F034115

PHOSPHORUS

Lambda User Controlled Infrastructure for European Research

Integrated Project

Strategic objective:
Research Networking Testbeds

Deliverable reference number: D5.8

Resilient Grid Networks

Due date of deliverable: 2008-12-31
Actual submission date: 2008/12/31
Document code: Phosphorus-WP5-D5.8

Start date of project:
October 1, 2006

Duration:
30 Months

Organisation name of lead contractor for this deliverable: **University of Leeds (ULEEDS)**

Project co-funded by the European Commission within the Sixth Framework Programme (2002-2006)		
Dissemination Level		
PU	Public	
PP	Restricted to other programme participants (including the Commission	
RE	Restricted to a group specified by the consortium (including the	
CO	Confidential, only for members of the consortium (including the Commission Services)	

Abstract

As the use of Grid services grows, the provisioning of resilient services should be considered as a major aspect in the design of Grid networks. Characteristics of these systems, such as high heterogeneity, complexity and scalability, create many technical challenges in this respect. This deliverable deals with several aspects of providing resiliency in the context of Grid environments. Fault-tolerance schemes applicable to both Grid computational and network resources are presented. Novel approaches for supporting resilient network design and resilient traffic engineering against network resource failures are proposed and evaluated through simulation. Furthermore, novel scheduling approaches that incorporate different fault-tolerance schemes against Grid resource failures are proposed. Finally, a joint resilience study for addressing both network and Grid resilience resources is presented.

List of Contributors

Kostas Katrinis	AIT
George Markidis	AIT
Kostas Georgakilas	AIT
Anna Tzanakaki	AIT
Emmanuel Varvarigos	CTI
Panagiotis Kokkinos	CTI
Konstantinos Manousakis	CTI
Konstantinos Christodoulopoulos	CTI
Jens Buysse	IBBT
Chris Develder	IBBT
Marc De Leenheer	IBBT
Maria Chtepen	IBBT
Taisir El-Gorashi	ULeeds
Jafaar Elmirghani	ULeeds

Table of Contents

List of Figures

List of Tables

0 Executive Summary

Providing fault-tolerance in a distributed, heterogeneous environment such as the Grid environment, while optimizing resource utilization and job execution times, is a challenging task. To accomplish this, failures of both network and Grid resources (computational/storage) should be considered. This deliverable presents the outcome of collaborative research, addressing various problems associated with providing resilience in the context of Grid networks. In Grid computing, fault-tolerant schemes can be broadly classified as network-based fault tolerant schemes and Grid-resource fault tolerant schemes. Network-based fault-tolerant schemes overcome network components failures and QoS degradation using resilient schemes implemented in the network infrastructure. Grid resource failures are addressed by the Grid-resource fault tolerant schemes implemented at the application or middleware level. Joint schemes addressing both network and Grid-resource failures are also considered.

Fault-tolerance schemes against network failures are addressed in Sections 2 and 3. In Section 2, resilient network design is considered by proposing offline resilient routing and wavelength assignment (RWA) algorithms where lightpaths used by all connection requests are jointly optimized. Two resilient RWA algorithms are presented, which are based on linear program (LP)-relaxation formulations and perform joint minimization of the number of wavelengths utilized by the primary and backup paths under dedicated path 1+1 protection. Due to the LP formulation these algorithms can scale to large networks and traffic loads. Physical impairment-aware resilient network design is also investigated and addressed by proposing an algorithm that is based on an integer linear programming (ILP) formulation.

In Section 3, resilient traffic engineering is investigated by introducing online RWA algorithms. Dual link failures are investigated by quantifying the extent of the catastrophic consequences of such failures in networks that have been engineered to handle only single link failures. Backup paths are highly susceptible to physical layer impairments as they are commonly longer than the primary paths. To overcome this issue, novel online RWA algorithms that jointly address resilience and physical layer performance are presented. As providing a high level of resiliency to all connections in the network is expensive and tends not to scale well, providing different levels of resilience to different traffic types in accordance with their requirements is a vital issue in Grid environments. Various RWA schemes for provisioning lightpaths with disparate protection requirements in a dynamically provisioned WDM network are investigated with the aim to enhance the backup resource utilization and improve the network performance. The problem of efficiently grooming low-speed connections while satisfying their resilience requirements is investigated by considering two schemes that explore different ways of grooming primary and backup paths: Differentiated Resilience at Lightpath (DRAL) level and Differentiated Resilience at Connection (DRAC) level. Simulation

results compare the performance of different traffic classes with different traffic requirements under DARL and DRAC. Simulation results show that DRAL trades the bandwidth efficiency in routing each connection request for the savings in grooming ports usage while DRAC is highly sensitive to changes in the number of grooming ports. Furthermore, differentiated resilience is studied in the context of MPLS networks, where a resilience scheme that allows anycast flows to survive any link or server failure is proposed. Rerouting of the lower traffic class is implemented to reduce the blocking probability experienced by higher traffic classes. Simulations examined the blocking probability under different number of servers and find that increasing the number of servers decreases the blocking probability; however, this decrease tends to get smaller as the number of server increase.

Grid-resource fault tolerant schemes for Grid resources are the focus of Section 4. Two techniques are often applied to introduce fault-tolerance against computational/storage resource failures: job checkpointing and job replication. It is shown that neither of these techniques in their pure form is able to cope with unexpected load and failure conditions in a Grid environment. Therefore, several algorithms are proposed that dynamically adapt their parameters based on history statistics and current status of the Grid environment. Furthermore, a novel hybrid scheduling approach is introduced to enable dynamic alternation between the two techniques, depending on the system load. The case study results show that (i) checkpointing overhead can be successfully minimized by dynamically adapting checkpointing frequency; (ii) adaptive replication can even lower overhead for low and variable loads; (iii) the hybrid approach successfully combines the best of both over a broad load range. Another resource related problem in Section 4 considers data consolidation (DC) and resiliency issues. DC arises when a task requests for its execution datasets that are stored in more than one storage sites. Data replication resilience features are combined with a number of DC techniques in order to provide fault-tolerance. We show that the efficiency with which the DC schemes handle the network congestion caused by the applied resiliency techniques strongly affects the number of tasks that are successfully scheduled.

In Section 5, a resilience scheme that jointly addresses network and Grid resilience is considered. Computational resource resilience schemes are integrated into network protection, by implementing the anycast principle that allows relocation of jobs to other servers. Different network protection scenarios are compared by conducting an ILP study to compute the routes to other servers where jobs are relocated. The results confirm that shared protection schemes require considerably less resources than dedicated, but the amount of network capacity required can even be further lowered by providing backup paths to alternative destinations (i.e. relocation). In the considered case study, this comes at the price of a small increase of required Grid server capacity.

1 Introduction to Grid Resilience

Grid computing is emerging as a promising computing paradigm for connecting distributed computational and storage resources via heterogeneous, autonomously managed environments. As the use of Grid services grows in popularity, provisioning of resilient services should be considered as a major aspect in the design of Grid networks.

In traditional networks, resiliency supports the ability of overcoming various kinds of failures by introducing certain aspects to the network design and traffic engineering.

In a Grid environment a job execution might be affected by possible faults in the involved network or among Grid resources. This often leads to many significant consequences, including delay or failure of time-critical executing jobs. Building a fault-tolerant Grid system involves resolving a number of significant technical challenges. A Grid environment includes a high number of heterogeneous resources, which we usually categorize as network resources (links, switches, transmitter-receivers, etc) and Grid resources (computational and storage resources). These resources interact in order to create an execution platform for different tasks. The complexity and distribution of resources are major obstacles to the development of a resilient Grid. This has to do with the fact that resources in a Grid environment are more prone to failures than resources in traditional computing platforms; the high heterogeneity of resources in a such a Grid network creates many failure possibilities, including not only independent failures of each element but also those resulting from interactions between them [Medeiros03]. Additional difficulties arise in the process of diagnosing these failures in such an environment. Therefore, dealing with failure scenarios in a Grid network is challenging.

Following the above classification of resources in a, Grid network, fault-tolerant schemes can be broadly classified as network-based schemes and Grid resource-based schemes. Network-based fault-tolerant schemes can overcome both catastrophic and QoS degradation failures that are related to the networking resources. However, these schemes do not address computational resource failures. These are addressed by the Grid resource-based fault-tolerant schemes implemented at the application, middleware, or Operating System (OS) level. The applications that are related to Phosphorus project are mostly Data-Grid applications, that is applications in which the transferring of datasets is the crucial (or among the most crucial) part of a task. To this end, network resource resilience is particularly important for such Grid networks. In Section 1.1 we present a classification of the network resource resilience method and then in Section 1.2 we consider resiliency issues related to Grid resources (computation and storage resources).

.

1.1 Network Resources Resilience Methods Classification

In general, a network is referred to as survivable if it provides some ability of recovering operations that are disrupted by one or multiple catastrophic failures at any of its components. Phosphorus project focuses on lambda Grids that is Grid networks that are based on optical technology. Thus, in the following paragraphs we will focus our study on optical network resiliency. Typically, in optical networks failures include link (fiber) interruptions or optical switch/router failures [Ou05] [Grover03]. To this end, in an optical Grid network we need to design effective methods to recover from single or multiple failures of these kinds.

A task in a Grid network may consist of a number of subtasks that can include the transferring of data from one resource to another, the execution of a program at a computation resource, etc. The objective of the protection and restoration schemes is to reroute/re-schedule the affected task related traffic, using any form of redundancy provided by the network. Even though failures cannot be avoided, quick detection, identification, and restoration make the Grid network more reliable and increase the end-user confidence.

1.1.1 Protection vs. Restoration

There are two types of fault-recovery mechanisms: protection and restoration. Protection is referred to a system involving 100% redundant hardware to switch traffic or jobs from failed to protection facilities. In terms of communication resources [Ou05][Grover03] this means that for every data transfer there is a dedicated alternate path that transfers exactly the same data set to the same or another specified destination. Restoration is a dynamic process that involves the identification of the failure and actions to circumvent it. For communication resources, restoration assumes a network based on reconfigurable switches (cross-connects in optical networks) and spare capacity, where a higher layer restoration algorithm, along with a suitable control plane, is used to reroute traffic around failures.

In general, dynamic restoration methods are more efficient, because they do not allocate spare equipment in advance and they provide resilience against different kinds of failures. On the other hand, protection methods have faster recovery times, and can guarantee recovery from disrupted services. Note that, protection is static and cannot circumvent any type of failure that has not been taken into account in the designing process, while restoration is dynamic and thus more flexible to address different types of problems.

1.1.2 Number of Failures Recovered: Single Failure vs. Multiple

Part of recent work on survivability focuses on the recovery from a single failure, where one failure is repaired before another failure is assumed to occur in the network. This is known as the assumption of single failure scenario. Nevertheless, multiple, near simultaneous failures are also possible in a realistic

network, and appropriate recovery methods should be designed to accommodate this cases as well. Therefore, when considering multiple failures, resource availability depends intimately on the precise detail of the failures (locations, repair times, etc.), and on the volume of recovery resources that are allocated (i.e. protection or restoration, dedicated protection or shared). Moreover, there is the possibility that the failures will occur within a fixed zone range. This type of failure model is called group failure. Note that single or multiple failures can refer to any type of resource in a Grid network, i.e. links, switches, computation, storage resources, etc.

