three paths](image)

Figure 4.1: Examples of non-disjoint paths.

We take the scenario of Fig. 4.1(a) for example when we transmit data from node $N_1$ to node $N_4$ along two paths. The primary path contains trunks $L_1$, $L_2$ and $L_3$ and the first alternative path contains trunks $L_1$, $L_4$, $L_5$ and $L_3$. They contain common trunks $L_1$ and $L_3$. We first transmit data along the primary path and the first alternative will be used only in the case that $L_2$ is congested while $L_1$ and $L_3$ still have free channels. For the cases that at least one of the trunks $L_1$ and $L_3$ congested, both the primary path and the first alternative path fail. Therefore, the offered load to the first alternative path is

$$a(1, m, p) = \rho_{m, p} b_{2, p}(0)(1 - b_{1, p}(0))(1 - b_{3, p}(0)).$$  \hspace{1cm} (4.17)

Then we consider the scenario of Fig. 4.1(b), in which a second alternative containing trunks $L_1$, $L_4$, $L_6$ and $L_7$ is added to the case of Fig. 4.1(a). The second alternative path has a common trunk $L_1$ with the primary path and two common trunks $L_1$ and $L_4$ with the
first alternative path. The offered load to the second alternative path can be introduced by
the congestion of trunk \( L_3 \) or/and the simultaneous congestion of \( L_2 \) and \( L_5 \).

In this case, we have

\[
a(2, m, p) = \rho_{m, p}(1 - b_{1, p}(0))(1 - b_{4, p}(1)) \\
\times ((b_{3, p}(0)) + (1 - b_{3, p}(0))b_{2, p}(0)b_{5, p}(1)).
\]

Equation (4.18) demonstrates how the offered load on alternate path is derived conditioning on the congestion state of the common trunks. For a realistic network such equations need to be individually written for each alternative path. By comparison, equation (4.1) applies to all alternate paths in the network. Nevertheless, the calculation procedure of trunk blocking probability and network blocking probability are the same with disjoint paths. A numerical example with non-disjoint paths is presented in Section 4.4-L. We assume disjoint paths in default scenario for simplicity. Those who are interested in the application to the cases with non-disjoint paths can calculate the offered load on each trunk as we do in the examples and follow the remaining steps of the algorithms.

### 4.3.4 The max(EFPA, service-based OPCA) approximation

The accuracy of the above three approximations depends on the combined effect of overflow error and path error introduced, in which overflow error causes underestimation and path error causes overestimation. Both of the errors can be affected by different network parameter values, which therefore affect the accuracy of the approximations and make different approximations the most accurate under different scenarios. The difference in behavior of EFPA versus service-based OPCA under different scenarios give rise to a new approximation based on choosing the maximal value of the EFPA and service-based OPCA blocking probability approximations designated. As demonstrated empirically in Section 4.4 can lead to an accurate approximation in most scenarios, and almost always it seems to be a conservative approximation.
4.4 Numerical Results

In this section, we compare the performance of OPCA, service-based OPCA and EFPA in approximating the network blocking probabilities of multiservice classes. To this end, we will consider a wide range of scenarios. However, one scenario, which we call a default scenario where the offered load for each OD pair is the same and has a symmetrical network topology will receive much attention. In all cases considered, we also provide intuitive explanations to the discrepancies between the approximations and simulation results for the network blocking probabilities as they vary according to various effects. In particular, we consider effects such as the effect of the service rates and bandwidth requirements of both classes. We then consider design factors such as: the number of channels per trunk, the maximum allowable number of alternate paths, and the effect of trunk reservation. We also discuss the robustness of the approximations to the shape of the holding time. We will consider symmetric and asymmetric scenarios, and networks of various topologies, including the Next Generation Core Optical Network (CORONET).

The performance results are compared based on simulations unless the running times are prohibitive. Error bars for the 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. In any case, the length of the confidence interval is always less than 10% of the mean value measured.

4.4.1 Default scenario

There is one network scenario that we repeatedly use in many experiments with the same set of parameter values, or possibly with small variations. It is convenient to present it once in this section as a default scenario and throughout the section only to point out the deviations from this default scenario.
Our default scenario is a 2-class 6-node fully meshed network where each trunk has 50 channels. Its traffic load is characterized by call arrivals of both classes following Poisson processes, and the holding time of both classes are exponentially distributed with mean holding time equal to 1. In our 6-node fully meshed network, there are in total 15 different OD pairs (or equivalently 30 directional OD pairs). The threshold of class 1 traffic is 38 channels (76% of trunk capacity) and the threshold of the class 2 traffic is 40 channels (80% of trunk capacity). The maximum allowable number of alternate paths is set to 4 for both classes. The bandwidth requirements are 2 and 5 for class 1 and class 2 calls, respectively.

In our default scenario, the traffic for all directional OD pairs is the same, in which case for each directional OD pair, the offered arrival rates of class 1 and class 2 are $\rho_1$ and $\rho_2$, respectively. Then, the total arrival rate in the network is $30(\rho_1 + \rho_2)$.

