AGH University of Science and Technology
Faculty of Electrical Engineering, Automatics, Computer Science and Electronics

Ph.D. Thesis

Andrzej Szymański

Grade-of-Service-Based Routing Strategies for Optical Networks

Supervisor:

Prof. dr hab. inż. Andrzej Jajszczyk

AGH University of Science and Technology
Faculty of Electrical Engineering, Automatics, Computer Science and Electronics
Department of Telecommunications

Al. Mickiewicza 30, 30-059 Kraków, Poland
tel. +48 12 617 39 37
fax. +48 12 634 23 72

http://www.agh.edu.pl
http://www.eaiie.agh.edu.pl
http://www.kt.agh.edu.pl

Reviewers:

prof. dr hab. inż. M. Pióro[1]
prof. dr hab. inż. Z. Papir[2]

[1] Warsaw University of Technology
[2] AGH University of Science and Technology

ISBN 978-83-88309-62-5

Cover and layout design by Rafał Stankiewicz

Printed in Poland
by *Drukarnia GS Sp. z o.o., ul. Zabłocie 43, Kraków*

To my friends.

Acknowledgements

First and foremost, I would like to express my gratitude to Artur Lasoń and Jacek Rząsa for their invaluable help, brilliant ideas and numerous discussions during my research.

I would also like to thank Piotr Żuraniewski and Lucjan Janowski for hints and remarks regarding the statistical analysis of the obtained results, and Rafał Stankiewicz for his support with LaTeX.

Finally, I would like to thank my supervisor, prof. Andrzej Jajszczyk, for his patience during my research and determination to present the results at the top telecommunication conferences and journals.

Abstract

The dissertation is focused on optical wavelength routed networks with dynamic provisioning of lightpaths. The possibility of differentiating Grade of Service among different classes of lightpath requests is thoroughly investigated throughout this work. As a result, a set of mechanisms and routing strategies that allow an optical network operator to achieve differentiated lightpath blocking probability is proposed. Detailed algorithms are presented in two versions, one suitable for networks with the centralised computation model, and the other for networks with the distributed computation model. The performance of the proposed mechanisms and strategies is investigated using discrete-event computer simulation in two network topologies. Simulation results show that all proposed mechanisms and strategies managed to achieve the stated goals in assumed network conditions and successfully delivered GoS-differentiated services. However, the costs of implementing those strategies, being the decreased performance of low priority requests and increased complexity of control procedures, were quite different. Based on the obtained performance results, two candidates, the global capacity threshold and the path capacity threshold, have been identified as the preferred mechanisms to be implemented in future optical networks, due to their superiority in performance.

Streszczenie

Niniejsza rozprawa doktorska jest poświęcona zagadnieniu jakości obsługi zgłoszeń w komutowanych sieciach optycznych. W pracy przedstawiono mechanizmy pozwalające na zróżnicowanie prawdopodobieństwa blokady żądań zestawienia ścieżki optycznej należących do różnych klas. Przedstawiono także strategie zbudowane przy użyciu opisanych mechanizmów. Szczegółowe algorytmy zostały przedstawione w dwóch wersjach: dla sieci, w których dobór trasy i przydział długości fali odbywa się w sposób scentralizowany oraz dla sieci ze sterowaniem rozproszonym. Wydajność zaproponowanych mechanizmów i strategii została przebadana za pomocą symulacji komputerowej w dwóch topologiach sieciowych. Uzyskane wyniki pozwalają na stwierdzenie, że wszystkie zaproponowane mechanizmy osiągnęły zadany cel, jakim jest zróżnicowanie prawdopodobieństwa blokady żądań zestawienia ścieżki optycznej należących do różnych klas w założonych warunkach sieciowych. Dane uzyskane w czasie symulacji pokazują również, że koszty zastosowania opisanych mechanizmów, takie jak: zwiększenie prawdopodobieństwa blokady żądań niskiego priorytetu oraz wzrost złożoności procedur sterowania siecią, są różne. Wskazano dwa mechanizmy, które są preferowane do zastosowania w przyszłych komutowanych sieciach optycznych ze względu na wydajność. Są to: mechanizm weryfikacji liczby ciągłych nośnych optycznych dostępnych w sieci (*global capacity threshold mechanism*) oraz mechanizm weryfikacji liczby ciągłych nośnych optycznych dostępnych w ścieżce (*path capacity threshold mechanism*).

Contents

List of Symbols

$a(\psi, p)$	The path capacity of path p in network state ψ
$a(\psi, p, j)$	The path capacity of path p on wavelength λ_j in network state ψ
a^j	The number of links on which wavelength λ_j is free in the whole network
a_e^j	The number of fibres on link e on which wavelength λ_j is free
$a_e^j(\psi)$	The number of fibres on link e on which wavelength λ_j is free in a network state ψ
a_{ij}	The number of available wavelengths on link e_{ij}
b_e^j	The number of fibres on link e on which wavelength λ_j is busy
c	A default edge cost
$c(p)$	A weight of path p
c_e^j	A cost of an edge e on wavelength λ_j
c_c	A default cost of using a wavelength converter
d_{iD}	The shortest path distance from node i to destination node D
e_{ij}	A link connecting nodes i and j
f_e	The number of fibres on link e
h	The number of lags tested

h_j	The hop count of path p_j
l	The number of links in the physical network
n	The number of nodes in the physical network
p	A path (a concatenation of links)
p_i	The i-th path in a set of potential paths P_{SD}
q_j	The preference of node j
s	The standard deviation of a sample
$t_{\alpha/2}$	The $(1-\alpha/2)$ quantile of the t-Student distribution with $N-1$ degrees of freedom
w	The number of wavelengths in the network
w_{ij}	The number of wavelengths on link e_{ij}
$\bar{x}$	The sample mean
x	A sample
$z_{\alpha/2}$	The $(1-\alpha/2)$ quantile of the normal distribution
A	A set of wavelengths considered for lightpath setup
A^p	A set of free wavelengths on path p
A^p_{Si}	A set of continuous free wavelengths on links from source node S to an intermediate node i on path p
A_0	The set of unused (free) wavelengths on the first link of path p
A_{ij}	A set of free wavelengths on link e_{ij}
D	A destination node of a lightpath request
E	The set of edges (links) in the physical topology
G	A graph representing the physical network topology
H_0	The null hypothesis
H_a	The alternative hypothesis
$L'(D,j)$	The relative capacity loss for destination D and wavelength λ_j

$L'_i(D,j)$	The relative capacity loss for destination D and wavelength λ_j obtained from a neighbouring node i
$L(p,j)$	The relative capacity loss resulting from setting up a lightpath request on path p and wavelength λ_j
M	The number of samples
N	The number of observations in a sample
P	A set of paths in the network
$P'(p^*)$	The set of paths which share common links with path p^*
P_{SD}	A set of paths from a source node S to a destination node D
Q_{LB}	The Ljung-Box test statistics
R	A lightpath request
S	A source node of a lightpath request
T	A configurable integer threshold
V	The set of vertices (nodes) in the physical topology
$V(\psi'(j))$	An optimisation function
$1-\alpha$	The confidence coefficient
α	The significance level
α_i	A constant
$\chi^2_{1-\alpha,h}$	The α quantile of the chi-square distribution with h levels of freedom
$\delta_{i,j}$	A calculated metric of wavelength i on path j
λ_i	The i-th wavelength
μ	The population mean
$\hat{\rho}$	The sample autocorrelation at lag j
σ	The population standard deviation
ψ	The current network state

$\psi'(j)$	A network state resulting from setting up the current lightpath request on wavelength λ_j
Ω	A set of wavelengths in the network
Ω_{ij}	The set of wavelengths on link e_{ij}

1 Introduction

The groundbreaking invention of laser [39], [55] and subsequent works by Kao and Hockham on low loss optical fibres opened up a significant chapter in the history of telecommunications. In a short time, in 1962, an idea of multiplexing independent data streams using different optical wavelengths appeared [31]. However, this concept, currently known as Wavelength Division Multiplexing (WDM), had to wait till the early 1990's before its widespread implementation in telecommunication networks. The two main drivers for its rapid development were the unprecedented growth of the total traffic volume carried by modern networks and the invention of optical amplifiers [31].

Introduction of WDM into optical transport systems opened up new possibilities. In 1989 Chlamtac, Ganz and Karmi proposed a concept of a lightpath, a "direct communication path" between two (not necessarily adjacent) nodes, established by allocating the same wavelength throughout the path of the transmitted data [14]. The concept of a lightpath decreased significantly the necessary processing power of the intermediate nodes, as they did not have to perform an electro-optic conversion and did not have to process the data stream in the electrical domain. A network using this mode of operation is termed a wavelength routed optical network.

The connection-oriented nature of the wavelength routed network requires that the lightpath must be established before the data transfer phase and may be disconnected afterwards. In order to establish a lightpath its path must be determined and a free wavelength must be assigned. This problem is known as the Routing and Wavelength Assignment (RWA) problem. The requirement of allocating the same wavelength on all links may be relaxed by using wavelength converters. However, this is not the case in this dissertation.

Initially, the set of lightpaths in the network was static, and the lightpaths were set up manually. However, progress in photonic switching technologies opened up the ability to set-up and tear down lightpaths on demand. In 2001 the Telecommunication Standardization Sector of the International Telecommunication Union (ITU-T) presented the recommendation "Architecture for the automatically switched optical network (ASON)" [24] which describes the reference architecture for the control plane of the optical network. In 2004 the Internet Engineering Task Force (IETF) proposed the Generalized Multi-Protocol Label Switching (GMPLS) Architecture [40] which may be used to dynamically create and remove lightpaths in the optical network. In both architectures, two protocols are considered for the actual exchange of the lightpath setup messages and for resource allocation: RSVP-TE [7] and CR-LDP [3].

Optical networks play an important role in today's networking. An increase in the available capacity promotes the introduction of new services, each having different network requirements. On one hand, this requires high capacity, on-demand provisioned connections, on the other, providing connections with the appropriate quality guarantees.

The quality of the lightpath may be considered from three viewpoints: Quality of Service (QoS), Quality of Resilience (QoR) and Grade of Service (GoS). Quality of Service (QoS) deals with connection parameters that affect data flow after the lightpath is established (see [29] for details). Quality of Resilience (QoR) describes the availability of the connection with regard to different kinds of network and equipment failures (see [15] for details). Grade of Service (GoS) includes all parameters applicable to connection setup.

There are two GoS parameters that are especially visible: the connection setup time and the connection blocking probability. The blocking probability is the probability that a lightpath request will be blocked due to some reason, resulting from either policy requirements or lack of resources. The connection setup time reflects the time needed from the moment a lightpath request is generated to the moment the lightpath is completely set up and ready for data transfer. Both parameters are related, as having the lower blocking probability usually means putting more effort and time in finding the right path and wavelength on which the lightpath may be set up.

This work deals with Grade of Service issues in the optical network with a strong focus on one parameter: the connection blocking probability. A set of mechanisms which allow a network operator to achieve differentiated GoS in the optical network is proposed and thoroughly investigated throughout the dissertation. The evaluation is performed on two reference networks using discrete-event computer simulation.

1.1 Scope and thesis

The dissertation is focused on optical wavelength routed networks with dynamic provisioning of lightpaths. It is assumed that the network topology is given, and does not change in time. The network nodes have no wavelength conversion capabilities.

Lightpaths are set-up and torn down on demand. Additionally, existing lightpaths cannot be rerouted or disconnected to free resources for new connections. Lightpath requests belong to two classes: high priority and low priority. When a lightpath request arrives, the network needs to choose a path and a wavelength for the request, and set up the request. The network may decide to block the request, as well.

The following hypothesis is investigated in the dissertation:

> It is possible to differentiate the blocking probability of lightpath requests in optical networks using moderately complex routing strategies.

To prove this hypothesis a set of GoS mechanisms is proposed and thoroughly investigated in the dissertation.

1.2 Publications

Some of the results presented here were already published in the form of two conference and two journal papers:

> [63] A. Szymanski, A. Lason, J. Rzasa and A. Jajszczyk, "Route Management Strategies for Grade of Service Differentiation in Optical Networks," *IEEE International Conference on Communications, ICC '06*, Istanbul, Turkey, pp. 2453 – 2458, 11 – 15 June 2006.
>
> [64] A. Szymański, A. Lasoń, J. Rząsa, A. Jajszczyk, "Strategie zarządzania doborem trasy umożliwiające różnicowanie jakości obsługi w przełączanych sieciach optycznych", *Przegląd Telekomunikacyjny, Wiadomości Telekomunikacyjne*, Vol. 80, Iss. 2–3, pp. 64–67, 2007.
>
> [61] A. Szymanski, A. Lason, J. Rzasa and A. Jajszczyk, "Grade-of-Service-based Routing in Optical Networks" *IEEE Communications Magazine*, Vol. 45, No. 2, pp. 82 – 87, February 2007.
>
> [62] A. Szymanski, A. Lason, J. Rzasa and A. Jajszczyk, "Performance Evaluation of the Grade-of-Service-based Routing Strategies for Optical Networks" *IEEE International Conference on Communications, ICC '08*, Beijing, China, pp. 5252 – 5257, 19 – 23 May 2008.

All of the presented papers underwent a thorough review process before their final publication.

The paper [63] proposes three mechanisms: link capacity threshold, path capacity threshold and wavelength pools and investigates their performance in a PanEU network using the shortest path routing and the first fit wavelength assignment.

The paper [64] also investigates the mechanisms presented earlier in [63]. Additionally, it provides the first classification of the grade-of-service-based routing strategies in the optical networks.

The paper [61] presents a tutorial overview and comparison of the grade-of-service-based routing strategies with emphasis on mechanisms that provide differentiated call blocking probability for different classes of lightpath requests.

Finally, the paper [62] introduces two more mechanisms: first link capacity threshold and global capacity threshold and a mechanism based on fixed-alternate routing with a different number of paths in each class. Then a set of strategies using the mechanisms is presented and investigated.

1.3 Structure of the dissertation

The rest of the dissertation is structured as follows. Chapter 2 presents the work related with the investigated subject. Chapter 3 describes in detail the proposed mechanisms and strategies. Chapter 4 deals with performance assessment methodology and tools. Chapter 5 presents the performance of the individual mechanisms. Chapter 6 presents a comparison of the strategies built using the proposed mechanisms. Finally, Chapter 7 concludes the dissertation.

2 Related work

Inherently, GoS in a wavelength routed optical network is directly related to the Routing and Wavelength Assignment (RWA) algorithm. Routing and wavelength assignment is a specific problem for optical networks, for a good review of other problems related to design and operation of communication networks see [50].

Depending on the way a network operates, the RWA problem may be either static or dynamic. Several Routing and Wavelength Assignment algorithms for optical networks were proposed in the past; for a good review see [74, 75].

A static RWA is used when all lightpaths are to be set up at the beginning of network operation. Usually, the set of lightpaths is given in the form of a logical topology connectivity matrix. The static RWA problem is often formulated as a Mixed Integer Linear optimisation problem where the optimisation target is to set up all requested lightpaths minimising the use of network resources. This is very useful during the network dimensioning phase, as it helps finding the required number of fibres on each link and the required number of wavelengths in the network. The static RWA is NP-hard (for the proof see [13]), so the optimal solution is in most cases not available. Thus, many heuristic methods have been proposed for solving it. As in this dissertation the focus is on dynamically provisioned connections, the static RWA will not be investigated in a greater detail.

In the dynamic RWA problem the physical network topology, including the number of available wavelengths and the number of fibres on each link is given. The lightpath requests are not known in advance, but arrive during the network operation. The task is to find an optimal path and assign the best wavelength for a lightpath request. If the algorithm cannot find sufficient resources, the lightpath request is blocked, otherwise it is accepted. Each lightpath occupies network resources during the holding time, which is unknown at the set-up time, and releases them afterwards. This is called a dynamic traffic scenario. Some papers (for example [73]) assume an incremental traffic scenario, where the holding time is infinite. However, this approach is used far less frequently.

The most common performance measure of a dynamic RWA algorithm is the lightpath blocking probability, computed as the ratio of the number of lightpath requests that were blocked to the overall number of lightpath requests. Birman and Kerschenbaum show that the blocking probability of a lightpath request grows fast with the number of hops on the path from the source to the destination [8]. This makes room for algorithms that try to protect resources for multi-hop lightpaths. A measure which gives the insight on the grade of service (GoS) fairness of the RWA algorithm is the blocking probability computed for each source-destination (S-D) pair [8]. An important issue for a dynamic RWA algorithm is also its stability. It is important that the algorithm does not deteriorate in time, i.e. the resources do not become fragmented [13].

Lee and Li note, that a lightpath request may be blocked due to two reasons: the bandwidth constraint, i.e. there is no free capacity on a path from the source to the destination, and the wavelength continuity constraint, i.e. there is enough bandwidth on the path but its use requires a wavelength converter which is not available [34]. Birman and Kershenbaum enumerate three methods for minimising the blocking probability associated with wavelength continuity constraint: using wavelength converters, reallocating the existing connections or using an appropriate RWA algorithm [8].

For each lightpath request the dynamic RWA algorithm needs to find a path and choose a wavelength on which the request can be accepted. It is common to divide this problem into two subproblems: routing, and wavelength assignment. Mokhtar and Azizoglu note that "both subproblems may be solved sequentially in any order, jointly or in an alternating fashion" [44]. Hence, two cases will be distinguished. A sequential case, where both subproblems are solved separately, and an integrated case, where both subproblems are solved jointly as one problem. The case when routing and wavelength assignment occur alternately will be dealt with in the class of sequential problems.

In the sequential approach it is necessary to solve one of the subproblems first, and based on the result solve the other one. This creates two possibilities: routing followed by wavelength assignment or wavelength assignment followed by routing. Spaeth terms this "path priority scheme" and "wavelength priority scheme," respectively [57]. It is important to note that the division into routing algorithms and wavelength assignment algorithms stems from the obtained result: a path or a wavelength. In many cases routing algorithms take into account wavelength usage information, and even more, try to use a path that has continuous free wavelengths. Nonetheless, they do not choose the wavelength on which the lightpath is to be established. On the other side, wavelength assignment algorithms may use such measures as the resource usage on one or more paths from source to destination, but they do not decide on which path the currently processed request is to be set up.

The integrated approach does not divide between routing and wavelength assignment. Instead, it solves the whole problem in one step, which usually improves the quality of the solution, but increases the complexity of the algorithm.

The last issue is the amount of information used for routing and wavelength assignment decisions and the credibility of the information. Most of the algorithms presented in this chapter assume the centralised approach in which all decisions are taken in one place. In this case the information about the network state perfectly matches the current network state. However, if those algorithms are run in a distributed environment, the algorithms may suffer from increased blocking due to inadequate network state information and resource allocation collisions resulting from many lightpaths being set up simultaneously. Some of the algorithms presented in this chapter are prepared to deal with distributed computing, by either assuming uncertainty of network state data or by tight coupling with signalling and resource reservation.

2.1 Notation

The physical network topology is given and it does not change in time. It is represented by the graph $G(V, E)$ where V is the set of n nodes ($n = |V|$) and E is the set of l directed edges (links), $l = |E|$. Ω is the set of wavelengths available on each link, $w = |\Omega|$ is the number of wavelengths on each link. Some algorithms presented in this chapter assume uneven number of wavelengths on the links, for them Ω_{ij} is the set of wavelengths available on link e_{ij}, w_{ij} is the number of wavelengths on the link, $w_{ij} = |\Omega_{ij}|$. Lightpath requests are given in the form of source-destination (S-D) pairs, (S, D) where $S, D \in V, S \neq D$.

2.2 Routing

There are two major classes of routing algorithms: static and adaptive [44]. Static algorithms do not change their routing procedure in time. Adaptive algorithms use the network state information for taking routing decisions.

Fixed routing, which belongs to the class of static algorithms, computes exactly one path for each S-D pair at the beginning of network operation. The path remains the same (fixed) throughout the network life, assuming no network failures, and all lightpath requests between a given source and destination are routed alongside this path regardless of the network state. If there are no free resources on the path then the request is blocked.

In alternate routing an ordered set of paths is prepared in advance for each S-D pair. The number of paths in the set and the search order may be fixed, or may be chosen adaptively. The former case is often termed fixed-alternate routing. If a lightpath request cannot be accepted on any of those paths, it is rejected.

Finally, adaptive routing searches for a path taking the current network state into consideration. Its operation is not restricted to a predefined set of paths, thus, it is sometimes referred to as unconstrained routing.

Fixed routing

In fixed routing a single path is assigned to each source-destination pair and this path does not change in time [44]. A lightpath request that cannot be set up on this path is blocked. This class of routing algorithms is very simple, does not require any network state data, but it suffers from high blocking if any of the links on the S-D path is congested. Additionally, it may lead to significant disproportions in GoS for different S-D pairs [8].

One of the special cases of fixed routing is the shortest path routing. It is one of the simplest algorithms, used often as a reference routing algorithm in comparisons (see [13], [59]). The inputs to this algorithm are the source and the destination nodes of a lightpath request and the topology on which the request is to be routed. The algorithm does not use information about the currently available link capacities or the current network traffic. Each link in the topology is assigned the unit metric or, in rare cases, a metric that reflects the real link (fibre) length. The alternative name "least hop routing" is sometimes used to emphasise that the particular instance of the shortest path routing is using the unit length for all links, but this terminology is not consistently used throughout the literature. The path is chosen according to the shortest path algorithm (Dijkstra algorithm [16], for example), minimising the use of network resources.

Paper [46] proposes a different approach to fixed routing, called the Static Bias. In this method a set of fixed paths is obtained from a static RWA optimisation using the linear programming techniques. The input to the static RWA problem is a set of lightpaths corresponding to the mean traffic in the network. The precomputed paths are then used as the fixed paths for setting up dynamic lightpath requests. This approach offers $10-40\,\%$ better blocking probability compared to the shortest path routing.

Alternate routing

In alternate routing a set of paths is assigned to each S-D pair. Depending on the algorithm used, the set may be searched in a fixed order (fixed-alternate routing) or in an adaptive order. The authors of [53] report that using alternate routing may be more beneficial than using fixed routing and wavelength conversion.

Alternate routing has long been studied in the context of classic telephone networks. Since 1950s the telephone networks used hierarchical alternate routing with fixed sequences of alternative choices. With the progress in signalling and programming capabilities of exchanges, a non-hierarchical alternate routing methods were proposed. A good survey of those methods may be found in [49]. Due to similar nature, optical networks inherited some ideas from telephone networks, with fixed-alternate routing, least-loaded path routing and trunk reservation being some examples.

Several different alternate routing algorithms for optical networks appeared in the past. The paper [57] proposes an alternate routing algorithm called "Alternative Central Routing on partially Link-Disjoint paths," (ACR-LD). The set of alternate paths for each S-D pair is computed as follows. The first (original) path is the shortest path. Then, one link of the original path is removed from the topology and a shortest path is calculated, resulting in an alternative path that is partially link-disjoint with the original path. Then, the removed link is added back to the topology. This calculation is repeated for all links of the original path. Finally, a completely link-disjoint path is calculated and added to the set. The simulations performed on an 18-node netwok show that the scheme performs much better than fixed routing.

In [58] another alternate routing variant is proposed in which the set of alternate paths contains k completely link-disjoint paths. Spaeth terms this approach "Alternative Source Routing on Completely Disjoint Paths," (ASR-CD).

The paper [22] proposes another alternate routing method, termed "alternate routing with limited trunk reservation," where connections that require more hops are provided with a greater number of alternate paths. Additionally, this scheme assumes that a given number of free wavelengths should be left on the path, which further equalises the blocking probabilities between short and long lightpaths. The comparison with fixed routing and alternate routing with two paths (primary and secondary) on a torus network with 16 nodes and 4 or 8 wavelengths shows that alternate routing with limited trunk reservation outperforms the other two methods in terms of fairness and overall blocking probability, especially if its parameters are well tuned.

In Least Congested Path (LCP) routing [10] a set of paths to be considered for each communicating S-D pair is computed. For each link a congestion level expressed in the number of free wavelengths on the link is obtained. The congestion level of a path is the congestion level of the most congested link on that path (i.e. the link which has the minimum number of free wavelengths). LCP chooses the path which is least congested, if there are no resources on the chosen path, the next least congested path is chosen.

The “Fixed Paths Least Congestion” (FPLC) routing was proposed in [35]. This is an extension of the LCP method, in which the congestion level of the path is the number of continuous idle wavelengths on the path. The simulations as well as analytical analysis show, that FPLC outperforms shortest path routing and routing based on k shortest paths with fixed search order, especially for low loads. A variant of this approach, FPLC-N(k), also proposed in [35], examines only the first k links on each path. FPLC-N(k) reduces the amount of information needed for deciding which path to choose, at a cost of a slightly worse performance than FPLC.

Adaptive routing

In the Adaptive Dynamic Routing with x pre-calculated paths (ADRx), proposed by Spaeth [57], [58], a set of x pre-calculated paths is considered for satisfying the lightpath request. If, however, the request cannot be accommodated on any of those paths, a new path is calculated dynamically. The dynamic calculation takes into account all links, excluding those loaded above a given threshold. The simulations performed on an 18-node network show that this scheme outperforms alternate routing with 2 paths. The simulations also show a general trend, that differences in the blocking probability between fixed routing, fixed-alternate routing and adaptive routing (ADRx) are mostly present in the low load region.

An extension to this approach limits the number of hops on any path. The limit may be absolute, relative to the length of the shortest path between S-D nodes or adaptive. The first one poses unnecessary restrictions for low network load and may even block some node pairs if the network is sufficiently large. The relative limit is more flexible, but still introduces unnecessary blocking for low load region. The adaptive limit considers network load on the links. The presented simulation results show, however, that gains from using those mechanisms are close to the simulation error margin.

The paper [77] proposes the Adaptive Least Loaded Routing scheme (ALLR-k), which was designed to overcome problems with LLR and FPLC. The set of paths to be considered while lightpath setup is not static, as with LLR and FPLC. Each node maintains up to six precomputed paths to every other node: a shortest path, a pair of disjoint shortest paths, a maximum bandwidth shortest path and a pair of disjoint maximum bandwidth shortest paths.

Disjoint paths are computed using Suurballe’s algorithm [60]. Shortest paths are computed using hop count as a metric. For Maximum Bandwidth paths the authors propose a piece-wise linear increasing and convex cost function that depends on the average utilisation of links during the last path computation period. The lightpath is set up on one of those paths. The simulations show that ALLR-k performs similarly or better than FPLC.

The authors of [73] introduce the Near-maximum Available Wavelengths (NAW) algorithm. The rationale behind this algorithm is to find a path with the greatest number of available continuous wavelengths to prevent exhausting resources. As the authors note, an important matter is the computational complexity of the algorithm, which should scale well with the number of links and wavelengths. NAW proceeds similarly to the well-known Dijkstra shortest path algorithm, but uses a different representation of the node and link weights. A link is attributed a binary vector which reflects the currently available wavelengths on the link. A similar vector is attributed to each node and reflects the availability of continuous free wavelengths on the best path from the source node to the given node. The number of available wavelengths is the node weight. The NAW algorithm selects a path from the source node to all other nodes maximising the node weights. The algorithm performed best with the First Fit wavelength assignment scheme and it provided about 50% reduction of the blocking probability, compared to the Least Loaded and k-shortest path algorithms. However, the simulation environment used in [73] was different from the one commonly used in that the lightpaths were never released (incremental traffic). Thus, in the dynamic scenario the performance figures may be different.

The drawback of the algorithm is the requirement of posessing the global network state information in the node performing computation. Although OSPF extensions are able to provide that, the management and signalling overhead might be excessive. Subsequent algorithms: NUW_S, NUW_U and HOP_NAW have this requirement relaxed and require reduced amount of information about the network state. The second problem, arising in some networks is that this algorithm focuses on load balancing and not on keeping paths short, thus in the high load conditions too many resources may be used, which leads to a higher blocking probability.

In NUW_S algorithm [73] the source node sends a request message towards the destination node. As this request passes intermediate nodes, it builds a path p on which the lightpath is to be established and, simultaneously, it collects information about the continuous free wavelengths available on the path. In node i the next hop node j for the request message is chosen based on the preference value q_j computed using the collected wavelength usage information (which wavelengths are free from the source node to node i), available wavelengths on the considered next hop link e_{ij} and the total count of available wavelengths from node j to all nodes k which are closer to the destination than node j. The preference value is expressed as:

$$q_j = |(A^p_{Si} \cap A_{ij})| \cdot \sum_{d_{kD} < d_{jD}} a_{jk} \qquad (2.1)$$

where A^p_{Si} is the set of continuous free wavelengths on the path from the source node S to an intermediate node i on the path p, A_{ij} is the set of free wavelengths on the link e_{ij}, $|(A^p_{Si} \cap A_{ij})|$ is the number of continuous wavelengths from the source node to node j, a_{jk} is the number of free wavelengths on link e_{jk} (this number is zero if the link e_{jk} does not exist), d_{jD} is the shortest distance (least hop distance) from node j to the destination node.

NUW_U [73] works similarly to the NUW_S algorithm, but differs in the way a node preference is computed. The reader is referenced to [73] for details. Simulation studies performed in the referenced paper show, that the NUW_U algorithm performs better than NUW_S, i.e. it provides a lower blocking probability and shorter paths. Both NUW_S and NUW_U algorithms perform well in high load conditions, compared to fixed algorithms, because they are able to find a feasible path when the network is short of resources.

The Hop_NAW [73] algorithm is based on sending several request messages from the source to the destination, each of them collecting the information about the wavelength availability on its path. The path with the greatest number of wavelengths is chosen for the request. Additionally, Hop_NAW uses crankback technique if a given request is blocked. The Hop_NAW algorithm does not require any information about the network state, beyond that provided by a standard link-state routing protocol (i.e. topology), but instead places a greater burden on signalling. The performance of Hop_NAW is similar to NAW.

The paper [17] proposes another two adaptive routing algorithms: "Available wavelengths" (AW) and "Total Wavelengths and Available Wavelengths" (TAW). Both algorithms use the shortest path algorithm with a non-standard dynamic link metric, based on the number of the available wavelengths on each link.

For the AW algorithm[17]:

$$c_{ij} = \begin{cases} -log(1 - \frac{1}{a_{ij}}) & ; a_{ij} > 1, \forall(i,j) \in E \\ 1 & ; a_{ij} = 1, \forall(i,j) \in E \end{cases} \tag{2.2}$$

For the TAW algorithm:

$$c_{ij} = -log\left(1 - \left(1 - \frac{a_{ij}}{w_{ij}}\right)^{a_{ij}}\right); \forall(i,j) \in E \tag{2.3}$$

where a_{ij} is the number of available wavelengths on link e_{ij}, and w_{ij} is the total number of wavelengths on link e_{ij}. The weight formula is based on calculating the probability of a given wavelength being free on the link.

Adaptive Unconstrained Routing (AUR) proposed in [44] first orders wavelengths in some manner, then starting from the first one tries to satisfy the request using shortest-path routing. The shortest path computation considers the wavelength usage in the network. The following four variants of AUR were proposed:

- AUR/PACK orders wavelengths starting from the wavelengths that are used on the greatest number of links in the network,
- AUR/SPREAD orders wavelengths starting from the wavelengths that are used on the smallest number of links in the network,
- AUR/RANDOM randomly orders wavelengths.
- AUR/FIXED orders wavelengths according to their physical (wavelength) order.

An additional case is the AUR/Exhaustive approach, referenced also in Section 2.4, where a shortest path is searched on each wavelength in turn and the shortest one of them is used.

The above approaches were compared with the Fixed Routing/First Fit approach. The tests were performed on two network architectures: ARPA2 network with 21 nodes and 26 links, and a randomly generated network with 15 nodes and 32 links. For single fibre networks with 4 and 8 wavelengths, the best algorithm was AUR/Exhaustive, then AUR/Pack, AUR/Random, AUR/Spread. The AUR/Exhaustive method outperformed others but it required a significant computational effort due to the fact that the shortest path computation had to be run for each wavelength. AUR/PACK and AUR/Fixed gave similar results, with AUR/PACK performing slightly better. The results were also compared with Fixed shortest path routing with First Fit wavelength assignment. All AUR algorithms outperformed the Fixed Routing, although the difference diminished with increasing load. Then a comparison between fixed routing with First Fit wavelength assignment, alternate routing with two paths and First Fit wavelength assignment, AUR/Pack and AUR Exhaustive was made, showing that the performance gap between fixed routing and AUR may be significantly reduced just by introducing an alternate path. Thus, alternate routing might be seen as a viable compromise between simple fixed routing and complex AUR.

In [26] the authors propose a distributed routing method called "alternate-link routing." In this method, the path choice for a request is performed by choosing the outgoing link in the current node and forwarding the request to the next-hop node for processing. Each node maintains a routing table, which contains an ordered list of outgoing links to reach each destination. The version of the algorithm investigated in paper [26] maintained two outgoing links per destination in each node. The first one was chosen according to the shortest path policy, the second by choosing the shortest path with the first link removed from the topology. A proper outgoing link for an arriving connection request was chosen according to either the shortest-path-first policy, where the first link, corresponding to the shortest path, was preferred or the least-congested policy when the link leading to the maximum number of free continuous wavelengths on the path was preferred. The presented algorithm outperforms fixed routing and fixed-alternate routing at low and moderate loads.