A number of papers have considered multiple failures. In [Schupke01] a framework for the recovery of multiple failures is defined. The framework distinguishes between a horizontal and a vertical approach according to the recovery scope of a single layer or of multiple layers, respectively. The horizontal approach includes network partitioning, pre-computed recovery (before first failure), re-computed recovery (after first failure) and re-restoration (after secondary failures). The vertical approach includes recovery at higher layer, recovery at lower layer and central reconfiguration. [Schupke03] considered the use of p-cycles at the presence of multiple failures. Link p-cycles protect single fiber cut failures. Multiple failures can be recovered, if each failure is on a different p-cycle. Multiple failures can also be recovered using cycle reconfiguration. Special multiple failures (also node failures) on a single p-cycle can be recovered by failure signaling, even if reconfiguration is ineffective. Finally, this paper presents the loss values in a single p-cycle depending on the working capacity of the protected links. In [Tacca06] an MPLS-TE scheme for rapid local fault detection and recovery in networks that may be hit by multiple concurrent failures was presented. In the scheme, the detection of a PFP (Probable Failure Pattern) is achieved via BFD (Bidirectional Forwarding Detection) sessions originating at the PLR (Point of Local Recovery). In [Chandak0] it was shown that recovery from a dual-link failure, using an extension of link protection for single link failure, results in a constraint, referred to as BLME constraint, whose satisfiability allows the network to recover from dual-link failures without the need for broadcasting the failure location to all nodes. [Fumagalli04] proposed a probabilistic approach to efficiently handling multi-failure scenarios in MPLS networks that make use of local recovery. The objective of the proposed approach is to provide a way to control the number of bypass tunnels that are required to handle multi-failure patterns, while monitoring the expected length and outage probability of the chosen bypass tunnels.

1.1.3 The Scope of Recovery Procedure: Local vs. Global recovery

This classification is driven by the process of restoration in data networks. In global recovery, the traffic is rerouted through a backup route (backup path or protection path) once a link failure occurs on the working path (primary path). The primary and backup path for a connection must be link disjoint so that no single link failure can affect both of these paths. In local recovery, the traffic is rerouted only around the failed link. While global recovery leads to efficient utilization of backup resources and lower end-to-end propagation delay for the recovered route, link recovery provides faster recovery switching time. Recently, sub-path recovery is proposed by dividing a primary path into a sequence of segments and protecting each segment separately or by dividing the whole network into different domains, and a path segment in one domain must be recovered by the resources in the same domain. Compared with path recovery, sub-path recovery can achieve high scalability and fast recovery times but sacrifices some of its efficiency.

1.1.4 Protection Issues

1.1.4.1 *The Allocation of Backup Resources in Protection Schemes: Dedicated vs. Shared*

Protection schemes can be dedicated or shared. In dedicated protection, sharing is not allowed between backup resources, while in shared protection, backup resources can be shared among different resilience demands. In data networks links and switches are the possible points of failure. Dedicated protection implies that sharing is not allowed between backup paths, while shared protection allows backup capacity to be shared on some paths as their protected segments (links, sub-paths, and paths) are mutually diverse or not in the same shared risk groups (SRGs) [Ou05] [Grover03]. Shared Risk Groups (SRGs) express the risk relationship that associates the devices under the assumption of a single failure. An SRG may consist of all the optical channels in a single fiber, all of the optical channels through all the fibers wrapped in the same cable.

In dedicated protection if traffic is transmitted simultaneously on both primary and backup paths from the source node to the destination node this scheme is referred to as 1+1 protection. If traffic is only transmitted on the primary path and the source and destination nodes both switch over to the backup path when the primary path is cut (link or switch failure), this scheme is referred to as 1:1 protection. 1+1 protection provides very fast recovery and does not require signaling between the two end nodes. In 1:1 protection, the backup path can be used to carry low-priority traffic during normal operation, and when the failure occurs high-priority traffic is switched to the backup path. Shared protection scheme can be extended to M:N protection where M primary paths may share N backup paths.

Different recovery schemes based on different classifications (protection/restoration, path/link/segment, dedicated/shared) have been studied in literature. [Ramamurthy99] examines different approaches to recover single-link failures in an optical network based on two basic survivability paradigms: path protection/restoration, and link protection/restoration. An Integer Linear Program is formulated to determine the capacity requirements for the above protection schemes for a static traffic demand. The numerical results obtained for a representative network topology and for random demands indicate that shared-path protection provides significant savings in capacity utilization over dedicated-path and shared-link protection schemes. On the other hand dedicated-path protection provides marginal savings in capacity utilization over shared-link protection. In [Ramamurthy99b], a model of protection switching times for the various protection schemes was formulated. The authors in [Sahasrabuddhe02] consider two fault management techniques in an IP-over-WDM network: protection at the WDM layer and restoration at the IP layer. These fault-management techniques were mathematically formulated and heuristics were developed in order to find efficient solutions in typical networks. The characteristics of these techniques (e.g., maximum guaranteed network capacity in the event of a fiber failure and the recovery time) were analyzed. [Shenai05] presents two hybrid survivability approaches that combine the positive effects of restoration with those of protection. The proposed approaches make use of available or collected network state information, such as link load, to identify critical links or segments in the network that are then proactively protected. The overall goal of the proposed approaches is to improve the restoration efficiency by providing a tradeoff between proactive protection and dynamic restoration. Experimental results show that under high loads, both the proposed approaches maintain a consistent restoration efficiency of at least 10%, or higher, when compared to the

basic restoration scheme. [Tapolcai08] provides a thorough study on Shared Segment Protection (SSP) under the GMPLS-based recovery framework, where an effective survivable routing algorithm for SSP is proposed. The tradeoff between the cost of resources and the restoration time was extensively studied by simulations under highly dynamic traffic. In [Limal99] an algorithm for link restoration of networks without wavelength translation was presented. The algorithm was studied in details and compared in terms of complexity against a classical path assignment algorithm. The work in [Wang02] compared path, sub-path (segment) and link restoration techniques for fault management in an IP-over-WDM network using GMPLS control signaling. Results showed that both sub-path and link restoration can provide faster restoration than path restoration; however, this comes at lower a restoration success rate. Authors in [Wang02] assert that this rate can be improved by employing specific traffic engineering extensions to LDP. [Ho04] focuses on the problem of dynamic survivable routing for segment shared protection (SSP) in mesh communication networks, namely by provisioning bandwidth guaranteed tunnels where a connection is settled by concatenating a series of protection domains, each of which contains a working and protection segment pair.

1.1.4.2 *Network Topology*

This classification is applicable to data networks [Grover03]. Protection schemes can be classified as ring protection and mesh protection. Ring protection methods include Automatic Protection Switching (APS) and Self-Healing Rings (SHR). Both ring protection and mesh protection can be further divided into two groups: path protection and link protection.

Three ring architectures have been widely deployed: two fiber unidirectional path-switched ring (UPSR), two-fiber bidirectional line-switched rings (BLSR/2) and four-fiber bidirectional line-switched rings (BLSR/4). In UPSR 1+1 path protection scheme is implemented. The protection scheme in a UPSR is easy to implement, requires no communication between the nodes, and provides fast failure recovery. However, this architecture is not capacity efficient since half of the capacity is devoted to protection purposes. A BLSRs allows protection bandwidth to be shared between spatially separated connections. BLSRs are more efficient than UPSRs in protecting distributed traffic pattern since the protection capacity in the ring is shared among all the connections. However BLSRs are more complex than UPSRs since BLSRs require signaling between the nodes for coordinating multiple protection mechanisms.

Networks based on a single physical ring are favored for their fast restoration. To achieve the restoration speed in an irregular mesh-based network topology, one solution is to cover the whole physical mesh network using multiple logical links (ring cover scheme). Another solution is to design a set of protection cycles (p-cycles) which cover all the links that need to be protected.

Resilient Packet Ring (RPR, IEEE 802.17) is designed with a protection mechanism aiming at restoring traffic on the ring in case of a link or node failure. In [Kvalbein04], RPR protection was evaluated with respect to service disruption, packet reordering and packet loss. Different error scenarios were simulated, with both steering and wrapping protection. The work in [A.Rahman06] focuses on the development of optical cross add and drop multiplexer (OXADM) to provide survivability through restoration against failure, such as a cable cut, in ring optical networks. Two types of restoration schemes were proposed to assure

uninterrupted data flows by means of linear/multiplex protection and ring protection. In [Gumaste05] a protection strategy for light-trail networks for ring topologies was proposed. The multi-point flow model associated with light-trail networks leads to a new set of problems in the area of protection and restoration. A hardware scheme to protect light-trails was proposed. In [Tsuboi03], a novel ring architecture is proposed that is suitable for asymmetric traffic and allows for protection using a single fiber ring.

1.1.5 Recovery Issues

Recovery mechanisms can be either centralized or distributed. In a centralized control system, failures are addressed by a single entity that performs the appropriate actions to restore the failures one-by-one. In this way, contention on the various resources is avoided. However, centralized control schemes may affect the restoration time, i.e. the average time needed to successfully circumvent the problem. In a distributed control system, there are distributed entities dedicated to restoring failures, e.g. the monitoring or resource broker system in a Grid network, the source switch of a connection in data networks, etc. When the failure is identified, the distributed restoration scheme assigns the failed tasks or the blocked connection to other available and working resources. However, since in a distributed architecture information about the utilization of the resources is not always up-to-date, this assignment may lead to contention. For example, in a data network, if the restoration is distributed, each interrupted connection can be restored following either a pre-computed route or a dynamically computed route. Since the connections are restored in a distributed manner, it is possible for more than one connection to attempt to allocate the same network link. Although such contentions can be resolved through restoration retries, the performance may be affected. Comparing these control mechanisms, a centralized restoration scheme may achieve better performance, since it has the global control and can target at the global optimization for the resources usage.

A large number of papers in literature have considered centralized recovery algorithms such as [Liu01], [Doverspike03], [Bhandari99],[Dunn94] and [Bouillet02]. However, a fewer number of papers have addressed distributed recovery algorithms. [Ramamurthuy01] compares centralized vs. distributed online provisioning approaches in optical networks. Primarily, it is found that the centralized provisioning approach does not scale as the size of the network increases, due to the large amount of information required to be maintained at the centralized management entity and the associated computational and signaling overhead. Also it is found that the distributed provisioning approach requires more capacity to achieve the same blocking as the centralized provisioning approach, and the blocking in the distributed provisioning approach can be reduced with retries. [Ramamurthy99b] proposes a distributed control protocol for path and link restoration, assuming a fully distributed control network. In [Komine90] a new distributed restoration algorithm based on message flooding is introduced. The algorithm is able of restoring multiple-link and node failures, using multi-destination flooding and path route monitoring. [Qiao02] introduces a novel framework, known as Distributed Partial Information Management (DPIM) where several major challenges in achieving efficient shared path protection under distributed control with only partial information are addressed. Among

the issues addressed in this distributed framework, the amount of partial information about existing active and backup paths maintained and exchanged is addressed, including a good estimate of the bandwidth needed by a candidate backup path and allocating it via distributed signaling. [Li03] proposes an efficient restoration path selection algorithm for restorable connections over shared bandwidth in a fully distributed MPLS/GMPLS architecture, known as Full Information Restoration. This algorithm uses signaling protocol extensions to distribute and collect additional link state information. It uses bandwidth most efficiently as it keeps accurate information about the amount of reserved bandwidth that must be reserved on each link in the network to restore any single link failure. A distributed scheme, known as Resource Aggregation for Fault-Tolerance (RAFT), was proposed by [Dovrolis98]. The distributed scheme implemented a Fault Management Table (FMT) on each link to keep track of all the traffic flows which their backup paths pass this link. The shared spare capacity can be calculated with this FMT.