### 4.4.2 Network Blocking probabilities for the classes

![Figure 4.2: Network blocking probabilities for (a) class 1 calls and (b) class 2 calls. The offered load of class 2 is 1 Erlang.](image)

We first consider the default scenario. In Fig. 4.2, we present results for the network blocking probabilities obtained by OPCA, service-based OPCA, EFPA and simulation for class 1, as a function of class 1 offered load. We observe in the figure that all the three
approximations tend to underestimate the network blocking probability when the offered load is low. This is due to the fact that in a fully meshed network with low traffic load, and therefore less overflows, long paths will be very rare. Note that in a fully meshed network the primary path contains only one trunk and hence does not introduce any path error. Accordingly, overflow error will dominate path errors causing underestimation of network blocking probability.

In the surrogate model of OPCA, where the maximum allowable number of overflow is \( D \), when a junior call, which has overflowed \( d_1 \) times, encounters and preempts a senior call, which has overflowed \( d_2 \) times and \( d_1 < d_2 \), the senior call is overflowed as a result of the contention, but its remaining number of allowable overflows is limited to no more than \( D - d_2 \). In the real model under the same circumstances, the junior call will overflow and its remaining number of allowable overflows is \( D - d_1 \) times, which is more than the allowable number of the overflowed call in the surrogate model of OPCA. The preemptive priority of junior calls over senior calls in OPCA and service-based OPCA implies smaller number of allowable overflows and therefore less proportion of overflowed traffic in the total offered load in the network \([53]\). Since the surrogate model of OPCA gives preemptive priority to new calls over overflowed calls of any class, while service-based OPCA gives preemptive priority to new calls over overflowed calls of the same class, new calls in the surrogate model of OPCA obtain higher level of priority than those in the surrogate model of service-based OPCA. Therefore, the surrogate model of OPCA exhibits lower proportion of overflow traffic than the service-based OPCA, and the service-based OPCA exhibits lower proportion of overflow traffic than the original (real) model where no priority is given to junior calls. This leads to lower overflow error and therefore lower underestimation of network blocking probability in this case for class 1 traffic of OPCA than service-based OPCA and of service-based OPCA than EFPA. This explains higher estimation of network blocking probability by OPCA than by service-based OPCA, and the lowest estimation of EFPA.
Furthermore, we observe that as the traffic load increases, the underestimation by all the network blocking probability approximation methods is reduced. This is consistent with the fact that in high load, overflow probability increases. Then, more and more overflows imply the use of longer and longer alternate paths, and therefore path error increases. As observed, the path error in cases of high traffic load may cancel out the overflow error and in this way may improve the approximations. Because EFPA exhibits higher path error than OPCA and service-based OPCA, EFPA may outperform them as the offered load increases, as shown in Fig. 4.2(a). Since service-based OPCA is more accurate than EFPA when the traffic load is small and EFPA is generally more accurate than service-based OPCA when the traffic load is heavy, we conservatively consider max(EFPA, service-based OPCA) as our approximation of choice rather than EFPA and service-based OPCA over the whole range of traffic load.

In Fig. 4.2(b), we present results for the network blocking probability obtained by OPCA, service-based OPCA, EFPA and simulations for class 2 traffic and we observe certain similar performance behaviors and trends of the approximations as observed for the class 1 traffic. One noticeable difference is that the network blocking probability obtained by service-based OPCA is more accurate for class 2 traffic than for that of class 1. This is because the bandwidth requirement of class 2 overflow traffic is larger than that of class 1 overflow traffic, and therefore it is more difficult for overflowed traffic of class 2 to find free channels to transmit once it is preempted. Therefore, in the surrogate model of service-based OPCA, network blocking probability of class 2 traffic is higher, closer to the results obtained by simulation in this case.

We also observe that the network blocking probability by OPCA exceeds the simulation result as the traffic increases. In the surrogate model of OPCA, where the large overflowed traffic of class 2 can be preempted by the class 1 calls that require lower-bandwidth, the performance of the more bandwidth hungry class will be lower and the network blocking probability of class 2 traffic predicted by OPCA is further increased to
more than the result obtained by simulation. This inaccuracy is more severe when the
offered load of class 1 traffic that requires lower bandwidth in this case is larger than
that of class 2 traffic that requires higher bandwidth, as shown in Fig. 4.3(b). We ob-
serve that when the ratio of offered load of class 1 and that of class 2 is 10:1, the result of
OPCA is significantly more than that of the simulation. This increase in network blocking
probability evaluation of OPCA also happens if the bandwidth requirement of class 2 far
exceeds that of class 1, as shown in Fig. 4.4(b). As demonstrated, OPCA can significantly
overestimate the network blocking probability under the scenarios when the offered load
of the class that require low-bandwidth far exceeds that of the class that requires high-
bandwidth, or when the difference of bandwidth requirements by the two classes is large.
The high sensitivity of network blocking performance of OPCA to these parameters ad-
versely affects its robustness in the case of multiservice circuit switched networks and
therefore OPCA will not be further considered in this section.

Here we consider an NSF network with 13 nodes and 16 bidirectional trunks, as shown
in Fig. 4.5 and the maximum allowable number of alternate paths $D = 1$ for each OD pair
in this network. In Fig. 4.6 we present the network blocking probabilities for class 1
and class 2 traffic in the NSFNet while maintaining all the other parameter values as in
Figure 4.4: Network blocking probabilities for (a) class 1 calls and (b) class 2 calls. The bandwidth requirement of class 2 is 8.