2.3 Wavelength assignment

In a Random wavelength assignment a wavelength is chosen at random among the set of candidates [8]. The simulations performed in [8] indicate that this method favours short lightpaths, and its performance is significantly degraded for long lightpaths. This method is outperformed by other wavelength assignment methods, as reported in [8], [30], [32], [44], [59]. Some papers report (see [19] and [26]) that for distributed lightpath setup, when multiple lightpaths are being set-up simultaneously, random wavelength assignment might perform better than other approaches due to the fact that randomness prevents collisions and race conditions between parallel lightpath setups. Random wavelength assignment is commonly used as a reference method and for mathematical analysis of blocking probabilities.

The First Fit wavelength assignment, termed originally PACK[1] [14], and in some papers Incr/Incr [8] or FIXED [44], requires that the wavelengths are ordered in some way, for example $\lambda_1, \lambda_2, ...\lambda_w$. The ordering is consistent throughout the whole network, and does not change in time. For each lightpath request the algorithm sequentially scans the wavelengths starting from λ_1 and the first wavelength which can accommodate the lightpath request is chosen. A general idea behind this algorithm is to maximise the utilisation of wavelengths, and to ensure that a maximum number of new lightpaths can be allocated [13].

The Incr/Decr wavelength assignment scheme was proposed in [8] and is based on a similar idea as the First Fit algorithm. The lightpath requests which occupy only one link (single-hop) are assigned wavelengths starting from the lowest available wavelength (λ_1), whereas lightpath requests that occupy more than one link (multi-hop) are assigned wavelength starting from the last available wavelength (λ_w). The comparison of the Incr/Decr algorithm with First Fit (Incr/Incr) algorithm performed in [8] suggests that there is no substantial difference between those methods.

In the Centralised Lightpath Allocation (CLA) method [14], sometimes referred to as Max Wave policy [22], PACK [44] or Most Used [59], [74], the wavelength occupied on the greatest number of links in the whole network is used for satisfying the request. If this is not possible, the next most used wavelength is considered. The performance of this method is very close to, but slightly better

[1]Please note that some algorithms, proposed by different authors, bear the same name. For example PACK is used in context of First Fit and Centralised Lightpath Allocation, this name appears also as the name of the integrated RWA strategy.

than that of First Fit [22]. The purpose of this algorithm is to maximise the utilisation of already used wavelengths and avoid using lightly loaded wavelengths.

The SPREAD algorithm [44], sometimes called Least Used [74], proceeds the opposite way to PACK, trying to balance the utilisation of wavelengths. The algorithm sorts wavelengths according to their utilisation, i.e. the number of links (fibres) on which a given wavelength is used. Then, the algorithm tries to set up the request on the least loaded wavelength, and if it fails the next least used wavelength is considered. As noted in [74], this algorithm is not preferred in practice, since its performance degrades quickly. It tends to service only lightpath requests spanning few hops, while blocking longer lightpath requests.

The Least Loaded wavelength selection algorithm [30], suitable mostly for multi-fibre networks, chooses a wavelength $\lambda_j, \lambda_j \in A^p$ so that $\min_{e \in p}[f_e - b_e^j]$ is maximised. In case of a tie, the minimum index j is chosen. f_e is the number of fibres on link e, b_e^j is the number of fibres on link e on which wavelength λ_j is busy, A^p is the set of available wavelengths on path p, and p is the path on which the lightpath request is to be established. [74] notes that this approach reduces to First Fit in single fibre networks.

The Minimum Sum algorithm [30] chooses wavelength λ_j for which $\sum_{e \in p} \frac{b_e^j}{f_e}$ achieves minimum, in case of ties, the minimum j is chosen.

The goal of the Max Sum [59] algorithm is to choose wavelength λ_j for the lightpath so that the network is in a good state $\psi'(j)$ after the lightpath is established. This is achieved by maximising the function $V(\psi'(j))$ over all possible resulting states $\psi'(j)$. ψ is the current network state, $\psi'(j)$ is the network state resulting from establishment of the current lightpath request on wavelength λ_j.

The link capacity of link e on wavelength λ_j in state ψ, $a_e^j(\psi)$, is the number of fibres on which wavelength λ_j is free on link e. The capacity of path p on wavelength λ_j, $a(\psi, p, j)$, is the number of fibres on which λ_j is unused on the most heavily used link along the path, $a(\psi, p, j) \equiv min_{e \in p} a_e^j(\psi)$. The path capacity of path p in state ψ is defined as

$$a(\psi, p) \equiv \sum_{j=1}^{w} a(\psi, p, j) \tag{2.4}$$

where w is the number of wavelengths on a link.

The Max Sum ($M\Sigma$) algorithm chooses wavelength λ_j to maximise $V(\psi'(j))$ defined as:

$$V(\psi'(j)) = \sum_{p \in P} a(\psi'(j), p) \tag{2.5}$$

where P is the set of all paths in the network. In other words, the algorithm chooses wavelength λ_j in a way to maximise the sum of available capacities on all paths after the request is set up.

Two other algorithms, Max Weighted Sum ($MW\Sigma$) and Max Min, are presented in [59], but their performance is not investigated. The algorithms differ from the Max Sum algorithm in a way $V(\psi'(j))$ is defined,

$$\mathrm{MW\Sigma} : V(\psi'(j)) = \sum_{p \in P} c(p) a(\psi'(j), p) \tag{2.6}$$

$$\mathrm{Max\_Min} : V(\psi'(j)) = \min_{p \in P} a(\psi'(j), p) \tag{2.7}$$

where $c(p)$ is an arbitrarily chosen weight of path p.

The authors of [19] propose another wavelength assignment scheme, suitable for multifibre networks. The Maximum-Availability method chooses the wavelength which is available on the maximum number of fibres on path p. In other words, it chooses wavelength λ_j for which the path capacity of path p on wavelength λ_j in the current network state ψ, $a(\psi, p, j)$ achieves maximum. If several wavelengths have the same path capacity, one of them is chosen randomly. The presented studies show that the presented method outperforms First Fit and Random wavelength assignment for multifibre networks.

The Min Product [25] algorithm chooses the wavelength for which the product $\prod_{e \in p} \{a_e^j\}$ is minimal. a_e^j is the number of fibres on which wavelength λ_j is busy on link e. In case of a tie a minimum value of j is chosen. [74] notes that this approach is suitable for multifibre networks, and it reduces to First Fit in single fibre networks.

The Relative Capacity Loss [76] algorithm is based on Max Sum. The goal of this approach is to choose a wavelength which has the minimum impact on capacity of other paths, similarly to the Max Sum algorithm. However, this approach takes into consideration the amount of resources left for a given path and avoids allocation that would lead to blocking of some paths.

The term capacity of path p on wavelength λ_j, $a(\psi, p, j)$, was introduced in the description of the Max Sum approach. The Relative Capacity Loss due to lightpath setup on path p and wavelength λ_j, $L(p, j)$ is defined as follows:

$$L(p, j) = \frac{a(\psi, p, j) - a(\psi', p, j)}{\sum_i a(\psi, p, i)} \tag{2.8}$$

The lightpath request is set up on λ_j for which the $\sum_{p \in P'(p^*)} L(p, j)$ is minimal. $P'(p^*)$ is the set of paths which share common links with path p^*.

The authors of [76] also propose a Weighted RCL approach that is suitable for networks with a nonuniform traffic intensity matrix. In the algorithm, the weights $c(p)$ that are assigned to paths are proportional to the traffic intensity on those paths. Then the RCL algorithm is modified to set up the request on wavelength λ_j for which the $\sum_{p \in P'(p^*)} L(\psi, p, j) \cdot c(p)$ is minimal.

The main motivation for the Distributed Relative Capacity Loss (DRCL) algorithm [74] was to adapt RCL for its use in networks with distributed control. The algorithm is similar to RCL in that it calculates the capacity loss due to the lightpath set up on wavelength λ_j. The capacity loss in RCL was calculated upon the set of all path that had common links with the current lightpath. DRCL takes into account capacity loss on paths originating at the same source node S as the current lightpath. The algorithm is based on the Bellman-Ford algorithm [6]. Each node computes the table of triplets (wavelength, destination, relative capacity loss) which is exchanged with neighbouring nodes.

Capacity loss $L'(D, j)$ for destination D and wavelength λ_j is calculated in node S the following way:

1. $L'(D, j) = 0$ if there is no path from node S to destination D on wavelength λ_j
2. $L'(D, j) = 1/a_{SD}$ if the path from S to D is a single link (S is a neighbour of D), a_{SD} is the number of currently available wavelengths on this link.
3. $L'(D, j) = max(1/a_{Si}, L'_i(D, \lambda)$ if the path from S to D goes through neighbouring node i; $L'_i(D, j)$ is the value of relative capacity loss obtained from node i, a_{Si} is the number of currently available wavelengths on the link to i.

The capacity loss for a path destined to D on wavelength λ_j is calculated as the sum of relative capacity loss values for other destinations.

2.4 Integrated RWA algorithms

In the AUR/EXHAUSTIVE method [44], a shortest path is computed for each wavelength, taking into consideration the wavelength usage in the network. The calculation includes only the links that have a given wavelength free. The shortest path from the computed ones is chosen for the lightpath setup.

Dynamic Routing presented in [22] is based on two algorithms: alternate routing with limited trunk reservation and the Max Wave wavelength assignment policy. Given the lightpath request for an S-D pair, the algorithm calculates:

1. for each wavelength λ_i, the number of links a^i on which the wavelength is free in the whole network,
2. the hop count, h_j, and the number of free wavelengths, A^{p_j}, for each of the alternate paths $p_j \in P_{SD}$ connecting nodes S and D.

For networks with full wavelength conversion A^{p_j} will be the number of free wavelengths on the busiest link. For networks without wavelength conversion this will be the number of continuous (free) wavelengths.

The algorithm uses the constants α_1 and α_2 to calculate the weighted sum of the variables: $\delta_{i,j} = \alpha_1 a^i + (1-\alpha_1)\{\alpha_2(w - A^{p_j}) + (1-\alpha_2)h_j\}$. The (path, wavelength) pairs are considered starting from the smallest $\delta_{i,j}$. If there is more than one pair with the same $\delta_{i,j}$, pairs with smallest a^i are preferred in the first place. If the choice is still not unique, a pair with the largest $\alpha_2(w - A^{p_j}) + (1 - \alpha_2)h_j$ is preferred. If it is not possible to establish the connection on the selected combination of the path and wavelength, the next $\delta_{i,j}$ is considered. For the case of $\alpha_1 = 0$ the wavelength may be chosen according to the First Fit rule.

The Least Loaded Routing (LLR) [30] algorithm simultaneously chooses a path and wavelength. The path is chosen from k shortest paths. The algorithm chooses path p and wavelength λ_j that satisfy $\max_{p,j} \min_{e \in p} f_e - b_e^j$ where f_e is the number of fibres on link e, b_e^j is the number of fibres on link e on which wavelength j is busy, and p is the path chosen from the set of k shortest paths between the source and the destination. Ties are broken using the Most Used rule, if the tie still cannot be broken, the shortest path is chosen.

The Min Sum Routing (MSR) algorithm [30] computes for each (path, wavelength) pair the following cost value $\sum_{e \in p} \frac{b_e^j}{f_e}$ and chooses the pair with the minimum cost.

The algorithms LLR and Min Sum Routing were compared in [30] on a multifibre 30-node mesh network. Increasing the number of alternate paths taken into consideration results in lowering the blocking probability. The blocking probability drops much faster for LLR than for MSR, but the difference diminishes significantly with increasing load. The simulations show that considering many alternate paths gives good results for single fibre networks, and as the network contains more fibres per link the benefit of considering many (more than three) alternate paths diminishes. Nevertheless, it is beneficial to use more than one path for setting up a request. The simulations also show, that for multifibre networks with alternate paths the benefit of wavelength conversion is small.

[34] introduces the concept of an auxiliary wavelength graph. The auxiliary graph is built upon the original network topology, but includes the details of the wavelength conversion. The network is modeled as w physical networks, each representing the specific wavelength. The planes are interconnected according to the wavelength conversion capabilities in each node. This enables the use of standard path search algorithms (for example shortest path) for simultaneous routing and wavelength assignment. The auxiliary wavelength graph must be changed after each lightpath setup/release to reflect current availability of wavelengths and converters. The algorithm presented in [34] uses shortest path search on an auxiliary graph. Each free edge associated with a wavelength on a physical link has some finite cost c, each free wavelength converter edge has some finite cost c_c. Occupied wavelengths and converters are reflected in the infinite weight. Ties are broken using the smallest wavelength index. The algorithms, presented

in [12] and [36] allow for reducing the computational complexity. Moreover, the algorithm presented in [36] is suitable for distributed computing.

The Layered-Graph method, presented in [72] is an extension of the Auxiliary Wavelength Graph to multifibre networks, but assumes no wavelength conversion. In this method the physical network topology is converted to a layered graph, where each wavelength occupies a separate wavelength plane. The topology of each wavelength plane reflects the physical network topology. A virtual source and destination nodes exist for each physical nodes. Each of them is connected to the corresponding nodes in the wavelength plane using unidirectional edges. These are the only common points for the wavelength planes.

Two strategies are suggested. The first one (PACK) assumes, that as long as a given wavelength is free on at least one fibre on a given link, the corresponding edge has a constant weight. The weight changes to ∞ when the wavelength is occupied on all fibres on the link. The second (SPREAD) one assumes that the weight of edge where the corresponding wavelength is free on all fibres is the initial cost. The cost increases by a constant value with each busy fibre on a given wavelength on the link. Finally, the weight changes to ∞ when the wavelength is occupied on all fibres on the link.

Routing and wavelength assignment is done by searching for the shortest path on the wavelength graph with edge weights set up according to PACK or SPREAD approaches. If there are many concurrent paths with the same length, the most used wavelength and link is used for PACK, and the least used link and wavelength for SPREAD.

The SPREAD method outperformed Fixed routing with First Fit wavelength assignment, Alternate routing with 2 paths with Random wavelength assignment, and FPLC. The comparison on the NSFNET topology shows, that SPREAD outperforms PACK, and the difference increases with the number of fibres. The method may be used for networks with uneven number of fibres on links, and with different cost for each link.

The paper [4] utilises a similar approach as the Layered Graph Method but proposes a different cost metric. The metric of each edge in a wavelength graph, c_e^j is defined as follows:

$$c_e^j = \begin{cases} \frac{1}{a_e^j} & \text{if} a_e^j \neq 0, \\ \infty & \text{if} a_e^j = 0 \end{cases} \tag{2.9}$$

Based on the proposed cost metric, three different RWA schemes are proposed. In a total-cost based selection path p on wavelength λ_j which has the minimum cost $\delta_{j,p}$ is chosen. The cost $\delta_{j,p}$ is calculated in the following way:

$$\delta_{j,p} = \sum_{e \in p} c_e^j \tag{2.10}$$

For the balanced cost-based selection the cost $\delta_{j,p}$ is calculated the following way:

$$\delta_{j,p} = h_p \cdot \sum_{e \in p} c_e^j \tag{2.11}$$

where h_p is the hop count (number of links) of path p.

For the future cost-based selection the cost $\delta_{j,p}$ is calculated the following way:

$$\delta_{j,p} = \sum_{e \in p} c_e'^j \tag{2.12}$$

where

$$c_e'^j = \begin{cases} \frac{1}{a_e^j - 1} & \text{if} a_e^j > 1, \\ \infty & \text{if} a_e^j \leq 1 \end{cases} \tag{2.13}$$

and if all paths calculated using this method have a weight of ∞ then the path is chosen according to the total-cost based selection scheme.

The presented results show that the future cost-based selection scheme outperforms the other two algorithms and it also outperforms fixed routing with First Fit wavelength assignment.

2.5 QoS and GoS related work

This section presents previous works on QoS and GoS routing in optical networks. Quality of Service (QoS) deals with connection parameters that affect data flow after the lightpath is established. Grade of Service (GoS) includes all parameters applicable to connection setup, such as connection blocking probability and setup time.

The paper [8] introduces the concept of wavelength reservation. The purpose of the presented algorithm is to equalise the blocking probability between requests of short and long lightpaths. Some wavelengths are reserved on some links for the exclusive use by the specific traffic. The reserved links and wavelengths were chosen manually, based on blocking probabilities for each S-D pair. The authors note that “Reserving wavelengths puts up barriers to sharing of the wavelength resource. Thus while enhancing the performance of one traffic stream it is expected to decrease the global network performance.”

The paper [8] also introduces the concept of protecting threshold. This concept is based on the trunk reservation techniques used in circuit switched networks and is used for protecting resources for lightpaths spanning multiple hops from single hop lightpaths, which are easier to establish. The single hop lightpath will not be established if the number of free wavelengths on a given link is less than the specified threshold.

The authors of [53] use fixed-alternate routing with wavelength reservation. Alternate paths are used only if the number of continuous wavelengths on them is greater than the specified threshold. The presented results suggest that this mechanism is beneficial in some cases as it prevents consuming too much resources by the alternate (longer) paths.

In [11] the authors analyse the proportional differentiation among traffic classes and propose three algorithms: intentional blocking, intentional termination, and a hybrid algorithm that is a combination of the two previously mentioned. All algorithms constantly monitor blocking probabilities in all classes of lightpath requests and either intentionally block some requests, or intentionally terminate some existing lightpaths to achieve the proper relationship between blocking probabilities in each class. As a result, the higher priority class the lower the blocking probability.

In [65] the authors analyse the performance of an algorithm in which each class of lightpath requests is given a specific range of wavelengths to choose from. This concept is further evaluated using a continuous-time Markov chain and computer simulation.

In [45] the authors propose the use of absorption probability as a measure of grade of service in the optical network. The absorption probability at time t is the probability that at least one lightpath request will be blocked before time t. In the paper the authors analyse the concept using transient analysis of the Markov chain model of the network and compare the analytical results with the results obtained from computer simulation.

In [52] the authors propose a model to evaluate the bit error rate (BER) of the lightpath during the routing and wavelength assignment. The lightpath is set up only if it meets the specified QoS, expressed in terms of BER.

In [66] the authors propose an impairment constraint-based routing scheme, in which a path computed using k-shortest path routing is assessed according to a given model. The path that does not pass the validation phase is removed from the set of k paths and the calculation is repeated. The paper discusses physical impairments present in the optical networks: polarisation mode dispersion (PMD), amplified spontaneous emission (ASE) noise, crosstalk, chromatic dispersion and filter concatenation and analyses their impact on connection quality.

In [27] the authors propose three methods for routing and wavelength assignment that take into account the QoS parameters of the lightpath: the least quality resource allocation, the minimal wavelength shifting, and alternate routing for overloaded multiwavelength resources. The methods rely on the definition of appropriate cost functions in the transformed network graph. The concept of service-specific wavelength-resource graphs is further investigated in [28].

In [29] the authors propose a method for constraint-based path selection for dynamic RWA in the optical networks based on flooding probe messages, processing the messages in an appropriate way in the intermediate nodes, and choosing the most suitable path in the destination node. While traversing the network the probe messages collect information regarding signal degradation in traversed elements. The collected information, including physical impairments, reliability, policy and traffic conditions allows the choice of the path that meets the necessary QoS and QoR constraints.

In [37] the authors propose routing and wavelength assignment based on a genetic algorithm, taking into account the polarisation mode dispersion (PMD) and amplified spontaneous emission (ASE). The algorithm also chooses the location of wavelength converters and compensator nodes.

In [2] the authors propose two wavelength assignment schemes which take into account the four wave mixing (FWM) effect. A similar RWA approach is presented in [20] and [21]. The authors propose models allowing the estimation of FWM effect along a candidate path-wavelength pair. The admission of a lightpath request is based on the expected QoS on a candidate path-wavelength pair. One of the proposed routing methods also takes into consideration the impact of the lightpath setup on existing lightpaths and prevents setup if the QoS of existing lightpaths would not be satisfactory.

The authors of [41] propose another method in which the algorithm selects the best path in terms of noise figure and checks whether the path satisfies the required QoS. The authors provide a model to calculate noise figure based on parameters of the network elements carrying the optical signal (fibres, optical amplifiers, input and output taps).

The authors of [51] propose two adaptive RWA algorithms called “highest Q factor” and “max min Q factor” which take the bit error rate (BER) of the current connection request and BERs of the existing connections into account while making both routing and wavelength assignment decisions. The BER is calculated based on estimated crosstalk effects.

The authors of [23] propose two methods for RWA which include physical impairments in the calculations. The methods are based on calculating a candidate path and assessing, using the provided model, the path quality. If the candidate path has insufficient quality the next path is considered. The first RWA method used in the paper is based on AUR/Exhaustive, the second is based on AUR/FIXED. If the path cannot be found, or it does not meet the QoS criteria, the next wavelength is considered. The model used for assessing the quality of a candidate lightpath takes into consideration the Polarisation Mode Dispersion (PMD) and the Optical Signal-to-Noise Ratio (OSNR).

3 Proposed mechanisms and strategies

This chapter provides an overview, classification and presentation of the proposed Grade-of-Service routing mechanisms and strategies. The chapter begins with a section that contains a proposed classification of the mechanisms and introduces the notation conventions which will be used throughout the remaining part of the chapter. The next sections, Section 3.2, 3.3 and 3.4, present the proposed mechanisms. Finally, Section 3.5 present strategies built upon the presented mechanisms. The strategies are then evaluated in Chapters 5 and 6.

3.1 Overview and classification

A Grade-of-Service routing strategy is built on one or more mechanisms that provide differentiated services in an optical network. There are at least three groups of those mechanisms [61]:

(a) mechanisms which preserve network resources for high priority requests,

(b) mechanisms in which a different routing algorithm, path metrics or a set of candidate paths is considered for each class of requests,

(c) mechanisms in which an existing lower class lightpath may be preempted or rerouted if the resources it uses are needed for a higher class lightpath.

The focus of this dissertation is on the first two groups of mechanisms. Thus, in the next two sections these mechanisms are thoroughly presented alongside with detailed algorithms. The third group, based on preemption, will only be briefly discussed.

The algorithms are presented in two versions: one suitable for networks with the centralised computation model and one for networks with the distributed computation model. These are the two architectures implementing the Path Computation Element (PCE) [18]. The distributed version of the algorithms is prepared to interact with the RSVP-TE signalling [7].

The descriptions use a common set of symbols. All algorithms operate on a lightpath request R from node S to node D. Request R is either a low or a high priority request. Each link carries w wavelengths in each direction and $\Omega = \{\lambda_1, \ldots, \lambda_w\}$ is the set of wavelengths in a network. p denotes a path (a concatenation of links), which will be used to setup request R. A is the set of wavelengths considered for the lightpath setup. $|A|$ is the number of elements in A. A_0 is the set of unused (free) wavelengths on the first link of path p. T is a configurable, integer threshold parameter that determines how much resources are preserved for high priority requests.

3.2 Mechanisms which preserve network resources for high priority requests

The general idea behind this group of mechanisms is to provide a notion of resources that are protected from being used by the low priority lightpaths. Effectively, this increases the chances that future high priority requests will find sufficient resources in the network at the time they arrive. The preservation of resources is a kind of "an educated guess," since future high priority requests are not known at the time resources are preserved.

One of the possible ways of preserving resources for future higher priority lightpaths is to define a threshold. The threshold describes the amount of resources that should remain free for future high priority requests. If the amount of free resources is less than or equal to a given threshold, the network rejects low priority requests. The higher the threshold, the greater amount of resources preserved, and less resources available for low priority requests. Consequently, increasing the threshold value decreases the blocking probability of high priority requests and, simultaneously, increases the blocking probability of low priority requests. Thus, it is necessary to achieve a proper balance between the aggressiveness of resource preservation and the resulting penalty imposed on the low priority requests.

The degradation of the blocking probability of the low priority requests is especially important for low and moderate offered traffic. In such conditions the network could (potentially) work with a non-GoS-aware mechanisms, and all classes of requests, including the high priority class would get a sufficiently low blocking probability. If the GoS mechanism is used to extend the amount of the offered traffic that can be accommodated by the network without increasing the blocking in the high priority class above a required level, and for the majority of time the network remains in the low traffic region, then this is the region of the most aggressive comparison between strategies, and the place where the heated debate may arise on whether it is beneficial to use this kind of mechanisms or not.

Three of the presented mechanisms were introduced earlier in [61] and [63], one was presented earlier in [61]. In contrast to the earlier versions, the algorithms discussed here may be used with any routing algorithm and any wavelength assignment algorithm.

All of the presented mechanisms, except one, work according to a similar schema:

1: Find the first path to the destination, according to the chosen routing algorithm;

2: Accept or decline the path using the chosen GoS assurance mechanism;

3: If the path is accepted and there is at least one continuous wavelength on the path, assign the wavelength and set up the lightpath;

4: Otherwise, choose the next path, if available, and return to step 2;

5: If there is no next path available, block the request.

Depending on the control plane solution used, there may exist three options: either all decisions regarding the lightpath setup are taken by a single centralised PCE which maintains the exact network state (stateful PCE), or the decisions are taken by a single centralised PCE which is stateless, or they are taken in a distributed manner. In the first case, the PCE possesses all information required to setup the lightpath, including the current state of the available resources in the network. However, in the second and third cases there is no central database that collects the status of all resources. Hence, the PCE must use other means to gather information about the availability of the resources.

The first source of information might be the routing updates carrying the current status of all links in the network. However, this information may be more or less outdated, depending on the delay between updates and the way they are triggered. Only one of the presented algorithms requires this kind of information to be available.

The second source of information is the lightpath setup process which includes resource reservation. Usually, backward reservation is used for that purpose, since it is more efficient (see [19]). RSVP-TE [5] is one of the protocols that implement this paradigm. In this protocol resource reservation is performed in two stages. In the first one a PATH message is sent in the forward direction and collects wavelength availability along the chosen path. The destination node, upon receiving the PATH message, chooses the wavelength and responds with a RESV message which reserves the chosen wavelength along the path. This constitutes the second stage of the lightpath setup. Most of the presented algorithms are prepared to be performed while processing the PATH message, i.e., during the first stage of the lightpath setup. This gives an additional benefit of having more precise information regarding the available resources in the network, compared to the information carried in routing updates.

The numerical complexity of the presented algorithms is estimated solely as the computational cost of adding the presented mechanisms to the network that does not implement GoS mechanisms. Thus, the numerical complexity includes neither the complexity of calculating a candidate path nor a complexity of wavelength assignment after a candidate path is approved. In the subsequent formulas the following notation is used: n will denote the number of nodes in the network; l will denote the number of links in the network; w will denote the number of available wavelengths. The value of l may be substituted with $n(n-1)$ for the worst case of a full mesh network. However, most networks are more than 2 and less than 3-connected, thus real-life l seems to be a linear function of n;

3.2.1 Wavelength pools (pool)

This mechanism was first investigated in [65], where the authors presented the concept and analysed it using both a continuous-time Markov chain and computer simulation. The version presented here is a slight modification of the original concept. The whole set of wavelengths available on each link is divided into two pools: the first, common pool may be used by all lightpath requests. The second (preserved) pool may be used only by high priority lightpath requests and is used if it is not possible to set up a request using the common pool. Parameter T describes here the number of wavelengths in the preserved pool.

The acceptance of a candidate path for a low priority request depends on the availability of free wavelengths on the path in the common pool. If there is at least one wavelength free, the candidate path is accepted, otherwise it is rejected. For the high priority requests a candidate path is always accepted. It is important to note, however, that this mechanism requires cooperation from the wavelength assignment algorithm, especially for the high priority requests, since the requests must be established using the common pool, if possible. The preserved pool should only be used as the last resort.

The mechanism requires almost no additional effort from the control plane, since it only affects wavelength choice in the destination node. During lightpath setup the PATH message collects wavelength usage information on the whole path. Based on this information the destination node chooses the wavelength on which the request will be set up. The described algorithm may be implemented in this step, rejecting the candidate path if a low priority request is served and there are no free continuous wavelengths in the common pool. A detailed algorithm description for this mechanism is presented below. The numerical complexity of the algorithm may be estimated as $O(lw)$ since to compute the available number of wavelengths on a candidate path it is necessary to check all links on the path, the number of which is upper-bounded by l, and on each link it is necessary to check w wavelengths to remove busy wavelengths from the set of available wavelengths. Then, for a low priority request it is necessary to remove the wavelengths corresponding to the preserved pool from the set. Finally, the number of elements in the resulting set of available wavelengths needs to be counted, and this set may contain, at maximum, w elements.

The algorithm for the wavelength pools strategy (pool)

Networks with a centralised control:

1: Compute the first candidate path p from node S to node D according to the preferred routing algorithm.

2: Compute the set of available wavelengths $A \subset \Omega$ so that each $\lambda_i \in A$ is free on all links on path p.

3: If R is a low priority request then remove wavelengths $\lambda_i : i > w - T$ from set A.

4: If $A = \emptyset$ then decline the candidate path p and go to step 7.

5: Choose wavelength $\lambda_j \in A$ according to the preferred wavelength assignment algorithm.

6: Set up request R on path p and wavelength λ_j and FINISH.

7: Compute the next candidate path p; if it exists go to step 2, otherwise block request R and FINISH.

Networks with a distributed control:

1: In node S, compute the first candidate path p from node S to node D according to the preferred routing algorithm.

2: In node S:

2.1: if R is a low priority request then send an explicitly routed PATH message to node D along path p that contains an initial set of the available wavelengths $A = A_0 \setminus \{\lambda_{w-T+1} \dots \lambda_w\}$;

2.2: if R is a high priority request then send an explicitly routed PATH message to node D along path p that contains an initial set of the available wavelengths $A = A_0$.

3: While traversing path p set A in the PATH message is updated and the wavelengths that are busy on a traversed link are removed from set A. If, at any intermediate node, $A = \emptyset$ then the candidate path is rejected and a PathErr message is sent towards node S; go to step 6.

4: In node D, receive the PATH message:

4.1: if $A = \emptyset$ then the candidate path is rejected and a PathErr message is sent towards node S; go to step 6;

4.2: otherwise λ_j is chosen among the available wavelengths ($\lambda_j \in A$) according to the preferred wavelength assignment algorithm and a RESV message is sent towards node S.

5: In node S, on receiving the RESV message the lightpath is established, so FINISH.

6: Compute the next candidate path p from node S to node D, if it exists go to step 2, otherwise block request R and FINISH.

In the presented algorithm $A_0 \setminus \{\lambda_{w-T+1} \dots \lambda_w\}$ is the set of free wavelengths from the common pool which are free on the first link of the candidate path p.

3.2.2 First link capacity threshold (flcap)

This algorithm is one of the simplest algorithms proposed. It works by checking whether there are sufficient resources on the first link of a candidate path. For low priority requests a candidate path is accepted if the number of free wavelengths on the first link of the path is greater than the specified threshold T. On the other hand, if the number of free wavelengths on the first link of a candidate path is less than or equal to T then the candidate path is rejected. For high priority requests the candidate path is always accepted.

The mechanism is very simple, and not very effective. Its main drawback is a very limited scope of resources that it preserves (just one link) and the fact that it does not pay any attention to the wavelength continuity which is a major contributor to the blocking probability in the optical networks that do not have wavelength conversion capabilities.

Since the number of free wavelengths on the first link of the lightpath should be locally accessible in the source node, the algorithm does not rely on any kind of mechanism that distributes the information about currently available resources throughout the network. The numerical complexity of the presented algorithm may be estimated as $O(w)$, since it is only necessary to check the set of w wavelengths on a single link, and this information is available in the local node. The formal description of the mechanism is presented below.

The algorithm for the first link capacity threshold strategy (flcap)

Networks with a centralised control:

1: Compute the first candidate path p from node S to node D according to the preferred routing algorithm.

2: If R is a high priority request then go to step 4.

3: If the number of free wavelengths on the first link is less than or equal to T then decline the candidate path p and go to step 8.

4: Compute the set of available wavelengths $A \subset \Omega$ so that each $\lambda_i \in A$ is free on all links on path p.

5: If $A = \emptyset$ then decline the candidate path p and go to step 8.

6: Choose wavelength $\lambda_j \in A$ according to the preferred wavelength assignment algorithm.

7: Set up request R on path p and wavelength λ_j and FINISH.

8: Compute the next candidate path p; if it exists go to step 2, otherwise block request R and FINISH.

Networks with a distributed control:

1: In node S, compute the first candidate path p from node S to node D according to the preferred routing algorithm.

2: In node S,

2.1: If R is a low priority request and the number of free wavelengths on the first link is less than or equal to T ($|A_0| \leq T$) then decline the candidate path and go to Step 6.

2.2: Send an explicitly routed PATH message to node D along path p that contains an initial set of available wavelengths $A = A_0$,

3: While traversing path p set A in the PATH message is updated and the wavelengths that are busy on the traversed links are removed from set A. If, at any intermediate node, $A = \emptyset$ then the candidate path is rejected and a PathErr message is sent towards node S.

4: In node D, receive the PATH message:

4.1: if $A = \emptyset$ then the candidate path is rejected and a PathErr message is sent towards node S;

4.2: otherwise λ_j is chosen among the available wavelengths ($\lambda_j \in A$) according to the preferred wavelength assignment algorithm and a RESV message is sent towards node S.

5: In node S, on receiving the RESV message the lightpath is established, so FINISH.

6: Compute the next candidate path p from node S to node D; if it exists go to step 2, otherwise block request R and FINISH.