1.1.6 Differentiated Network Resilience

Although offering a flat (equal) resilience service to all connection requests in a WDM network is the simplest approach in terms of implementation and pricing complexity, there are certain arguments indicating the inefficiency/inflexibility of a flat resilience scheme. Providing 100% guaranteed resilience to all types of traffic supported by existing and future networks may be unnecessary and wasteful in terms of resource utilization, resulting in cost inefficiencies. For example, some Grid applications (e.g., simulations), do not require the same level of resilience with real-time business transactions. Even for applications sharing inherently the same requirements in terms of fault-tolerance, the various service users may not be willing to be charged equally; instead, some users may choose to trade-off assured fault-tolerance for receiving a service discount.

It follows from the above that a more efficient resilience scheme capable of supporting a variety of applications would be a scheme that provides different level of network survivability to different traffic types in accordance with the respective Service Level Specifications (SLS) maximizing the network utilization [Zhang02]. Therefore in a Grid environment an important requirement will be to provide differentiated survivability services to different types of traffic enabling higher priority demands to exploit higher network availability [Fuma06][Pandi06].

1.1.6.1 *Related Work*

Network resilience can be defined as the capability of the network to provide continuous service in the presence of failures. Due to the large amount of traffic at the lightpath level, resilience mechanisms are highly critical and many schemes have been proposed to address this issue [Moh00]. It is possible to implement a single restoration algorithm that is able to be used “preemptively”, before a failure occurs as part of a predesigned protection method and dynamically after the occurrence of a failure not previously considered. The advantages offered by the pre-designed protection method compare with dynamic restoration are the shorter restoration times and the 100% restoration guarantee.

A classification of the pre-designed protection method is performed based on link or path protection schemes. In the link based method the failed link is replaced by a new path which is merged with the unaffected portion of the primary path, to constitute the backup path. This method constraints the choice of the backup paths and requires more spare resources than the path-based method [Iras98], which computes a complete end-to-end backup path from the source to the destination of the failed primary path. In the path-based method, wavelength channels on the backup path can be either dedicated or shared. If dedicated the wavelength channels assigned to a specific backup path cannot be assigned to other backup paths whereas in the shared method, backup paths can share wavelength channels under the single link failure assumption, if their primary paths are link-disjoint. This is known as backup multiplexing and provides improved resource utilization. Specifically in [Rama01], it was shown that the total resource requirement for the dedicated backup method is 260-265% of the requirement without lightpath protection, and it can be reduced to 186-195% by considering backup multiplexing.

1.1.7 Multi-layer and Multi-domain Network Resilience

1.1.7.1 *Single Layer vs. Multilayer Recovery Process*

The basic recovery approach is to recover the affected services in the lowest possible layer. As such, the survivability is provided as close as possible to the origin layer of the failure. With recovery at the lowest layer, every survivable layer reserves some resources for the recovery purposes. A single-layer recovery mechanism cannot acquire resources in other layers in real time without breaching the independent operation of different layers. However, this requires dedicated resources at each survivable layer and thus at each layer the actual utilization of the resources is decreased.

A second multilayer recovery approach consists of recovering disrupted traffic close to the origin of the failure, independent of the layer. A higher-layer recovery scheme can resolve failures happening in layers below. To understand this approach we give the following example which is driven by data networks. A transport network often carries several service classes with different reliability requirements. It is generally easier to provide reliability grades when the survivability schemes reside in higher layers. Since a single recovery scheme suffices to protect the service against failures occurring in every (lower) layer, the implementation complexity of internetworking between different schemes can be avoided. In this case internetworking merely involves activating the responsible recovery scheme as fast as possible at the appropriate network layer. On the other hand, the finer switching granularity of the higher layers complicates rerouting in the event of lower-layer failures, because many entities are then affected at the same time.

1.1.7.2 *Optimal Selection of the Layer of Recovery*

The evolution of the future network infrastructure towards a converged all-IP architecture mandates partially the simplification of legacy and gradually obsolete parts of the protocol stack, leading to a direct IP-over-WDM network. Among others, the omission of intermediate Layer-2 protocols (e.g. ATM, SDH) simplifies the provision of recovery in case of failures: there is either the option of providing restoration of a failed

route at the IP-layer or make use of any optical resilience scheme at hand (pre-computed protection paths or ad-hoc provisioned restoration paths). In the following we discuss the pros and cons of each approach in the context of Grid-specific communication.

IP-layer recovery offers both the options of proactive protection or reactive restoration. Perhaps contrary to common expectation, restoration was the first recovery mechanism available in the evolution of the IP architecture and is embedded in standard routing protocols like OSPF and RIP. In the case of link-state routing protocols like OSPF, failure detection occurs at time scales multiple to link-state distribution periods (usually in the order of minutes). Upon detection, each router independently replaces all paths that entailed the failed link(s) with new paths that use sane links solely. An analogous scheme exists for routing with distance-vectors routing. Protection at the network layer is not inherently supported by the IP protocol suite, but instead provided implemented by means of the label switching infrastructure created by the MPLS protocol. In MPLS protection switching, apart from a working label switched path, labels may be distributed along paths that are disjoint to the primary path at the resource reservation phase. Upon failure detection, the ingress Label Switch Router is signalled to revert to a backup path, simply by changing the label swapping scheme. While MPLS protection switching yields faster recovery times (at the penalty of increased resource consumption), generally IP-layer resilience suffers for high recovery times, typically in the order of minutes. While this may be acceptable for some Grid applications, like computation-intensive experiments with relatively low bandwidth requirements or archival jobs, applications with real-time communication requirements (e.g. visualization) will not tolerate long recovery times. For these applications, employing recovery at the optical layer, where a failure can be usually restored in the sub-second order is necessary.

1.1.7.3 *Recovery across multiple domains*

The optical network infrastructure in support of a Grid may and in fact will most probably not be owned by a single administrative entity, but constitute rather a federation of domains that works in concert to fulfil the end-to-end QoS requirements posed by a Grid application [Staessens06]. This aggravates further the provision of survivable services, mainly due to the requirement for a control infrastructure that is scalable and optimal across various levels of heterogeneity.

[Develder08] proposed a scheme based on proxies towards routing in a multi-domain optical Grid network, introducing thus a hierarchy of routing services. Among others, the proposed scheme offers control plane scalability through aggregation of exchanged link state information and optimization over all interconnected domains of the Grid network. Otherwise, related work on the subject is scarce and surely merits a closer investigation, both generally and also in the specific context of Grids. The incorporation of the multi-layer nature of the network turns the problem even more interesting (and complex) and perhaps with a bigger span of options for the recovery part.

1.2 Grid Resource Fault-Tolerant Schemes

In this context, fault-tolerance is defined as the ability of a Grid application or task to continue valid operation after a Grid computation or storage resource fails. Grid resource-based fault-tolerant schemes are implemented at the application, middleware, or Operating System level [Valcarenghi08] and are mainly based on checkpointing ,replication and migration techniques.

Checkpointing [Mehnert-Spahn08] [Zhang01] is a technique that helps tolerate the errors leading to losing the effect of work of long-running applications. Checkpointing mechanisms takes a snapshot of the entire state of computing processes to allow, in case of failure, resuming the computing process from its last checkpoint on either the same or a different processor. Two main checkpointing approaches exist: coordinated checkpointing, where a coherent checkpoint is ensured for all cooperating processes, and communication independent checkpointing, where each process checkpoints its own state independently.

The general idea of replication [Ho06] is to create redundant copies in different locations. Replication reduces access latency and bandwidth consumption. It also facilitates load balancing and improves reliability by allowing the replicated component to continue to provide its service in spite of the failure of some of its copies, without affecting its clients. Selecting a site for the placement of replica is attributed to various factors such as number of requests for a particular file, bandwidth, read/write statistics and location of the resources.

Process migration [Du07] is the process of transferring *the* state of a process from one machine to the other so that an ongoing computation can be correctly continued. Process migration provides a number of desirable advantages in a Grid environment. Process migration facilitates remote processing if the local facilities are not sufficient and the desired resources not remotely accessible. Process migration helps balancing the load across the processors in a distributed system. Load-balancing can improve the performance of a system. *Fault-tolerance* is a main advantage of Service migration. Running processes can be transferred to another processor to provide *fault-tolerance.*

1.2.1 Related Work

A large number of papers have explored Grid-resource fault tolerant schemes in distributed computing environments. Aspects that have been explored include the design and implementation of fault detection services [Hwang03, Subbiah04], as well as the development of failure prediction [Derbal03, Zhang04, Schroeder06, Oliner07] and recovery strategies [Dogan02, Silva03, Camargo04]. The latter are often implemented through job checkpointing in combination with migration and job replication. Although both methods aim to improve system performance in the presence of failure, their effectiveness largely depends on tuning run-time parameters, such as the checkpointing interval and the number of replicas [Oliner05, Li03, Bossie06]. Determining optimal values for these parameters is far from trivial, for it requires good knowledge of the application and the distributed system at hand.

The next subsection gives a review of the research efforts considering checkpointing and replication.