Figure 4.5: 13-node NSFNet topology.

the default scenario. Although very close to each other, we observe that service-based OPCA still outperforms EFPA a little for both class 1 and class 2 traffic. For the blocking probability for class 2 traffic, OPCA also exceeds the simulation results with the increased offered load of class 1, which is similar to their behaviors in the 6-node fully meshed network.

4.4.3 The effect of multi-service rate

We illustrate here the effect of multi-service rate on network blocking probabilities. By increasing both the arrival rate and service rate of class 1 three times while keeping other parameter values unchanged in the default scenario, Fig. 4.7 shows the network blocking
Figure 4.6: Network blocking probabilities for (a) class 1 calls and (b) class 2 calls in NSFNet. The offered load of class 2 is 0.05 Erlang.

Figure 4.7: Network blocking probabilities for (a) class 1 calls and (b) class 2 calls. The offered load of class 2 is 1 Erlang. The service rate of class 1 and class 2 are 3 and 1, respectively.

probabilities of both classes obtained by service-based OPCA, EFPA and simulation for the multi-service rate scenario, compared to the simulation results of the default scenario in Section 4.4.2. We observe that in this case, the simulation results are very close to each other and their confidence intervals are overlapped which shows that the network blocking probabilities of the classes are insensitive to the service rate as long as the offered load remains the same. For service-based OPCA and EFPA, according to the state probability equations 4.5 and 4.14 in Section 4.3, the results only depend on the offered load on
the trunk and therefore, we can ignore the effect of service rate on the network blocking probabilities.

4.4.4 The effect of bandwidth requirement of both classes

![Graphs showing network blocking probabilities for different classes with varying bandwidth requirements.](image)

Figure 4.8: Network blocking probabilities for both classes. The offered load for class 2 is 1 Erlang. The bandwidth requirements of class 1 are 1 for (a) and (b), 4 for (c) and (d). The bandwidth requirement of class 2 remains 5.

Fig. 4.8 show the network blocking probabilities of both classes when the bandwidth requirements of class 1 connections are equal to 1 for (a) and (b) and to 4 for (c) and (d), respectively, while all the other parameter values are kept the same as in the default scenario.

Comparing Figs. 4.8(a) and (c), we observe that because of the preemptive priority (within a class) of service-based OPCA, overflowed traffic receives fewer opportunities to be admitted by service-based OPCA than by EFPA, and therefore the network blocking
Figure 4.9: Network blocking probabilities for both classes. The offered load for class 1 is 6 Erlangs. The bandwidth requirements of class 2 are 3 for (a) and (b), 8 for (c) and (d). The bandwidth requirement of class 1 remains 2.

The network blocking probability estimated by service-based OPCA is higher than by EFPA when the offered load is light and the network blocking probability is realistically acceptable (less than 0.001). When the offered load is sufficiently heavy, the network blocking probabilities by EFPA increases due to the increasing path error effect and can be higher than that of service-based OPCA. However, this effect normally occurs in the range when the network blocking probability is so high that it is beyond our region of interest.

When the bandwidth requirement of class 1 increases, the network blocking probability of class 1 estimated by service-based OPCA will further increase because it will be more difficult for the overflowed traffic of class 1 to find an available alternate path to complete service after it is preempted in the surrogate model of service-based OPCA. This in turn leads to the reduction of class 2 network blocking probability estimated by
service-based OPCA since the two traffic classes compete for the same pool of capacity, as shown in Figs. 4.8(b) and (d). We also observe that with no priority in the original model and in EFPA, network blocking probabilities by EFPA are not so sensitive to bandwidth requirement changes as service-based OPCA.

Fig. 4.9 show the network blocking probabilities of both classes when the bandwidth requirements of class 2 traffic are 3 for (a) and (b) and 8 for (c) and (d) while all the other parameter values are kept the same as in the default scenario in 4.4.1. We observe that with the increased bandwidth requirement of class 2, network blocking probability of class 2 traffic by service-based OPCA will increase and that of class 1 traffic will decrease, which is consistent with the figures in Fig. 4.8.

4.4.5 The effect of the number of channels per trunk

To examine the effect of the number of channels (wavelength channels) per trunk on network blocking probabilities and on the accuracy of EFPA and service-based OPCA, we increase now the number of channels per trunk to 100 in the default scenario we consider above. Accordingly, we set the thresholds 76 (76%) and 80 (80%), for class 1 and class 2, respectively, while all the other parameter values are kept the same as in the default scenario.

In Fig. 4.10, we provide the results obtained for the network blocking probabilities of the two traffic classes in the default scenario with 100 channels in each trunk. We observe that the accuracy of both service-based OPCA and EFPA are improved compared to the case of 50 channels per trunk shown in Fig. 4.2. The improvement in accuracy is achieved because of the following reasons.

1. When the number of channels per trunk increases, the variance of the overflow traffic decreases, leading to a lower Poisson error.
Figure 4.10: Network blocking probabilities for the classes in the default scenario with 100 channels each trunk. The offered load of Class 2 is 3 Erlangs.