3.2.3 Link capacity threshold (lcap)

This algorithm is an extension of the previous one, since given a candidate path it checks the number of free wavelengths on each link of the path. For a low priority request, if there are more than T free wavelengths on each link then the path is accepted. Otherwise the path is rejected. For high priority requests the path is always accepted. As a result, at least T wavelengths should remain free on each link (for future high priority lightpath requests) after the current low priority lightpath request is satisfied.

The numerical complexity of the algorithm may be estimated as $O(lw)$ since it is necessary to check all links on a candidate path, the number of which is upper bounded by l, and on each link it is necessary to check w wavelengths.

The presented mechanism has a broader scope of resources it preserves. However, it still does not pay any attention to wavelength continuity. In a busy network, an incoming high priority lightpath request might find that there are some wavelengths free on each link on a path, but none of them is continuously free on the whole path. The formal description of the algorithm for networks with both centralised and distributed control is presented below.

The algorithm for the link capacity threshold strategy (lcap)

Networks with a centralised control:

1: Compute the first candidate path p from node S to node D according to the preferred routing algorithm.

2: If R is a high priority request then go to step 4.

3: For each link e_i on path p:

3.1: if the number of free wavelengths on link e_i is less than or equal to T then decline the candidate path p and go to step 8.

4: Compute the set of available wavelengths $A \subset \Omega$ so that each $\lambda_i \in A$ is free on all links on path p.

5: If $A = \emptyset$ then decline the candidate path p and go to step 8.

6: Choose wavelength $\lambda_j \in A$ according to the preferred wavelength assignment algorithm.

7: Set up request R on path p and wavelength λ_j and FINISH.

8: Compute the next candidate path p; if it exists go to step 2, otherwise block request R and FINISH.

Networks with a distributed control:

1: In node S, compute the first candidate path p from node S to node D according to the preferred routing algorithm.

2: In node S, send an explicitly routed PATH message to node D along path p that contains an initial set of available wavelengths $A = A_0$,

3: While traversing path p:

3.1: set A in the PATH message is updated and the wavelengths that are busy on the traversed link are removed from set A;

3.2: if, at any intermediate node, $A = \emptyset$ then the candidate path is rejected and a PathErr message is sent towards node S; go to step 6;

3.3: if the low priority request is being served and if there are T or less free wavelengths on the traversed link then the candidate path is rejected and a PathErr message is sent towards node S; go to step 6.

4: In node D, receive the PATH message:

4.1: if $A = \emptyset$ then the candidate path is rejected and a PathErr message is sent towards node S; go to step 6;

4.2: otherwise λ_j is chosen among the available wavelengths ($\lambda_j \in A$) according to the preferred wavelength assignment algorithm and a RESV message is sent towards node S.

5: In node S, on receiving the RESV message the lightpath is established, so FINISH,

6: Compute the next candidate path p from node S to node D, if it exists go to step 2, otherwise block request R and FINISH.

3.2.4 Path capacity threshold (pcap)

In this algorithm the concept of preserving resources has been extended one step further, to include the wavelength continuity. The basis for accepting or declining a candidate path is the number of free continuous wavelengths on that path. This number is quite easily obtained while performing wavelength reservation for a lightpath setup, since the PATH message must collect wavelength usage on the candidate path. For a low priority request the candidate path is accepted if the number of free continuous wavelengths is greater than threshold T. Otherwise it is rejected. For high priority requests the candidate path is always accepted. Hence, at least T high priority requests may be later set up on that path.

This mechanism pays a great deal of attention to the wavelength continuity problem, which, as it will turn out later, gives very good results, while still requiring a low computational effort and reasonable control plane requirements.

The detailed version of the presented algorithm may be found below, it differs from the one presented earlier in [61] and [63] in that it allows an arbitrary routing and wavelength assignment algorithm, instead of the shortest path routing with first-fit wavelength assignment.

Similarly to the previous mechanism, the numerical complexity of the algorithm may be estimated as $O(lw)$ since it is necessary to check all links on a candidate path, which is upper-bounded by l, and on each link w wavelengths must be checked to remove busy wavelengths from the set of available wavelengths. Finally, the number of elements in the resulting set of available wavelengths needs to be counted, an this set may contain, at maximum, w elements.

The algorithm for the path capacity threshold strategy (pcap)

Networks with a centralised control:

1: Compute the first candidate path p from node S to node D according to the preferred routing algorithm.

2: Compute the set of available wavelengths $A \subset \Omega$ so that each $\lambda_i \in A$ is free on all links on path p.

3: If $A = \emptyset$ then decline the candidate path p and go to step 7.

4: If R is a low priority request and $|A| \leq T$ then decline the candidate path p and go to step 7.

5: Choose wavelength $\lambda_j \in A$ according to the preferred wavelength assignment algorithm.

6: Set up request R on path p and wavelength λ_j and FINISH.

7: Compute the next candidate path p; if it exists go to step 2, otherwise block request R and FINISH.

Networks with a distributed control:

1: In node S, compute the first candidate path p from node S to node D according to the preferred routing algorithm.

2: In node S, send an explicitly routed PATH message to node D along path p that contains an initial set of available wavelengths $A = A_0$.

3: While traversing path p set A in the PATH message is updated and the wavelengths that are busy on the traversed link are removed from set A;

3.1: if, at any intermediate node, $A = \emptyset$ then the candidate path is rejected and a PathErr message is sent towards node S; go to step 6;

3.2: else if R is a low priority request and $|A| \leq T$ then the candidate path is rejected and a PathErr message is sent towards node S; go to step 6.

4: In node D, receive the PATH message:

4.1: if $A = \emptyset$ then the candidate path is rejected and a PathErr message is sent towards node S;

4.2: else if R is a low priority request and $|A| \leq T$ then the candidate path is rejected and a PathErr message is sent towards node S;

4.3: else λ_j is chosen from the available wavelengths ($\lambda_j \in A$) according to the preferred wavelength assignment algorithm and a RESV message is sent towards node S.

5: In node S, on receiving the RESV message the lightpath is established, so FINISH,

6: Compute the next candidate path p from node S to node D, if it exists go to step 2, otherwise block request R and FINISH.

3.2.5 Global capacity threshold (gcap)

This is the most complex mechanism, out of the ones presented here. It pays extreme attention to the wavelength continuity constraint, taking into account not only the current path but also the paths that will be potentially used by future lightpath requests. However, the price paid for this is considerable and consists of the computational effort and signalling requirements.

The decision regarding the acceptance of the current path is based on the global state of the network, which must either be known or obtained during the evaluation process. The idea behind this mechanism is to preserve wavelength continuous paths for future high priority requests, so that after a low priority lightpath is set up, there are enough resources in the network to set up at least T high priority lightpaths between any pair of nodes, where T is a given threshold. High priority requests are immediately admitted by this mechanism, without any further checks.

In contrast to the previously mentioned mechanisms, this one does not accept a candidate path, but the exact path-wavelength pair. As a result, each path-wavelength pair must be evaluated separately. This requires slight modifications to the general algorithm presented at the beginning of Section 3.2:

1: Find the first path to the destination, according to the chosen routing algorithm;

   1a: Take the appropriate wavelength, according to the preferred wavelength assignment algorithm;

2: Accept or decline the path-wavelength pair using the global capacity threshold mechanism;

3: If the path-wavelength pair is accepted then set up the lightpath;

4: Otherwise;

   4a: Choose the next wavelength, if available and return to step 2;

   4b: If there is no next wavelength available, choose the next path, if available, and return to step 1a;

5: If there is no next path available, block the request.

Evaluating a candidate path-wavelength pair for a low priority lightpath request is performed in three steps. In the first step, the potential number of lightpaths that may be set up among each pair of nodes in the network is evaluated, assuming the current network state. In the second step the check is repeated, with the resources used by the candidate path-wavelength pair marked as busy. This way, for each potential source-destination pair there are two numbers: X_{sd}, the potential number of lightpaths that may be set up currently and Y_{sd}, the potential number of lightpaths that may be set up after the current lightpath request is established using the candidate path-wavelength pair. In step three, those numbers are compared against threshold T. If there exists a source-destination pair for which $Y_{sd} < X_{sd}$ and $Y_{sd} < T$, then the candidate path is rejected. Otherwise it is accepted. In other words, the candidate path-wavelength pair is rejected if setting up a request would reduce the potential number of lightpaths between any source-destination pair below T. The important word here is "to reduce" since the pairs that are not affected by the current lightpath setup are ignored, regardless of whether the number of potential lightpaths is already below T or not.

There is one important issue left: the algorithm to compute the potential number of lightpaths among the given pair of nodes. In the rest of the dissertation, this is the number of continuous free wavelengths on the shortest path between the nodes. The choice was based on two factors: the shortest path is used as the first (or the only one) candidate path, so effectively the "preserved" resources are likely to be used in the future and, second, further complication of the already complex mechanism is avoided.

The numerical complexity of this algorithm may be estimated in the following way. To evaluate a candidate path it is necessary to evaluate at most w wavelengths. For each wavelength there is a need to evaluate $n(n-1)$ communicating pairs. Assuming that the potential paths for each potential pair are precomputed and assuming the complexity of evaluating each s-d pair as $O(lw)$ (the complexity of the path capacity threshold mechanism), the resulting complexity is $O(n^2lw^2)$, which is considerably worse than the complexity of the other algorithms.

The algorithm for the global capacity threshold strategy (gcap)

Networks with a centralised control:

1: Compute the first candidate path p from node S to node D according to the preferred routing algorithm.

2: Compute the set of available wavelengths $A \subset \Omega$ so that each $\lambda_i \in A$ is free on all links on path p.

3: If R is a low priority request and $|A| \leq T$ then decline the candidate path p and go to step 8.

4: If $A = \emptyset$ then decline the candidate path p and go to step 8.

5: Choose wavelength $\lambda_j \in A$ according to the preferred wavelength assignment algorithm.

6: If R is a low priority request then assess the candidate path-wavelength pair (p, j) using the assessment algorithm (presented later in the text); if the candidate path-wavelength pair is accepted then go to step 7, otherwise remove wavelength j from set A and go to step 4.

7: Set up request R on path p and wavelength λ_j and FINISH.

8: Compute the next candidate path p; if it exists go to step 2, otherwise block request R and FINISH.

Networks with a distributed control:

1: In node S, compute the first candidate path p from node S to node D according to the preferred routing algorithm.

2: In node S, send an explicitly routed PATH message to node D along path p that contains an initial set of available wavelengths $A = A_0$.

3: While traversing path p set A in the PATH message is updated and the wavelengths that are busy on the traversed link are removed from set A;

3.1: if, at any intermediate node, $A = \emptyset$ then the candidate path is rejected and a PathErr message is sent towards node S; go to step 6;

3.2: else if R is a low priority request and $|A| \leq T$ then the candidate path is rejected and a PathErr message is sent towards node S; go to step 6.

4: In node D, receive the PATH message:

  4.1: if R is a low priority request and $|A| \leq T$ then the candidate path is rejected and a PathErr message is sent towards node S; go to step 6;

  4.2: if $A = \emptyset$ then the candidate path is rejected and a PathErr message is sent towards node S; go to step 6;

  4.3: if R is a high priority request then choose a wavelength index j using the preferred wavelength assignment algorithm so that $\lambda_j \in A$ and send a RESV message towards node S with the chosen wavelength index j; go to step 5;

  4.4: if R is a low priority request then choose a wavelength index j using the preferred wavelength assignment algorithm so that $\lambda_j \in A$ and assess a candidate path-wavelength pair (p, j) using the assessment algorithm (presented later);

  4.5: if the path-wavelength pair (p, j) is accepted then send a RESV message towards node S with the chosen wavelength index j; go to step 5;

  4.6: if the path-wavelength pair (p, j) is declined then remove wavelength λ_j from set A and go to step 4.2.

5: In node S, on receiving the RESV message the lightpath is established, so FINISH,

6: Compute the next candidate path p from node S to node D, if it exists go to step 2, otherwise block request R and FINISH.

The assessment algorithm, presented below, assumes that the global network state is known to the node performing the algorithm.

For a candidate path-wavelength pair (p, j):

1: Take the first potential pair of source S' and destination D' nodes;

- 1.1: compute the potential path p' from node S' to node D' according to the shortest path rule;
- 1.2: if paths p and p' are link-disjoint (i.e. have no common links) then go to step 1.5;
- 1.3: compute the set of available wavelengths $A' \subset \Omega$ so that each $\lambda_i \in A'$ is free on all links on path p';
- 1.4: if $|A' \setminus \{\lambda_j\}| < T$ and $|A' \setminus \{\lambda_j\}| \neq |A'|$ then decline the candidate path-wavelength pair (p, j) and FINISH;
- 1.5: take the next potential pair of source S' and destination D' nodes and go to step 1.1.

2: Accept the candidate path-wavelength pair (p, j) and FINISH.

In the presented assessment algorithm, $A' \setminus \{\lambda_j\}$ is the resulting set of the available wavelengths on path p' after the current request is set up on the path-wavelength pair (p, j), assuming that paths p and p' have at least one link in common.

3.3 Mechanisms in which a different routing algorithm or a set of candidate paths is considered for each class of requests

In this class of mechanisms the low and high priority requests are differentiated by using a different set of potential paths or a different routing algorithm for each class of requests. Low priority requests are served with a simple algorithm, which provides an inferior blocking probability, while high priority requests are served with a more advanced algorithm that provides a lower blocking probability.

One of the possible implementations of this mechanism is the fixed-alternate routing with a different number of paths for each class of requests. High priority requests are offered more paths while low priority requests are offered fewer paths to consider. This implementation has been used in the subsequent investigations.

There are two important issues regarding the presented mechanism. First, there is a problem with parameter tuning since the number of alternate paths in the network is usually small, given the low connectivity degree of the underlying network. As a result, it may be difficult, or even impossible to maintain the proper blocking probability of the high priority requests below a given value for all assumed traffic conditions.

Second, when assessing the performance of strategies implementing this mechanism it is important to note, that there are, in fact, two reference cases without differentiation. The first one uses a simple algorithm for all requests, the second uses the complex algorithm for all requests. In both reference cases the low and high priority requests are treated equivalently, but the overall network performance varies. This issue has been discussed in more detail in [61] and in Section 5.6.

The following algorithm is used to compute the set of alternate paths:

1: Set the metric of all links in the topology to 1;

2: The first path between two nodes is obtained by using the shortest path algorithm;

3: Set the metric to n for all links that belong to the first path; n is the number of nodes in the network;

4: Compute the second path using the shortest path algorithm;

5: Set the metric to n for all links that belong to the second path; n is the number of nodes in the network;

6: Compute the third path using the shortest path algorithm.

One might be tempted to compute alternate paths as link disjoint paths. However, due to the fact that the underlying topologies have the connectivity degree between 2 and 3, it is often impossible, and hence the version of the algorithm presented above, which allows re-using links from previously computed paths, but with a great penalty, greater than any path length in the network. For the same reason the number of alternate paths was restricted to three.

In the subsequent investigations six variants of fixed-alternate routing were used. Three of them did not provide differentiation, as they were using 1, 2, and 3 paths for each class of requests. In the remaining three routing variants, combinations of (1,2), (1,3) and (2,3) were used, where the numbers in brackets reflect the number of alternate paths for low and high priority requests, respectively.

3.4 Mechanisms that use preemption

The last set of mechanisms is based on lightpath preemption, where the network may tear down some existing low priority lightpaths if it cannot find sufficient resources for a high priority request.

Introduction of this set of mechanisms into the network is not straightforward as it violates the implicit assumption that the lightpath should not be disconnected by the network, once it is set up. Thus, these mechanisms were omitted from the studies presented in the dissertation.

In judging whether this kind of mechanisms may be introduced in the network one should think of three interrelated issues. First, the kind of applications that will use the low priority lightpaths. Second, the probability of the lightpath being disconnected. Third, the reason for disconnection.

It is possible to think about some applications that may tolerate lightpath disconnections, especially if they are designed to do so. Such an application cannot be a real-time application, and when interrupted it should be able to reconnect a lightpath and continue from the interruption point. Examples of such applications might be synchronization of large content databases, or a large backup application. The latter one is usually not sensitive to delays as long as the whole backup process fits into the target time window. A different kind of application that might use low priority lightpaths is a packet network in which a low priority lightpath is used to increase the available bandwidth between two packet routers. Disconnection of such a lightpath will cause some traffic loss, will cause buffer overflows in routers, but for some networks and services this might be tolerable.

The two latter issues, the probability of disconnection and the reason for disconnection are related with the traffic profile of a high priority class. With a high priority traffic occupying only a tiny fraction of the total network throughput, the probability of a low priority lightpath being disconnected is quite small. If the high priority class is required for emergency communications and life-saving functions, such as telemedicine, which require extremely high robustness and availability regardless of traffic conditions in the network, the reasoning behind using preemption starts to be sound.

Although the mechanisms based on preemption were not considered in this dissertation, they are an interesting subject. Thus, this is one of the possible directions of further studies in the field of grade of service in the optical networks.

3.5 Strategies

Based on the presented mechanisms several strategies were defined and investigated. Some of them employ just one mechanism, the rest is a mix of two mechanisms. Single mechanism strategies based on resource preservation employ one of the mechanisms that preserve resources and use fixed-alternate routing with equal number of paths (1, 2 or 3) for each class of requests. The alternate paths are searched for using the algorithm presented in Section 3.3. Single mechanism strategies based on fixed-alternate routing are based solely on a different number of paths in each class.

The mixed strategies differentiate lightpath requests using two mechanisms simultaneously: a mechanism that preserves resources and fixed-alternate routing with a different number of paths for each class of requests. The following combinations of the number of alternate paths for low and high priority requests were used: (1,2), (1,3) and (2,3).

The following notation is used to describe strategies: *altXY-Z*, where X is the number of paths for a low priority request, Y is the number of paths for a high priority request, Z is the abbreviated name of the preservation mechanism. For example *alt13-pcap* is the strategy based on path capacity threshold mechanism with one path for low priority requests and 3 paths for high priority requests. Additionally, the notation *altXY* is used for single mechanism strategies based on fixed-alternate routing.

4 Performance assessment

To assess the performance of the presented mechanisms and strategies a set of steady-state discrete-event computer simulations has been performed. The aim of the performed simulations is to obtain estimates of the mean values of some parameters, of which the most important ones are the blocking probabilities of low and high priority lightpath requests.

There are various issues connected with conducting such an experiment. These issues range from statistical questions, whether the obtained results are credible and, thus, may be used for further reasoning, to technical questions on the simulator architecture and other design choices. These problems will be dealt with in the subsequent sections.

4.1 Statistical foundations

It is generally preferred to use estimators with confidence intervals since they provide some measure of reliability. The computed confidence intervals contain the unknown value of the population mean μ with a given probability $(1-\alpha)$, termed the confidence coefficient. If this value is expressed as a percentage, another term, confidence level, is used. Typically, statistical calculations are performed at the 90%, 95% or 99% confidence level. In this study, the 95% confidence level is assumed.

For a sample x that contains N individual and independent observations $x_n, n = 1 \dots N$, the sample mean $\bar{x}$ is the unbiased point estimator of the population mean μ

$$\hat{\mu} = \bar{x} = \frac{1}{N}\sum_{n=1}^{N} x_n \tag{4.1}$$

If the sampled population is approximately normally distributed, the distribution of the sample mean is the t-Student distribution with $N-1$ degrees of freedom. Thus, the confidence interval of the population mean μ may be estimated using the following formula:

$$\mu \in \left[\bar{x} \pm t_{\alpha/2}\left(s/\sqrt{N}\right)\right] \tag{4.2}$$

where $t_{\alpha/2}$ is the $(1-\alpha/2)$ quantile of the t-Student distribution with $N-1$ degrees of freedom and s is the sample standard deviation:

$$s = \sqrt{\frac{1}{N-1}\sum_{n=1}^{N}(x_n - \bar{x})^2} \tag{4.3}$$

Although the calculations depend on the fact that the sampled population is approximately normal, the computed confidence interval is rather insensitive to moderate departures from normality [43]. However, for a large sample size, according to the Central Limit Theorem, the distribution of the sample mean is approximately normal, regardless of the distribution of the sampled population [43]. Thus, the confidence interval may be specified in the following way:

$$\mu \in \left[\bar{x} \pm z_{\alpha/2}\left(\sigma/\sqrt{N}\right)\right] \tag{4.4}$$

where $z_{\alpha/2}$ is the $(1-\alpha/2)$ quantile of the normal distribution. The unknown parameter σ, being the standard deviation of the population, may be substituted with the sample standard deviation s. As a result:

$$\mu \in \left[\hat{\mu} \pm z_{\alpha/2}\left(s/\sqrt{N}\right)\right] \tag{4.5}$$

Usually, a sample is considered large for $N \geq 30$ [43]. In the performed simulations the sample size was chosen to be 31, which satisfies this rule.

4.2 Simulation type and data collection methods

The calculations presented in the previous section require each sample to contain independent observations. A set of independent observations may be gathered running a simulation several times, each time using differently seeded pseudo-random number generators. This simulation technique is known as the independent replications method [67]. However, each simulation must reach steady state before data gathering is started. Thus, a great deal of computational effort corresponding to the transient phase of the simulation is wasted at the beginning of each run.

To avoid this problem, another method was used. In the method of nonoverlapping batch means [48], [67], each case is simulated in a single, long run. The run is divided into a transient phase and a steady-state phase. The latter one is subsequently divided into a series of nonoverlapping intervals, called batches, hence the name of the method. Data obtained in each batch are treated as if they were independent and are used for further statistical processing. Consequently, for each simulated case there is only one warm-up period, which increases the efficiency of the simulation. Furthermore, any residual effects of a transient state become less pronounced as the simulation length increases.

4.2.1 Estimation of the transient phase length

Usually it is assumed that data obtained while the network is in transient, nonstationary phase, should be discarded. A quick and very rough method of estimating the length of the transient period is the observation of system parameters. When they start to stabilise, the steady state is assumed. In the presented simulations the blocking probability of the lightpath requests starts oscillating after several units of time. Thus, a cautious assumption of the simulation warm-up period of 1200 time units has been chosen for most simulations.

The length of the transient period has been verified using a method derived from the Central Limit Theorem and described in [48] and [67]. The statistical foundation for this method is that if a distribution of a given parameter is stationary, and its true mean and standard deviation are equal to μ and σ, then its sample mean follows normal distribution with mean μ and standard deviation $\sigma_{\bar{x}} = \sigma/\sqrt{M}$, where M is the number of samples in x. Therefore, a plot of $log(\sigma_{\bar{x}})$ against $log(M)$ should follow a straight line with a slope of about -0.5.

To perform the test, the assumed warm-up period has been divided into M nonoverlapping subperiods. However, to perform this study the samples gathered from the subperiods must be independent. To achieve independent samples, a large number N of replications of the tested warm-up period has been performed with differently seeded pseudo-random number generators. The observations x_{mn} for $m = 1, \ldots, M$ and $n = 1, \ldots, N$ are then converted into the sequence of sample means:

$$\bar{x}_m = \sum_{n=1}^{N} x_{mn} \tag{4.6}$$

As this sequence contains M independent normally distributed random variables, the standard deviation may be computed as:

$$\sigma_{\bar{x}} = \sqrt{\frac{1}{M-1} \sum_{m=1}^{M} (\bar{x}_m - \bar{x})^2} \tag{4.7}$$

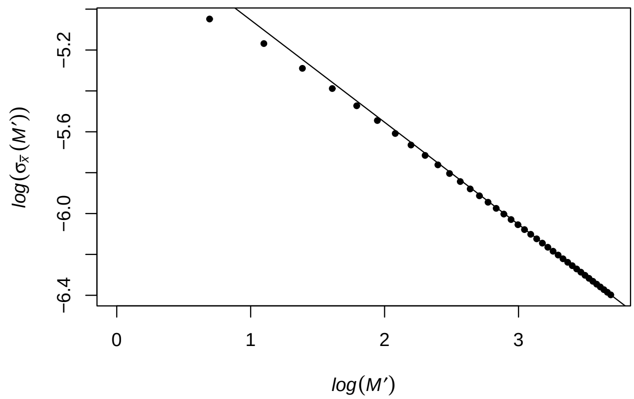

Figure 4.1: A sample plot of $log(\sigma_{\bar{x}}(M'))$ against $log(M')$ obtained for the blocking probability in the low priority class, PanEU network, pcap mechanism, $T = 2$, fixed-alternate routing with 3 paths for each class of requests. The simulation lasted 1200 time units.

where

$$\bar{x} = \frac{1}{M} \sum_{m=1}^{M} \bar{x}_m \tag{4.8}$$

Taking into consideration the first M' out of M samples and plotting the relationship $log(\sigma_{\bar{x}}(M'))$ against $log(M')$ for $M' = 1, \ldots, M$ the desired plot is obtained. The example is shown in Fig. 4.1. For convenience a straight line with a slope of -0.5 has been added to the plot.

4.2.2 Validation of the batch length

The steady-state phase is divided into the set of nonoverlapping intervals (batches) of equal size and batch means are analysed as if they were independent. However, the obtained observations are not independent in general. This must be taken into account by choosing batch sizes sufficiently large to avoid correlation between batches, as the correlation between individual samples usually diminish with the increased distance between them [67].

The Ljung-Box test [38] was applied to get an insight on the autocorrelation of the batch means of the blocking probabilities in the low and high priority classes. This is quite a strong test which tests the overall randomness based on a number of lags. For that reason this test is sometimes called a “portmanteau” test. The test is defined as follows:

Let H_0 be the hypothesis, that the data are random, and H_a hypothesis that the data are not random. The test statistic is then:

$$Q_{LB} = n(n+2) \sum_{j=1}^{h} \frac{\hat{\rho}^2(j)}{n-j} \tag{4.9}$$

where n is the sample size, $\hat{\rho}(j)$ is the sample autocorrelation at lag j, and h is the number of lags tested.

The hypothesis of randomness is rejected at the significance level α when $Q_{LB} > \chi^2_{1-\alpha,h}$ where the $\chi^2_{1-\alpha,h}$ is the α quantile of the chi-square distribution with h levels of freedom.

There are several cited rules for choosing the correct number of lags h. In the performed analysis $h = 8$, which is roughly 1/4 of the total number of samples, as recommended in [9].

For most simulations the batch length was chosen to be 1200 time units. On several occasions the results obtained from a simulation did not pass the Ljung-Box test. In those cases the simulation was repeated with an increased batch length, and unchanged number of batches. For technical reasons the length of the warm-up period was also increased to match the batch length. This process was repeated until there was no reason to reject the hypothesis that the data are random. This resulted in batch sizes from 2400 to 24000 time units.

4.2.3 Data collection

Four parameters were collected during each simulation: the blocking probability in the low priority class, the blocking probability in the high priority class, the total blocking probability and the network utilisation factor.

The blocking probability in the low priority class is computed within each simulation subperiod (batch) as the number of the blocked low priority lightpath requests divided by the number of the offered low priority lightpath requests. The blocking probability in the high priority class, and the total blocking probability are computed in the analogous way.

The network utilisation factor is more difficult to obtain. It represents the mean utilisation of an average link in the network. In the network of l links with w wavelengths on each link there are $2lw$ spans to utilise. A bidirectional lightpath routed through l' links utilises $2l'$ spans.

Fig. 4.2 shows the sample plot of the network utilisation in time. The average network utilisation may be computed as the shaded area on the figure divided by the batch length in time units and divided by the total number of of spans. Three variables are needed to do that: `last_load_change` holding the time the network

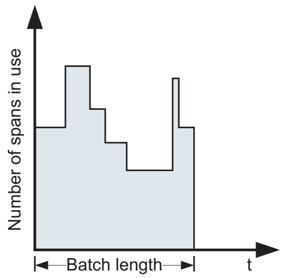

Figure 4.2: Sample network utilisation in time

load was last changed due to lightpath setup or teardown, `last_load` holding the current number of spans in use, and `acc_load` holding the value corresponding with the shaded area on the figure. At the beginning of the simulation all variables are set to zero. At the beginning of each batch the `last_load_change` is set to the current time, and `acc_load` is set to zero. On each successful lightpath setup, teardown, or at the end of the batch, the value of the `acc_load` is increased with:

$$(\texttt{current\_time} - \texttt{last\_load\_change}) \times \texttt{last\_load}$$

Then, `last_load` is increased (or decreased) by the number of spans used by the currently set-up (or torn down) lightpath, $2l'$, and the `last_load_change` is set to current time.

At the end of the batch the `acc_load` variable holds the value corresponding to the shaded area in Fig. 4.2. The value of the `acc_load` is divided by the batch length to obtain the average number of spans in use. This number is subsequently divided by the total number of spans, $2lw$, to obtain the average network utilisation.

4.3 Simulation details

4.3.1 Control plane

The algorithms in the previous section are presented in two versions: one for networks with centralised control and one for networks with distributed control. These are the two architectures implementing the Path Computation Element (PCE) [18]:

1. "centralised computation model" refers to a model whereby all paths in a domain are computed by a single, centralised PCE,

2. “distributed computation model” refers to the computation of paths in a domain being shared among multiple PCEs.

The centralised computation model with a stateful PCE [18] has been chosen for the simulation studies. In this model all lightpath requests are served on a first-in-first-out base by a single entity, which has a perfect knowledge about the network state. The motivation for such a decision was the complexity of the simulator, which is far lower in the centralised case, and the fact that in the distributed network, or with stateless PCE, there are parameters, which affect the network performance, but are quite hard to estimate. An example might be the service time of the signalling messages in each node.

It is also worth noting that the authors of [56] performed the comparison of performance between the centralised and the distributed model. With the increasing ratio of the lightpath interarrival time to the lightpath setup time (service time) the discrepancies between the performance of the centralised vs. distributed lightpath setup diminish. When the ratio approaches 100 the differences are negligible. In the simulated Pan-EU network the lightpath interarrival times of the order of $16 - 30$ units of time were used. Assuming the unit of time is 1 s, this would require service times not greater than 160 ms.

Additionally, Shen and Ramamurthy [56] note, that in networks with centralised control the lightpath setup time has a negligible impact on the network performance. Thus, the lightpath setup time was assumed to be zero.

4.3.2 Tools

The simulation environment was built using the OMNeT++ simulator version 2.3 [68]. The OMNeT++ simulator provided a generic framework for a discrete event simulation system, which was augmented by the author with the code necessary to simulate the selected functionality of the optical network control plane. Most of the code was written in C++, the remaining part was written in the internal OMNeT++ language, the NED language, used for defining the simulation modules and their interconnections. The statistical processing of the gathered data was done in the R statistical package [71]. Additionally, the processed data were exported into the PostgreSQL database [70] and processed using a set of SQL queries. The latter step greatly simplified the process of finding the top N algorithms according to given criteria.

4.3.3 Simulator architecture

The simulator consists of two blocks, the generator and the controller, connected in a bidirectional way, see Fig. 4.3. The controller module is responsible for processing the lightpath requests, the generator module is responsible for generating

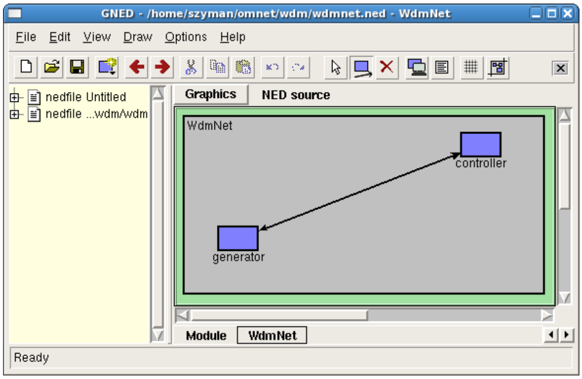

Figure 4.3: The internal simulator topology

the lightpath requests and for tearing down the existing lightpaths according to the assumed traffic pattern. The generator alone acts on behalf of all the nodes in the simulated network.

4.3.4 Lightpath request generation

The generator block uses four independent pseudo-random number generators for each simulated node. Two of them are responsible for generating the interarrival times between lightpath requests in the low and high priority class, the remaining two are responsible for generating the holding times of the requested lightpaths. The default pseudo-random number generators in version 2.3 of OMNeT++, which are linear congruential generators with a cycle length of $2^{31} - 2$, have been replaced with Mersenne-Twister generators [42] with a period length of $2^{19937} - 1$ and good randomness in up to 623 dimensions. This is a well-known generator suggested in [47]. An additional Mersenne-Twister generator with a different period length ($2^{11213} - 1$) was used to seed all other generators before the first use. The author did not implement the pseudo-random generator code on his own. Instead, a specialised library, the Boost Random library [69] was used. The library is equipped with an internal validation suite which was used to verify the correctness of the implementation.

4.3.5 Routing and wavelength assignment

The controller module is responsible for processing the lightpath requests and gathering statistical data. The physical network topology is assumed to be constant during the simulation. Thus, at the beginning of a simulation the controller module computes and stores the alternate paths for each source-destination pair to speed up the simulation. Currently, three alternate paths are calculated and stored. The calculations are done using the Dijkstra algorithm implementation from the Boost Graph library [69]. The set of alternate paths is computed according to the rules given in 3.3.

Wavelength assignment is done using the first-fit algorithm [14] (the algorithm was originally termed PACK, the name first-fit is currently used throughout the literature, [74]). In this algorithm the lowest-index wavelength that is free on a given path is used to satisfy the lightpath request. Although better algorithms exist [59], the first-fit algorithm is commonly found in the literature and its performance in networks without wavelength conversion is better than the other commonly used algorithm, namely the random wavelength assignment algorithm [59]. Additionally, it requires minimal amount of information about the network state and has a low computational complexity.