1.2.1.1 *Checkpointing*

Different techniques are proposed in literature to address the checkpointing overhead and scalability concerns. Incremental checkpointing is a well researched technique that reduces data stored during checkpointing to only blocks of memory modified since the last checkpoint [Agarwal04]. In [Chakravorty07] a checkpointing-based fault-tolerance protocol for MPI jobs is presented, which lowers the overhead during normal execution and allows fast crash recovery by using the ideas of message logging and object-based processor virtualization. The latter limits the re-execution to only the failed processor and allows to distribute the failed work among the other processors. Clearly, this approach is only applicable to homogeneous environments. Another important checkpointing technique that has been addressed by several researches [Young74, Gelenbe79, Tantawi84] is based on calculating of the optimal checkpointing frequency and is called the optimal checkpoint interval problem. Analytical solutions are provided only under specific system assumptions. For example, it is often assumed that interoccurrence times of failures and repairs for each resource are independent and exponentially distributed. However in practice failures tend to cluster in time, while being caused by a relatively small set of computational nodes [Zhang04] [Schroeder06] [Oliner07]. As optimal solutions are not generally applicable, static sub-optimal solutions are developed. For example, a min-max checkpoint placement approach is introduced in [Ozaki04] where the sub-optimal checkpoint sequence is determined under uncertain circumstances in terms of the system failure time distribution. As the system and application parameters presumably change over time, cooperative (adaptive) checkpointing optimization approachs were considered. A cooperative checkpointing approach, introduced in [Oliner06, Oliner06a], addresses system performance and robustness issues by allowing the application programmer, the compiler and the runtime system to jointly decide on the necessity of each checkpoint. Another set of cooperative checkpointing schemes is proposed in [Xiang06] where the checkpointing interval is dynamically adjusted to achieve a timely job completion in the case of failure according to the remaining job execution time, time left before the deadline and the expected remaining number of failures before job termination. The latter implies that the system failure distribution should be known in advance. In [Katsaros07], dynamic checkpointing interval reduction was considered only if it leads to a computational gain, defined as the sum of the differences between the means for fault-affected and fault-unaffected job response times. In [Li07] yet another adaptive fault management scheme (FT-Pro) is discussed. This approach optimizes application execution time by considering the failure impact and the prevention costs. FT-Pro supports three prevention actions: skip checkpoint, take checkpoint and migrate. The appropriate action is selected based on the predicted frequency of failure. Therefore, the effectiveness of FT-Pro strongly depends on the quality of this prediction.

1.2.1.2 *Replication*

Similar to deciding upon the best checkpointing interval, developing an algorithm to calculate the optimal number of job replicas is a complicated issue which has been addressed in several studies [Li03, Hou94]. However, a number of restrictions were enforced on the execution environment, job interdependency, *etc*. Most of the replication-based fault-tolerant algorithms have assumed a fixed number of job duplicates. However, dynamic algorithms have recently received attention. In [Silva03] a dynamic replication-based method, called Workqueue with Replication (WQR), is introduced where a single copy of a job is distributed to random idle resources in FCFS (First Come First Served) order. When the job queue is empty and the

system has free resources, replication is activated to cope with varying availability of hosts. However, if a system is heavily loaded for a long period during peak hours when most of the failures in distributed environments tend to occur [Zhang04], the replication process is significantly delayed or even it is not activated at all as the WQR failure prevention is turned off by definition. In [Choi06] a group-based dynamic replication mechanism for peer-to-peer grid computing environments is proposed. This approach determines the amount of replication taking into account the reliability of each volunteer group, which is a group of resources with similar properties. In [Chtepen09] the algorithms introduced dynamically vary the number of job replicas dependent on the system load, the group-based approach determines the amount of replication taking into account the reliability of each volunteer group, which is a group of resources with similar properties.

1.2.1.3 *Combined approaches*

Several papers [Ziv97] [Pradhan94] describe schemes that combine checkpointing and job replication to deal with transient fault detection. Transient faults are often hard to detect because they do not result in a resource crash but only in a job state modification, which however can lead to wrong output. Therefore, duplicate jobs are executed on different nodes and their state is compared to track faults. The checkpointing mechanism, in turn, serves two purposes: preservation of a job state, to reduce the fault-recovery time; and state comparison of job replicas. To our knowledge, no work combining checkpointing and replication was performed thus far with the objective of achieving better resource utilization and improving job execution time.

2 Resilient Network Design

Phosphorus' emphasis is on network-related aspects of Grid computing, and thus its focus is on applications that are heavily dependent on communication resources. More specifically, for the Phosphorus testbed five applications have been designed, namely: WISDOM, KoDaVis, TOPS, DDSS and INCA [D3.1], from which three (KoDaVis, DDSS and INCA) have none or little requirements for computation resources, while the remaining two (WISDOM and TOPS) have also some computation part. To this end, network resilience is particularly important for the Phosphorus project and also for other Grid networks of this type. To address failures related to the network resources in this and the following section we will focus on resiliency issues that are related to the optical underlined network.

Resilient network design refers to the allocation of the necessary networking resources in order to serve a predefined or a predicted set of connection requests, providing at the same time resiliency. This allocation is performed either by reserving exiting resources or by adding additional resources if the existing ones are not enough.

Optical networks rely on wavelength division multiplexing (WDM) to efficiently exploit the available bandwidth. WDM enables different connections to be established concurrently through a common set of fibres, subject to the distinct wavelength assignment constraint; that is, the connections sharing a fibre must occupy distinct wavelengths. In the absence of wavelength conversion, a lightpath must be assigned a common wavelength on every link it traverses; this restriction is referred to as the wavelength continuity constraint. The problem of setting up lightpaths by routing and assigning wavelengths to them, so as to minimize the network resources used or maximize the traffic served, is called the routing and wavelength assignment (RWA) problem. The RWA problem is usually considered under two alternative traffic models. When the set of connection requests is known in advance (for example, given in the form of a static traffic matrix) the problem is referred to as *offline* or *static* RWA, while when the connections arrive randomly and are served on a one-by-one basis the problem is referred to as *online* or *dynamic* RWA.

Part of this section presents offline RWA resilient algorithms, as opposed to online RWA resilient algorithms, which are investigated in Section 3. Offline RWA is known to be an NP-hard optimization problem. Thus, offline RWA is more difficult as a combinatorial (algorithmic) problem than online RWA, since it has to jointly optimize the lightpaths used by all the connection requests,

instead of optimizing the provisioning of a single connection request, in the same way that the multi-commodity flow problem is more difficult than the shortest path problem in conventional networks.

Since offline RWA is NP-hard, obtaining an optimal solution might require non-polynomial computational effort in the worst case. To overcome this difficulty, much of the previous work on the RWA problem has focused on developing efficient heuristic methods. To make the problem computationally tractable, a common approach is to decouple the routing and the wavelength assignment problem to its two constituent sub-problems, namely by first finding a route for all connection requests and then searching for an appropriate wavelength assignment to the calculated routes. Note that both sub-problems are NP-hard: the routing problem for a set of requested connections corresponds to the multi-commodity flow problem, while the wavelength assignment problem corresponds to the graph coloring problem, both being NP-hard. Therefore, for each of the constituent sub-problems of RWA heuristic algorithms have to be applied. Algorithms developed for routing [Ramamurthy02] and wavelength assignment [Zang02], can then be combined and produce solutions for the joint RWA problem. However, such decomposition techniques suffer from the drawback that the optimal solution of the (joint) RWA problem might not be included in the solutions provided by the algorithms that address the two subproblems in which the RWA problem is decomposed.

Often, offline RWA is formulated as an integer linear program (ILP). Since the associated integer linear programs are very hard to solve, the corresponding relaxed linear programs (LP) have been used to get bounds on the optimal value that can be achieved for the desired objective function. Since fractional flows (fractional lightpaths) that the LP algorithms return are not physically realizable in WDM optical networks, the LP solution must be converted to an integral one, which approximates the optimal value of the LP objective; that usually happens by utilizing appropriate rounding techniques. An RWA LP formulation proposed in [Ozdaglar03], [Christodou08] has been shown to produce optimal or near optimal integer solutions for all RWA instances. Space reduction is usually achieved by forcing lightpaths to be routed through a restricted subset of candidate paths, linear on the size of the network. By selecting an appropriately large (but constant) number of candidate paths, the space of RWA solutions constructed is expected to be representatively large and contain an optimal RWA solution with large probability.

A number of exact ILP formulations for the design of survivable WDM networks have been proposed [Zhang04, Rama99a, Rama99b, Zang03, Qiao02, Zhang03, Bouillet07, Aneja07]. In [Zang03] authors address the RWA problem in a network with path protection under duct-layer constraints. Off-line algorithms for static traffic were developed to combat single-duct failures. Authors in [Qiao02] proposed an ILP-based model to jointly compute the shared protected primary-backup path pair for dynamic traffic. The model takes both network resource usage and backup distance into consideration. The problem of computing a pair of link disjoint paths in WDM network with the wavelength continuity constraint is NP-complete [Zhang03]. Several heuristics, LP (Linear Programming)-based have been proposed to solve the problem. In [Sahin00], the routing problem and the wavelength-assignment problem are solved separately. Several routing heuristics are developed and a vertex-coloring approach is used to solve the wavelength-assignment problem for various protection and restoration schemes in [Sahin00].

In Section 2.1 we present two routing and wavelength assignment formulations that provide dedicated path 1+1 protection to connections requests. The objective of the algorithms is to jointly minimize the number of wavelengths utilized by the primary and protected paths. The proposed algorithms are based on an LP relaxation formulation [Ozdaglar03], [Christodou08] that tends to yield exact integer solutions for a large number of RWA instances.

The majority of network design approaches proposed in the literature assume an ideal physical layer, where the quality of the optical signal is assumed to remain unchanged from source to destination. In reality, however, signal quality is significantly affected by physical impairments introduced by transmitting fibres and optical components, such as amplifier spontaneous emission noise (ASE), chromatic dispersion (CD), polarization mode dispersion (PMD), filter concatenation (FC) and nonlinear effects such as self- and cross-phase modulation (SPM, XPM), and four-wave-minxing (FWM). All these impairments may degrade the received signal quality to the extent that correct signal decoding to become totally impossible.

Section 2.2 addresses jointly the problem of designing optimally capacitated WDM networks and mitigating the negative effects of physical impairments. The latter is achieved through the placement of 3R regenerators at selected nodes of the network. The minimization of the number of utilized regenerators, together minimizing the working and spare capacity of the network, are modeled and formulated as an integer linear programming problem, whose solution yields an optimal design for any input network instance.

2.1 Survivable Routing and Wavelength Assignment

As it was mentioned in Section 1.1 a network is referred to as survivable if it provides some ability to recover ongoing functions disrupted by a catastrophic failure of any of its components. Moreover, in Section 1.1 the various network-related recovery schemes were classified into protection vs. restoration, dedicated vs .shared, as well as link vs. path vs. subpath (segment)-based techniques (Figure 1).

In this section we present two recovery schemes for the offline RWA problem. The proposed survivable RWA algorithms (to be referred to as S-RWA) provide (i) single primary paths for the connection demands that do not require protection, (ii) dedicated 1+1 link disjoint primary-backup path pairs for the demands that require protection, and the corresponding wavelength assignments, for a given set of connection requests known a-priory.

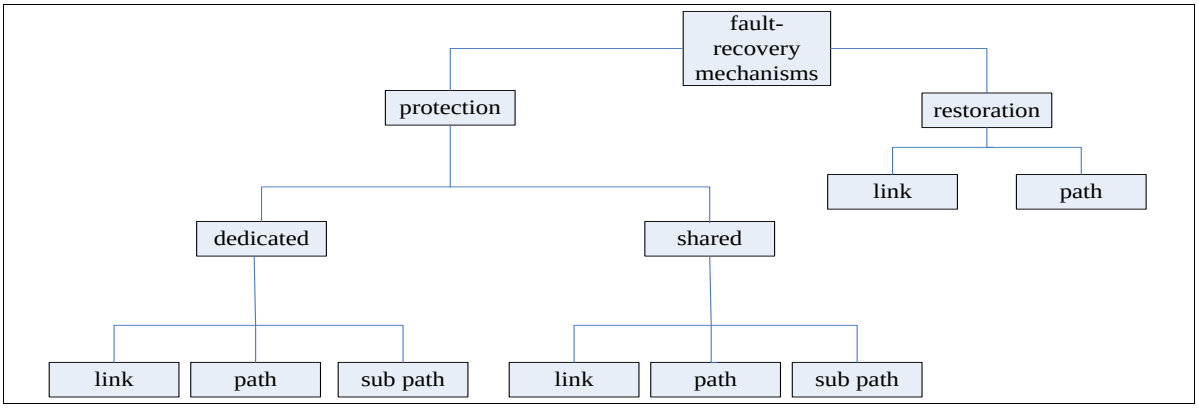


Figure 1: Fault recovery classification.