2. The increase of the number of channels per trunk also reduces the proportion of overflowed traffic and therefore reduces the overflow error, which also increases the accuracy of EFPA.

We also observe that, in general, service-based OPCA is superior to EFPA and it is sandwiched between EFPA and the simulation results.

Notice also that for the network blocking probabilities evaluation of both classes, service-based OPCA still outperforms EFPA in the case of 100 channels per trunk when the traffic is light. This together with the improved accuracy of EFPA as the number of channels per trunk increases from 50 to 100, provide some evidence that service-based OPCA can be accurate as the network capacity scales upwards and performs even better than for networks with lower capacity.

4.4.6 The effect of maximum allowable number of alternate paths

Here we examine how the network blocking probabilities are affected by the maximum allowable number of alternate paths $D$ which limits how many times traffic can overflow. Traffic that already overflowed $D$ times is not allowed to overflow again and will be blocked and cleared from the network. For single class networks with light traffic, on one
hand, increasing $D$ means more opportunities to overflow which may reduce the network blocking probabilities, but on the other hand, increasing $D$ implies that calls use longer paths in alternate routes which leads to inefficiency which in turn may even increase the network blocking probabilities especially when the network is congested. In general, the maximum number of allowable alternative paths $D$ should be set appropriately to reserve channels for the primary path traffic and prevent the network from being congested by overflowed calls that take long routes.

Fig. 4.11(a) and (b) demonstrates the effect of maximum allowable number of alternate paths on the network blocking probabilities of both classes obtained by simulation, EFPA and service-based OPCA. The offered traffic load of class 1 and class 2 are 5 Erlangs and 1 Erlang, respectively. We change the maximum allowable number of alternate paths, while keeping all the other parameter values the same as in the default scenario.

We observe that there is a clear benefit for both classes, in the present scenario case, to increase $D$ to at least 3. After that, the rate of decrease in the network blocking probabilities of both classes slow down as $D$ increases, due to the inefficiency caused by the long alternate paths.
Figure 4.12: Network blocking probabilities of both classes in the default scenario. The threshold of class 2 $T_2$ remains 40 (80%).

Figure 4.13: Network blocking probabilities of both classes in the default scenario. The threshold of class 1 $T_1$ remains 38 (76%).

### 4.4.7 The effect of trunk reservation

We again consider the default scenario with the offered traffic load equal to 5 Erlangs and 1 Erlang for class 1 and class 2, respectively. We change the threshold of class 1, $T_1$, while keeping all the other parameter values the same as in the default scenario.

For this case, Fig. 4.12(a) illustrates the effect of $T_1$ on the network blocking probability of class 1 traffic.

We observe that increasing $T_1$, in the present case, reduces the network blocking probability of class 1 traffic by allowing overflowed traffic of class 1 to use more resources,
which in turn increases the network blocking probability of class 2 because they compete for the same pool of capacity, as shown in Fig. 4.12(b). Fig. 4.13 shows the network blocking probabilities when as we vary $T_2$, with the similar trends and behaviors as the Fig. 4.12.

### 4.4.8 The effect of the shape of the holding time distribution

![Figure 4.14: Network blocking probabilities of both classes, considering different service time distributions in the default scenario. The offered load for class 2 is 1 Erlang.](image)

The results presented above are based on the assumption that the holding times of the traffic of both classes are exponentially distributed. It is therefore important to examine the robustness of the approximations to the shape of the holding time distribution. To this end, we compare the results obtained under the exponential assumptions versus results obtained under heavy-tailed holding time distribution, where we maintain the same mean for the two alternatives. The use of heavy-tailed holding times are justified because such connections may represent traffic demands associated with individual application flows, and it has been established that Internet flow size distributions are heavy tailed [108–110].

In particular, we consider our heavy-tailed holding times, denoted $h$, to follow a Pareto distribution with a complementary distribution function (CDF) that takes the form:
\[
Prob(h > x) = \begin{cases} 
(\delta / x)^{\gamma}, & x \geq \delta \\
1, & \text{otherwise.}
\end{cases}
\] (4.19)

where \( \delta \) (seconds) is the scale parameter (minimum holding time) and \( \gamma \) is the shape parameter of the Pareto distribution. The mean of \( h \) is given by

\[
E(h) = \begin{cases} 
\infty, & 0 < \gamma \leq 1 \\
\delta \gamma / (\gamma - 1), & \text{otherwise.}
\end{cases}
\] (4.20)

For \( 0 < \gamma \leq 2 \), the variance \( Var(h) = \infty \). In our simulation we set \( \delta = 0.5 \) and \( \gamma = 2 \) for both classes. All the other parameter values are kept the same as in the default scenario.

Fig. 4.14(a) and (b) show the simulation results for network blocking probabilities of both classes traffic for the default scenario with the following four cases of service distribution:

1. Exp-Exp – holding times of both classes are exponentially distributed
2. Exp-Pareto – holding time of class 1 is exponentially distributed while that of class 2 is Pareto distributed
3. Pareto-Pareto – holding times of both classes are Pareto distributed
4. Pareto-Exp – holding time of class 1 is Pareto distributed while that of class 2 is exponentially distributed.