4.3.6 Lightpath setup and teardown

Upon request, the controller performs the lightpath setup using a given GoS strategy, a set of precomputed alternate paths and a wavelength assignment algorithm. The simulator may work with either unidirectional lightpaths or bidirectional lightpaths. In the presented studies only bidirectional lightpaths were used. Bidirectional lightpaths are set up using the same wavelength in both directions to avoid resource fragmentation.

Although searching for a free wavelength is done only in one direction, the simulator checks whether the wavelength in the other direction is free. If not, an error is reported since with bidirectional lightpaths this situation should never happen. This step was implemented to catch hidden errors in the simulator code.

On success, the resources used by the computed lightpath are marked as busy and a confirmation is sent to the generator module. On failure, no resources are reserved and the generator module is notified about the lightpath setup failure.

When lightpath teardown is requested by the generator module, the controller releases the resources used by the affected lightpath. However, before marking resources as free the controller checks whether they are marked as busy. This mechanism is also implemented to catch potential mistakes in the code, since freeing resources that are already free would be a sign of a serious flaw in the simulation code. After releasing resources the controller informs the generator module about the disconnected lightpath.

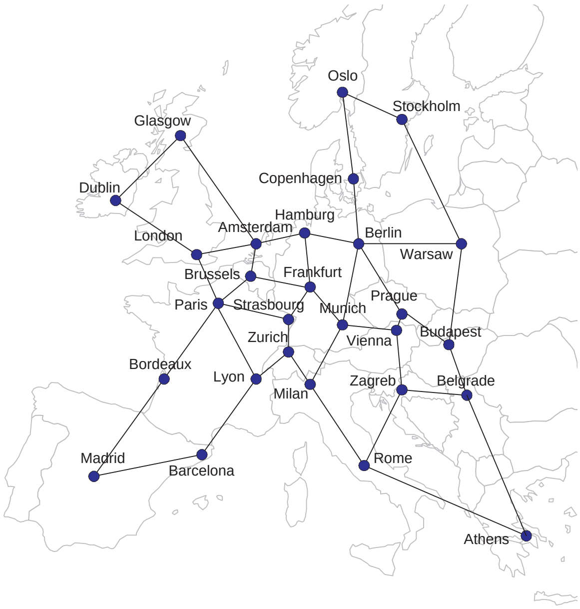

Figure 4.4: Reference network topology used in simulations: PanEU network.

4.4 Network architectures and parameters

To assess the behaviour of the presented strategies two networks were used: the 28 nodes Pan-European reference network (see [33]) shown in Fig. 4.4 and the 14 nodes NSF network (see [54]) shown in Fig. 4.5. In both networks each link consists of two fibres, one in each direction, each fibre is assumed to carry 80 wavelengths and the network employs no wavelength conversion.

The offered traffic is generated based on a uniform traffic matrix. In the base case each node pair generates bidirectional lightpath requests with mean intensity of 0.033 lightpath per unit time for the Pan-EU network and 0.165 lightpath per unit time for the NSF network. The interarrival time was exponentially distributed. A given fraction (10 − 50%) of the offered traffic belongs to the high priority class, while the remaining traffic belongs to the low priority class. The lightpath holding time is also exponentially distributed with the mean of 10 units time, independent of the class. As a result, the total network load in the base case is $28 \times 27 \times 0.033 \times 10 = 249.48\,\text{Erl}$ for the Pan-EU network and $14 \times 13 \times 0.165 \times 10 = 300.3\,\text{Erl}$ for the NSF network. To observe network behaviour under different loads, this base case traffic is scaled by the factor

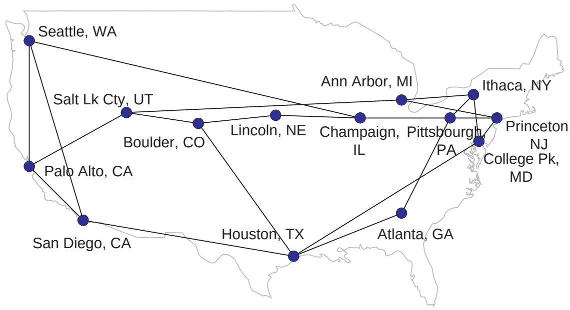

Figure 4.5: Reference network topology used in simulations: NSF network.

ranging from 1.0 to 1.8. The scaling factor is used in the subsequent text and figures as the offered load intensity measure.

The simulations have been performed for increasing values of T, starting from 0, when both traffic classes are treated equivalently, until the blocking probability of the high priority requests dropped below 0.005. The value of 0.005 is arbitrarily chosen as the optimisation target for choosing the right value of T. Any reasonable value may be assumed here, as required by the network operator policy and requirements. The decision to perform a linear search for an optimal value of parameter T, instead of using more efficient binary search [1] was mainly motivated by the fact that the author wanted to gather detailed data for plots showing the relationship between values of parameter T and the achieved network performance.

5 Evaluation of the single mechanism strategies

This chapter is the first one dealing with the performance of the strategies proposed in Chapter 3. The evaluated strategies are a subset of strategies proposed in Section 3.5, namely the strategies based on a single Grade-of-Service routing mechanism only. The evaluation details, including the simulated network topologis and the tools used for simulation were described in Chapter 4.

5.1 A strategy based on the wavelength pools mechanism (pool)

In this mechanism the whole set of wavelengths is divided into two disjoint pools: a common pool which is used by all classes of requests and the preserved pool. The preserved pool contains T wavelengths, may be used solely by the high priority requests and is used only when the common pool is exhausted on a given path. To evaluate this mechanism two strategies were built, the alt11-pool based on shortest path routing and the alt33-pool based on fixed-alternate routing with three paths for each class of requests.

Figs. 5.1 and 5.2 show the blocking probability in the low and high priority classes as a function of parameter T in the PanEU network using the alt11-pool strategy. The results are presented for three values of the offered load when the fraction of the high priority traffic is equal to 50%. The figures show that the pool mechanism is able to achieve statistically significant differences in the blocking probability of low and high priority requests. The amount of differentiation may be regulated by choosing the value of parameter T. For $T = 0$ there is no differentiation, for $T \geq 40$ the blocking probability of the high priority requests drops below 0.005 for load 1.8, at the expense of the increased blocking probability in the low priority class, compared to the case without differentiation ($T = 0$).

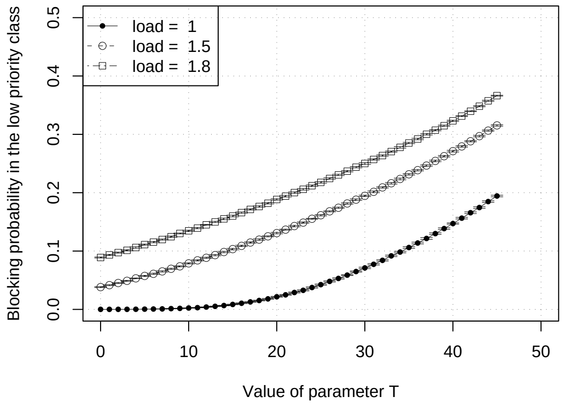

Figure 5.1: A strategy based on the wavelength pools mechanism (pool) and shortest path routing, PanEU network, 50% of traffic is the high priority traffic. The blocking probability in the low priority class as a function of parameter T is presented for three values of the offered load.[2]

Figs. 5.3 and 5.4 show the corresponding data for the PanEU network using the alt33-pool strategy. The obtained results are quite similar and the only difference is that fixed-alternate routing, being better algorithm than shortest path routing, results in lower blocking probabilities in both traffic classes. As a result it is sufficient to use lower values of T to achieve the requested blocking probability in the high priority class. In the considered scenario the blocking probability of high priority requests lower than 0.005 was achieved with $T = 26$. Lower values of T result in more resources left for the low priority traffic, and, as a consequence, a lower blocking probability of the low priority requests.

Figs. 5.5 and 5.6 show three-dimensional plots of the blocking probability in the high priority class as a function of the offered load and the fraction of the high priority traffic. The first figure was obtained with shortest path routing, the second one with fixed-alternate routing and three paths for each traffic class. The results are similar for both cases. The blocking probability raises with increasing the overall offered traffic and with increasing fraction of the high priority traffic in the network, which is quite an obvious result.

[2]Please note that the confidence intervals are marked on all figures, but their length is in most cases comparable to the line width.

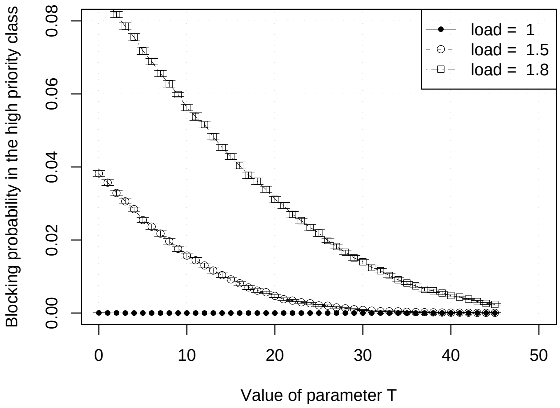

Figure 5.2: A strategy based on the wavelength pools mechanism (pool) and shortest path routing, PanEU network, 50% of traffic is the high priority traffic. The blocking probability in the high priority class as a function of parameter T is presented for three values of the offered load.

However, the corresponding plots for the low priority class, shown in Figs. 5.7 and 5.8 reveal an interesting difference between the behaviour of the evaluated mechanism with shortest path routing and fixed-alternate routing. For shortest path routing the blocking probability of the low priority requests raises gradually with increasing load but is independent of the fraction of the high priority traffic. This can be easily seen in Figs. 5.9 and 5.10. For fixed-alternate routing the blocking probability of the low priority requests also raises gradually with increasing load but decreases with increasing amount of the high priority traffic in the network (see Figs. 5.11 and 5.12). This phenomenon may be attributed to the specific order of operation in the analysed strategy.

The division of the set of available wavelengths into two disjoint pools creates, in fact, two independent networks. In the shortest path routing case, all requests, regardless of their class, are served using the first network, corresponding with the common pool. If there are insufficient resources in the common pool then the low priority requests are blocked, and the high priority requests create an overflow traffic directed to the second network. This traffic is never directed back to the first network. Thus, changes in the fraction of the high priority traffic do not affect the blocking probability in the first network. In a general case, the blocking probability in the common pool network needs not necessarily

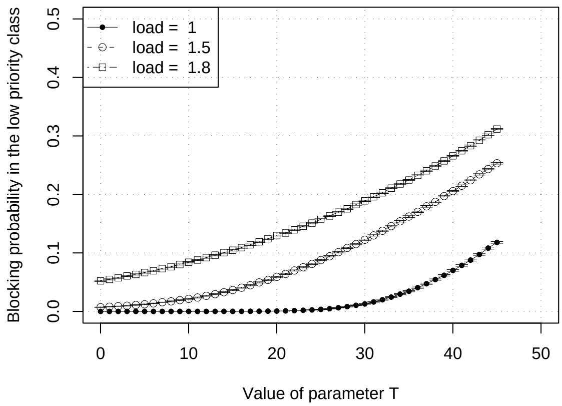

Figure 5.3: A strategy based on the wavelength pools mechanism (pool) and fixed-alternate routing with 3 paths for each class, PanEU network, 50% of traffic is the high priority traffic. The blocking probability in the low priority class as a function of parameter T is presented for three values of the offered load.

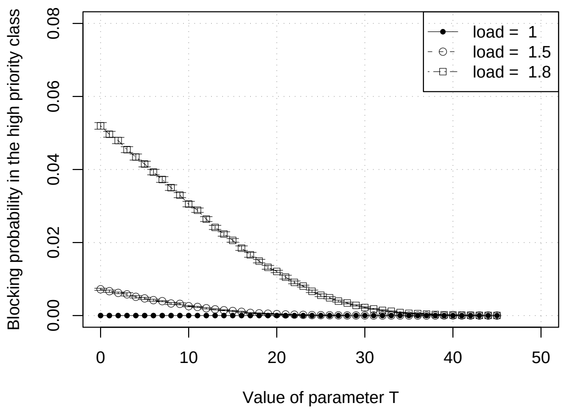

Figure 5.4: A strategy based on the wavelength pools mechanism (pool) and fixed-alternate routing with 3 paths for each class, PanEU network, 50% of traffic is the high priority traffic. The blocking probability in the high priority class as a function of parameter T is presented for three values of the offered load.

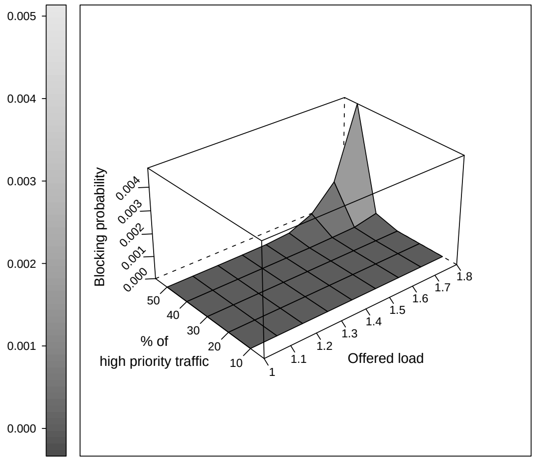

Figure 5.5: A strategy based on the wavelength pools mechanism (pool) and shortest path routing, PanEU network, $T = 40$. The blocking probability in the high priority class as a function of the offered load and the fraction of the high priority traffic.

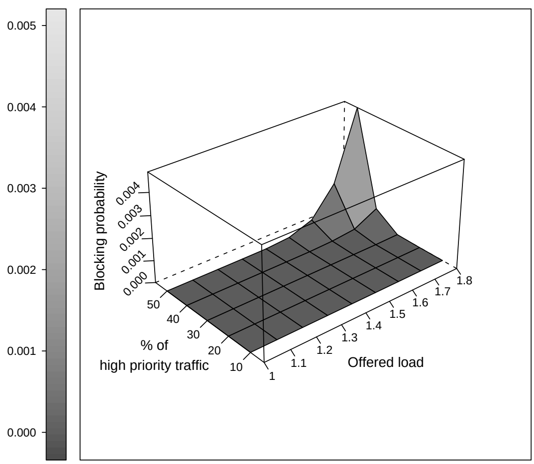

Figure 5.6: A strategy based on the wavelength pools mechanism (pool) and fixed-alternate routing with 3 paths for each class, PanEU network, $T = 26$. The blocking probability in the high priority class is presented as a function of the offered load and the fraction of the high priority traffic.

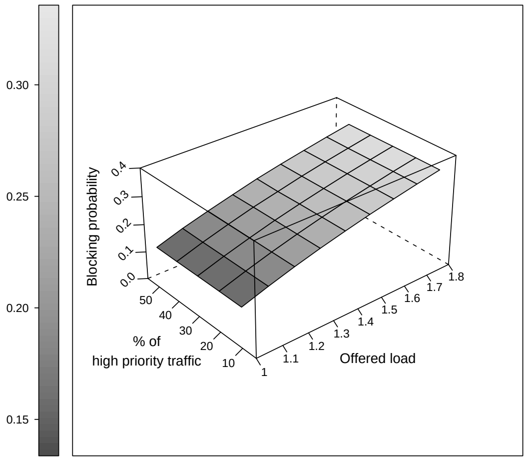

Figure 5.7: A strategy based on the wavelength pools mechanism (pool) and shortest path routing, PanEU network, $T = 40$. The blocking probability in the low priority class is presented as a function of the offered load and the fraction of the high priority traffic.

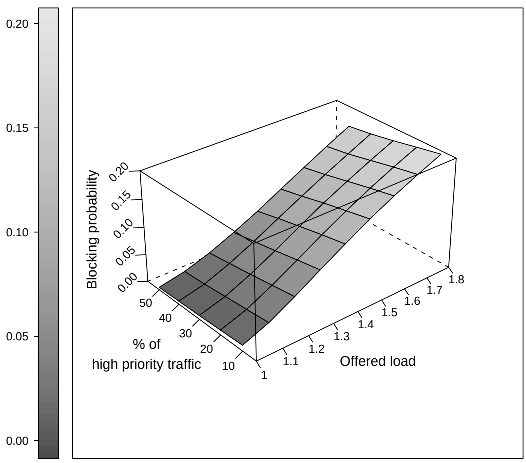

Figure 5.8: A strategy based on the wavelength pools mechanism (pool) and fixed-alternate routing with 3 paths for each class, PanEU network, $T = 26$. The blocking probability in the low priority class is presented as a function of the offered load and the fraction of the high priority traffic.

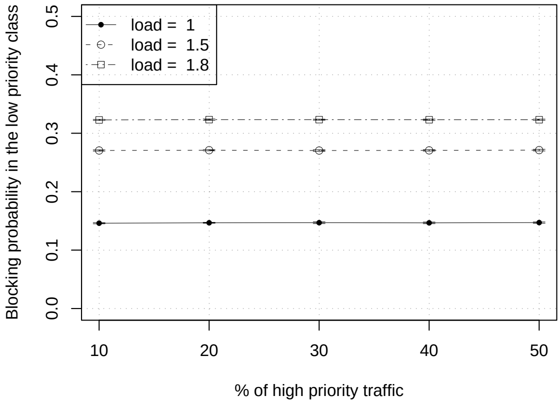

Figure 5.9: A strategy based on the wavelength pools mechanism (pool) and shortest path routing, PanEU network, $T = 40$. The blocking probability in the low priority class is presented as a function of the fraction of the high priority traffic, for three values of the offered load.

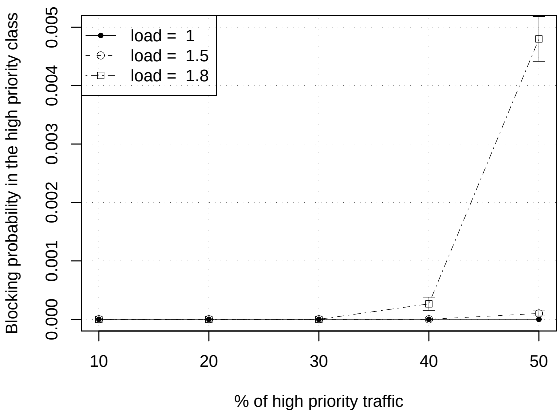

Figure 5.10: A strategy based on the wavelength pools mechanism (pool) and shortest path routing, PanEU network, $T = 40$. The blocking probability in the high priority class is presented as a function of the fraction of the high priority traffic, for three values of the offered load.

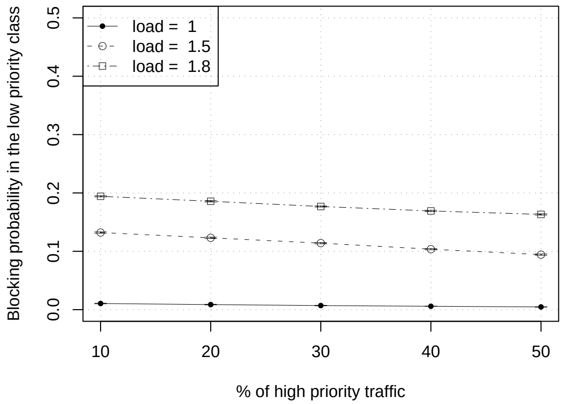

Figure 5.11: A strategy based on the wavelength pools mechanism (pool) and fixed-alternate routing with 3 paths for each class, PanEU network, $T = 26$. The blocking probability in the low priority class is presented as a function of the fraction of the high priority traffic, for three values of the offered load.

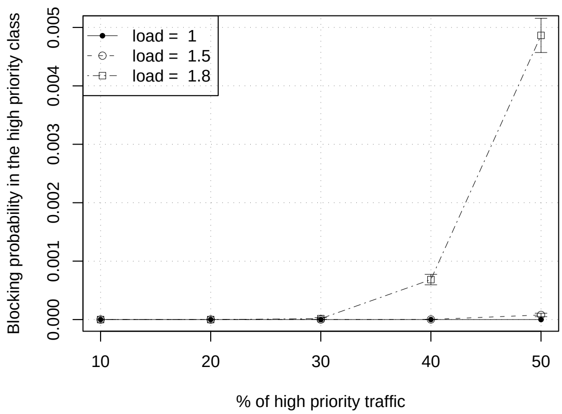

Figure 5.12: A strategy based on the wavelength pools mechanism (pool) and fixed-alternate routing with 3 paths for each class, PanEU network, $T = 26$. The blocking probability in the high priority class is presented as a function of the fraction of the high priority traffic, for three values of the offered load.

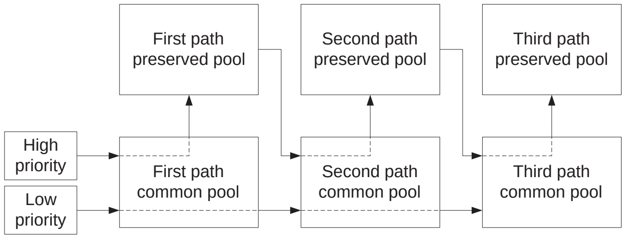

Figure 5.13: Order of operation with the pool mechanism when fixed-alternate routing is used.

be the same as blocking probability of the low priority requests. However, in the presented scenario the low priority requests may be regarded as a random sample of all the requests directed to the common pool network and thus their blocking probability is an estimate of the blocking probability in the common pool network, hence the observed phenomenon.

The situation is quite different with fixed-alternate routing. The low priority requests are served using the common network, but the high priority requests are tried in an alternating way on the common pool and the preserved pool (see Fig. 5.13). Thus, for constant overall traffic, the greater fraction of the high priority traffic, the lesser the traffic overflowing to the common pool on the second and next paths, hence lower blocking probability in the low priority class.

The presented results were obtained in the PanEU network. The corresponding results obtained in the NSF network are quite similar and are shown in Appendix A in Figs. A.1 – A.12.

5.2 A strategy based on the first link capacity threshold mechanism (flcap)

In this mechanism a candidate path for a low priority request is assessed according to the number of free wavelengths on the first link on the path. If the number of free wavelengths on the first link is greater than the specified threshold T and there is at least one continuous free wavelength on the path then the low priority request is admitted, else it is rejected. The high priority requests are always accepted, assuming the availability of a free wavelength on the candidate path. Two strategies: alt11-flcap and alt33-flcap were built to assess this mechanism. The first one is based on the shortest path routing, the second one on the fixed-alternate routing with three paths for each class.

Figs. 5.14 and 5.15 show the blocking probability in the low and high priority classes as a function of parameter T in the PanEU network using shortest path routing. The results are presented for three values of the offered load and the fraction of the high priority traffic is equal to 50%. The flcap strategy is able to achieve statistically significant differences in the blocking probability of low and high priority requests. The amount of differentiation may be regulated by choosing the value of parameter T. For $T = 0$ there is no differentiation, for $T \geq 53$ the blocking probability of the high priority requests drops below 0.005 for load 1.8. Increasing the value of T has a negative effect on the blocking probability of the low priority requests.

Figs. 5.16 and 5.17 show the corresponding data for fixed-alternate routing with three paths for each class. The obtained results are quite similar. Using fixed-alternate routing results in lower blocking probabilities in both traffic classes. The necessary value of parameter T to achieve the blocking probability of the high priority requests lower than 0.005 is reduced to $T = 48$.

Figs. 5.18 and 5.19 show the three-dimensional plots of the blocking probability in the high priority class as a function of the offered load and the fraction of the high priority traffic. The first figure was obtained with shortest path routing and the second one with fixed-alternate routing and three paths for each traffic class. The results are similar for both cases. The blocking probability raises with increasing overall offered traffic and with increasing fraction of the high priority traffic in the network. The corresponding plots for the low priority class are shown in Figs. 5.20 and 5.21.

The presented results were obtained in the PanEU network. The corresponding results obtained in the NSF network are quite similar and are shown in Appendix A in Figs. A.13 – A.20.

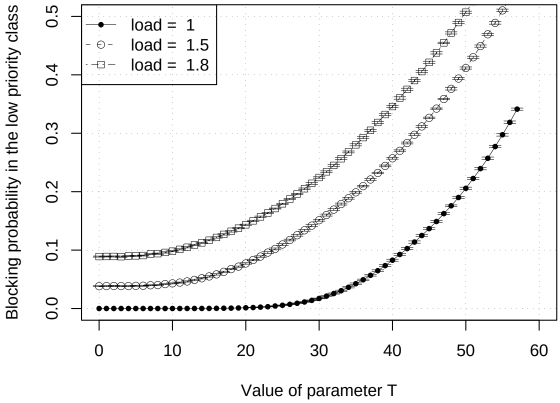

Figure 5.14: A strategy based on the first link capacity threshold mechanism (flcap) and shortest path routing, PanEU network, 50% of traffic is the high priority traffic. The blocking probability in the low priority class is presented as a function of parameter T, for three values of the offered load.

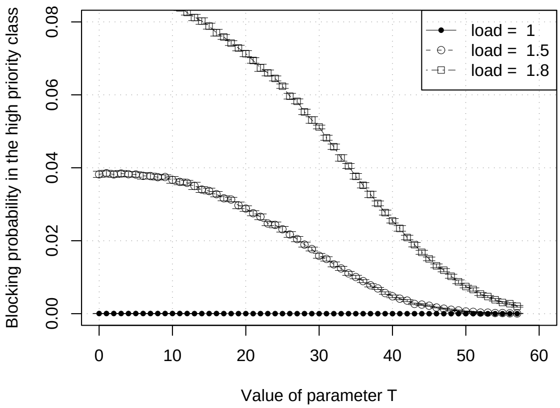

Figure 5.15: A strategy based on the first link capacity threshold mechanism (flcap) and shortest path routing, PanEU network, 50% of traffic is the high priority traffic. The blocking probability in the high priority class is presented as a function of parameter T, for three values of the offered load.

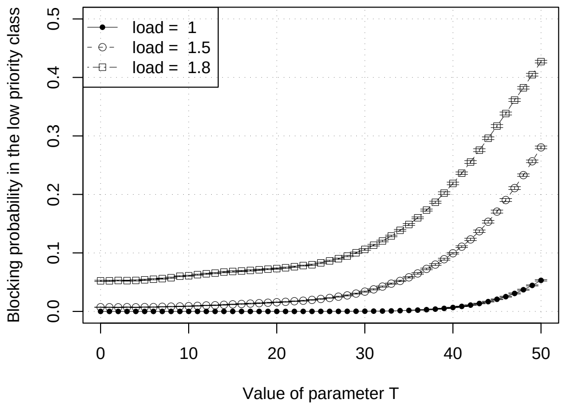

Figure 5.16: A strategy based on the first link capacity threshold mechanism (flcap) and fixed-alternate routing with three paths for each class, PanEU network, 50% of traffic is the high priority traffic. The blocking probability in the low priority class is presented as a function of parameter T, for three values of the offered load.

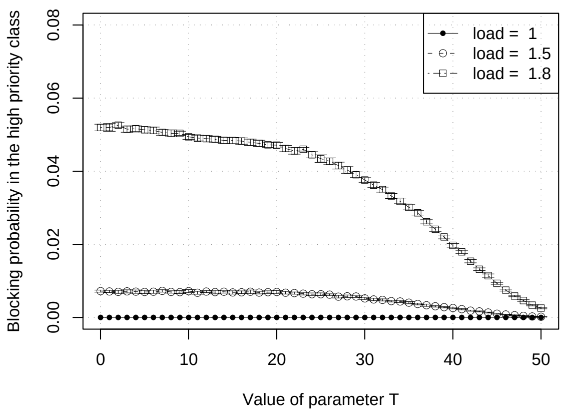

Figure 5.17: A strategy based on the first link capacity threshold mechanism (flcap) and fixed-alternate routing with three paths for each class, PanEU network, 50% of traffic is the high priority traffic. The blocking probability in the high priority class is presented as a function of parameter T, for three values of the offered load.

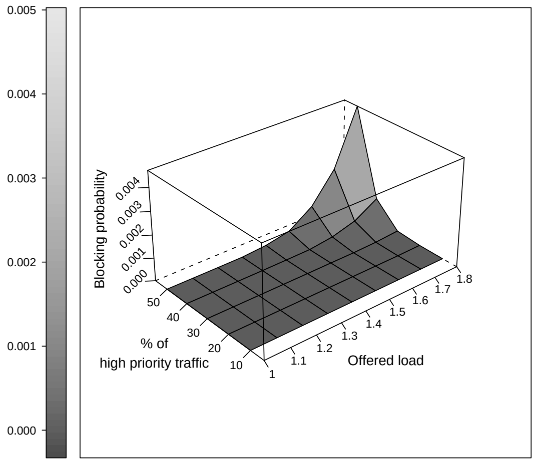

Figure 5.18: A strategy based on the first link capacity threshold mechanism (flcap) and shortest path routing, PanEU network, $T = 53$. The blocking probability in the high priority class is presented as a function of the offered load and the fraction of the high priority traffic.

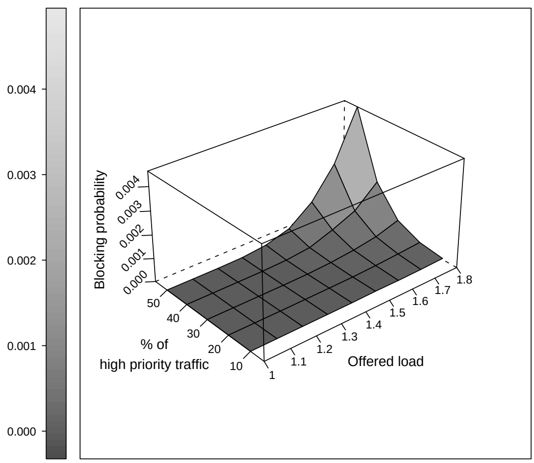

Figure 5.19: A strategy based on the first link capacity threshold mechanism (flcap) and fixed-alternate routing with three paths for each class, PanEU network, $T = 48$. The blocking probability in the high priority class is presented as a function of the offered load and the fraction of the high priority traffic.

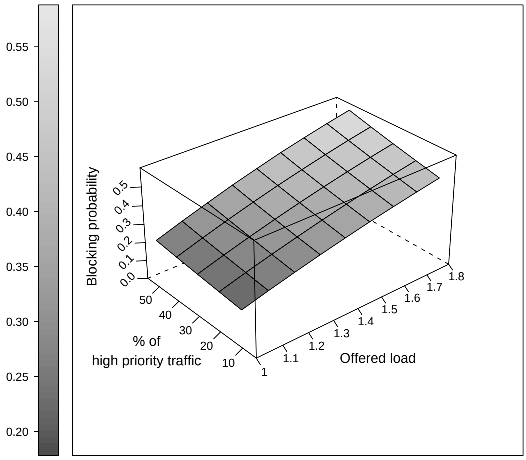

Figure 5.20: A strategy based on the first link capacity threshold mechanism (flcap) and shortest path routing, PanEU network, $T = 53$. The blocking probability in the low priority class is presented as a function of the offered load and the fraction of the high priority traffic.

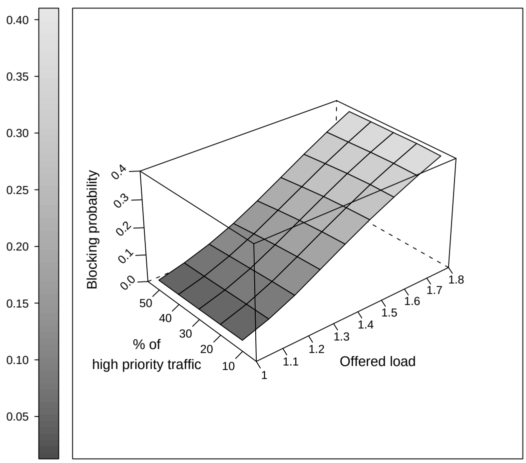

Figure 5.21: A strategy based on the first link capacity threshold mechanism (flcap) and fixed-alternate routing with three paths for each class, PanEU network, $T = 48$. The blocking probability in the low priority class is presented as a function of the offered load and the fraction of the high priority traffic.

5.3 A strategy based on the link capacity threshold mechanism (lcap)

In this mechanism a candidate path for a low priority request is assessed according to the number of free wavelengths on each link on the path. If the number of wavelengths on each link is greater than the specified threshold, T, and there is at least one continuous free wavelength on the path then the low priority request is admitted, else it is rejected. The high priority requests are always accepted, assuming the availability of a free wavelength on the candidate path. Two strategies: alt11-lcap and alt33-lcap were built to assess this mechanism. The first one is based on the shortest path routing, the second one on the fixed-alternate routing with three paths for each class.

Figs. 5.22 and 5.23 show the blocking probability in the low and high priority classes as a function of parameter T in the PanEU network using shortest path routing. The results are presented for three values of the offered load when the fraction of the high priority traffic is equal to 50%. The figure shows that the lcap strategy is able to achieve statistically significant differences in the blocking probability of low and high priority requests. The amount of differentiation may be regulated by choosing the value of parameter T. For $T = 0$ there is no differentiation, for $T \geq 25$ the blocking probability of the high priority requests drops below 0.005 for load 1.8, at the expense of the increased blocking probability in the low priority class, compared to the case without differentiation ($T = 0$).

Figs. 5.24 and 5.25 show the corresponding data for the PanEU network using fixed-alternate routing with three paths for each class. The obtained results are quite similar. However, as with the previous strategies, it is sufficient to use lower values of T to achieve the requested blocking probability in the high priority class. In the considered scenario the blocking probability of high priority requests lower than 0.005 was achieved with $T = 18$.