2.1.1 S-RWA Algorithm

A network topology is represented by a connected, simple graph $G=(V,E)$. V denotes the set of nodes, which we assume not to be equipped with wavelength conversion capabilities. E denotes the set of (point-to-point) single-fibre links. Each fibre is able to support a common set $C=\{1,2,,\ldots,W\}$ of W distinct wavelengths. The static version of RWA assumes an a-priori known traffic scenario given in the form of a matrix of nonnegative integers Λ, called the traffic matrix. Then, Λ_{sd} denotes the number of requested connections from source s to destination d (source-destination pair (s,d)). We are also given the set of the demands that require protection in a form of a matrix Λ^{1+1}, so that $\Lambda_{sd}^{1+1} \leq \Lambda_{sd}$ corresponds to the number of demands that require protection. Note that in the case of $\Lambda_{sd}^{1+1} < \Lambda_{sd}$ some connection demands between (s,d) pair do not require protection. For each requested connection the algorithm selects the primary path and, for the connections that require protection, the algorithm also selects the backup lightpath with the objective of minimizing the number of wavelengths utilized in the whole network. We present two LP-relaxation formulations for 1+1 protection in WDM networks (which will be referred to as Survivable-RWA algorithms).

The proposed Survivable-RWA algorithms consist of four phases:

1. The first (pre-processing) phase computes a set of candidate primary and backup paths to route the set of the requested connections. Due to the fact that we use two different algorithms to compute the sets of primary and backup candidate paths we end with two different formulations and two different Survivable-RWA algorithms. The first phase takes polynomial time since it is based on variations of k-shortest path algorithms.

2. In the second phase of the algorithm we formulate the survivable RWA problem as a linear program (LP). We present two different LP formulations that use the two different sets of primary and backup candidate paths, calculated in phase 1 of the algorithm. The corresponding linear programs are solved using the Simplex algorithm that is generally

considered efficient for the great majority of all possible inputs. If the instance is feasible and the solutions are integer, the algorithm terminates by returning the corresponding optimal solution in the form of routed lightpaths and assigned wavelengths, and blocking equal to zero (meaning that all connections can be served with the given number of wavelengths). If the instance is feasible but the solutions are fractional we proceed to phase 3. If the instance is infeasible, meaning that it cannot be solved with the given number of wavelengths, we proceed to phase 4.

3. The third phase uses iterative fixing and rounding techniques in order to obtain an integer solution. The maximum number of fixing and rounding iterations is the number of connection requests which is polynomial on the size of the input.

4. The fourth phase handles the infeasible instances (cases that the given number of wavelengths is not enough so as to serve all connection requests), so that some (since all is not possible) requested connections are established. Infeasibility is overcome by iteratively increasing the number of available wavelengths and re-executing phases 2 and 3 until a feasible solution is obtained. At the end of phase 4 we have to select which connections should be blocked so as to reduce the number of wavelengths to the given.

2.1.1.1 *Path Selection (Phase 1)*

In the first phase we use two alternative methods to compute the set of candidate paths that are passed as input to the two different LP relaxation formulations (to be presented in the next subsection). In fact the choice of the alternate routes greatly affects the solutions of the LPs, as well as their execution times [Zang03]. In [Suurballe74] Suurballe's algorithm solves the general problem of finding *K* link-diverse paths in a network. The work in [Bhandari99] is a good reference on diverse routing algorithms in mesh networks.

Single set of *k* disjoint paths

In this algorithm for each source-destination pair (*s,d*) we calculate a single set of *k* candidate paths from which the primary and the backup paths are then chosen. In particular, for each (*s,d*) pair we find *k* edge-disjoint paths between nodes *s* and *d*, such that every path pair of this set p_i, p_j , $i, j \in 1,...,k$, $i \neq j$, is edge disjoint, that is, there are no common links contained in the paths p_i and p_j. We denote the set of candidate edge disjoint paths for (*s,d*) pair as D_{sd}. If there are no *k* edge-disjoint paths between an (*s,d*) pair, we find the maximum possible number of edge-disjoint paths for this pair. A simple heuristic to find diverse (link-disjoint) paths is presented in [Bouillet07, chapter 6.4], and several ILP formulations in [Phung07].

Distinct sets (separate sets for the primary and backup paths)

In this variation, *k* candidate primary paths for each requested connection (*s,d*) are identified using a variation of the *k*-shortest path algorithm. This set corresponds to the set of candidate primary paths for connection (*s,d*) and is denoted by P_{sd}. For each path $p \in P_{sd}$, a set B_{sd}^{p} of candidate

diverse backup (protection) paths are then computed (Figure 2). Note that a path $p \in P_{sd}$ is edge disjoint with every path in B_{sd}^{p}, but paths in B_{sd}^{p} are not necessarily edge disjoint with each other. The cardinality of set B_{sd}^{p} is *m.* Thus, the number of paths for every source destination pair (*s,d*) that are given as input in the RWA algorithm (phase 2) is *k*m.*

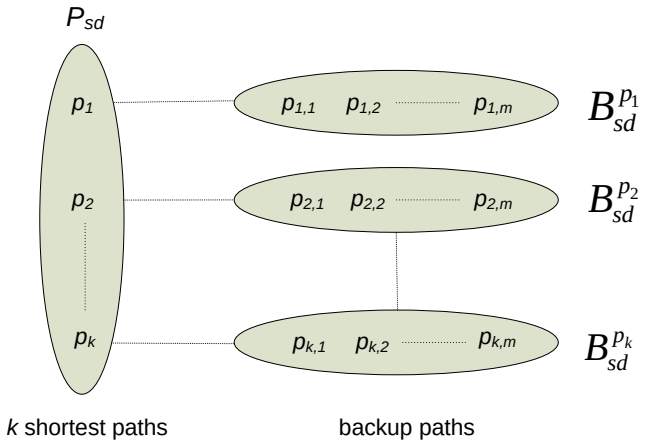


Figure 2: Relation between the sets Psd and Bsd.

2.1.1.2 *LP-relaxation formulations (phase 2)*

In this section we present two different LP relaxation formulations to address the problem of RWA with dedicated path protection. We will refer to these algorithms as S-RWA I and the S-RWA II. The first formulation S-RWA I uses as input the single set of edge disjoint alternative paths, while the second formulation S-RWA II uses as input the separate sets for the primary and the backup paths, as presented in the previous subsection (phase 1). The proposed LP-relaxation formulations aim at minimizing the maximum resource usage, in terms of wavelengths used on network links. Let $F_l=f(w_l)$ denote the flow cost function, an increasing function on the number of lightpaths w_l traversing link *l*; the actual formula is presented in the next subsection. The LP objective is to minimize the sum of all F_l values.

Formulation I

The following notation describes the inputs to the LP-relaxation formulation for the case of a single set of disjoint paths.

Input

- $s,d \in V$: network nodes
- $w \in C$: an available wavelength
- $l \in E$: a network link
- $p \in D_{sd}$: a candidate primary and backup path
- Λ_{sd} **:** the number of requested connections from node *s* to node *d*

- $\Lambda_{sd}^{1+1} \leq \Lambda_{sd}$ the number of requested connections from node *s* to node *d* that require protection

Variables:

- λ_{pw}: an indicator variable, equal to 1 if path *p* occupies wavelength *w*, else 0.
- F_l: the flow cost function value of link *l*

Objective

$$\textbf{minimize}: \sum_{l} F_l$$

Subject to the following constraints:

- Distinct wavelength assignment constraints,

  $$\sum_{p|l\in p} \lambda_{pw} \leq 1, \text{ for all } l \in E, \text{ for all } w \in C.$$

  No wavelength can be utilized more than once.

- Incoming traffic constraints,

  $\sum_{p\in D_{sd}} \sum_{w} \lambda_{pw} = \Lambda_{sd}$, for all (*s,d*) pairs that do **not** require a backup path ((*s,d*) $\in \Lambda$ and (*s,d*) $\notin \Lambda^{1+1}$).

  The number of lightpaths that is assigned to each connection that does not require protection is equal to the corresponding demand in the traffic matrix.

- Incoming traffic and protection constraints,

  $\sum_{p\in D_{sd}} \sum_{w} \lambda_{pw} = 2 \cdot \Lambda_{sd}$, for all (*s,d*) pairs that require a backup path ((*s,d*) $\in \Lambda^{1+1}$ **)**.

  The number of lightpaths that is assigned to each connection that requires protection is equal to the double of the corresponding demand in the traffic matrix.

- Primary/Protection path constraints,

  $$\sum_{w} \lambda_{pw} \leq \Lambda_{sd}, \text{ for all paths } p$$

  In this way, if a demand requires protection, the primary and protection paths are selected so as to utilize different paths from the set of candidate paths, ensuring in this way that the primary and backup path are disjoint.

- Flow cost function,

  $$F_l \geq f\ w_l\ = f\ \sum_{p|l\in p} \sum_{w} \lambda_{pw}$$

- The integrality constraints are relaxed to $0 \leq \lambda_{pw} \leq 1$.

  To this end, in the proposed formulation the variables can take non-integer (fractional) values and the problem can be solved using a Linear Program algorithm (and in particular Simplex). Note that non-integer solutions for the flow variables are not acceptable, since a

connection has to be served by an integer number of lightpaths and is not allowed to bifurcate between alternative paths or wavelength channels. Thus, the problem has to be formulated as an Integer Linear Program (ILP) but since ILP is NP-hard (which is considered intractable), we relax the constraints that define integer variables and solve the LP-relaxation (which is tractable) instead. If the LP-relaxation yields integer solutions then we are sure that we have found an optimal solution of the ILP problem. Otherwise we have to round the fractional variables and the optimality might be lost. It is due to these constraints that the formulation is called LP-relaxation.

Formulation II

The following notation describes the inputs to the LP-relaxation formulation for the case of separate sets of primary and backup paths.

Input

- $s,d \in V$: network nodes
- $w \in C$: an available wavelength
- $l \in E$: a network link
- $p \in P_{sd}$: a candidate primary path
- $b_p \in B_{sd}^p$: a candidate backup path for primary path *p*
- Λ_{sd} **:** the number of requested connections from node *s* to node *d*
- Λ_{sd}^{1+1} the number of requested connections from node *s* to node *d* that require protection

Variables:

- λ_{pw}: an indicator variable for primary paths, equal to 1 if primary path *p* occupies wavelength *w*, else 0
- F_l: the flow cost function value of link *l*
- $b_{b_p w}$**:** an indicator variable for backup paths, equal to 1 if path b_p is the protection (backup) of path *p* and occupies wavelength *w*, else 0

Objective

$$\textbf{minimize}: \sum_{l} F_l$$

Subject to the following constraints:

- Distinct wavelength assignment constraints for primary and backup paths,
  $$\sum_{p|l\in p} \lambda_{pw} + \sum_{p} \sum_{b_p|l\in b_p} b_{b_p w} \le 1, \quad \text{for all } l \in E, \text{ for all } w \in C.$$
  No wavelength can be utilized more than once.