The four curves are very close to each other and their confidence interval overlap, which shows that network blocking probabilities of both classes are not very sensitive to the shape of the holding time distribution in the present case.
4.4.9 The asymmetrical cases

All the results we have presented are for the symmetrical models, where for each OD pair traffic is sent for both classes, the offered loads for all OD pairs are identical, and the network topology is symmetrical as well. However, in reality, core network topologies are normally not symmetrical and the traffic between some OD pairs have a very different profile than others, because the OD pairs can be very different, e.g., data-centers, core routers, or LHCOPN node, with different traffic profiles.

Therefore, in this section, we study the performance of EFPA and service-based OPCA in asymmetrical cases.

![Network blocking probabilities for both classes](image)

Figure 4.15: Network blocking probabilities of both classes, while 20 OD pairs will only send the class 1 traffic and the rest 10 OD pairs will only transmit class 2 traffic in a 6-node fully-meshed network with 50 channels in each trunk. The offered load for class 2 is 1 Erlang.

For the 6-node fully meshed network, the total 30 OD pairs are divided into two groups, in which 20 OD pairs only transmit class 1 traffic, and the remaining 10 OD pairs will only transmit class 2 traffic. All other parameter values are the same as in the default scenario.

We observe that in this case, the network blocking probabilities estimated by EFPA and service-based OPCA, as shown in Fig. 4.15 are closer to each other than in the symmetrical case shown in Fig. 4.2. This is because the service-based OPCA can benefit
more from the congestion information exchanged when senior calls are preempted, in the symmetrical case than in the asymmetrical case. This benefit is more prominent in symmetrical network because with evenly distributed offered load, all the trunks have overflowed calls from all the other trunks and the congestion information of all trunks spreads efficiently to all the other trunks in the network. The asymmetry reduces the advantage of the service-based OPCA and makes its results closer to those of EFPA. Nevertheless, we still observe that service-based OPCA presents better results than EFPA also in this asymmetrical case.

We further study the performance of EFPA and service-based OPCA in a 13-node NSFNet that has both asymmetrical offered load and asymmetrical topology, shown in Fig. 4.5. We choose all possible OD pairs with shortest path routing, where a tie is broken randomly. There are $12 \times 13 = 156$ OD pairs in the 13-node NSFNet and we set randomly chosen 104 OD pairs out of the total of 156 to only send class 1 traffic, and the remaining 52 send only class 2 traffic, while keeping all the other parameter values the same as in the default scenario.

![Figure 4.16: Network blocking probabilities of both classes, while 104 OD pairs will only send the class 1 traffic and the rest 52 OD pairs will only transmit class 2 traffic in 13-node NSFNet with 50 channels in each trunk. The offered load for class 2 is 0.1 Erlang.](image)

We observe similar results in Fig. 4.16, where EFPA and service-based OPCA are very close to each other but service-based OPCA still outperforms EFPA slightly.

107
4.4.10 The CORONET

We present here that max(EFPA, service-based OPCA) is applicable to large scale networks such as the Coronet, shown in Fig. 4.17 while simulation results are computationally prohibitive for such large scale networks. With all of the 9900 OD pairs in the network and one alternative path for each OD pair, the network blocking probabilities for both classes traffic can be obtained by service-based OPCA and EFPA within reasonable running times, as shown in Fig. 4.18. The running times used to calculate the network blocking probabilities in the Coronet are about 33.31 seconds and 41.73 seconds by EFPA and service-based OPCA, respectively, obtained using MATLAB 7.6.0 executed on a desktop PC with IntelR CoreTM 2 Quad @ 3 GHz CPU, 4 GHz RAM and 32-bit operating system.

4.4.11 The three classes case

All the results we have presented so far are for the cases with two service classes in different scenarios. Here we present the cases with three service classes.

We again consider a 6-node fully meshed network where each trunk has 50 channels. The bandwidth requirements are 2, 4 and 5 for class 1, class 2 and class 3 calls, respec-
Figure 4.18: Network blocking probabilities for (a) class 1 calls and (b) class 2 calls. The offered load of class 2 is 0.001 Erlang.

Figure 4.19: Network blocking probabilities for (a) class 1 calls, (b) class 2 calls and (c) class 3 calls. The offered load of class 1 and class 2 are 3 Erlangs and 1 Erlang, respectively. The thresholds are 38, 42 and 43 for class 1, class 2 and class 3 traffic, respectively.

The maximum allowable number of alternate paths is set to 4 for all classes. We observe in Fig. 4.19 that blocking probability for class 3 calls obtained by service-based OPCA is much higher than obtained by EFPA and very close to the simulation results. This is because the bandwidth requirement for class 3 calls are much larger and the overflowed class 3 calls will hardly be served again in the surrogate model of service-based OPCA.
which leads to the high blocking probability by service-based OPCA. As a result, many resources are left for class 1 and class 2 calls, which causes relatively low blocking probability of class 1 and class 2 calls by service-based OPCA. Nevertheless, service-based OPCA still outperforms EFPA in general.