Figs. 5.26 and 5.27 show the three-dimensional plots of the blocking probability in the high priority class as a function of the offered load and the fraction of the high priority traffic. The first figure was obtained with shortest path routing and the second one with fixed-alternate routing and three paths for each traffic class. The results are similar for both cases. The blocking probability raises with increasing overall offered traffic and with increasing fraction of the high priority traffic in the network.

The corresponding plots for the low priority class are shown in Figs. 5.28 and 5.29.

The presented results were obtained in the PanEU network. The corresponding results obtained in the NSF network are quite similar and are shown in Appendix A in Figs. A.21 – A.28.

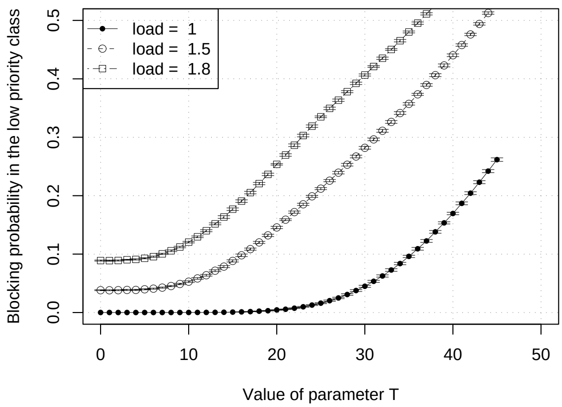

Figure 5.22: A strategy based on the link capacity threshold mechanism (lcap) and shortest path routing, PanEU network, 50% of traffic is the high priority traffic. The blocking probability in the low priority class is presented as a function of parameter T, for three values of the offered load.

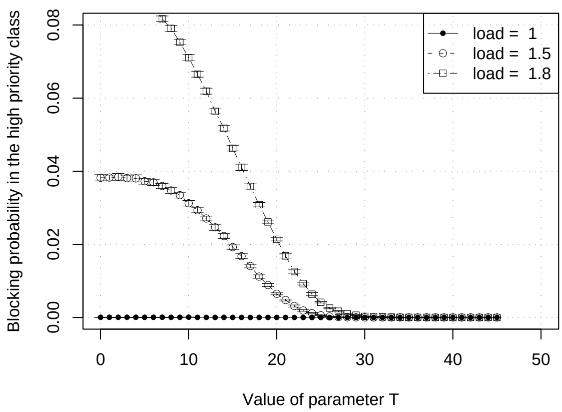

Figure 5.23: A strategy based on the link capacity threshold mechanism (lcap) and shortest path routing, PanEU network, 50% of traffic is the high priority traffic. The blocking probability in the high priority class is presented as a function of parameter T, for three values of the offered load.

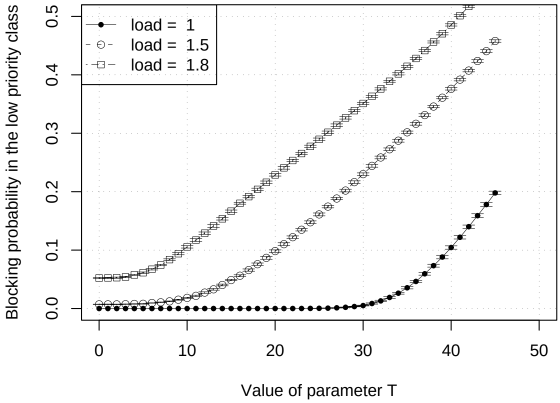

Figure 5.24: A strategy based on the link capacity threshold mechanism (lcap) and fixed-alternate routing with three paths for each class, PanEU network, 50% of traffic is the high priority traffic. The blocking probability in the low priority class is presented as a function of parameter T, for three values of the offered load.

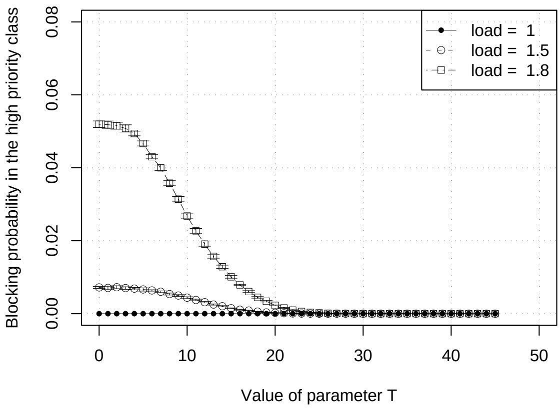

Figure 5.25: A strategy based on the link capacity threshold mechanism (lcap) and fixed-alternate routing with three paths for each class, PanEU network, 50% of traffic is the high priority traffic. The blocking probability in the high priority class is presented as a function of parameter T, for three values of the offered load.

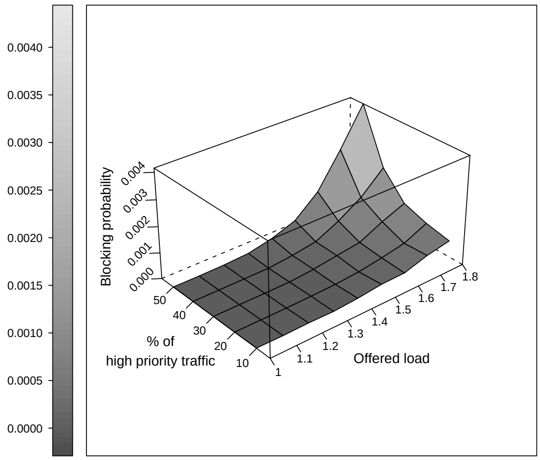

Figure 5.26: A strategy based on the link capacity threshold mechanism (lcap) and shortest path routing, PanEU network, $T = 25$. The blocking probability in the high priority class is presented as a function of the offered load and the fraction of the high priority traffic.

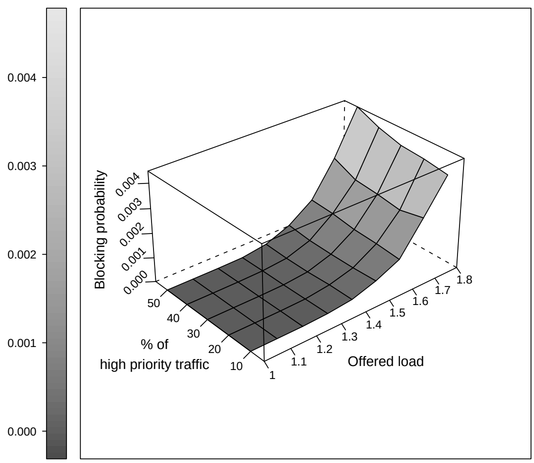

Figure 5.27: A strategy based on the link capacity threshold mechanism (lcap) and fixed-alternate routing with three paths for each class, PanEU network, $T = 18$. The blocking probability in the high priority class is presented as a function of the offered load and the fraction of the high priority traffic.

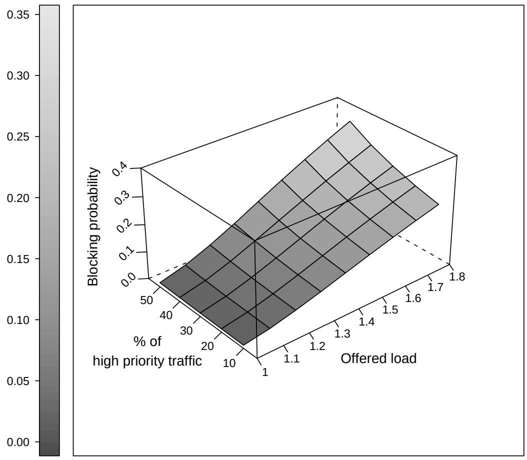

Figure 5.28: A strategy based on the link capacity threshold mechanism (lcap) and shortest path routing, PanEU network, $T = 25$. The blocking probability in the low priority class is presented as a function of the offered load and the fraction of the high priority traffic.

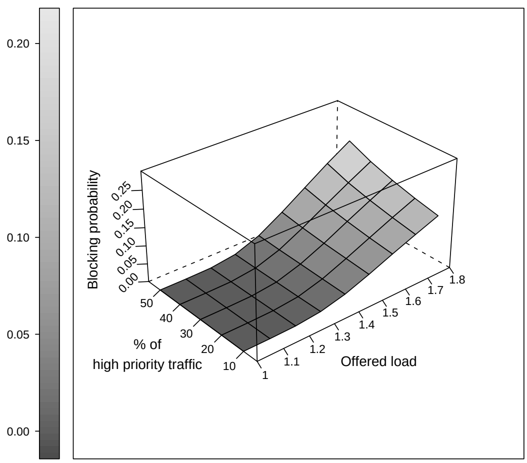

Figure 5.29: A strategy based on the link capacity threshold mechanism (lcap) and fixed-alternate routing with three paths for each class, PanEU network, $T = 18$. The blocking probability in the low priority class is presented as a function of the offered load and the fraction of the high priority traffic.

5.4 A strategy based on the path capacity threshold mechanism (pcap)

In this mechanism the candidate path for a low priority request is assessed according to the number of continuous free wavelengths on the path. If the number of wavelengths on the path is greater than the specified threshold, T, then the low priority request is admitted, else it is rejected. The high priority requests are always accepted, assuming the availability of a free wavelength on the candidate path. Two strategies: alt11-pcap and alt33-pcap were built to assess this mechanism. The first one is based on shortest path routing, the second one on the fixed-alternate routing with three paths for each class.

Figs. 5.30 and 5.31 show the blocking probability in the low and high priority classes as a function of parameter T in the PanEU network using shortest path routing. The results are presented for three values of the offered load when the fraction of the high priority traffic is equal to 50%. The figures show that the pcap strategy is able to achieve statistically significant differences in the blocking probability of low and high priority requests. The amount of differentiation may be regulated by choosing the value of parameter T. For $T = 0$ there is no differentiation, for $T \geq 8$ the blocking probability of the high priority requests drops below 0.005 for load 1.8, at the expense of the increased blocking probability in the low priority class, compared to the case without differentiation ($T = 0$).

Figs. 5.32 and 5.33 show the corresponding data for the PanEU network using fixed-alternate routing with three paths for each class. In the considered scenario the blocking probability of the high priority requests lower than 0.005 was achieved with $T = 3$.

Figs. 5.34 and 5.35 show the three-dimensional plots of the blocking probability in the high priority class as a function of the offered load and the fraction of the high priority traffic. The first figure was obtained with shortest path routing and the second one with fixed-alternate routing and three paths for each traffic class. The results are similar for both cases. The blocking probability raises with increasing overall offered traffic and with increasing fraction of the high priority traffic in the network.

The corresponding plots for low priority class are shown in Figs. 5.36 and 5.37.

The presented results were obtained in the PanEU network. The corresponding results obtained in the NSF network are quite similar and are shown in Appendix A in Figs. A.29 – A.36.

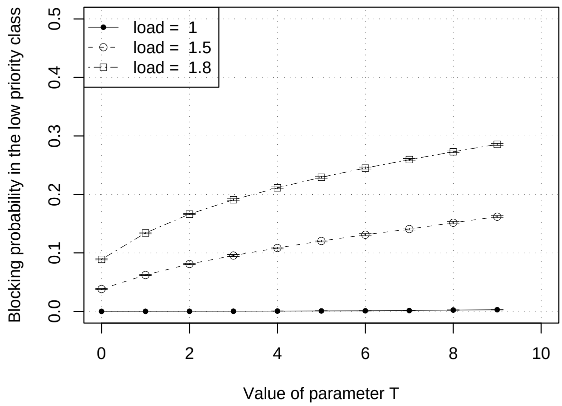

Figure 5.30: A strategy based on the path capacity threshold mechanism (pcap) and shortest path routing, PanEU network, 50% of traffic is the high priority traffic. The blocking probability in the low priority class is presented as a function of parameter T, for three values of the offered load.

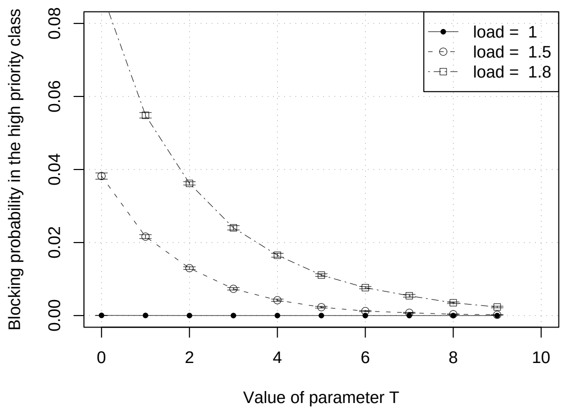

Figure 5.31: A strategy based on the path capacity threshold mechanism (pcap) and shortest path routing, PanEU network, 50% of traffic is the high priority traffic. The blocking probability in the high priority class is presented as a function of parameter T, for three values of the offered load.

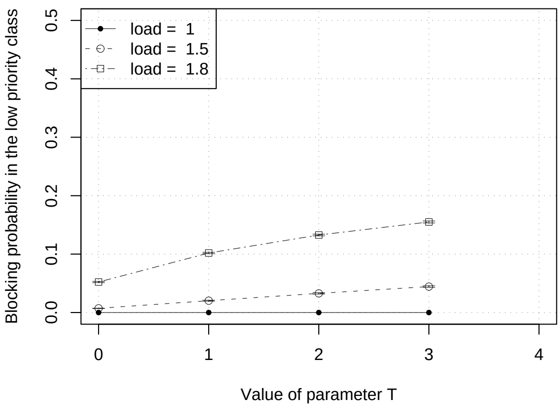

Figure 5.32: A strategy based on the path capacity threshold mechanism (pcap) and fixed-alternate routing with three paths for each class, PanEU network, 50% of traffic is the high priority traffic. The blocking probability in the low priority class is presented as a function of parameter T, for three values of the offered load.

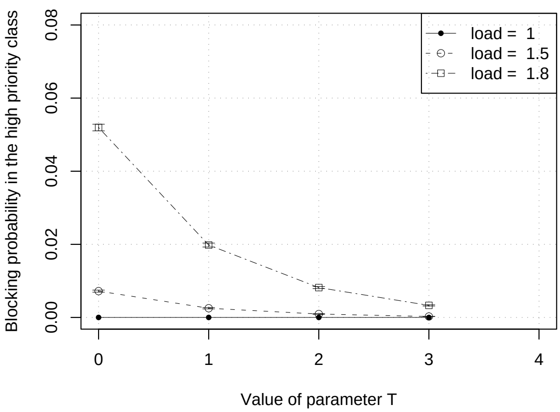

Figure 5.33: A strategy based on the path capacity threshold mechanism (pcap) and fixed-alternate routing with three paths for each class, PanEU network, 50% of traffic is the high priority traffic. The blocking probability in the high priority class is presented as a function of parameter T, for three values of the offered load.

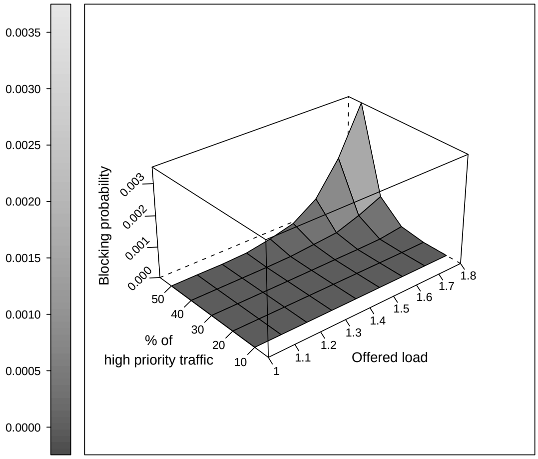

Figure 5.34: A strategy based on the path capacity threshold mechanism (pcap) and shortest path routing, PanEU network, $T = 8$. The blocking probability in the high priority class is presented as a function of the offered load and the fraction of the high priority traffic.

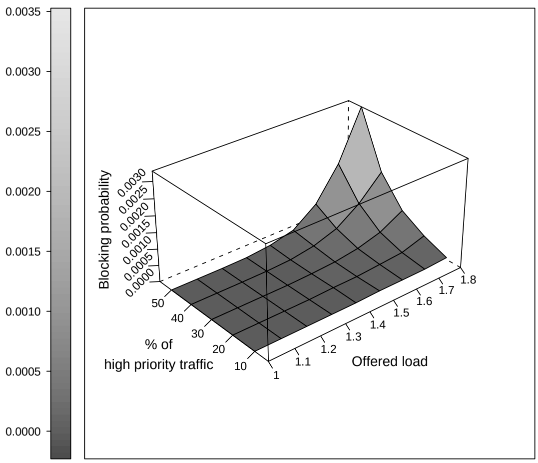

Figure 5.35: A strategy based on the path capacity threshold mechanism (pcap) and fixed-alternate routing with three paths for each class, PanEU network, $T = 3$. The blocking probability in the high priority class is presented as a function of the offered load and the fraction of the high priority traffic.

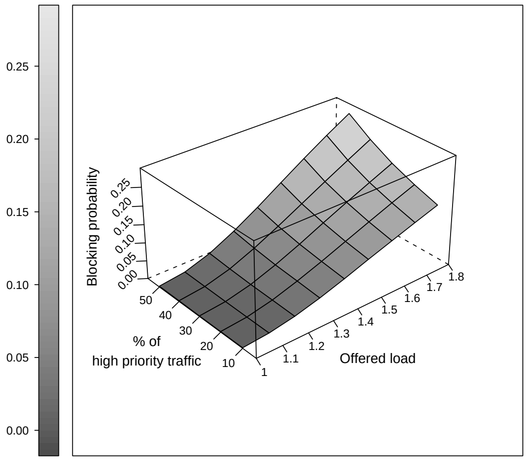

Figure 5.36: A strategy based on the path capacity threshold mechanism (pcap) and shortest path routing, PanEU network, $T = 8$. The blocking probability in the low priority class is presented as a function of the offered load and the fraction of the high priority traffic.

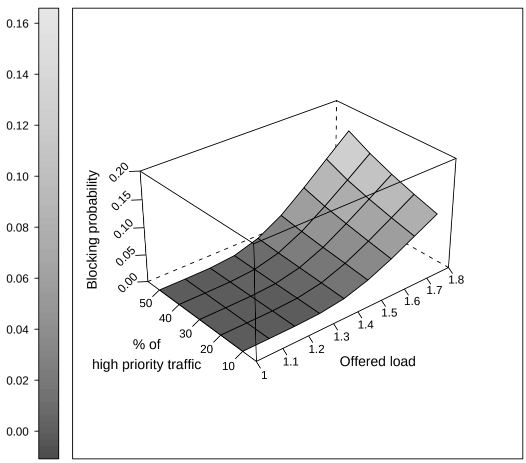

Figure 5.37: A strategy based on the path capacity threshold mechanism (pcap) and fixed-alternate routing with three paths for each class, PanEU network, $T = 3$. The blocking probability in the low priority class is presented as a function of the offered load and the fraction of the high priority traffic.

5.5 A strategy based on the global capacity threshold mechanism (gcap)

In this mechanism a candidate path for a low priority request is assessed to determine whether necessary resources are left between each source and destination nodes for future high priority requests. The high priority requests are always accepted, assuming the availability of a free wavelength on the candidate path. Two strategies: alt11-gcap and alt33-gcap were built to assess this mechanism. The first one is based on shortest path routing, the second one on the fixed-alternate routing with three paths for each class.

Figs. 5.38 and 5.39 show the blocking probability in the low and high priority classes as a function of parameter T in the PanEU network using shortest path routing. The results are presented for three values of the offered load when the fraction of the high priority traffic is equal to 50%. The figures show that the gcap strategy is able to achieve statistically significant differences in the blocking probability of low and high priority requests. The amount of differentiation on the value of parameter T. For $T = 0$ there is no differentiation, for $T \geq 6$ the blocking probability of the high priority requests drops below 0.005 for load 1.8.

Figs. 5.40 and 5.41 show the corresponding data for the PanEU network using fixed-alternate routing with three paths for each class. As with the previous mechanisms fixed-alternate routing provides lower blocking probabilities in both classes and results in lower necessary value of parameter T. In the considered scenario the blocking probability of high priority requests lower than 0.005 was achieved with $T = 2$.

Figs. 5.42 and 5.43 show the three-dimensional plots of the blocking probability in the high priority class as a function of the offered load and the fraction of the high priority traffic. The first figure was obtained with shortest path routing and the second one with fixed-alternate routing and three paths for each traffic class. The results are similar for both cases. The blocking probability raises with increasing overall offered traffic and with increasing fraction of the high priority traffic in the network.

The corresponding plots for the low priority class are shown in Figs. 5.44 and 5.45.

The presented results were obtained in the PanEU network. The corresponding results obtained in the NSF network are quite similar and are shown in Appendix A in Figs. A.37 – A.44.

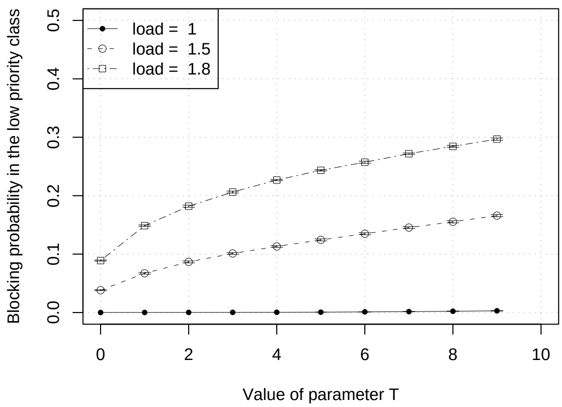

Figure 5.38: A strategy based on the global capacity threshold mechanism (gcap) and shortest path routing, PanEU network, 50% of traffic is the high priority traffic. The blocking probability in the high priority class is presented as a function of parameter T, for three values of the offered load.

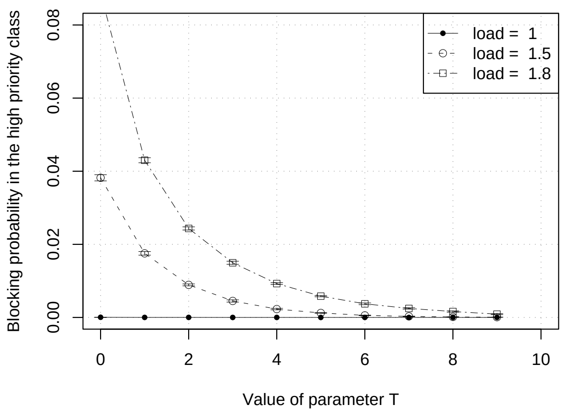

Figure 5.39: A strategy based on the global capacity threshold mechanism (gcap) and shortest path routing, PanEU network, 50% of traffic is the high priority traffic. The blocking probability in the high priority classes is presented as a function of parameter T, for three values of the offered load.

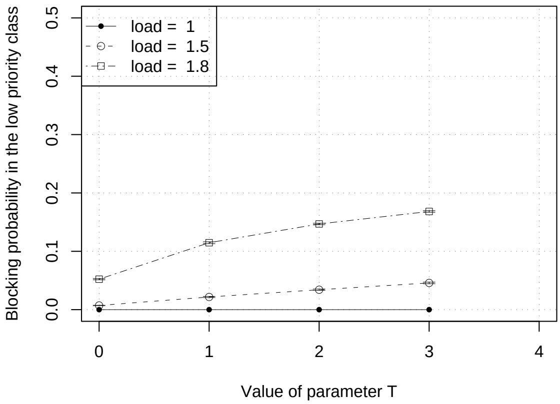

Figure 5.40: A strategy based on the global capacity threshold mechanism (gcap) and fixed-alternate routing with 3 paths for each class, PanEU network, 50% of traffic is the high priority traffic. The blocking probability in the low priority class is presented as a function of parameter T, for three values of the offered load.

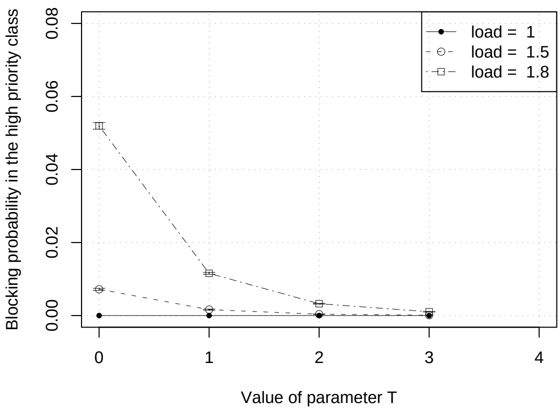

Figure 5.41: A strategy based on the global capacity threshold mechanism (gcap) and fixed-alternate routing with 3 paths for each class, PanEU network, 50% of traffic is the high priority traffic. The blocking probability in the high priority class is presented as a function of parameter T, for three values of the offered load.

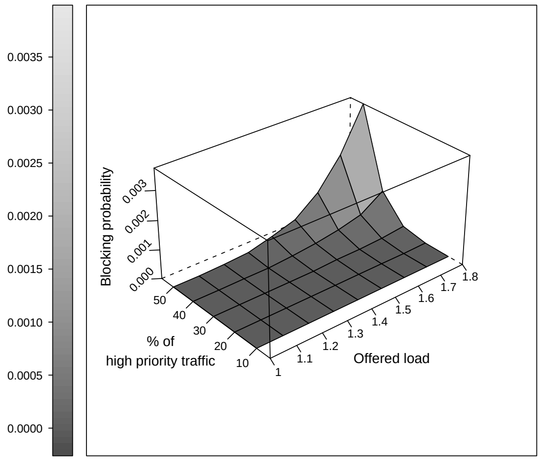

Figure 5.42: A strategy based on the global capacity threshold mechanism (gcap) and shortest path routing, PanEU network, $T = 6$. The blocking probability in the high priority class is presented as a function of the offered load and the fraction of the high priority traffic.

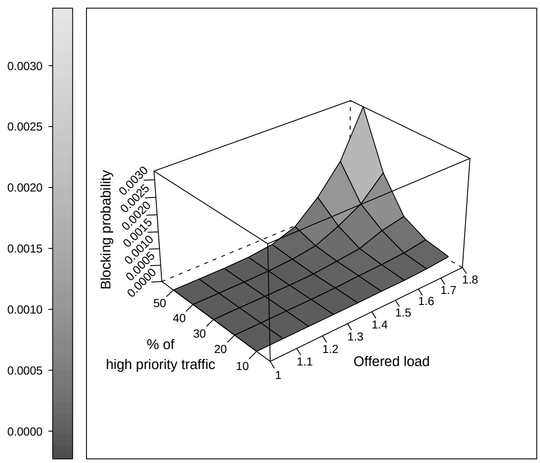

Figure 5.43: A strategy based on the global capacity threshold mechanism (gcap) and fixed-alternate routing with 3 paths for each class, PanEU network, $T = 2$. The blocking probability in the high priority class is presented as a function of the offered load and the fraction of the high priority traffic.

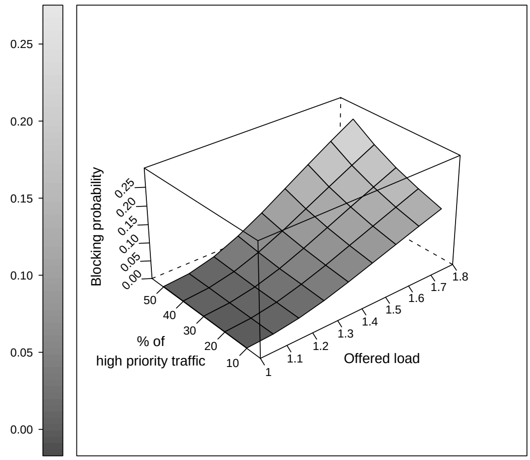

Figure 5.44: A strategy based on the global capacity threshold mechanism (gcap) and shortest path routing, PanEU network, $T = 6$. The blocking probability in the low priority class is presented as a function of the offered load and the fraction of the high priority traffic.

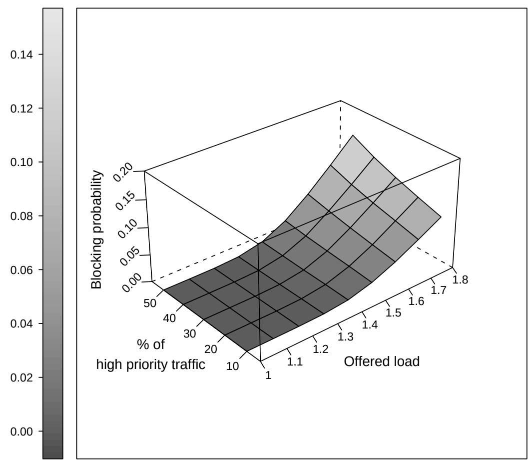

Figure 5.45: A strategy based on the global capacity threshold mechanism (gcap) and fixed-alternate routing with 3 paths for each class, PanEU network, $T = 2$. The blocking probability in the low priority class is presented as a function of the offered load and the fraction of the high priority traffic.

5.6 A strategy based on alternate routing

In the alternate paths mechanism low and high priority requests are offered a different number of alternate paths to try. In the simulation studies the maximum number of alternate paths was limited to three, due to the fact that the considered networks are more than two and less than three-connected topologies. The strategy has been investigated with the following combinations of the number of alternate paths for low and high priority requests: (1,2), (1,3), (2,3). Additionally, three cases without differentiation, (1,1), (2,2) and (3,3) were tested as a reference.

Figs. 5.46 and 5.47 show the blocking probability in the low and the high priority classes as a function of the offered load for two cases in which a single shortest path was used for low priority requests and more paths were used for high priority requests compared to the case when both classes were served using shortest path routing. Increasing the number of alternate paths for high priority requests resulted in lowering the blocking probability of the high priority requests at the expense of slightly increased blocking probability in the low priority class.

On the other hand, it is possible to take as a reference the case where all requests were served using three alternate paths, and reduce the number of paths available for the low priority requests. The result is shown in Figs. 5.48 and 5.49.

This illustrates two different viewpoints, from which the strategy based on the alternate paths mechanism may be assessed. The viewpoints are associated with the choice of the reference scenario. The first possibility is to choose a scenario with a weak algorithm for each class and observe the improvements gained from using a better algorithm in the high priority class. The second possibility is to start with a good algorithm in each class and deliberately decrease performance of the low priority class to make the network less utilised, so that the high priority requests have ample resources to choose from.

Figs. 5.50 and 5.51 show the three-dimensional plots of the blocking probability in the low and the high priority classes as a function of the offered load and the fraction of the high priority traffic in the network. As it was expected, the blocking probabilities in the low and the high priority classes rise gradually with increasing offered traffic and with increasing fraction of the high priority traffic.

The alternate paths mechanism, however successful in differentiating the low and high priority class, was unable to achieve the required (0.005) blocking probability of the high priority requests in some of the considered cases. Thus, in the subsequent studies it is only used in combination with other mechanisms and never used alone.

The presented results were obtained in the PanEU network. The corresponding results obtained in the NSF network are quite similar and are shown in Appendix A in Figs. A.45 – A.50.

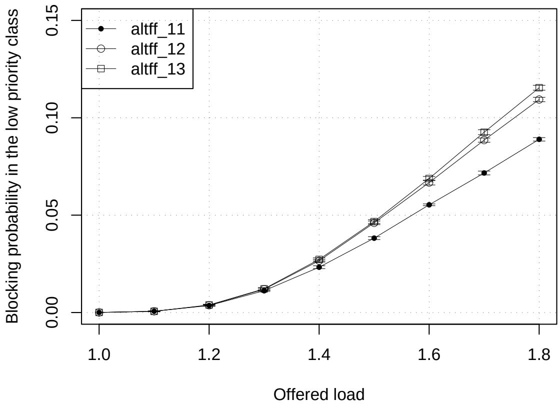

Figure 5.46: A strategy based on the alternate paths mechanism with 1 path in the low priority class and varying number of paths in the high priority class. PanEU network with 50% of high priority traffic. The blocking probability in the low priority class is presented as a function of the offered load.

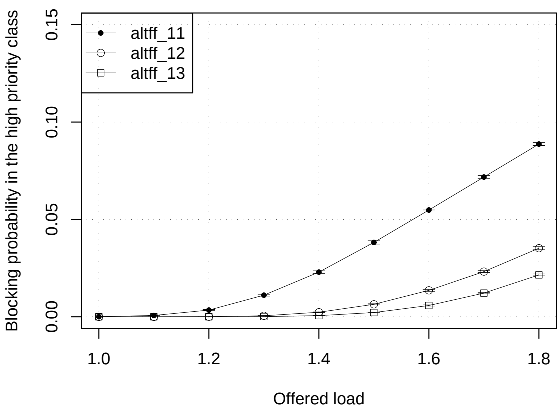

Figure 5.47: A strategy based on the alternate paths mechanism with 1 path in the low priority class and varying number of paths in the high priority class. PanEU network with 50% of high priority traffic. The blocking probability in the high priority class is presented as a function of the offered load.