- Incoming traffic constraints (only for primary paths),
  $$\sum_{p\in P_{sd}} \sum_{w} \lambda_{pw} = \Lambda_{sd}, \text{ for all } (s,d) \text{ pairs } \in \Lambda_{sd}$$
  The number of primary lightpaths assigned to each connection is equal to the corresponding demand in the traffic matrix.

- Protection constraints,

  $\sum_{w} \lambda_{pw} = \sum_{w} \sum_{b_p} b_{b_p w}$, for all (*s*,*d*) pairs $\in \Lambda_{sd}^{1+1}$

  If a primary path $p \in P_{sd}$ is selected then a backup path is selected from the corresponding backup set $b_p \in B_{sd}^{p}$. Otherwise no backup path from this set is selected.

- Flow cost function,

  $$F_l \geq f\ w_l\ = f\left(\sum_{p|l\in p} \sum_{w} \lambda_{pw} + \sum_{p} \sum_{b_p|l\in b_p} \sum_{w} b_{b_p w} \right)$$

- The integrality constraint is relaxed to $0 \leq \lambda_{pw}, b_{b_p w} \leq 1$.

Flow Cost Function

The variable F_l expresses the cost of congestion on link *l*, for a specific routing of the connections. We choose F_l to be a properly increasing function $f(w_l)$ of the number of lightpaths ($w_l = \sum_{p|l\in p} \sum_{w} \lambda_{pw}$ for the first formulation, and $w_l = \sum_{p|l\in p} \sum_{w} \lambda_{pw} + \sum_{p} \sum_{b_p|l\in b_p} \sum_{w} b_{b_p w}$ for the second formulation) crossing link *l*. $F_l = f(w_l)$ is also chosen to be convex (instead of linear), implying a greater degree of 'undesirability', when a single link becomes highly congested. More specifically, for the rest of this study we utilize the following flow cost function:

$$F_l = f\ w_l\ = \frac{w_l}{W + 1 - w_l}.$$

The above (nonlinear) function is inserted to the LP in the approximate form of a piecewise linear function; i.e., a continuous non-smooth function, that consists of *W* consecutive linear parts ([Ozdaglar03], [Christodou08]). The piecewise linear function is constructed as follows: we begin with $F_l(0)=0$, and iteratively set, for $i=1,\ldots,W$, $F_l{}^i(w_l)=a_i \cdot w_l+\beta_i$, $i-1 \leq w_l \leq i$, where $a_i=F_l(i)-F_l(i-1)$ and $\beta_i=(i-1)\cdot F_l(i)-i\cdot F_l(i-1)$.

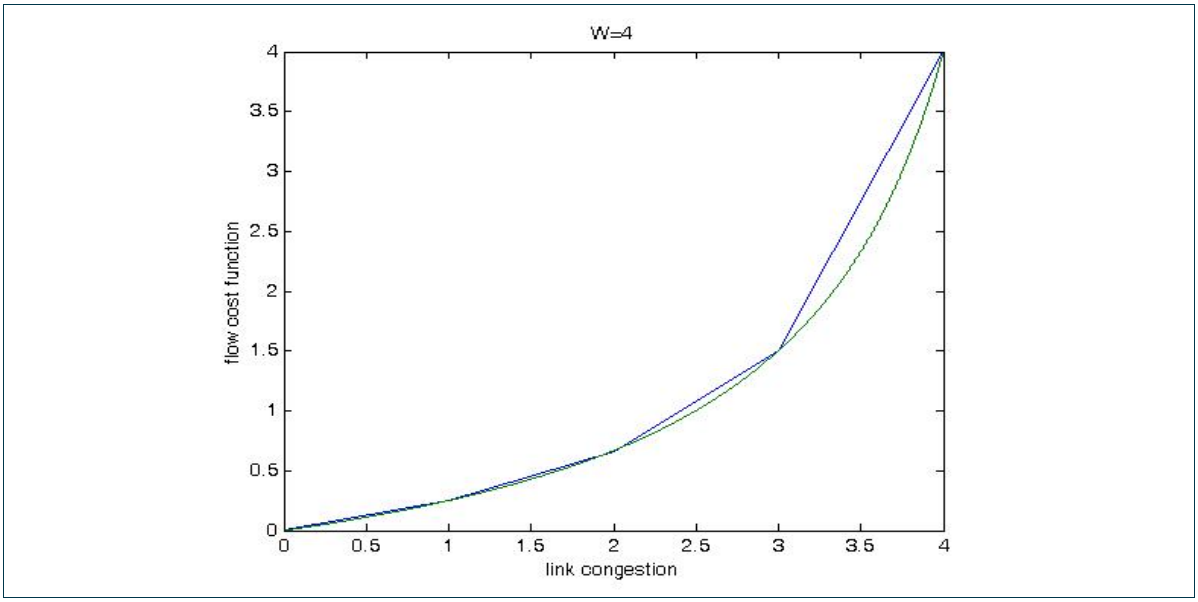


Figure 3: The flow cost function $F_l=f(w_l)$ (curved line) and the corresponding piecewise linear function.

Figure 3 shows how we transform the non-linear cost function to a piecewise linear. Observe that the piecewise linear function is exactly equal to $F_l=f(w_l)$ at integer argument values (w_l=1,2,…,W) and greater at other (fractional) argument values. Inserting a sum of such piecewise linear functions to the LP objective, results in the identification of integer optimal solutions by Simplex, in most cases. This is because the vertices of the polyhedron defined by the constraints tend to correspond to the corner points of the piecewise linear function and thus consist also of integer components. Since the Simplex algorithm moves from vertex to vertex of that polyhedron there is a higher probability of obtaining integer solutions than other using methods.

Number of Variables and Constraints

Table 1 shows the number of variables and the number of constraints of the proposed algorithms in the case that all the requests need to be protected. Let N=|V| be the number of network nodes, W=|C| the number of available wavelengths, L=|E| the number of links, and k and m be the parameters in the corresponding algorithms that calculate the candidate paths (phase 1). We also let ρ be the *traffic load*, defined as the ratio of the total number of connection requests over all possible source-destination pairs, that is,

$$\rho = \frac{\sum_{(s,d)\,pairs} \Lambda_{sd}}{N \cdot (N-1)},$$

where Λ_{sd} is the number of lightpaths that have to be established for source-destination pair (s,d).

For all the formulations the number of variables increases with the traffic load, the number of alternate paths, the number of network nodes and the number of available wavelengths per link.

We can also observe that the number of variables of the S-RWA II algorithm increases with the number of *k* (alternate paths) and *m* (number of protection paths for each path *p*). The number of the constraints for both formulations S-RWA I and S-RWA II are equal. The constraints of the formulations differ on the number of inequalities and the number of equalities they require.

Formulation	Number of Variables	Number of constraints		N: number of nodes W: number of wavelengths L: number of links k: number of shortest paths for each connection m: number of protection paths for each path ρ: load (percentage of total connections) Constraints: 1: incoming traffic constraints 2: distinct wavelength assignment constraints 3: flow cost function constraints 4: primary/protection disjoint constraints 5: demand protection constraints
		=	≤	
RWA without protection	$k \cdot \rho \cdot N^2 \cdot W+L$	$(\rho \cdot N^2)_1$	$(L \cdot W)_2+ (L \cdot W)_3$	
S-RWA I	$k \cdot \rho N^2 \cdot W+L$	$(\rho \cdot N^2)_1$	$(L \cdot W)_2+(L \cdot W)_3+ (k \cdot \rho \cdot N^2)_4$	
S-RWA II	$(k+1) \cdot m \cdot \rho \cdot N^2 \cdot W+L$	$(\rho \cdot N^2)_1+ (k\rho N^2)_5$	$(L \cdot W)_2+(L \cdot W)_3$	

Table 1: Number of variables and constraints for the proposed S-RWA algorithms.

2.1.1.3 *Iterative fixing and rounding (phase 3)*

The proposed algorithms are based on LP-relaxation formulations. Thus, in these formulations the variables can take non-integer (fractional) values so as to be able to solve the problem with Simplex algorithm (which is considered to be fast for the majority of the input instances). As explained, non-integer solutions for the flow variables are not acceptable, since a connection has to be served by an integer number of lightpaths. Although the piecewise linear flow cost function used in these formulations (see the previous section) is defined so as to make Simplex find a solution that consist of integer variables, there are cases that the solution returned by Simplex still has some fractional variables [Christodou08].

Thus, if we do not obtain integers solutions by the execution of Simplex we employ the following iterative fixing and rounding methods. We start by fixing variables, that is making the integer variables of the previous LP solution constants, and solve the reduced remaining problem. When this process cannot be further pursued we continue with the rounding process that is we round a single variable and solve the remaining problem. Note that the number of fixing and rounding iteration that can be performed is equal to the number of connection requests and thus polynomial on the size of the input.

Fixing Variables

In the case that all the solutions of the LP-relaxation are not integer, we remove the solutions that are integer (we assume that for these connection requests we have found the lightpath that would serve them) and solve the reduced remaining problem. This is equivalent to making these variables constants and solving the initial LP problem with these additional equality constraints. In the latter case we only need to add equality constraints in the end of the LP tableau, so we do not need to create a new tableau. Fixing variables do not change the objective cost that is returned by the LP, so we move from the previous solution to a solution with equal or more integers with the same objective. Thus, if we finally reach to an integer solution we are sure that this is one among the optimum solutions. On the other hand, fixing variables is not guaranteed to return an integer solution if one exists, since the integer solution might consist of different integer variables than the ones gradually fixed by this process. When we reach a point that the process of fixing does not increase the integrality of the solution, we move to the rounding process.

Rounding a single variable

Having a set of non-integer solutions we have to choose one variable to round. Choosing this variable is not a trivial task, since this might result in an increase in the objective. Thus, we want to round the variable that results in the smallest (or none) increase of the objective. This is the variable whose derivate is smaller with respect to the used objective function. Instead of finding this variable, something that would require additional calculations, we round the variable that is closest to 1 and we continue with the fixing process. Rounding is inevitable in the case that there is no integer solution with the same objective as the LP relaxation of the RWA instance. While fixing variables helps us move to more integer solutions with the same objective, rounding helps us move to a higher objective and search for an integer solution there. Note that if we reach an integer solution only by fixing the variables we are sure that we have found an optimum integer solution. However, by the time that we round a single variable we are not sure anymore that we will find an optimum.