4.4.12 The cases with non-disjoint paths

Here we present the results of our algorithms when we relax the disjointedness assumption and the paths of a same OD pair contain some common trunks. As discussed, the only difference between evaluating blocking probability in this case and in the cases based on disjoint paths is the computation of the trunk offered load, which are affected by the common trunks and their positions along the paths and therefore need to be calculated case by case.

![Network blocking probabilities for (a) class 1 and (b) class 2 calls. The offered load of class 2 is 1 Erlang.](image)

Figure 4.20: Network blocking probabilities for (a) class 1 and (b) class 2 calls. The offered load of class 2 is 1 Erlang.

In the default scenario described in Section 4.4.1, we assign five disjoint paths for each OD pair. To relax the disjointedness assumption, we add two alternative paths to each OD pair and they both have two common trunks which are also contained in the five paths. Fig. 4.20 show that in this case, when we relax the disjointedness assumption, the
results obtained by service-based OPCA are still more accurate than by EFPA most of time, and max(EFPA, service-based OPCA) is still applicable to this case involving non-disjoint paths. If we consider an approximation, in which the non-disjoint paths of a same OD pair are treated as if they were disjoint, the results of both EFPA and service-based OPCA will be lower, as shown in Fig. 4.20. This is because we ignore the dependency and give the alternative paths more overflowed offered load that in fact should be blocked, and in this way, we underestimate the network blocking probabilities.

4.4.13 Moment Matching

As discussed, traffic offered by an overflow stream is known to have higher peakedness than a Poisson process. The error introduced by assuming them to be Poisson processes can be reduced by moment matching [33, 39, 54], which resort to processes whose moments match those of the overflow streams. Here we implement one of the moment matching approaches in [39] in both EFPA and service-based OPCA.

Table 4.1: Network blocking probability for class 1 in 6-node fully mesh network with moment matching implemented in EFPA and service-based OPCA.

<table>
<thead>
<tr>
<th>Offered load of class 1</th>
<th>EFPA</th>
<th>EFPA with moment matching</th>
<th>service-based OPCA</th>
<th>service-based OPCA with moment matching</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.3</td>
<td>0.0000341</td>
<td>0.00000366</td>
<td>0.00000490</td>
<td>0.00000527</td>
</tr>
<tr>
<td>4.8</td>
<td>0.0002106</td>
<td>0.0002361</td>
<td>0.0002889</td>
<td>0.0003149</td>
</tr>
<tr>
<td>5.2</td>
<td>0.0007877</td>
<td>0.0008884</td>
<td>0.0009953</td>
<td>0.0010929</td>
</tr>
<tr>
<td>5.5</td>
<td>0.0019289</td>
<td>0.0020963</td>
<td>0.0022383</td>
<td>0.0024608</td>
</tr>
<tr>
<td>6.5</td>
<td>0.0167839</td>
<td>0.0185148</td>
<td>0.0155129</td>
<td>0.0163552</td>
</tr>
<tr>
<td>8</td>
<td>0.0650663</td>
<td>0.0670232</td>
<td>0.0603895</td>
<td>0.0618303</td>
</tr>
<tr>
<td>9</td>
<td>0.1010129</td>
<td>0.1018540</td>
<td>0.0963064</td>
<td>0.0980423</td>
</tr>
</tbody>
</table>

Tables 4.1 and 4.2 show the results obtained by EFPA and service-based OPCA with moment matching with increase of class 1 offered load while the offered load of class 2
Table 4.2: Network blocking probability for class 2 in 6-node fully mesh network with moment matching implemented in EFPA and service-based OPCA.

<table>
<thead>
<tr>
<th>Offered load of class 1</th>
<th>EFPA</th>
<th>EFPA with moment matching</th>
<th>service-based OPCA</th>
<th>service-based OPCA with moment matching</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.3</td>
<td>0.0000315</td>
<td>0.0000341</td>
<td>0.0001243</td>
<td>0.0001344</td>
</tr>
<tr>
<td>4.8</td>
<td>0.0002201</td>
<td>0.0002499</td>
<td>0.0006679</td>
<td>0.0007335</td>
</tr>
<tr>
<td>5.2</td>
<td>0.0009122</td>
<td>0.0010449</td>
<td>0.0021566</td>
<td>0.0023869</td>
</tr>
<tr>
<td>5.5</td>
<td>0.0024201</td>
<td>0.0026637</td>
<td>0.0046690</td>
<td>0.0051738</td>
</tr>
<tr>
<td>6.5</td>
<td>0.1295776</td>
<td>0.1343897</td>
<td>0.1259148</td>
<td>0.1292402</td>
</tr>
<tr>
<td>8</td>
<td>0.2130164</td>
<td>0.2159933</td>
<td>0.2060681</td>
<td>0.2103931</td>
</tr>
</tbody>
</table>

remains 1 Erlang. Comparing with the results obtained by the original EFPA and service-based OPCA, we observe small increase of accuracy in both EFPA and service-based OPCA by moment matching. However, the benefit is not so obvious due to the other assumptions that cause errors in the approximations, which is consistent with results in [53].