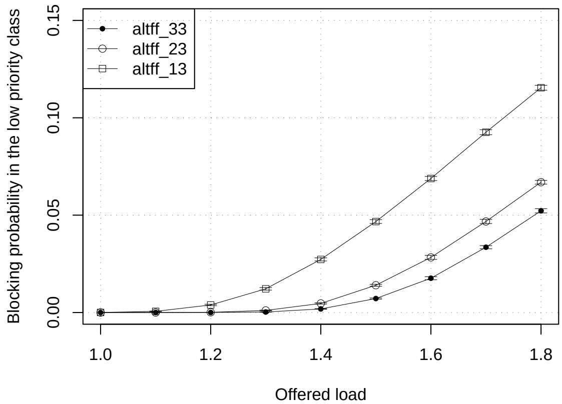

Figure 5.48: A strategy based on the alternate paths mechanism with 3 paths in the high priority class and varying number of paths in the low priority class. PanEU network with 50% of high priority traffic. The blocking probability in the low priority class is presented as a function of the offered load.

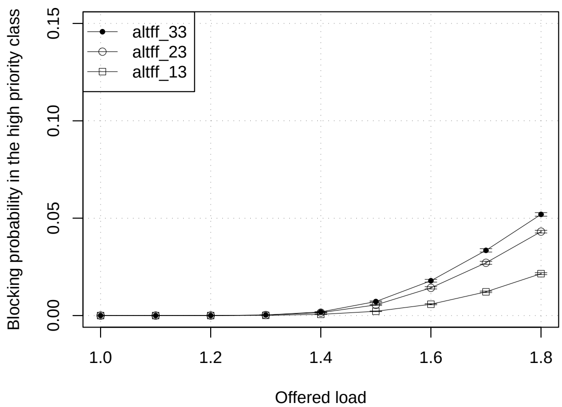

Figure 5.49: A strategy based on the alternate paths mechanism with 3 paths in the high priority class and varying number of paths in the low priority class. PanEU network with 50% of high priority traffic. The blocking probability in the high priority class is presented as a function of the offered load.

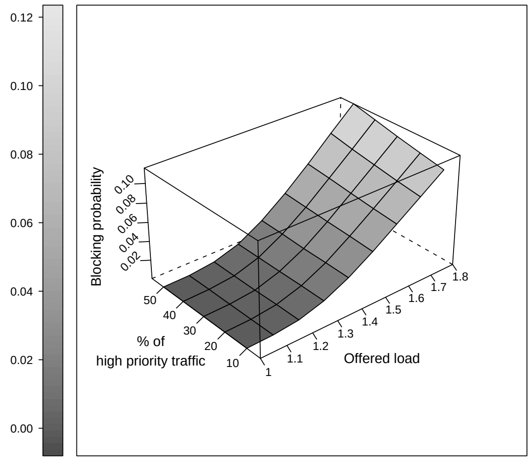

Figure 5.50: A strategy based on the alternate paths mechanism with 1 path in the low priority class and 3 paths in the high priority class. PanEU network. The blocking probability in the low priority class is presented as a function of the offered load and the fraction of the high priority traffic.

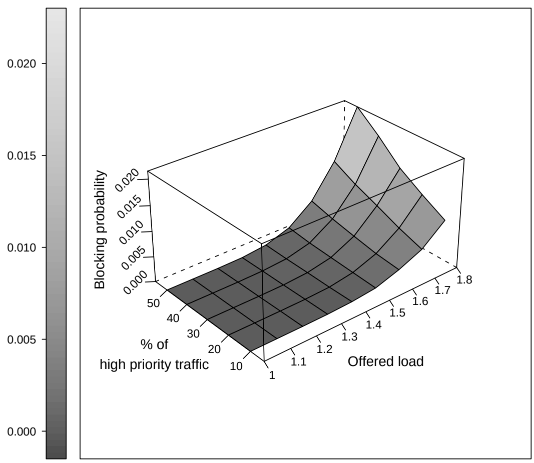

Figure 5.51: A strategy based on the alternate paths mechanism with 1 path in the low priority class and 3 paths in the high priority class. PanEU network. The blocking probability in the high priority class is presented as a function of the offered load and the fraction of the high priority traffic.

6 Comparison of strategies

Several strategies need to be investigated. Fifteen of those strategies are based on a single mechanism performing resource preservation. Here, the routing algorithm delivers the same number of paths for each class of requests. In the remaining fifteen strategies the lightpath requests are differentiated using two mechanisms, one based on resource preservation and one based on a different number of alternate paths in each class. The strategies based solely on a different number of paths were excluded from comparisons since they could not achieve the requested blocking probability of 0.005 in some of the assumed traffic conditions.

6.1 Comparison methodology

For each strategy it is possible to choose a different value of parameter T, which regulates the amount of resources preserved for the high priority traffic. As a result, this value affects the balance between the blocking probability of the low and high priority requests. To be able to compare the strategies, there is a need to have a reference criterion, to which the strategies would be normalised. Two assumptions were made:

> **Assumption 7.1:** The mean blocking probability of the high priority requests should be below 0.005 for each of the 45 cases of the offered traffic considered.

> **Assumption 7.2:** The value of T should remain constant with changing traffic conditions, i.e., in all 45 investigated cases of the offered traffic.

The traffic cases are based on the offered load in the range of 1.0 – 1.8 in steps of 0.1 and a variable fraction of the high priority traffic, ranging from 10%

to 50% in steps of 10%. Assumption 7.1 reflects the operator's need to guarantee that the high priority requests are served in a satisfactory manner in changing traffic conditions. The choice of the value 0.005 as the maximum allowed blocking probability for high priority requests is arbitrary. Any reasonable value may be used here, as required by the network operator. The range of offered traffic, in which the blocking probability remains below the specified minimum, is also arbitrarily chosen, and should reflect the past network operator experience with the given network and the contracts signed with the network users. Assumption 7.2 reflects a requirement, that the network optimization is performed off-line, and relatively rare, compared to the rate of the traffic changes in the network. This is a convenient situation for the network operator since then it neither needs a fast optimization algorithm nor signalling extensions to quickly propagate the changing value of T to all nodes. This is a potential scenario of an ASON network with a variable load and fixed network parameters.

Consequently, a lower bound for parameter T is established since with increasing its value the amount of resources devoted to high priority requests increases, and as a result the blocking probability of the high priority requests decreases. However, increasing the value of parameter T increases the blocking probability of the low priority requests which is undesirable. Thus, it is reasonable to keep the value of T as low as possible to avoid unnecessary drop in performance of the low priority requests. As a result, for each strategy there is only one possible value of parameter T. In the remaining comparisons only this value of parameter T is used for a given strategy. The obtained values are presented in Tables 6.1 and 6.2, names *altXY* reference the alternate paths algorithm with X paths for low priority requests and Y paths for high priority requests.

Table 6.1: Values of parameter T for all strategies, PanEU network

Scenario	alt11	alt12	alt13	alt22	alt23	alt33
gcap	6	2	1	3	2	2
pcap	8	3	2	4	3	3
lcap	25	17	14	20	17	18
pool	40	23	16	29	24	26
flcap	53	38	32	50	46	48

The assumptions 7.1 and 7.2 have one more implication. The high priority class is served in a satisfactory manner in all assumed traffic conditions, for all considered strategies. Thus it is possible to disregard it in the subsequent discussions and focus on the comparisons of the blocking probability of the low priority requests, since this reflects the price paid for granting the high priority requests a superior grade of service.

Table 6.2: Values of parameter T for all strategies, NSF network

Scenario	alt11	alt12	alt13	alt22	alt23	alt33
gcap	5	2	1	3	2	2
pcap	6	2	2	3	2	2
lcap	16	11	8	15	13	14
pool	32	17	12	23	20	22
flcap	30	17	12	30	26	30

6.2 Comparison of single mechanism strategies

Figs. 6.1 and 6.2 present the comparison of the single mechanism strategies. The figures show the blocking probability of the low priority requests vs. offered load. The network is the PanEU network, with 20% of the high priority traffic using either alt11 or alt33 routing.

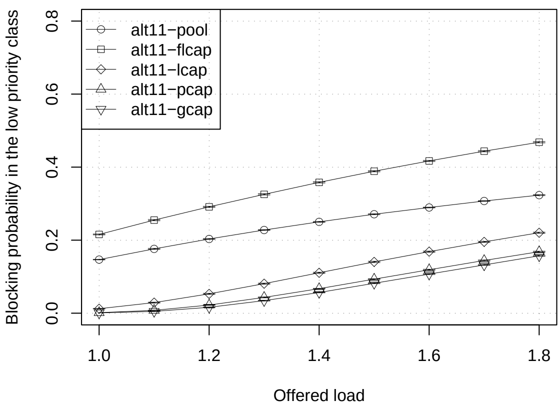

Figure 6.1: Comparison of resource preservation mechanisms for shortest path routing, PanEU network, 20% of traffic is the high priority traffic. The blocking probability in the low priority class as a function of the offered load.

The performance of the compared strategies varies greatly. Strategies based on gcap and pcap mechanisms seem to be the most promising ones, as they offer the lowest blocking probability. Their performance is quite close to each other, with a larger gap to the nearest successor, the lcap strategy. Strategies based on flcap and pool mechanisms achieve the highest blocking probabilities.

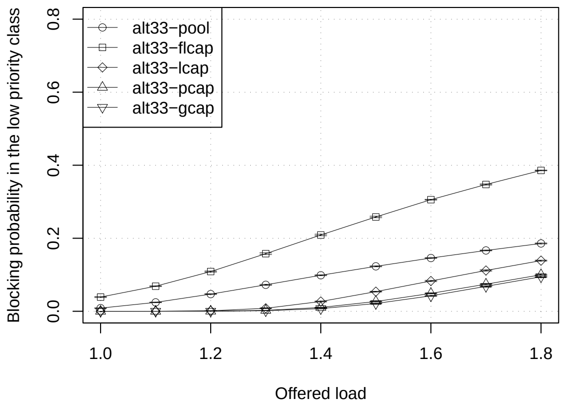

Figure 6.2: Comparison of resource preservation mechanisms for fixed-alternate routing with three paths in each class, PanEU network, 20% of traffic is the high priority traffic. The blocking probability in the low priority class as a function of the offered load.

An additional drawback of the strategies based on flcap and pool mechanisms is that their blocking probability remains quite high in the low load region which is undesirable from the operator's point of view since it cannot offer low blocking probability to all customers even in the off-peak times. The strategies based on gcap and pcap mechanisms do not seem to have this negative property as the blocking probability of those strategies falls quite fast to near zero with the decreasing offered load.

The blocking probability for strategies based on alt11 routing is significantly worse than the blocking probability for strategies based on alt33 routing. This is not a new observation, but it is a strong indication that limiting the number of paths has a negative impact on the achieved GoS and, thus, the shortest path routing should be avoided, if possible.

For complete overview, Figs. 6.3 and 6.4 present the total blocking probability and Figs. 6.5 and 6.6 present the network utilisation for the considered strategies.

6.3 Single vs. compound strategies

The next step is to analyse benefits gained from applying the strategies that use two GoS mechanisms simultaneously, and to compare their performance with the single mechanism strategies. Tables 6.3 and 6.4 present the blocking probability

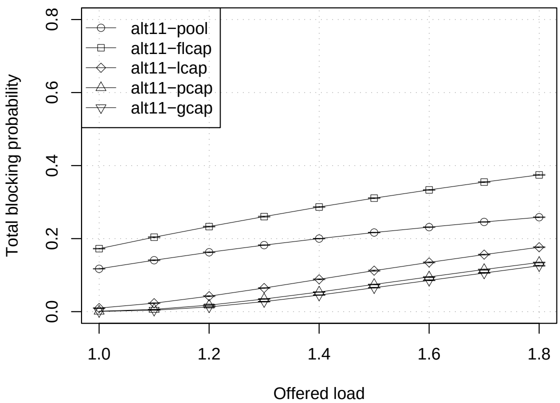

Figure 6.3: Comparison of resource preservation mechanisms for shortest path routing, PanEU network, 20% of traffic is the high priority traffic. The total blocking probability as a function of the offered load.

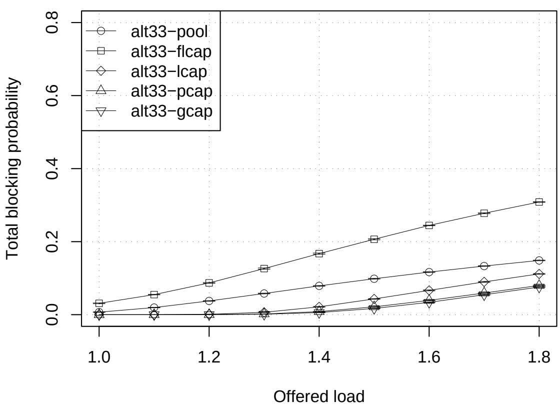

Figure 6.4: Comparison of resource preservation mechanisms for fixed-alternate routing with three paths in each class, PanEU network, 20% of traffic is the high priority traffic. The total blocking probability as a function of the offered load.

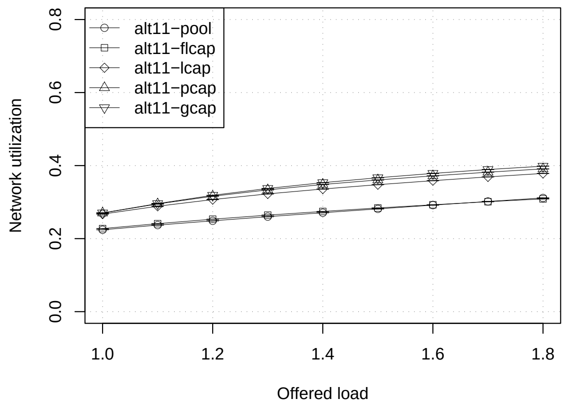

Figure 6.5: Comparison of resource preservation mechanisms for shortest path routing, PanEU network, 20% of traffic is the high priority traffic. The network utilization as a function of the offered load.

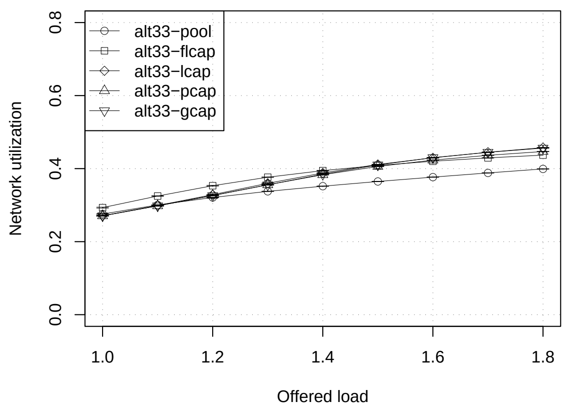

Figure 6.6: Comparison of resource preservation mechanisms for fixed-alternate routing with three paths in each class, PanEU network, 20% of traffic is the high priority traffic. The network utilization as a function of the offered load.

of the low priority requests for all considered strategies, implementing either a single mechanism or a mix of mechanisms. The results were obtained in the PanEU and NSF networks for a fixed load of 1.4 and 20% of the high priority traffic.

Table 6.3: Blocking probabilities of the low priority requests for the PanEU network with offered load of 1.4 and 20% of high priority traffic.

Scenario	alt11	alt12	alt13	alt22	alt23	alt33
gcap	0.057	0.038	0.033	0.015	0.012	0.007
pcap	0.067	0.042	0.038	0.019	0.016	0.010
lcap	0.111	0.055	0.041	0.043	0.026	0.027
pool	0.250	0.128	0.088	0.128	0.093	0.099
flcap	0.358	0.168	0.118	0.235	0.159	0.209

Table 6.4: Blocking probabilities of the low priority requests for the NSF network with offered load of 1.4 and 20% of high priority traffic.

Scenario	alt11	alt12	alt13	alt22	alt23	alt33
gcap	0.033	0.021	0.018	0.005	0.004	0.002
pcap	0.038	0.021	0.021	0.006	0.004	0.002
lcap	0.067	0.038	0.025	0.024	0.015	0.013
pool	0.201	0.084	0.056	0.088	0.066	0.070
flcap	0.174	0.060	0.034	0.145	0.093	0.154

There is no general answer to the question whether introducing two mechanisms that provide GoS differentiation is beneficial for the network. The answer depends on the preservation mechanism used. The flcap mechanism clearly benefits from adding a second mechanism and works best with alt13 routing. On the other hand, the gcap and pcap mechanisms work best with alt33 routing and introducing different number of paths for each class degrades the performance. To aid comparison, the results from Tables 6.3 and 6.4 are illustrated in a graphical form in Figs. 6.7 and 6.8 with alt22 cases removed for clarity of presentation.

If one looks at the performance of the flcap mechanism with alt11, alt12, alt13, i.e., with a constant number of paths for low priority requests and an increasing number of alternate paths for high priority requests, one will see that the performance of the network increases (the blocking probability decreases). However, a deeper look at the sequence alt13, alt23, alt33, (i.e., with a constant number of alternate paths for high priority requests and increasing number of paths for low priority requests,) reveals a major drop in performance. This might be an indication that the flcap mechanism is quite inefficient in protecting

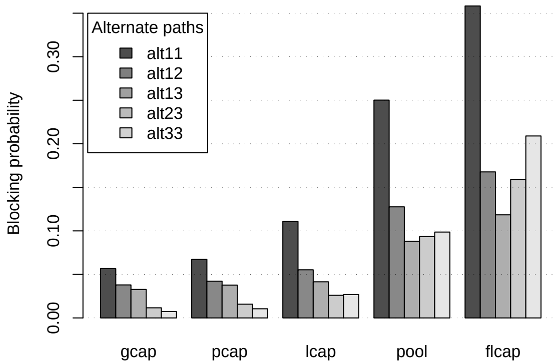

Figure 6.7: The comparison of the strategies for the PanEU network with an offered load of 1.4 and 20% of high priority traffic.

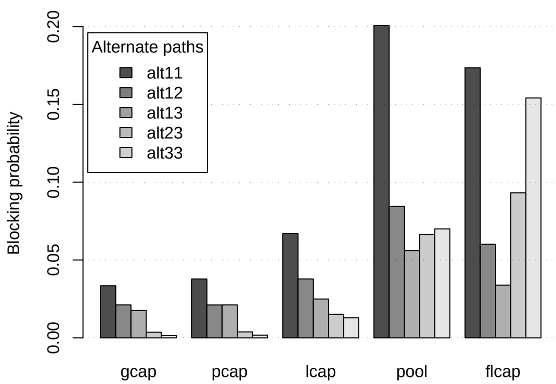

Figure 6.8: The comparison of the strategies for the NSF network with an offered load of 1.4 and 20% of high priority traffic.

resources. Increasing the number of paths for low priority requests increases the chances that such a request will find available resources and will be successfully set up. As a result, there are less resources left for high priority requests and as a countermeasure the amount of resources that are preserved (parameter T) must be increased, see Tables 6.1 and 6.2. This results in lowering the amount of resources for low priority traffic. A strategy that is inefficient preserves a great deal of resources since it cannot use them efficiently. Thus, it will decrease the overall network performance.

The gcap and pcap strategies do not show such a behaviour. The blocking probability decreases with improving the routing algorithm and the best results are achieved with the maximum considered number of paths for each class. The remaining strategies, pool and lcap, fall somewhere in the middle. At a first look they follow the flcap pattern, but, if evaluated against some other combinations of offered load and fraction of the high priority traffic they behave closer to the pcap and gcap algorithms.

6.4 Final evaluation

The following five comparisons have been prepared and presented to get the final insight on the performance of different strategies, each composed of a given routing variant and a preservation mechanism. In the first comparison, the number of cases in which a given routing variant offers the lowest blocking probability is evaluated, given the resource preservation mechanism. Each case corresponds to a combination of the offered load and the fraction of the high priority traffic. The result is presented in Figs. 6.9 and 6.10. The figures answer the following question: which routing variant is most successful for a given resource preservation mechanism?

It is quite clear that if gcap or pcap mechanisms are used in the network, the most successful routing scheme is alt33, as it offers the best performance in all cases. For lcap and pool the situation is not clear. For some cases the alt33 routing would be the best, for some others alt23 and alt13. For flcap the situation is almost clear, as in the majority of cases the best results are achieved with the alt13 routing.

The second comparison shows which preservation strategy performs the best if the network is restricted to use a particular variant of the fixed-alternate routing. The results are presented in Figs. 6.11 and 6.12. In the majority of the results the gcap algorithm is the best, followed by the pcap algorithm. However, its use in the network might not be preferred due to performance reasons. Thus, another comparison, which excludes gcap, is presented in Figs. 6.13 and 6.14.

The presented comparisons are a strong indication that, out of the presented preservation mechanisms, gcap and pcap might perform the best with any routing

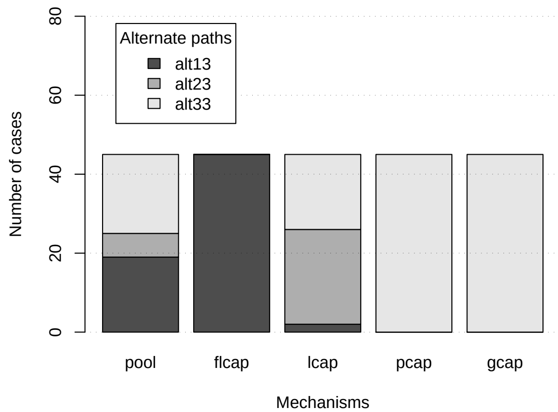

Figure 6.9: The number of cases in the PanEU network in which a given routing variant was the most successful one for a given preservation mechanism. The following routing variants: alt11, alt12 and alt22 were never the most successful ones, thus they are omitted.

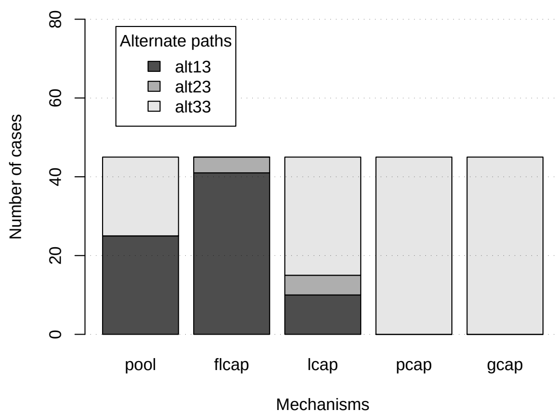

Figure 6.10: The number of cases in the NSF network in which a given routing variant was the most successful one for a given preservation mechanism. The following routing variants: alt11, alt12 and alt22 were never the most successful ones, thus they are omitted.

Figure 6.11: The number of cases for the PanEU network in which a given preservation mechanism is the most successful for a given routing variant. pool and flcap mechanisms were never the most successful ones, thus they are omitted.

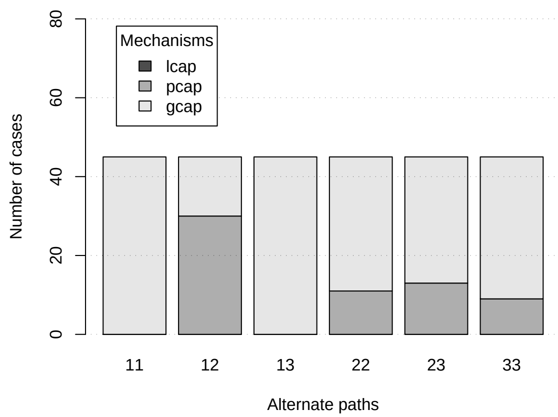

Figure 6.12: The number of cases for the NSF network in which a given preservation mechanism is the most successful for a given routing variant. pool and flcap mechanisms were never the most successful ones, thus they are omitted.

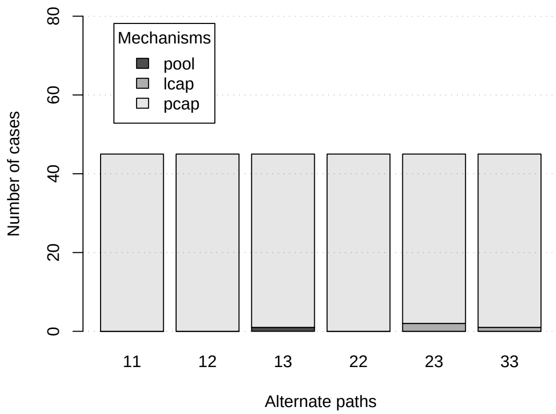

Figure 6.13: The number of cases for the PanEU network in which a given preservation mechanism is the most successful for a given routing variant when the gcap mechanism is excluded from comparisons. The flcap mechanism was never the most successful one, thus it is omitted.

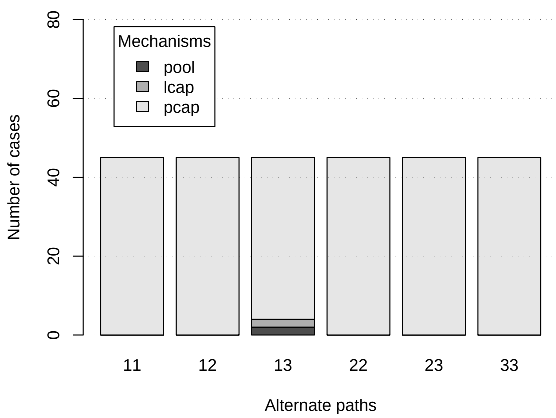

Figure 6.14: The number of cases for the NSF network in which a given preservation mechanism is the most successful for a given routing variant when the gcap mechanism is excluded from comparisons. The flcap mechanism was never the most successful one, thus it is omitted.

Table 6.5: The best strategies

Network	Strategy	No of cases
PanEU	alt33-gcap	45
NSF	alt33-gcap	36
	alt33-pcap	9

Table 6.6: The best strategies if gcap is not considered

Network	Strategy	No of cases
PanEU	alt33-pcap	44
	alt33-lcap	1
NSF	alt33-pcap	45

algorithm. On the other hand, the remaining algorithms are almost nonexistent in these comparisons, which supports the hypothesis that they are not good candidates for further consideration.

The final comparisons reflect the global view on the performance of different combinations of the presented mechanisms. The presented result shows the number of cases in which a given strategy was the best one, in all considered cases of the offered traffic. Table 6.5 presents the results.

The numbers show that the alt33-gcap strategy, combining the global capacity threshold mechanism and the fixed-alternate routing with three paths for each traffic class achieved the best results for the PanEU network, and achieved the majority of the best results (35 out of 45 cases) for the NSF network. However, this strategy has severe scalability issues, caused by its computational complexity, and also requires a great deal of information about the global network state, which complicates the control plane design. Thus, it might be interesting to know which strategy performs the best if the gcap mechanism is excluded from comparisons. The result is shown in Table 6.6.

In this comparison, the best result is obtained using the alt33-pcap strategy, combining the path capacity threshold mechanism with routing that provides three alternate paths for each request, both low and high priority. The strategy achieved the lowest blocking probability for low priority requests in all, but the one case, for the PanEU network and in all cases for the NSF network.

Taking all points together, gcap and pcap seem to be the strong candidates for implementing them in real networks as the sole GoS mechanism. An important observation is that pcap follows closely the performance of gcap (see Figs. 6.7 and 6.8), which might be a strong indication that in the overall cost-benefit comparison pcap might be preferred, due to its much lower complexity and a similar performance to gcap. As far as other algorithms are considered, lcap and pool offer a similar complexity to pcap, but significantly worse performance, and flcap delivers unacceptable performance with a slightly smaller complexity. As a result lcap, pool and flcap are questionable candidates for implementing in real networks.

7 Conclusions

The goal of the presented research was to investigate the possibility of differentiating Grade of Service in a wavelength routed optical network by using appropriate routing strategies. Six mechanisms were proposed, five of them were based on resource preservation, one on providing a different number of alternate paths for each class of requests. The mechanisms have been created taking into consideration the amount of information required for their operation in a distributed environment. For each mechanism a detailed algorithm was presented in two versions: one suitable for networks with a centralised computation model and one suitable for networks with a distributed computation model. Additionally, the computational complexity of each mechanism was estimated.

The presented mechanisms were investigated using a custom-built simulation tool based on the OMNeT++ simulator. A description of the simulation environment is followed by the discussion on the credibility of the obtained results.

The simulations have been performed in two reference networks: the PanEU reference network and the NSF network. Each proposed mechanism was able to provide differentiated Grade of Service for lightpath requests that belong to different classes, although with a different performance.

Based on the presented mechanisms a set of strategies was built, with some of the strategies incorporating two mechanisms simultaneously. The goal of this part of research was to find the most prospective mechanism or a combination of mechanisms to be used in future networks. The strategies were evaluated using computer simulation in several traffic scenarios, with varying overall traffic intensity and varying proportions of the high and low priority traffic. All proposed strategies managed to achieve the stated goals in assumed network conditions and successfully delivered GoS-differentiated services. However, the cost of implementing those strategies, being the decreased performance of low priority requests and increased complexity of control procedures, was quite different.

The two best strategies were based on either global capacity threshold and path capacity threshold mechanisms, the former performing slightly better than the latter at the cost of radically increased complexity and control plane requirements. These two mechanisms were the best ones with each variant of alternate paths routing that were tested. However, the best results were obtained when all requests, regardless of their class, were offered the maximum allowed number of paths. Introducing an additional level of GoS differentiation by offering a different number of paths for each class of requests resulted, for those two strategies, in a performance drop.

In an overall view, global capacity threshold and path capacity threshold are preferred mechanisms to implement them in future optical networks, due to their superiority in performance.

The GoS differentiation allows a network operator to deliver services tailored to customers' requirements, while simultaneously ensuring efficient resource utilisation. GoS differentiation is the first, but necessary step, for providing multiple services with GoS guarantees. Along with the development of new services delivered to customers and growing users' needs, the complexity of GoS mechanisms will increase in order to deal with more traffic classes and less predictable traffic patterns. The improved GoS mechanisms will provide more precise control over such GoS parameters as the blocking probability and connection set up time and their distribution among communicating node pairs. This seems to be the focus of the future work in this field.

7.1 Achievements and contributions

The achievements and contributions of the dissertation may be summarised as follows:

1. The dissertation introduced a classification of the Grade of Service mechanisms into mechanisms which preserve network resources for high priority requests, mechanisms in which a different routing algorithm, path metrics or a set of candidate paths is considered for each class of requests, and mechanisms in which an existing lower class lightpath may be preempted or rerouted if the resources it uses are needed for a higher class lightpath.

2. A set of five mechanisms that preserve network resources for high priority requests, and a mechanism that uses different number of alternate paths for each class of requests were proposed. For each of those mechanisms a detailed algorithms for both centralised computation model and distributed computation model was presented. Additionally, the numerical complexity of the proposed mechanisms was estimated.

3. A custom simulation tool was built, based on the OMNeT++ simulator.

4. The performance of each of the presented mechanisms was thoroughly investigated using computer simulation.

5. The benefits of using more than one mechanism were assessed and a prospective strategies were chosen for final comparison.

   Based on the presented results, the thesis stated in Chapter 1.1, ***It is possible to differentiate the blocking probability of lightpath requests in optical networks using moderately complex routing strategies***, was proved.

A Experiments: single mechanism strategies in the NSF network

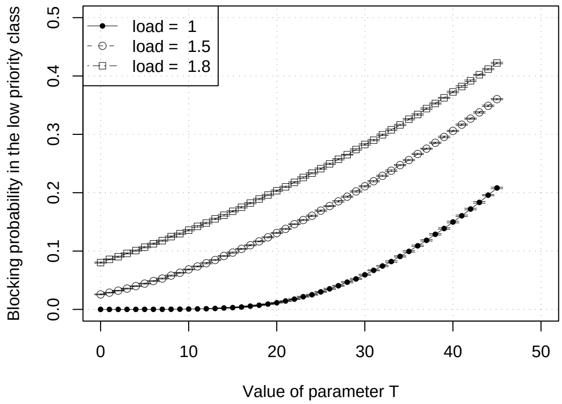

Figure A.1: Wavelength pools strategy (pool), NSF network, shortest path routing, 50% of traffic is the high priority traffic. The blocking probability in the low priority class as a function of parameter T, for three values of the offered load.

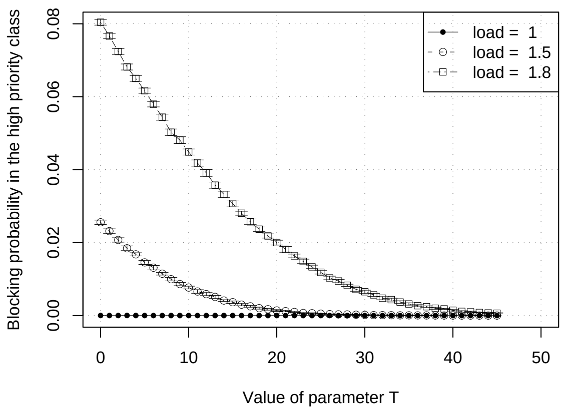

Figure A.2: Wavelength pools strategy (pool), NSF network, shortest path routing, 50% of traffic is the high priority traffic. The blocking probability in the high priority class as a function of parameter T, for three values of the offered load.

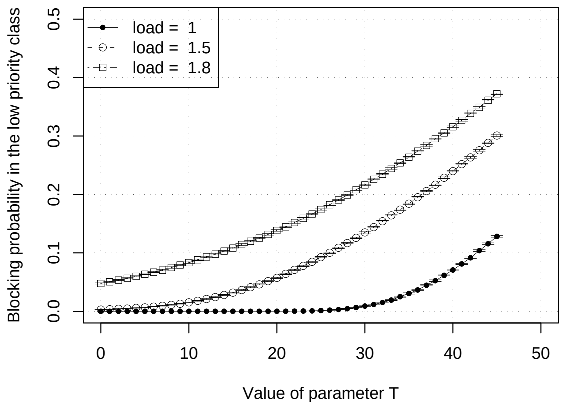

Figure A.3: Wavelength pools strategy (pool), NSF network, alternate routing with 3 paths in each class, 50% of traffic is the high priority traffic. The blocking probability in the low priority class as a function of parameter T, for three values of the offered load.