2.1.1.4 *Infeasible instances (phase 4)*

The fourth phase, finally, handles the infeasible instances, so that some (since all is impossible) additional requested connections can be established. Infeasibility is overcome by iteratively increasing the number of available wavelengths by 1 and re-executing the second phase. The resulting RWA solution must be converted to a final one that uses only W wavelengths; therefore, some wavelengths must be removed and the lightpaths occupying them have to be blocked. The removed wavelengths are those occupied by the minimum number of lightpaths, so as to block the minimum number of requested connections.

2.1.2 Simulation Results

To evaluate the performance of the proposed algorithms we performed a large number of simulation experiments. In our experiments we use the NSFnet topology (Figure 4) which has 14

nodes and 21 links (we assumed 42 directed links in our experiments). The results were averaged over 100 experiments corresponding to different random static traffic instances of a given traffic load. In particular, we performed experiments for traffic loads ρ=0.1 to 0.5. Each execution corresponds to a different instance of the RWA problem for the given traffic load and in order to fairly compare the algorithms we have produced the same 100 instances for all the examined algorithms. The algorithms presented in previous sections were implemented in Matlab, and LINDO library was used to solve the LP problems.

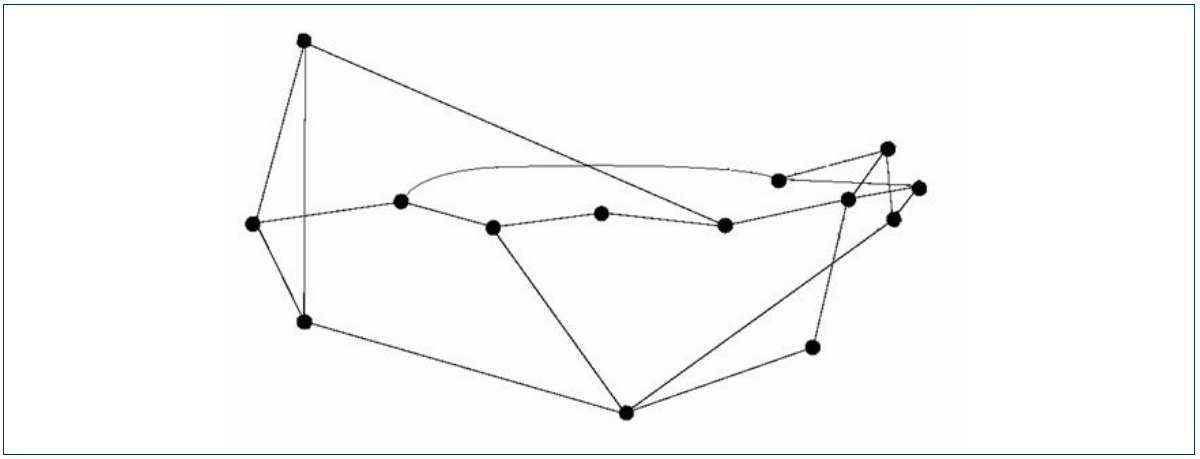


Figure 4: The NSFnet network with 14 nodes and 42 directed links.

In our simulations we assume that all the requests require protection. Following the notation used in the previous section, this corresponds to the case that primary traffic matrix Λ is equal to the backup matrix $\Lambda^{1+1} = \Lambda$. For example if the initial traffic load is given by ρ, then by protecting each request, the final traffic load corresponds to $2\cdot\rho$. If we cannot protect a requested connection then this request is blocked and no lightpath is established at all. In the following figures when we refer to the traffic load, the initial traffic load ρ (that corresponds to traffic matrix Λ) is considered.

Figure 5 presents the average blocking probability as a function of the number of available wavelengths for the two Survivable-RWA (S-RWA) algorithms for traffic load ρ=0.5. An obvious observation is that as the number of available wavelengths increases for a given traffic load, the blocking probability is decreased. Both algorithms need W=20 wavelengths in order to have zero blocking ratio. It is worth mentioning that the RWA algorithm without protection and for the specific experimental parameters of Figure 5 (W=14-21 and traffic load equal to 0.5) exhibits zero blocking. Of course in the case of the survivable RWA the traffic load is doubled since for each connection request we also request a backup path.

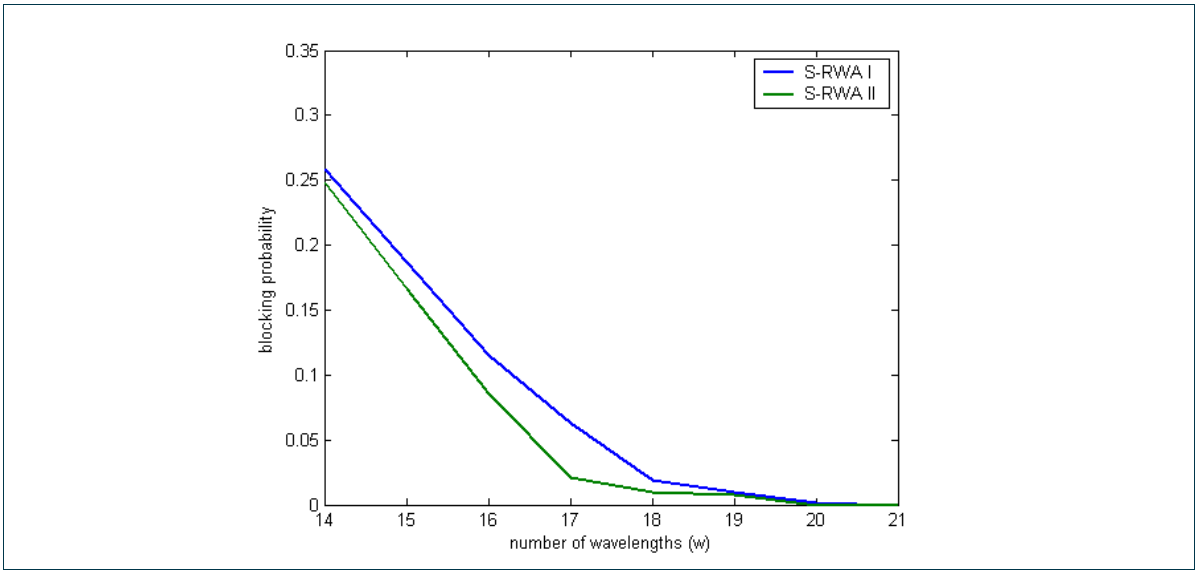


Figure 5: Blocking probability vs number of wavelengths for traffic load ρ=0.5.

Figure 6 presents the average blocking probability as a function of traffic load for the two Survivable-RWA (S-RWA) algorithms for constant number of available wavelengths (W=10). As the traffic load ρ is increased and the number of available wavelengths remains constrained and constant, the blocking probability increases. S-RWA I algorithm uses one set of disjoint candidate primary and backup paths to route the set of the requested connections. On the contrary, S-RWA II algorithm uses one set for the primary and one set for the backup paths to route the requested connections. The constraints of the corresponding LP-relaxation formulations are based on the corresponding sets of candidate paths (see phases 1 and 2 of the algorithm and Table 1). This is the main difference between the two algorithms. As depicted in Figure 5 and in Figure 6 the S-RWA II has slightly better performance. In fact the choice of the alternate routes significantly affects the solutions found by the LPs. The construction of the candidate paths for the S-RWA II algorithm gives a better space and thus the S-RWA II algorithm is able to find in many cases better solutions than S-RWA I. For the simulation we used for S-RWA I k=3, and for S-RWA II k=3, M=1.

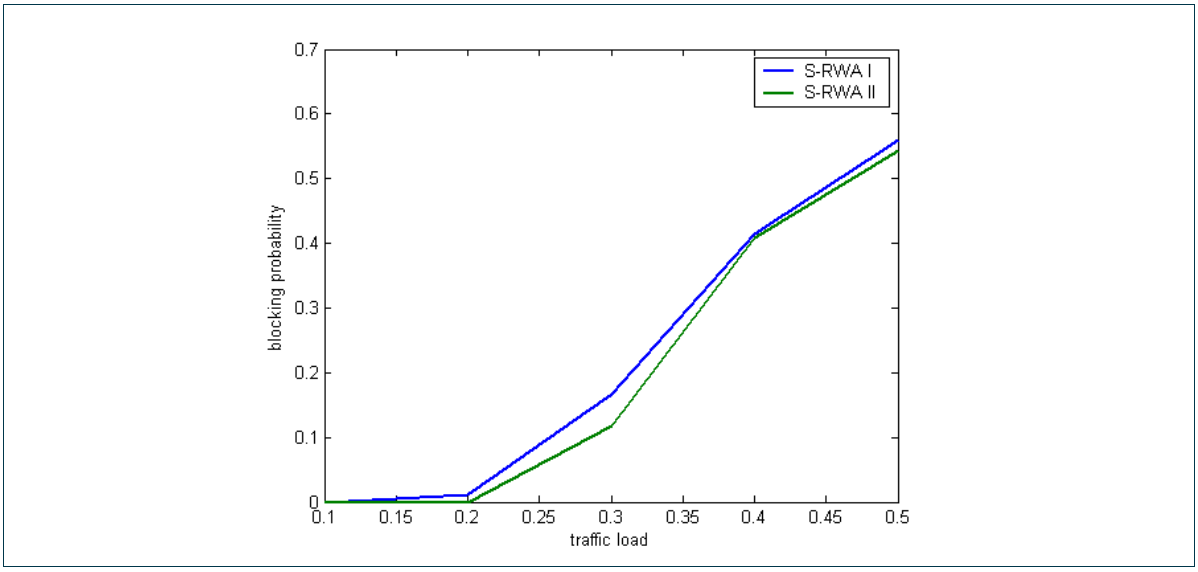


Figure 6: Blocking probability vs traffic load for 10 available wavelengths.

However, the choice of the alternate routes affects the execution times of the algorithms. Figure 7 shows the average execution times of the two S-RWA algorithms (in seconds) versus the traffic load ρ, assuming that W=10 wavelengths are available in the network. Note that by execution time we are referring to all the phases of the proposed S-RWA algorithms that include the execution of the Simplex algorithm in phase 2, the fixing and rounding iterations performed in phase 3, and the iterative increases of wavelengths in order to find a feasible solution of phase 4. From Figure 7 we can observe that S-RWA I algorithm exhibits lower average execution times when compared to S-RWA II. Note that the number of available wavelengths (W=10 for the experiments in Figure 6) for traffic load 0.3 to 0.5 are not enough to establish all the requested connections. For this reason the algorithms go to their fourth phase and increase the number of wavelengths in order to find a feasible solution. This vastly increases the whole execution time of the algorithms. For traffic loads 0.3 to 0.5 in which blocking is non-zero the execution time would decrease if we had more available wavelengths.