4.4.14 The effect of setup delay

Here we aim to investigate the effect of setup delay on the network blocking probability results of max(EFPA, service-based OPCA). The dependence of this effect on $D$ is also studied because deflected paths are often longer, so the setup delay becomes more significant as $D$ increases. Setup delay in circuit switched networks also depends on the end-to-end propagation delay and the handshaking algorithm during setup. The effect of setup delay on network blocking probability can be significant if the ratio of mean service duration per connection to the propagation delay is small because during part of the setup time, capacity is already reserved for the connection even though it is not yet used and cannot be used by other connections. In this subsection, we assume that for a
given connection, a setup delay of twice the propagation delay plus twice the processing delay to each node in a path is a conservative upper bound for the period that the entire path is reserved for the connection, it is neither used by this connection, nor by any other connections. This is a conservative assumption because it includes the time that a control packet travels to make reservations before actual reservations (and confirmations) are made. Our approach is to evaluate the network blocking probabilities twice, we use \( \text{max}(\text{EFPA, service-based OPCA}) \) to approximate the scenarios once the setup delay is added and once where it is ignored. The former gives us an upper bound for the blocking probability and the latter a lower bound. For each of the scenarios discussed below, for the cases where the setup delay is not included, we increase the mean service duration and reduce the arrival rate at the same rate so that the offered traffic and the blocking probability remain constant. Then for each of the above cases we also provide the blocking probability where setup delay is included.

Figure 4.21: Network blocking probabilities for (a) class 1 and (b) class 2 calls. The offered load are 5.5 erlangs and 1 erlang of class 1 and class 2, respectively. The propagation delay is set 0.001 second on every trunk and the processing delay is 0.0001 second in each node.

Fig. 4.21 provides the upper and lower bounds of network blocking probabilities for class 1 and class 2 calls based on \( \text{max}(\text{EFPA, service-based OPCA}) \) for the default scenario when the maximum number of alternative paths \( D = 4 \). The propagation delay is
set 0.001 second on every trunk and the processing delay is 0.0001 second in each node.

As the mean service duration and therefore the ratio of average holding time to average delay increases, the upper and lower bounds approach each other.

Figure 4.22: Network blocking probabilities for (a) class 1 and (b) class 2 calls. The offered load are 5.5 erlangs and 1 erlang of class 1 and class 2, respectively. The propagation delay is set 0.001 second on every trunk and the processing delay is 0.0001 second in each node.

In Fig. 4.22, we provide the network blocking probabilities for class 1 and class 2 calls when the maximum allowable number of alternative path is 0. As discussed, the average setup delay and its effect increase with the increased use of alternate paths. This can be demonstrated by comparing Fig. 4.21 and Fig. 4.22. In Fig. 4.22 the maximum number of alternative paths is 0 and the number of trunks along each primary path is always equal to 1 while in Fig. 4.21 the maximum number of alternative paths is 4 and the number of trunks along alternative path is 2. We observe that when the ratio of average holding time to average delay is 1000, the relative error is about 1.3% in Fig. 4.21 when $D$ is 4, while 0.6% in Fig. 4.22 when $D$ is 0.

In Fig. 4.23, we demonstrate the effect of the setup delay in the NSFNet, where the lengths of the trunks are different and therefore the propagation delay on them are different. We set the length of each trunk as shown in Fig. 4.5 where some of them are
obtained from [111] and others are approximated by the distances between the capitals of the states. In Fig. 4.23, we observe similar behavior of the upper and lower bounds to those in the default scenario.

### 4.4.15 Dimensioning

As discussed, blocking probability estimations are applied for dimensioning purposes for acceptable blocking probabilities such as $10^{-3}$ or $10^{-4}$. Here we illustrate that the error introduced by max(EFPA, service-based OPCA) is small in terms of error in dimensioning, even for the most inaccurate scenario of the 3-class case, discussed in Section 4.4.11.

In Fig. 4.24, we consider the 3-class case discussed in Section 4.4.11. We keep the ratio of the arrival rate 2, 1 and 1.5 for class 1, class 2 and class 3, respectively and increase the total offered load to the network. We dimension the network to find the number of channels per trunk required to keep the biggest blocking probability of the three classes below 0.001. Fig. 4.24 illustrate that the number of channels per trunk required approximated by max(EFPA, service-based OPCA) is very close to those obtained by simulation.
The relative errors of the approximation are less than 4% which is an acceptable error especially given the much larger errors in traffic prediction.

4.5 Conclusions

We have considered a circuit-switched multiservice multirate network with deflection routing and trunk reservation, and introduced two new approximations, OPCA and service-based OPCA, for the estimation of the network blocking probabilities of various traffic classes. We have explained the causes of the errors of the approximations and provided intuitive insights of their accuracy as compared to EFPA. Numerical results under a wide range of scenarios and parameter values have demonstrated that in most cases, service-based OPCA can estimate the network blocking probabilities reasonably well and is generally more accurate and more conservative than EFPA. We have also observed that OPCA can significantly overestimate the network blocking probabilities under certain scenarios and the performance of OPCA is not as robust as EFPA and service-based OPCA. Furthermore, we have proposed the more conservative max(EFPA, service-based
OPCA), which is more accurate than EFPA and service-based OPCA and more robust than OPCA. The results have also demonstrated the robustness of the approximations to the shape of the holding time distribution. Furthermore, we have shown that max(EFPA, service-based OPCA) is applicable to the network blocking probabilities estimation in large networks such as the Coronet, for which the simulation results are computationally prohibitive. Finally, we have shown that the relative error of max(EFPA, service-based OPCA) is acceptable when it is applied for the purpose of network dimensioning.
Chapter 5