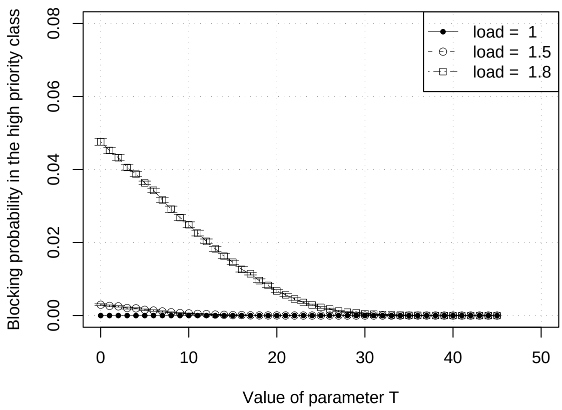

Figure A.4: Wavelength pools strategy (pool), NSF network, alternate routing with 3 paths in each class, 50% of traffic is the high priority traffic. The blocking probability in the high priority class as a function of parameter T, for three values of the offered load.

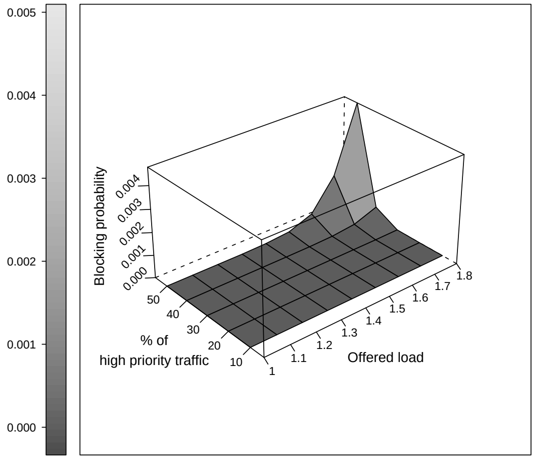

Figure A.5: Wavelength pools strategy (pool), NSF network, shortest path routing, $T = 32$. The blocking probability in the high priority class as a function of the offered load and the fraction of the high priority traffic.

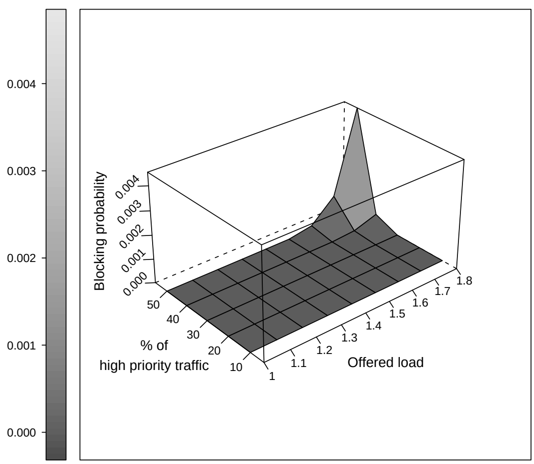

Figure A.6: Wavelength pools strategy (pool), NSF network, alternate routing with 3 paths in each class, $T = 22$. The blocking probability in the high priority class as a function of the offered load and the fraction of the high priority traffic.

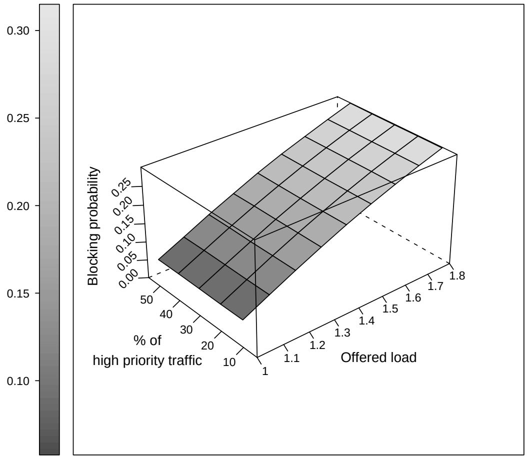

Figure A.7: Wavelength pools strategy (pool), NSF network, shortest path routing, $T = 32$. The blocking probability in the low priority class as a function of the offered load and the fraction of the high priority traffic.

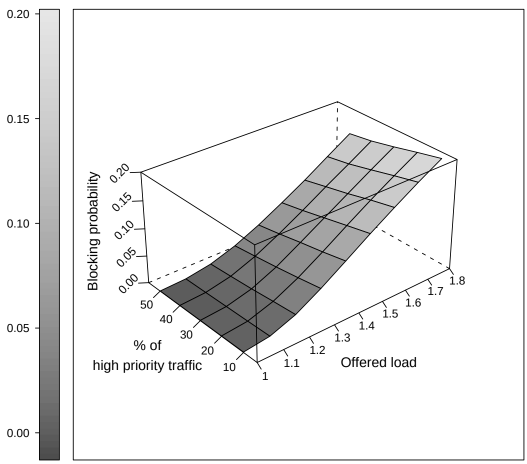

Figure A.8: Wavelength pools strategy (pool), NSF network, alternate routing with 3 paths in each class, $T = 22$. The blocking probability in the low priority class as a function of the offered load and the fraction of the high priority traffic.

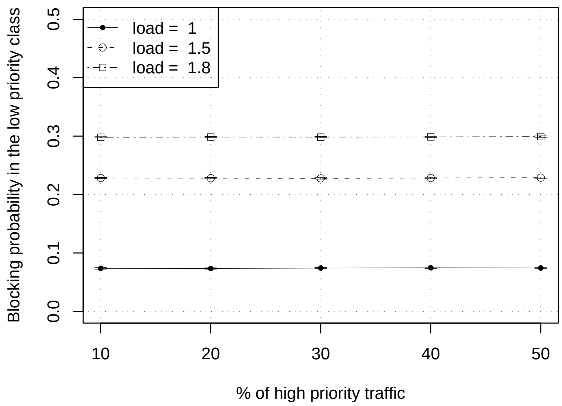

Figure A.9: Wavelength pools strategy (pool), NSF network, shortest path routing. The blocking probability in the low priority class as a function of the fraction of the high priority traffic, for three values of the offered load.

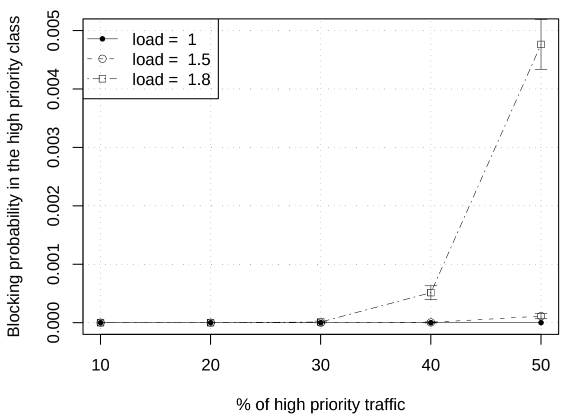

Figure A.10: Wavelength pools strategy (pool), NSF network, shortest path routing. The blocking probability in the high priority class as a function of the fraction of the high priority traffic, for three values of the offered load.

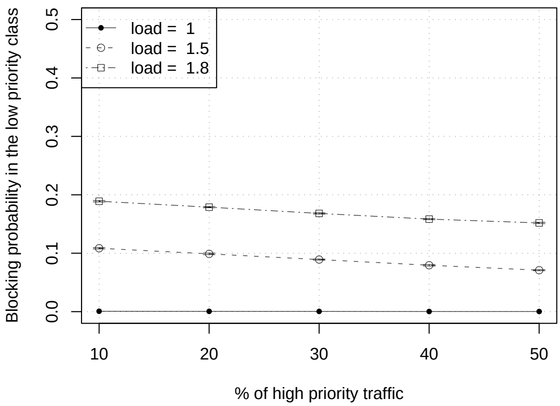

Figure A.11: Wavelength pools strategy (pool), NSF network, alternate routing with three paths for each class. The blocking probability in the low priority class as a function of the fraction of the high priority traffic, for three values of the offered load.

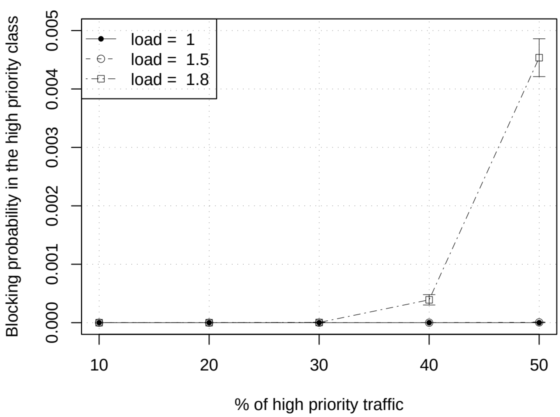

Figure A.12: Wavelength pools strategy (pool), NSF network, alternate routing with three paths for each class. The blocking probability in the high priority class as a function of the fraction of the high priority traffic, for three values of the offered load.

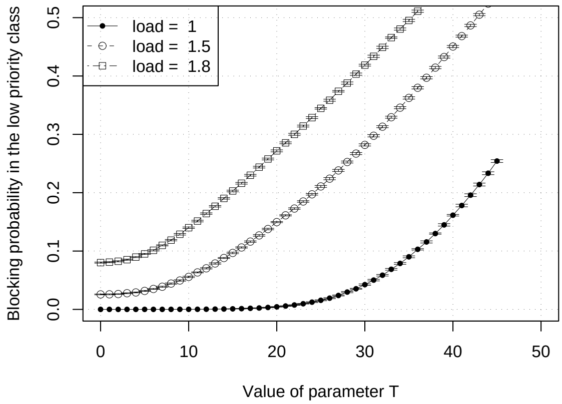

Figure A.13: First link capacity threshold strategy (flcap), NSF network, shortest path routing, 50% of traffic is the high priority traffic. The blocking probability in the low priority class as a function of parameter T, for three values of the offered load.

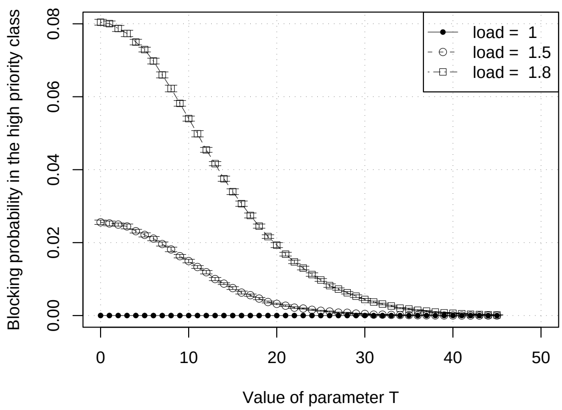

Figure A.14: First link capacity threshold strategy (flcap), NSF network, shortest path routing, 50% of traffic is the high priority traffic. The blocking probability in the high priority class as a function of parameter T, for three values of the offered load.

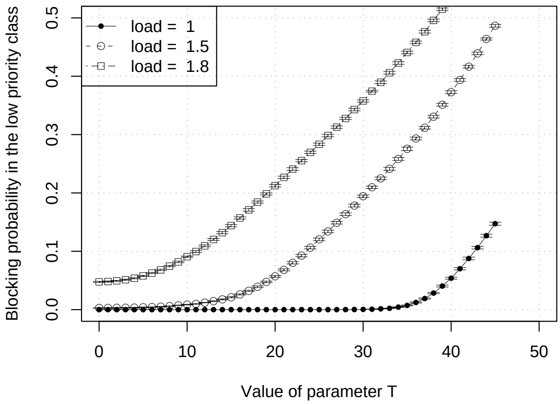

Figure A.15: First link capacity threshold strategy (flcap), NSF network, alternate routing with 3 paths in each class, 50% of traffic is the high priority traffic. The blocking probability in the low priority class as a function of parameter T, for three values of the offered load.

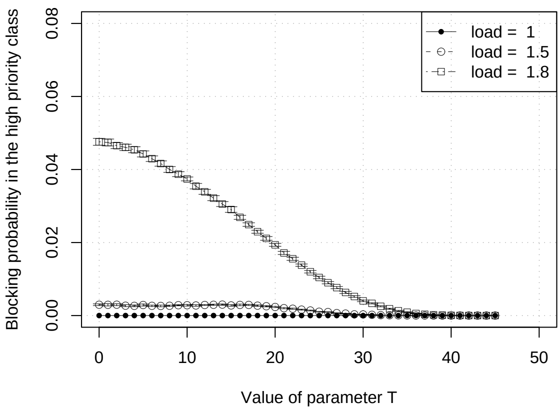

Figure A.16: First link capacity threshold strategy (flcap), NSF network, alternate routing with 3 paths in each class, 50% of traffic is the high priority traffic. The blocking probability in the high priority class as a function of parameter T, for three values of the offered load.

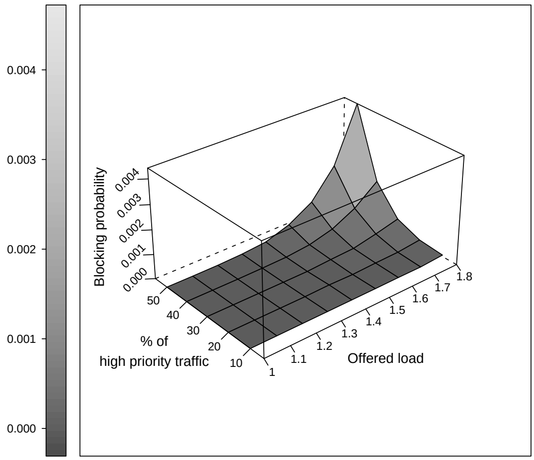

Figure A.17: First link capacity threshold strategy (flcap), NSF network, shortest path routing, $T = 30$. The blocking probability in the high priority class as a function of the offered load and the fraction of the high priority traffic.

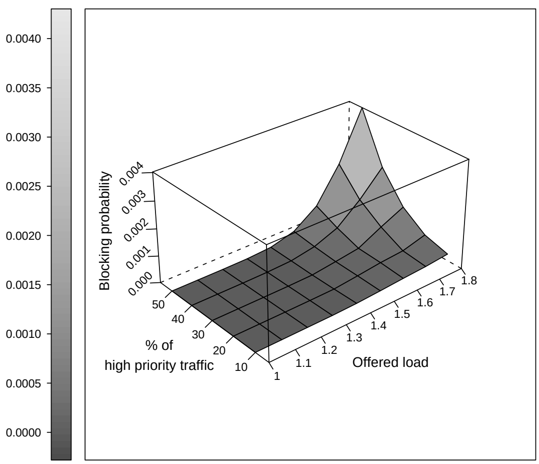

Figure A.18: First link capacity threshold strategy (flcap), NSF network, alternate routing with 3 paths in each class, $T = 30$. The blocking probability in the high priority class as a function of the offered load and the fraction of the high priority traffic.

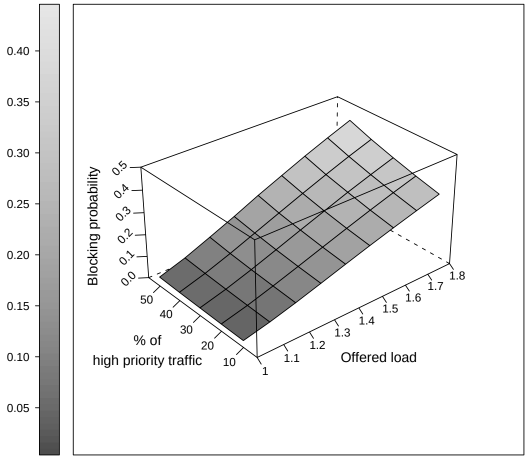

Figure A.19: First link capacity threshold strategy (flcap), NSF network, shortest path routing, $T = 30$. The blocking probability in the low priority class as a function of the offered load and the fraction of the high priority traffic.

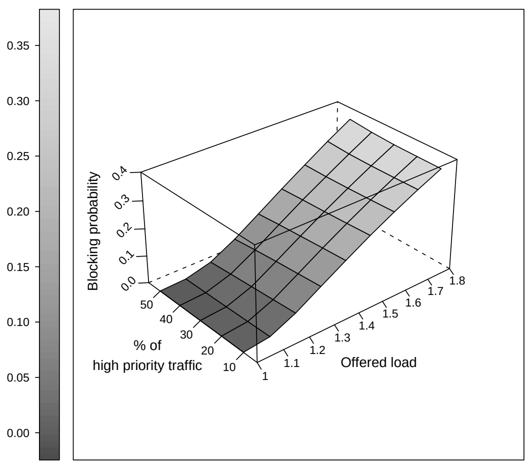

Figure A.20: First link capacity threshold strategy (flcap), NSF network, alternate routing with 3 paths in each class, $T = 30$. The blocking probability in the low priority class as a function of the offered load and the fraction of the high priority traffic.

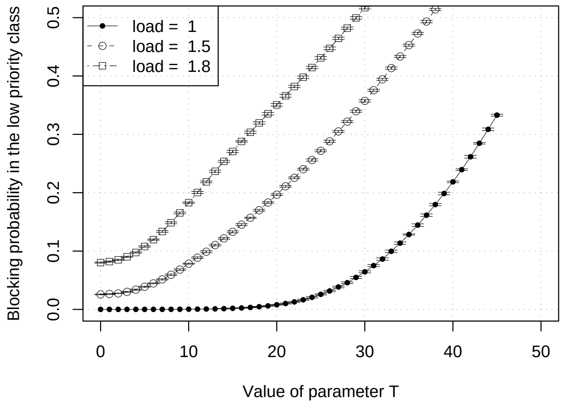

Figure A.21: Link capacity threshold strategy (lcap), NSF network, shortest path routing, 50% of traffic is the high priority traffic. The blocking probability in the low priority class as a function of parameter T, for three values of the offered load.

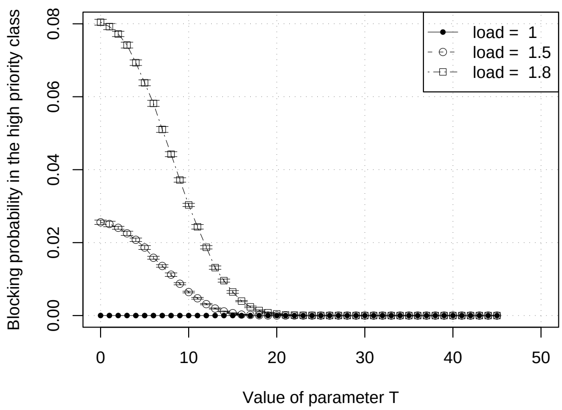

Figure A.22: Link capacity threshold strategy (lcap), NSF network, shortest path routing, 50% of traffic is the high priority traffic. The blocking probability in the high priority class as a function of parameter T, for three values of the offered load.

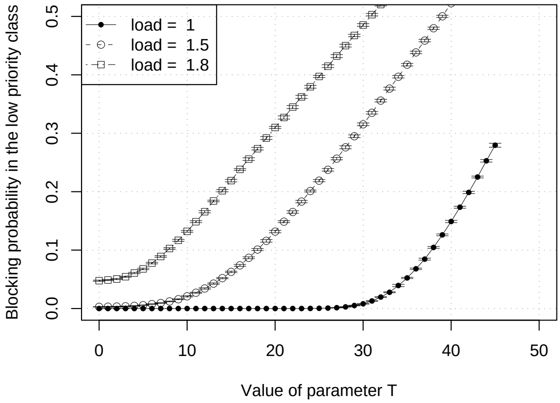

Figure A.23: Link capacity threshold strategy (lcap), NSF network, alternate routing with 3 paths in each class, 50% of traffic is the high priority traffic. The blocking probability in the low priority class as a function of parameter T, for three values of the offered load.

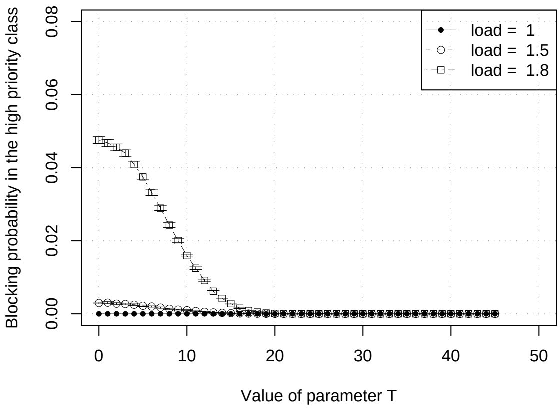

Figure A.24: Link capacity threshold strategy (lcap), NSF network, alternate routing with 3 paths in each class, 50% of traffic is the high priority traffic. The blocking probability in the high priority class as a function of parameter T, for three values of the offered load.

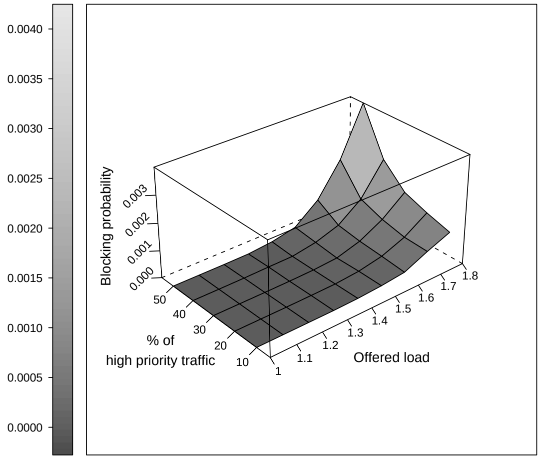

Figure A.25: Link capacity threshold strategy (lcap), NSF network, shortest path routing, $T = 16$. The blocking probability in the high priority class as a function of the offered load and the fraction of the high priority traffic.

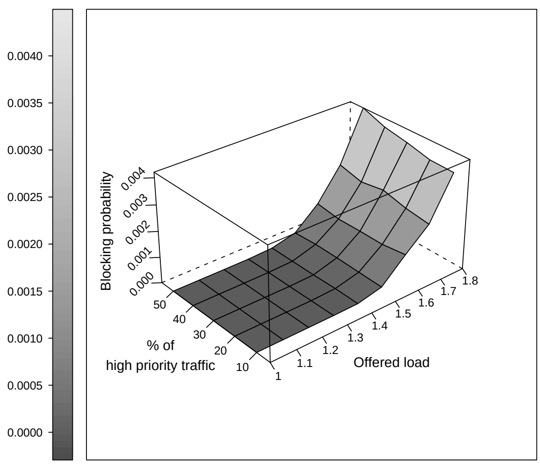

Figure A.26: Link capacity threshold strategy (lcap), NSF network, alternate routing with 3 paths in each class, $T = 14$. The blocking probability in the high priority class as a function of the offered load and the fraction of the high priority traffic.

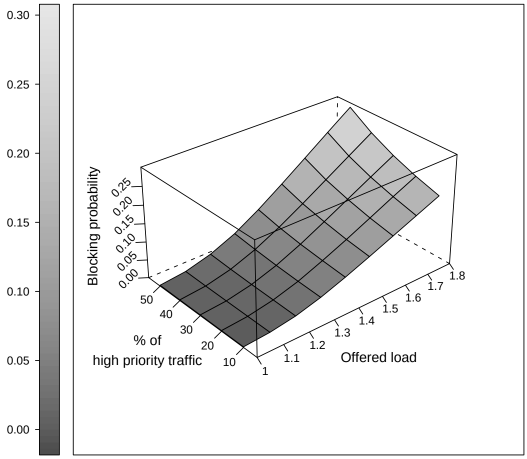

Figure A.27: Link capacity threshold strategy (lcap), NSF network, shortest path routing, $T = 16$. The blocking probability in the low priority class as a function of the offered load and the fraction of the high priority traffic.

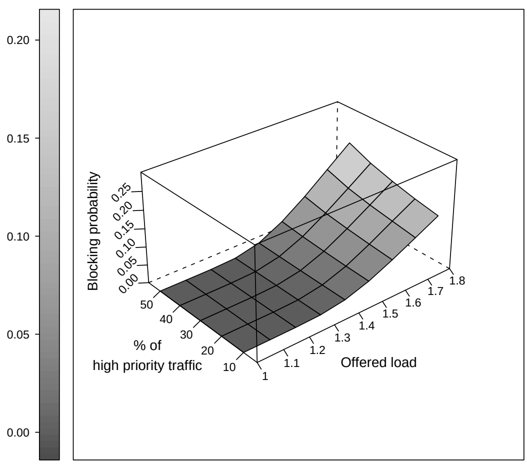

Figure A.28: Link capacity threshold strategy (lcap), NSF network, alternate routing with 3 paths in each class, $T = 14$. The blocking probability in the low priority class as a function of the offered load and the fraction of the high priority traffic.

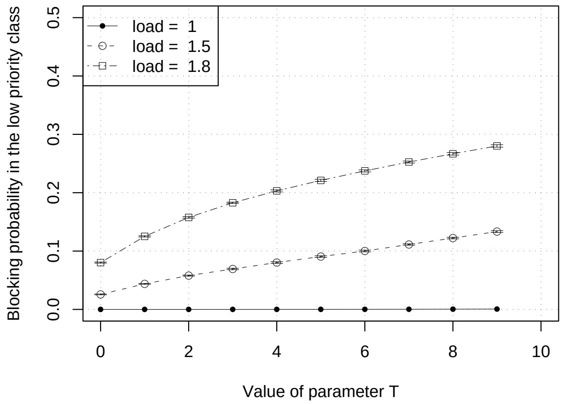

Figure A.29: Path capacity threshold strategy (pcap), NSF network, shortest path routing, 50% of traffic is the high priority traffic. The blocking probability in the low priority class as a function of parameter T, for three values of the offered load.

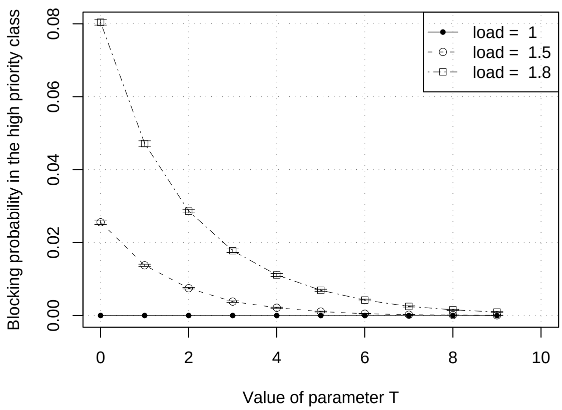

Figure A.30: Path capacity threshold strategy (pcap), NSF network, shortest path routing, 50% of traffic is the high priority traffic. The blocking probability in the high priority class as a function of parameter T, for three values of the offered load.

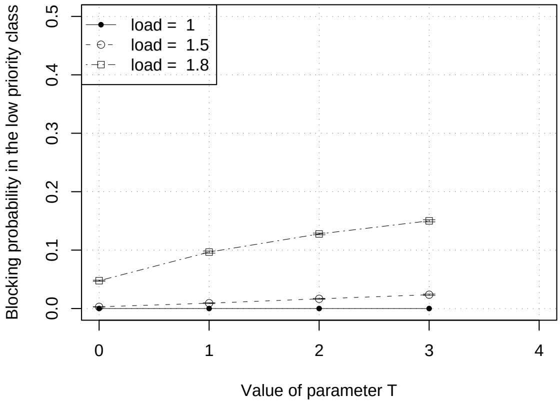

Figure A.31: Path capacity threshold strategy (pcap), NSF network, alternate routing with 3 paths in each class, 50% of traffic is the high priority traffic. The blocking probability in the low priority class as a function of parameter T, for three values of the offered load.

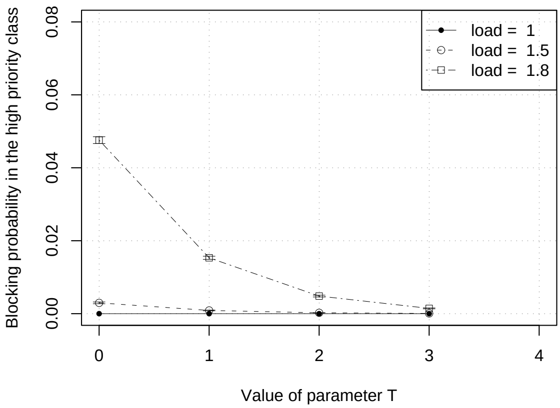

Figure A.32: Path capacity threshold strategy (pcap), NSF network, alternate routing with 3 paths in each class, 50% of traffic is the high priority traffic. The blocking probability in the high priority class as a function of parameter T, for three values of the offered load.

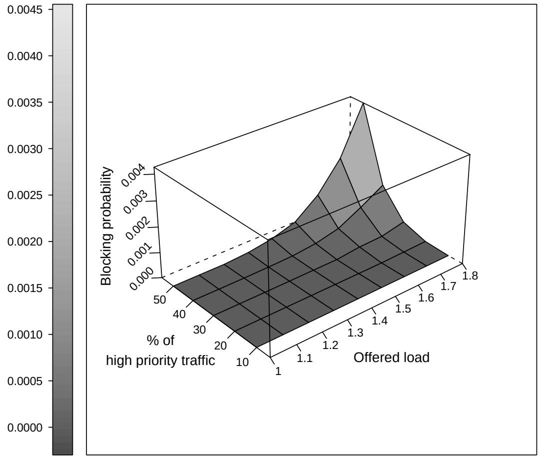

Figure A.33: Path capacity threshold strategy (pcap), NSF network, shortest path routing, $T = 6$. The blocking probability in the high priority class as a function of the offered load and the fraction of the high priority traffic.

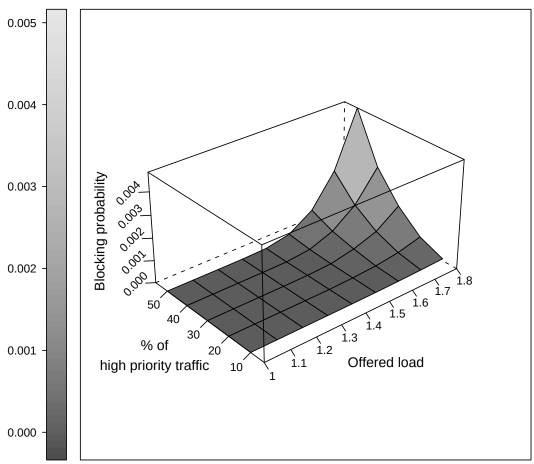

Figure A.34: Path capacity threshold strategy (pcap), NSF network, alternate routing with 3 paths in each class, $T = 2$. The blocking probability in the high priority class as a function of the offered load and the fraction of the high priority traffic.

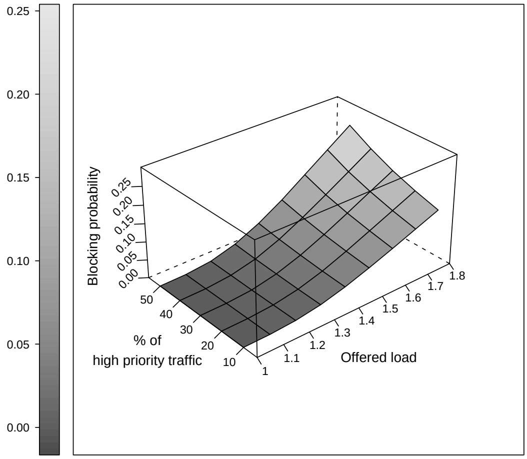

Figure A.35: Path capacity threshold strategy (pcap), NSF network, shortest path routing, $T = 6$. The blocking probability in the low priority class as a function of the offered load and the fraction of the high priority traffic.

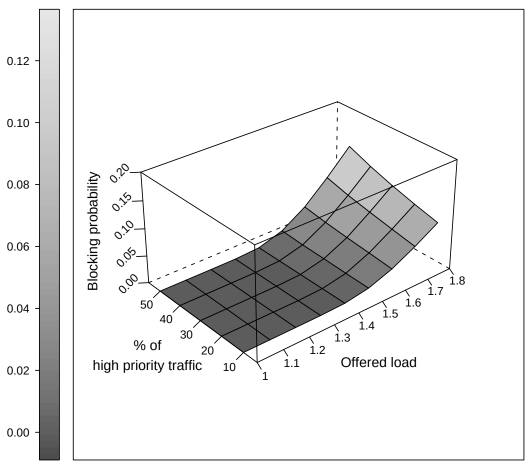

Figure A.36: Path capacity threshold strategy (pcap), NSF network, alternate routing with 3 paths in each class, $T = 2$. The blocking probability in the low priority class as a function of the offered load and the fraction of the high priority traffic.

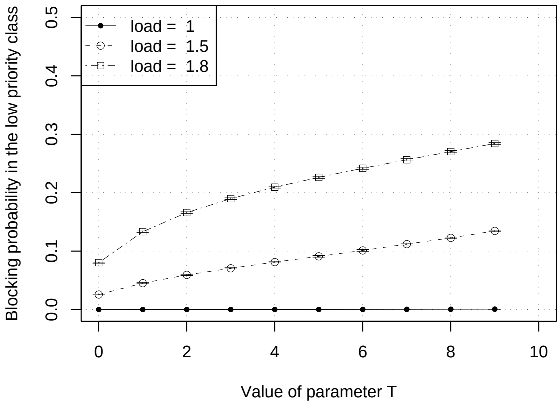

Figure A.37: Global capacity threshold strategy (gcap), NSF network, shortest path routing, 50% of traffic is the high priority traffic. The blocking probability in the low priority class as a function of parameter T, for three values of the offered load.

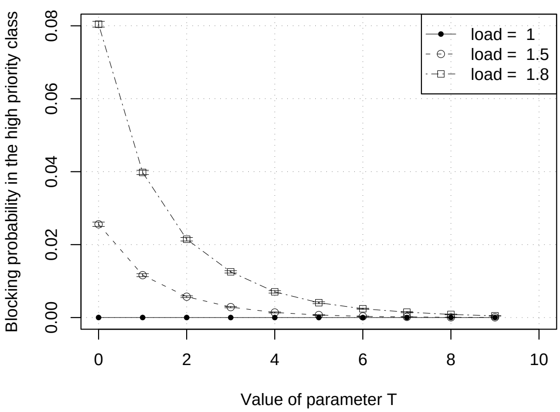

Figure A.38: Global capacity threshold strategy (gcap), NSF network, shortest path routing, 50% of traffic is the high priority traffic. The blocking probability in the high priority class as a function of parameter T, for three values of the offered load.

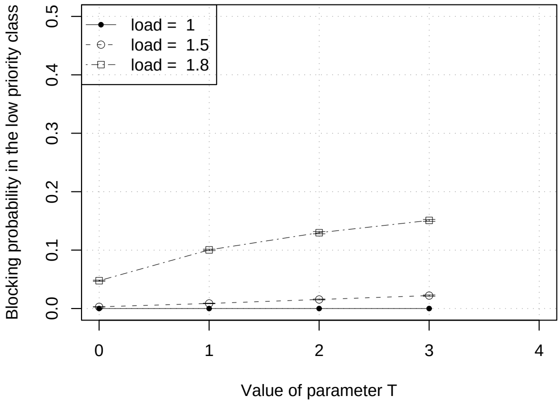

Figure A.39: Global capacity threshold strategy (gcap), NSF network, alternate routing with 3 paths in each class, 50% of traffic is the high priority traffic. The blocking probability in the low priority class as a function of parameter T, for three values of the offered load.

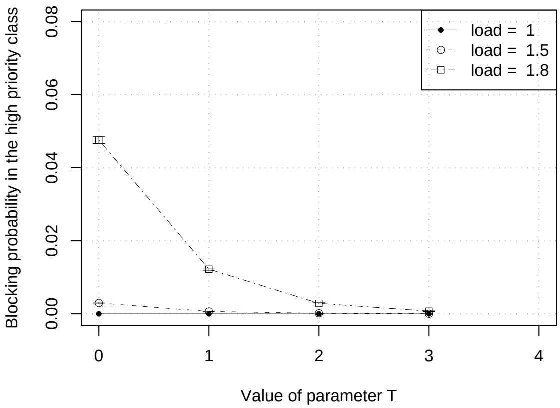

Figure A.40: Global capacity threshold strategy (gcap), NSF network, alternate routing with 3 paths in each class, 50% of traffic is the high priority traffic. The blocking probability in the high priority class as a function of parameter T, for three values of the offered load.

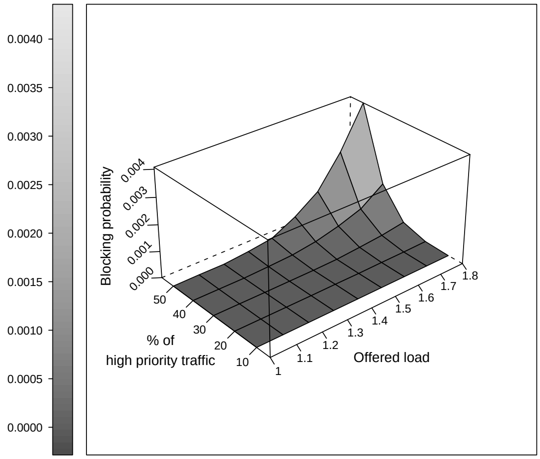

Figure A.41: Global capacity threshold strategy (gcap), NSF network, shortest path routing, $T = 6$. The blocking probability in the high priority class as a function of the offered load and the fraction of the high priority traffic.

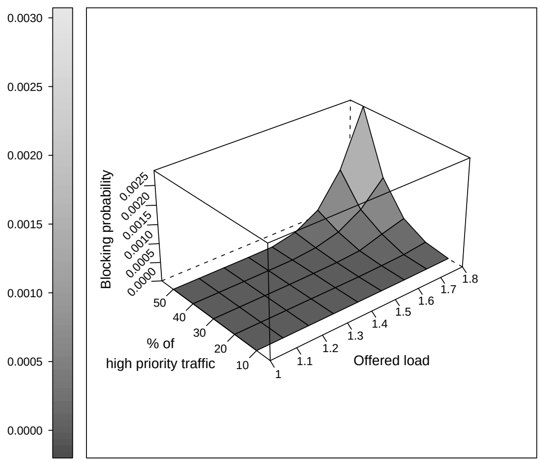

Figure A.42: Global capacity threshold strategy (gcap), NSF network, alternate routing with 3 paths in each class, $T = 2$. The blocking probability in the high priority class as a function of the offered load and the fraction of the high priority traffic.

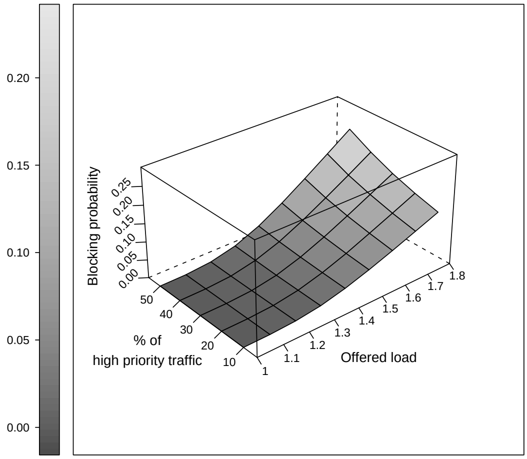

Figure A.43: Global capacity threshold strategy (gcap), NSF network, shortest path routing, $T = 6$. The blocking probability in the low priority class as a function of the offered load and the fraction of the high priority traffic.

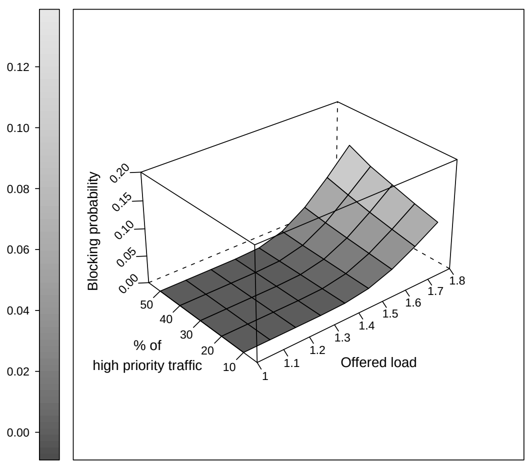

Figure A.44: Global capacity threshold strategy (gcap), NSF network, alternate routing with 3 paths in each class, $T = 2$. The blocking probability in the low priority class as a function of the offered load and the fraction of the high priority traffic.

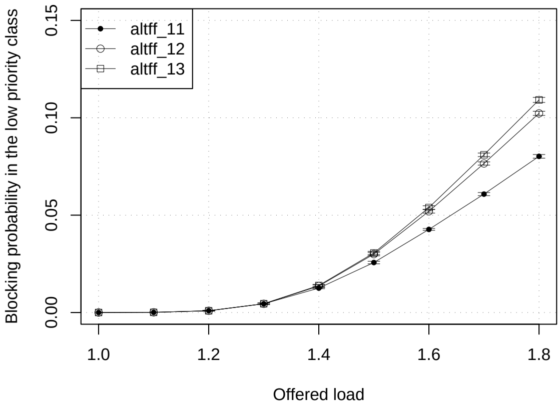

Figure A.45: Alternate paths strategy with 1 path in the low priority class and varying number of paths in the high priority class. NSF network with 50% of high priority traffic. The blocking probability in the low priority class as a function of the offered load.

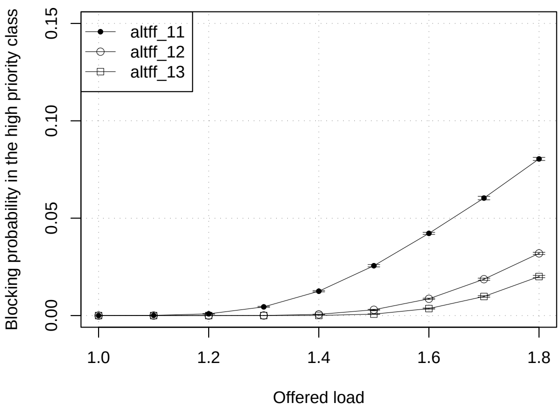

Figure A.46: Alternate paths strategy with 1 path in the low priority class and varying number of paths in the high priority class. NSF network with 50% of high priority traffic. The blocking probability in the high priority class as a function of the offered load.

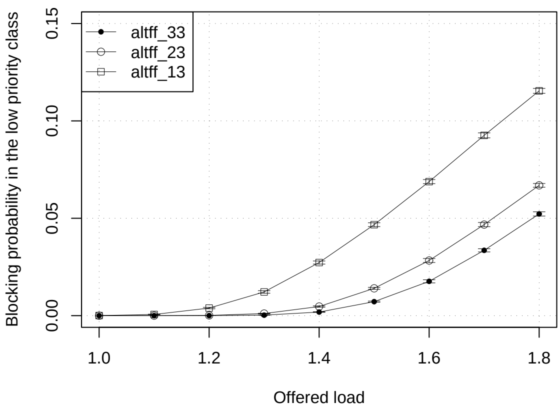

Figure A.47: Alternate paths strategy with 3 paths in the high priority class and varying number of paths in the low priority class. NSF network with 50% of high priority traffic. The blocking probability in the low priority class as a function of the offered load.

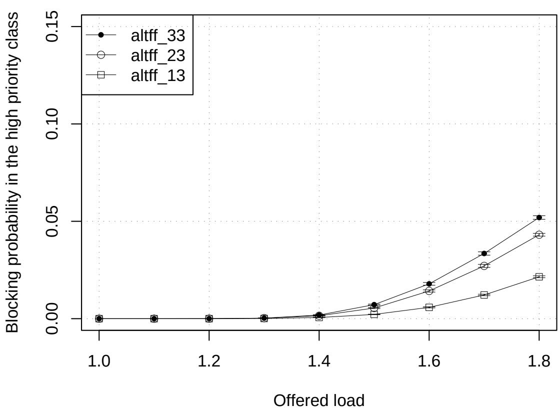

Figure A.48: Alternate paths strategy with 3 paths in the high priority class and varying number of paths in the low priority class. NSF network with 50% of high priority traffic. The blocking probability in the high priority class as a function of the offered load.

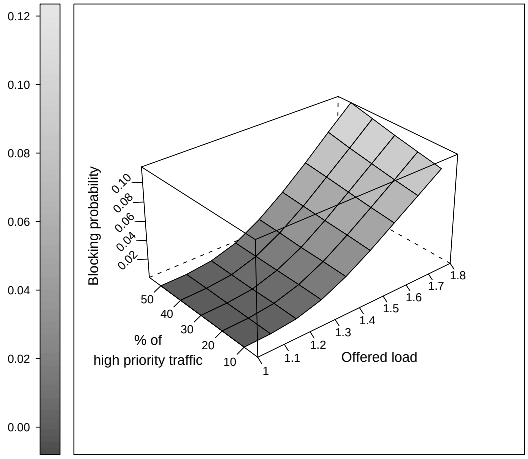

Figure A.49: Alternate paths strategy with 1 path in the low priority class and 3 paths in the high priority class. NSF network. The blocking probability in the low priority class as a function of the offered load and the fraction of the high priority traffic.

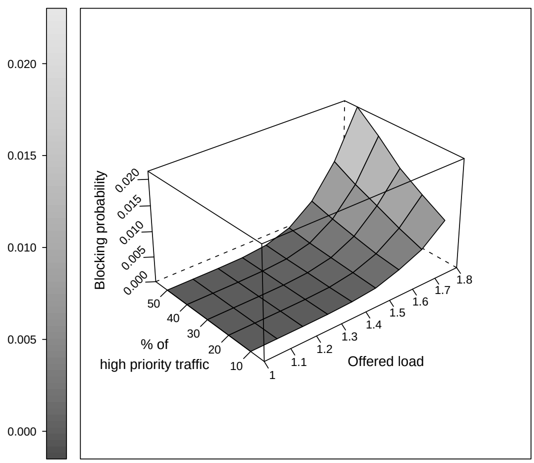

Figure A.50: Alternate paths strategy with 1 path in the low priority class and 3 paths in the high priority class. NSF network. The blocking probability in the high priority class as a function of the offered load and the fraction of the high priority traffic.

Bibliography

[1] A.V. Alfred, J.D. Ullman, J.E. Hopcroft, “Data Structures and Algorithms,” Addison-Wesley, Reading, Mass., USA, 1983. *(Cited on page 51.)*

[2] M. Ali, V. Leboucher, and D. Penninckx, “Intelligent lightpath selection schemes,” *European Conference on Optical Communication, ECOC ’02*, Copenhagen, Denmark, September 2002. *(Cited on page 22.)*

[3] P. Ashwood-Smith, ed., L. Berger, ed., “Generalized Multi-Protocol Label Switching (GMPLS) Signaling Constraint-based Routed Label Distribution Protocol (CR-LDP),” RFC 3472, January 2003. *(Cited on page 2.)*

[4] C. Assi, A. Shami, M.A. Ali, R. Kurtz, D. Guo, “Optical networking and real-time provisioning: an integrated vision for the next-generation Internet,” *IEEE Network*, Vol. 15, No. 4, pp. 36 – 45, July/August 2001. *(Cited on page 19.)*

[5] D. Awduche, “RSVP-TE: Extensions to RSVP for LSP Tunnels,” RFC 3209, December 2001. *(Cited on page 25.)*

[6] R. Bellman, “On a Routing Problem”, *Quarterly of Applied Mathematics*, Vol. 16, No. 1, pp. 87 – 90, 1958. *(Cited on page 17.)*

[7] L. Berger, “Generalized Multi-Protocol Label Switching (GMPLS) Signaling Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) Extensions,” RFC 3473, January 2003. *(Cited on pages 2 and 24.)*

[8] A. Birman, A. Kershenbaum, “Routing and wavelength assignment methods in single-hop all-optical networks with blocking,” *The Conference on Computer Communications, IEEE INFOCOM ’95*, Boston, Massachusetts, Vol. 1, pp. 431 – 438, 2 – 6 April, 1995. *(Cited on pages 6, 8, 14 and 20.)*

[9] G. E. P. Box and G. M. Jenkins, "Time Series Analysis: Forecasting and Control," Holden-Day, Oakland, USA, 2 edition, 1976. *(Cited on page 45.)*

[10] K. Chan, T. P. Yum, "Analysis of least congested path routing in WDM lightwave networks," *The Conference on Computer Communications, IEEE INFOCOM '94*, Toronto, Canada, Vol. 2, pp. 962 – 969, April 1994. *(Cited on page 9.)*

[11] Y. Chen, M. Hamdi, D. H. K. Tsang, "Proportional QoS over WDM Networks: Blocking Probability," *Sixth IEEE Symposium on Computers and Communications, ISCC '01*, Hammamet, Tunisia, pp. 210 – 215, 3 – 5 July 2001. *(Cited on page 21.)*

[12] I. Chlamtac, A. Farago, T. Zhang, "Lightpath (wavelength) routing in large WDM networks," *IEEE Journal on Selected Areas in Communications*, Vol. 14, Iss.5, pp. 909 – 913, June 1996. *(Cited on page 19.)*

[13] I. Chlamtac, A. Ganz, G. Karmi, "Lightpath Communications: An approach to high bandwidth optical WAN's," *IEEE Transactions on Communications*, Vol. 40, pp. 1171 – 1182, July 1992. *(Cited on pages 5, 6, 8 and 14.)*

[14] I. Chlamtac, A. Ganz, G. Karmi, "Purely optical networks for terabit communication," *The Conference on Computer Communications, IEEE INFOCOM '89*, Ottawa, Ontario, Canada, Vol. 3, pp. 887 – 896, 23 – 27 April 1989. *(Cited on pages 1, 14 and 49.)*

[15] P. Cholda, K. Wajda, A. Jajszczyk, J. Tapolcai, T. Cinkler, S. Bodamer, D. Colle, G. Ferraris, "Considerations about service differentiation using a combined QoS/QoR approach," *The 5th International Workshop on Design of Reliable Communication Networks, DRCN '05*, Island of Ischia, Italy, 16-19 October 2005. *(Cited on page 2.)*

[16] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, Vol. 1, pp. 269 – 271, 1959. *(Cited on page 8.)*

[17] T. Fabry-Asztalos, N.M. Bhide, K. M. Sivalingam, "Adaptive weight functions for shortest path routing algorithms for multi-wavelength optical WDM networks," *IEEE International Conference on Communications, ICC '00*, New Orleans, USA, pp. 1330 – 1334, 18 – 22 June 2000. *(Cited on page 12.)*

[18] A. Farrel, J.-P. Vasseur and J. Ash, "A Path Computation Element (PCE)-Based Architecture," RFC 4655, August 2006. *(Cited on pages 24, 46 and 47.)*

[19] F. Feng, X. Zheng, H. Zhang, "Performance Study of Distributed Wavelength Reservation Protocols within Both Single and Multi-Fiber WDM Networks,"

Photonic Network Communications, Vol. 6, Issue 2, pp. 95 – 103, September 2003. *(Cited on pages 14, 16 and 25.)*

[20] I. Fonseca, R. Almeida, M. Ribeiro, and H. Waldman, “Algorithms for FWM-aware routing and wavelength assignment,” *SBMO/IEEE MTT–S International Microwave and Optoelectronics Conference, IMOC ’03*, Foz do Iguacu, Brazil, Vol. 2, pp. 707 – 712, 20 – 23 September 2003. 20-23 September 2003 *(Cited on page 22.)*

[21] I. Fonseca, M. Ribeiro, R. Almeida, and H. Waldman, “Preserving global optical QoS in FWM impaired dynamic networks,” *Electronics Letters*, Vol. 40, No. 3, pp. 191 – 192, February 2004. *(Cited on page 22.)*

[22] H. Harai, M. Murata, H. Miyahara, “Performance of alternate rouing methods in all-optical switching networks,” *The Conference on Computer Communications, IEEE INFOCOM ’97*, Kobe, Japan, pp. 517 – 525, 7 – 12 April 1997. *(Cited on pages 9, 14, 15 and 17.)*

[23] Y. Huang, J.P. Heritage, B. Mukherjee, “Connection provisioning with transmission impairment consideration in optical WDM networks with high-speed channels,” *Journal of Lightwave Technology*, Vol. 23, No. 3, pp. 982 – 993, March 2005. *(Cited on page 22.)*

[24] ITU-T Rec. ITU-T G.8080/Y.1304, “Architecture for the automatically switched optical network (ASON)”. *(Cited on page 2.)*

[25] G. Jeong, E. Ayanoglu, “Comparison of wavelength-interchanging and wavelength selective cross-connects in multiwavelength all-optical networks,” *The Conference on Computer Communications, IEEE INFOCOM ’96*, San Francisco, CA, Vol. 1, pp. 156 – 163, 24 – 28 March 1996. *(Cited on page 16.)*

[26] J.P. Jue, G. Xiao, “An adaptive routing algorithm for wavelength-routed optical networks with a distributed control scheme,” *Ninth International Conference on Computer Communications and Networks, IEEE ICCCN ’00*, Las Vegas, Nevada, pp. 192 – 197, 16 – 18 October 2000. *(Cited on pages 13 and 14.)*

[27] A. Jukan, H.R. van As, “Resource allocation strategies with QoS-routing in optical networks,” *IEEE International Conference on Communications, ICC ’99*, Vancouver, British Columbia, Canada, Vol. 3, pp. 2048 – 2054, 6 – 10 June 1999. *(Cited on page 21.)*

[28] A. Jukan, H.R. van As, “Service-specific resource allocation in WDM networks with quality constraints,” *IEEE Journal on Selected Areas in Communications*, Vol. 18, No. 10, pp. 2051 – 2061, October 2000. *(Cited on page 21.)*

[29] A. Jukan, G. Franzl, "Path selection methods with multiple constraints in service-guaranteed WDM networks," *IEEE/ACM Transactions on Networking*, Vol. 12, No. 1, pp. 59 – 72, February 2004. *(Cited on pages 2 and 21.)*

[30] E. Karasan, E. Ayanoglu, "Effects of wavelength routing and selection algorithms on wavelength conversion gain in WDM optical networks," *IEEE/ACM Transactions on Networking*, Vol. 6, pp. 186 – 196, April 1998. *(Cited on pages 14, 15 and 18.)*

[31] H. Kogelnik, "High-capacity optical communications: personal recollections," *IEEE Journal of Selected Topics in Quantum Electronics*, Vol. 6, No. 6, pp. 1279 – 1286, November/December 2000. *(Cited on page 1.)*

[32] M. Kovacevic and A. Acampora, "Benefits of Wavelength Translation in All-Optical Clear-Channel Networks," *IEEE Journal on Selected Areas in Communications*, Vol. 14, pp. 868 – 880, June 1996. *(Cited on page 14.)*

[33] A. Kuchar, "Achievements of COST 266 Action and further prospects in research of advanced infrastructure for photonic networks," *The 6th International Conference on Transparent Optical Networks, IEEE ICTON '04*, Wroclaw, Poland, Vol. 1, pp. 37 – 42, 4 – 8 July 2004. *(Cited on page 50.)*

[34] K. C. Lee, V. O. K. Li, "A wavelength-convertible optical network," *IEEE/OSA Journal of Lightwave Technology*, Vol. 11, No. 5 – 6, pp. 962 – 970, May-June 1993. *(Cited on pages 6 and 18.)*

[35] L. Li, A.K. Somani, "Dynamic wavelength routing using congestion and neighborhood information," *IEEE/ACM Transactions on Networking*, Vol. 7, Iss.5, pp. 779 – 786, 1999. *(Cited on page 10.)*

[36] W. Liang and X. Shen, "Improved lightpath (wavelength) routing in large WDM networks," *IEEE Transactions on Communications*, Vol. COM-48, pp. 1571 – 1579, September 2000. *(Cited on page 19.)*

[37] M.A.C. Lima, A.C. Cesar, A.F.R. Araujo, "Optical network optimization with transmission impairments based on genetic algorithm," *SBMO/IEEE MTT–S Microwave and Optoelectronics Conference, IMOC '03*, Foz do Iguacu, Brazil, Vol. 1, pp. 361 – 365, 20 – 23 September 2003. *(Cited on page 22.)*

[38] G. M. Ljung and G. E. P. Box, "On a Measure of a Lack of Fit in Time Series Models," *Biometrika*, Vol. 65, pp. 297 – 303, 1978. *(Cited on page 44.)*

[39] T. H. Maiman, "Stimulated optical radiation in ruby masers," *Nature*, Vol. 187, pp. 493 – 494, 1960. *(Cited on page 1.)*

[40] E. Mannie, Ed. "Generalized Multi-Protocol Label Switching (GMPLS) Architecture," RFC 3945, October 2004. *(Cited on page 2.)*

[41] J. Martins-Filho, C. Bastos-Filho, E. Arantes, S. Oliveira, L. Coelho, J. de Oliveira, R. Dante, E. Fontana, and F. Nunes, "Novel routing algorithm for transparent optical networks based on noise figure and amplifier saturation," *SBMO/IEEE MTT–S Microwave and Optoelectronics Conference, IMOC '03*, Foz do Iguacu, Brazil, Vol. 2, pp. 919 – 923, 20 – 23 September 2003. *(Cited on page 22.)*

[42] M. Matsumoto and T. Nishimura, "Mersenne Twister: A 623-dimensionally equidistributed uniform pseudo-random number generator," *ACM Transactions on Modeling and Computer Simulation*, Vol. 8, No. 1, pp. 3 – 30, January 1998. *(Cited on page 48.)*

[43] J. T. McClave, F. H. Dietrich, "Statistics,", Collier Macmillan Publishers, London, 1998. *(Cited on page 42.)*

[44] A. Mokhtar, M. Azizoglu, "Adaptive wavelength routing in all-optical networks," *IEEE/ACM Transactions on Networking*, Vol. 6, pp. 197 – 206, April 1998. *(Cited on pages 6, 7, 8, 12, 14, 15 and 17.)*

[45] T. K. Nayak, K. N. Sivarajan, "A New Approach to Dimensioning Optical Networks," *IEEE Journal on Selected Areas in Communications*, Vol. 20, No. 1, January 2002. *(Cited on page 21.)*

[46] W. Van Parys, B. Van Caenegem, P. Demeester, "Reduction of Blocking in Arbitrary Meshed Wdm Networks through a Biased Routing Approach," *Optical Fiber Communication Conference and Exhibit, OFC '98*, San Jose, CA, USA, pp. 94, 22 – 27 February 1998. *(Cited on page 8.)*

[47] K. Pawlikowski, H.D.J. Jeong, J.S.R. Lee, "On credibility of simulation studies of telecommunication networks ," *IEEE Communications Magazine*, Vol. 40, No. 1, pp. 132 – 139, Jan 2002. *(Cited on page 48.)*

[48] K. Pawlikowski, "Steady-state simulation of queuing processes: a survey of problems and solutions," *ACM Computing Surveys*, pp. 123 – 170, June 1990. *(Cited on page 43.)*

[49] M. Pióro, "Design Methods for Non-hierarchical Circuit Switched Networks with Advanced Routing," Wydawnictwa Politechniki Warszawskiej, Warszawa, 1989. *(Cited on page 9.)*

[50] M. Pióro, D. Medhi, "Routing, Flow, and Capacity Design in Communication and Computer Networks," Morgan Kaufmann Publishers, San Francisco CA, 2004. *(Cited on page 5.)*

[51] Y. Pointurier, M. Brandt-Pearce, T. Deng, S. Subramaniam, "Fair QoS-Aware Adaptive Routing and Wavelength Assignment in All-Optical Networks," *IEEE International Conference on Communications, ICC '06*, Istanbul, Turkey, Vol. 6, pp. 2433 – 2438, 11 – 15 June 2006. *(Cited on page 22.)*

[52] B. Ramamurthy, D. Datta, H. Feng, J. Heritage, and B. Mukherjee, "Impact of transmission impairments on the teletraffic performance of wavelength-routed optical networks," *Journal of Lightwave Technology*, Vol. 17, No. 10, pp. 1713 – 1723, October 1999. *(Cited on page 21.)*

[53] R. Ramamurthy, B. Mukherjee, "Fixed-alternate routing and wavelength conversion in wavelength-routed optical networks," *IEEE/ACM Transactions on Networking*, Vol. 10, No. 3, pp. 351 – 367, Jun 2002. *(Cited on pages 9 and 21.)*

[54] R. Ramaswami, K. Sivarajan, "Design of Logical Topologies for Wavelength-Routed Optical Networks," *IEEE Journal on Selected Areas in Communications*, Vol. 14, pp. 840 – 851, June 1996. *(Cited on page 50.)*

[55] A.L. Schawlow, and C.H. Townes, "Infrared and Optical Masers," *Physical Review*, Vol. 112, Issue 6, pp. 1940 – 1949, 1958. *(Cited on page 1.)*

[56] L. Shen, B. Ramamurthy, "Centralized vs. distributed connection management schemes under different traffic patterns in wavelength-convertible optical networks," *IEEE International Conference on Communications, ICC '02*, New York, NY, USA, Vol. 5, pp. 2712 – 2716, 28 Apr – 2 May 2002. *(Cited on page 47.)*

[57] J. Spaeth, "Dynamic routing and resource allocation in WDM transport networks," *Computer Networks*, Vol. 32, pp. 519 – 538, May 2000. *(Cited on pages 6, 9 and 10.)*

[58] J. Spaeth, "Resource allocation for dynamic routing in WDM networks," *Proc. SPIE*, Vol. 3843, p. 235 – 246, August 1999 *(Cited on pages 9 and 10.)*

[59] S. Subramaniam,R.A. Barry, "Wavelength assignment in fixed routing WDM networks," *IEEE International Conference on Communications, ICC '97*, Montreal, Canada, Vol. 1, pp. 406 – 410, 8 – 12 June 1997. *(Cited on pages 8, 14, 15, 16 and 49.)*

[60] J. W. Suurballe and R. E. Tarjan, "A quick method for finding shortest pairs of disjoint paths," *Networks*, Vol. 14, No. 2, pp. 325 – 336, 1984. *(Cited on page 10.)*

[61] A. Szymanski, A. Lason, J. Rzasa and A. Jajszczyk, "Grade-of-Service-based Routing in Optical Networks" *IEEE Communications Magazine*, Vol. 45, No. 2, pp. 82 – 87, February 2007. *(Cited on pages 3, 4, 23, 25, 32 and 38.)*

[62] A. Szymanski, A. Lason, J. Rzasa and A. Jajszczyk, "Performance Evaluation of the Grade-of-Service-based Routing Strategies for Optical Networks" *IEEE International Conference on Communications, ICC '08*, Beijing, China, pp. 5252 – 5257, 19 – 23 May 2008. *(Cited on pages 3 and 4.)*

[63] A. Szymanski, A. Lason, J. Rzasa and A. Jajszczyk, "Route Management Strategies for Grade of Service Differentiation in Optical Networks," *IEEE International Conference on Communications, ICC '06*, Istanbul, Turkey, pp. 2453 – 2458, 11 – 15 June 2006. *(Cited on pages 3, 4, 25 and 32.)*

[64] A. Szymański, A. Lasoń, J. Rząsa, A. Jajszczyk, "Strategie zarządzania doborem trasy umożliwiające różnicowanie jakości obsługi w przełączanych sieciach optycznych", *Przegląd Telekomunikacyjny, Wiadomości Telekomunikacyjne*, Vol. 80, Iss. 2–3, pp. 64–67, 2007. *(Cited on pages 3 and 4.)*

[65] T. Tachibana, S. Kasahara, "QoS-Guaranteed Wavelength Allocation for WDM Networks with Limited-Range Wavelength Conversion," *IEICE Transactions on Communications*, Vol. E87-B, No. 6, pp. 1439 – 1450, June 2004. *(Cited on pages 21 and 26.)*

[66] I. Tomkos, D. Vogiatzis, C. Mas, I. Zacharopoulos, A. Tzanakaki, and E. Varvarigos, "Performance engineering of metropolitan area optical networks through impairment constraint routing," *IEEE Communications Magazine*, Vol. 42, No. 8, pp. S40 - S47, August 2004. *(Cited on page 21.)*

[67] J. Tyszer, "Object-Oriented Computer Simulation of Discrete-Event Systems," Kluwer Academic Publishers, Norwell, Mass., 1999. *(Cited on pages 42, 43 and 44.)*

[68] A. Varga, "Using the OMNeT++ discrete event simulation system in education," *IEEE Transactions on Education*, Vol. 42, No. 4, pp. 11, November 1999. *(Cited on page 47.)*

[69] "The BOOST C++ libraries," http://www.boost.org (Online, last visited 18.11.2008). *(Cited on pages 48 and 49.)*

[70] "PostgreSQL: The world's most advanced open source database, " http://www.postgresql.org (Online, last visited 18.11.2008). *(Cited on page 47.)*

[71] "The R Project for Statistical Computing," http://www.r-project.org (Online, last visited 18.11.2008). *(Cited on page 47.)*

[72] S. Xu, L. Li, and S. Wang, "Dynamic routing and assignment of wavelength algorithms in multifiber wavelength division multiplexing networks," *IEEE Journal on Selected Areas in Communications*, Vol. 18, pp. 2130 – 2137, October 2000. *(Cited on page 19.)*

[73] Y. Yoo, S. Ahn, C. S. Kim, "Adaptive Routing Considering the Number of Available Wavelengths in WDM Networks," *IEEE Journal on Selected Areas in Communications*, Vol. 21, No. 8, October 2003. *(Cited on pages 6, 11 and 12.)*

[74] H. Zang, J.P. Jue, B. Mukherjee, "A Review of Routing and Wavelength Assignment Approaches for Wavelength-Routed Optical WDM Networks," *Optical Networks Magazine*, Vol. 1 No. 1, pp. 47 – 60, Janury 2000. *(Cited on pages 5, 14, 15, 16, 17 and 49.)*

[75] H. Zang, J. P. Jue, L. Sahasrabuddhe, R. Ramamurthy, and B. Mukherjee, "Dynamic lightpath establishment in wavelength-routed WDM networks," *IEEE Communications Magazine*, Vol. 39, No. 9, pp. 100 – 108, September 2001. *(Cited on page 5.)*

[76] X. Zhang, C. Qiao, "Wavelength assignment for dynamic traffic in multi-fiber WDM networks," *The 7th International Conference on Computer Communications and Networks,* Lafayette, LA, USA, pp. 479 – 485, 12 – 15 October 1998. *(Cited on page 16.)*

[77] B. Zhou, H.T. Mouftah, "Adaptive least loaded routing for multi-fiber WDM networks using approximate congestion information," *IEEE International Conference on Communications, ICC '02*, New York, NY, USA, Vol. 5, pp. 2745 – 2749, 28 Apr – 2 May 2002. *(Cited on page 10.)*