Conclusion

5.1 Concluding Remarks

For a circuit-switched network with long-lived and short-lived connections where the long-lived connections can preempt the short-lived ones, EFPA and OPCA are applied with the quasi-stationary approximation to estimate the blocking probabilities. In most cases that are demonstrated, OPCA can estimate the blocking probabilities reasonably well, and generally, better than EFPA. As long-lived connections provide background traffic for short-lived ones, the ratio of their offered load also affects the accuracy of the approximations. Reduction of offered long-lived traffic together with increase of offered short-lived traffic will improve the accuracy of OPCA, while that of EFPA is not much improved. However, when the number of links on each trunk increases, the performance of EFPA is improved. Allowing more alternate path traffic, either by increasing the maximum allowable alternate paths or the long-lived traffic threshold, is beneficial under light traffic. However, when the network is fully occupied, it is important to restrict alternate path traffic. We have observed that the quasi-stationary approximation requires that the mean holding time of long-lived connections is at least 20 times longer than that of short-lived connections. Nevertheless, this is not a very restrictive requirement if long-lived
connections represent static connections and short-lived connections represent dynamic ones. We have also demonstrated that approximating blocking probability based on the exponential holding time assumption is not very sensitive to the shape of the holding time distribution, and is fairly accurate also for heavy-tailed holding time distributions. We have illustrated by numerical examples that, for small network, OPCA requires more computation time and more memory than EFPA, but the overall computing resources are manageable. For a large scale network such as the Coronet, we have demonstrated that OPCA is also applicable while the equivalent EFPA results are unattainable.

To evaluate the blocking probability of large CS networks with fixed routing, we have considered A-EFPA and EFPA implementation for CS networks based on asymptotic link blocking probability derivation. From the numerical results obtained using NSFNet and Internet2 networks, we observe that when link capacity is large, A-EFPA results are very close to those of EFPA, but A-EFPA saves approximately 99.9999% of the computing time. We observe consistent behaviors of A-EFPA and EFPA when they are applied in the Coronet. We have demonstrated very accurate calculations of the blocking probability using simulations, EFPA and A-EFPA. Complementing each other, these three methods are used for different ranges of parameter values.

We have considered a circuit-switched multiservice multirate network with deflection routing and trunk reservation, and introduced two new approximations, OPCA and service-based OPCA, for the estimation of the network blocking probabilities of various traffic classes. We have explained the causes of the errors of the approximations and provided intuitive insights of their accuracy as compared to EFPA. Numerical results under a wide range of scenarios and parameter values have demonstrated that in most cases, service-based OPCA can estimate the network blocking probabilities reasonably well and is generally more accurate and more conservative than EFPA. We have also observed that OPCA can significantly overestimate the network blocking probabilities under certain scenarios and the performance of OPCA is not as robust as service-based OPCA.
and EFPA. Furthermore, we have proposed max(EFPA, service-based OPCA), as an improvement over EFPA and service-based OPCA, and demonstrated that it gives accurate and conservative estimation of the network blocking probabilities. The results have also demonstrated the robustness of the approximations to the shape of the holding time distribution. Finally, we have shown that max(EFPA, service-based OPCA) is applicable to the network blocking probabilities estimation in large networks such as the Coronet, for which the simulation results are computationally prohibitive.

5.2 Future Research

The preemptive priority introduced in the surrogate model of OPCA imposes a virtual hierarchy on a non-hierarchical network. That is, instead of the original non-hierarchical network that suffers from mutual overflow, we now under OPCA have a hierarchical structure that avoids the mutual overflow related errors. However, to solve the problem of path error is still an open problem, especially in the cases of non-disjoint paths which have been used in practice. In fact, there are situations where they are necessary. However, using non-disjoint paths may lead to higher network blocking probability (as a result of fully occupied trunks being common to the primary and alternate path for example). Our approximation methods can also be used if the paths are not disjoint. However, at this stage the strong dependency between the trunks, in this case, will cause the approximation methods to underestimate the blocking probability. Development of algorithms to improve accuracy in the case of non-disjoint paths is still needed.

Networks with alternative routing may operate at multiple stable scenarios. One scenario is at low network blocking probability with most of the connections using primary paths. The other scenario usually happens when the network is fully occupied by many overflowed connections. The overflowed connections often take longer paths, use more resources and therefore cause other connections to overflow, which further aggravates the
situation and congests the network. To deal with the problem, we use trunk reservation to reserve certain number of links to primary path connections to avoid large number of overflowed connections in the network which may cause instability. In all the numerical examples that are presented in the networks with alternative routing, both EFPA and OPCA have converged to a unique solution. However, the convergence and uniqueness of the approximations in the cases with large amount of overflowed connections in the network is still worth studying.
Bibliography


127