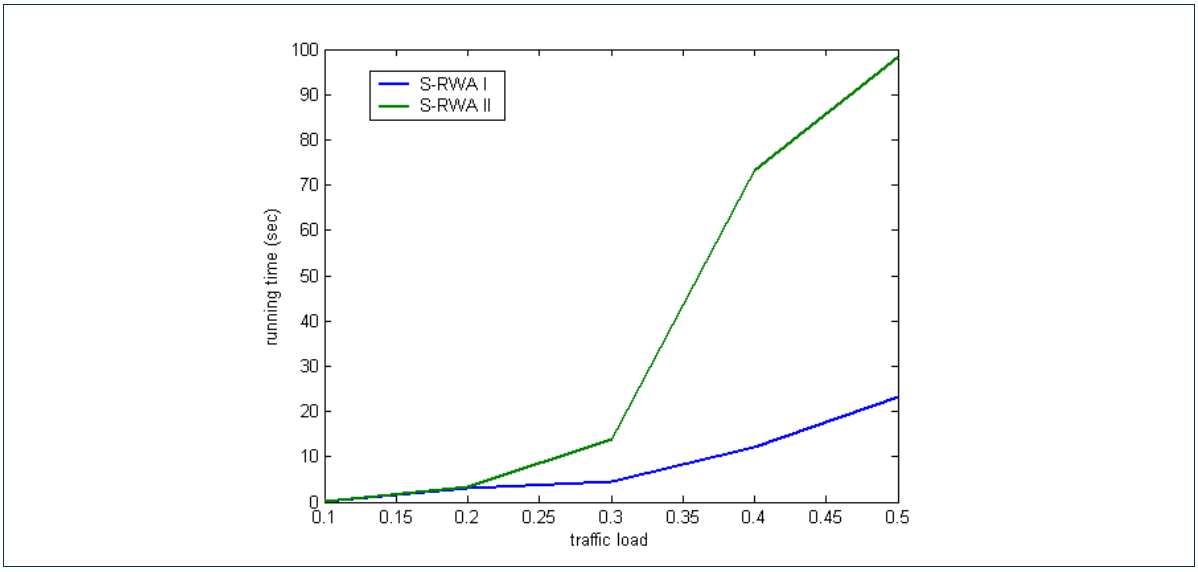


Figure 7: Total running time vs traffic load for *W* equal to 10 wavelengths.

2.1.3 Conclusions

In this section we presented two LP-relaxation formulations, namely S-RWA I and S-RWA II, for the problem of finding link-disjoint primary-backup lightpath pairs, assuming 1+1 protection for certain connection requests known in advance (offline or static traffic). The objective of the algorithms is to jointly optimize the utilization of wavelengths used by all the primary and backup lightpaths. The two proposed LP formulations differ in the space of the candidate paths that they use in order to choose the primary and the protection lightpaths. The complexity of the formulations was examined by presenting the number of variables and the number of constraints that they utilize. Our simulation results shows that the S-RWA II algorithm has better blocking probability, while the S-RWA I has better running time. Although we performed small scale experiments, the proposed algorithms are expected to scale and have acceptable execution time performance for large networks and high traffic loads, since they are based on LP-relaxation formulations which are solved using the Simplex algorithm which is generally considered efficient for the great majority of all possible inputs.

2.2 Physical Impairments Aware Resilient Network Design

In [D5.7] we modelled and proposed an integrated solution for jointly dimensioning an optical network and placing 3R regenerators to rectify lightpaths with unacceptable BER. Still, this work

addressed the design of an unprotected network, i.e. the imposition of extra capacity for recovery purposes was left to be done in later design phases. In this deliverable, we extend the work presented in [D5.7] to incorporate recovery from failures as well. More precisely, we address resilience by incorporating 1+1 protection for each connection request that is to be provisioned in the network being designed.

2.2.1 System Model

We employ here the same system and node/link architecture model as the one assumed in [D5.7]. For the sake of integrity, we review the main parts of this model and refer the reader to [D5.7] for further details.

A WDM (Wavelength Division Multiplexed) optical network in support of a Grid is modelled as a set of Optical Cross-connects (OXCs – referred to also as "nodes") interconnected by a set of unidirectional fibre links (referred to also as "links"). Bidirectional communication is achieved by installing two fibres between any two nodes in opposite directions. Each fibre comprises at maximum a constant number of discrete wavelengths *W* with bandwidth *B*, *both* depending on the WDM technology deployed by the lambda-switched Grid. We also limit the maximum number of fibres that can be installed in a duct by a small integer *maxfibre*.

The volume of traffic that the dimensioned network should be able to carry and recover in the case of single failures is expressed in the form of a two-dimensional traffic matrix Λ reflecting the estimated bandwidth demand between any two Grid sites. Each entry $\Lambda(i,j)$ of the matrix specifies the number of lightpaths that needs to be installed to carry the traffic generated at Grid site *i* and has Grid site *j* as its destination. Lightpaths are routed in our model using shortest paths, where path length is defined as the summation of the lengths (in kilometres) of all physical links comprising the path. Apart from the shortest path between the source and the destination of a traffic demand, we also consider alternative paths using a k-shortest path algorithm with the same definition of path length as above. Let *P* denote the set of candidate paths for a given source to a given destination.

Additionally to having *P* candidate paths for routing a demand from source to destination, we also compute *K* backup paths for each of the *P* primary paths. We compute each of the *K* backup paths such that each backup path is link-disjoint to the primary path it is protecting.

We assume the wavelength selective node architecture for each optical cross-connect (OXC), where N_f is the number of fibres deployed at a link incident to the node ($N_f \leq$ *maxfibre*). Each fibre of an input link carries maximally *W* wavelengths that are first demultiplexed. Subsequently, each wavelength λ_i is routed through the respective switch fabric and is either terminated, regenerated or routed to the appropriate output port, where all wavelengths are multiplexed and transmitted through the output fibre. Additionally, we assume that every wavelength selective switch at an OXC with *N* incident fibres is equipped with *N* additional ports for termination/regeneration

purposes, i.e. if *N* is the number of incident fibres to the OXC, then each wavelength specific switch is of size *2Nx2N*.

2.2.2 Cost Model

We model the cost scaling of all required optical equipment with linear functions, similar to the approach presented in [Caenegem98]. Throughout this work, we cope with the "greenfield" version of the network design problem, i.e. additionally to dimensioning the capacity of installed machinery and placing regenerators, our solution selects also the optimal set of links among Grid sites that need to be opened among all possible link set combinations. Thus, link opening costs are also considered.

The cost of installing a fibre link between two Grid sites involves the cost of digging the duct, leasing cost, fibre cost and maintenance cost. Intuitively, the cost of this process is proportional to the geographical distance covered by the link, since not only the installed hardware (pipes) and leasing cost is proportional to link length, but also the cost associated with the time required to complete the trenching (labour cost, equipment leasing) are proportional to link length. Let l_i be the length of link *i* and α stand for the trenching cost per link length unit (e.g. monetary unit/km). Then, the trenching cost α_i for link *i* is given by $\alpha_i = \alpha \cdot l_i$.

Unlike trenching cost, fibre cost involves two cost components: one proportional to link length and an additional fix cost per installed fibre. The length dependent cost is due to machinery installed per fibre span, like amplifiers, dispersion compensation fibre and pre-/post-compensation fibre, whereas the fixed cost component accounts for equipment installed at the termination points of the fibre, like de-/multiplexers. Let β_{span} and l_{span} stand for the cost per span and the span length respectively and β_{fix} stand for the fix cost of installing a fibre (all considered constant – uniform to all fibres). Then, the total cost β_i of installing a fibre on link *i* of length l_i is:

$$\beta_i = \frac{l_i}{l_{span}} \cdot \beta_{span} + \beta_{fix}$$

Furthermore, the cost of installing/activating a wavelength is the cost γ of the transmitter/receiver associated equipment, which is constant per wavelength. Thus, if γ_{ij} denotes the cost of installing/activating wavelength *j* on link *i*, then $\gamma_{ij}=\gamma$.

The last cost factor of the network design process is the switch fabric dimensions of the OXCs used.There are two factors that drive the cost of the switch fabric at a node: a) the number of MEMS switches required, which equals the maximum number of wavelengths deployed among all fibres incident to the node and b) the dimensions of each wavelength selective switch, which depends on the number of fibres incident to the node. Using the same reasoning as in [D5.7], our model assumes that the cost of switching is only governed by factor b), namely the dimensions of

each of the wavelength selective switches and is in fact independent from the cumulative number of switches at each node. Based on the above, we consider *S* distinct sizes of rectangular switches (e.g. 4x4, 8x8 etc.) and assign a constant cost per input port φ_s (*s=1..S*) to each switch of type s.

2.2.3 Physical Impairments

We used the analytical model presented already in [D5.3] for calculating the BER across a lightpath as a function of the Q-factor. The instantiation of the analytical model used in this work takes the following linear and non-linear impairments into consideration:

- Amplified Spontaneous Emission (ASE) noise
- Chromatic dispersion (CD)
- Self-Phase Modulation (SPM)
- Cross-Phase Modulation (XPM)
- Four Wave Mixing (FWM)

For a more thorough specification of the impairment model the reader is referred to the work presented in [D5.3] and [D.5.7].

2.2.4 Impairment-aware Resilient Network Design Formulation

Motivated by the linear relationship between installed capacity and associated cost, we formulate the problem of joint resilient network design and placement of regenerators as an ILP (Integer Linear Programming) problem.

Let *G = (V,E)* be the graph corresponding to the input topology, where *V* is the set of nodes (Grid sites) and *E* is the set of candidate links. Let also $f : E \to Q$ be the transformation that returns for each link the distance l_e between the two nodes that it is incident to. Given is also a |V| x |V| traffic matrix *Λ* with zero diagonal. In the following, instead of using index pairs to refer to elements of the incidence matrix of G and to elements of *Λ*, we instead serialize access by using integer indices. More precisely, we refer to links using the integer index e (1≤*e*≤|*E*|) and to traffic demands using the integer index $\underline{d}$ (1≤*d*≤*D*), *D* being the number of non-zero elements of T; otherwise, traffic demands requesting no lightpaths are ignored. Furthermore, we use the integer index *c* (1≤*c*≤*W*) to refer to each of the distinct wavelengths installed on a fibre. Let also h_d (*d*=1…*D*) be initialized to the number of wavelengths requested by demand *d*.

Towards modelling the dimensions of a node's switching fabric, we define two more constants: a) $\eta_{e,n} \in \{0,1\} (e = 1..|E|, n = 1..|V|)$,which is set to 1, if link *e* is incident to node *n,* and is zero otherwise and b) $\theta_s \in \mathbb{Z}^*_+ (s = 1..S)$ is an integer corresponding to one of the dimensions of an *nxn* switch (e.g. θ_s=*4* for a *4x4* switch), *S* being the number of possible switch dimensions considered by the problem.

In the following, we introduce the variables used in our ILP formulation, together with their problem-specific semantic:

- $x_{dpc} \in \mathbb{Z}_+$: Number of lightpaths that use the *p-th* shortest path serving demand *d* on wavelength *c*.
- $z_{dpckg} \in \mathbb{Z}$ **:** Number of lightpaths that use the *k*-th backup path on wavelength *g* to protect the *p-th* primary lightpath that serves demand *d* on wavelength *c.*
- $w_{ce} \in \mathbb{Z}_+$: Number of times wavelength *c* is used on link *e* (both by primary and backup paths).
- $y_e \in \mathbb{Z}_+$: Number of fibers installed on link *e*.
- $u_e \in \{0,1\} = \begin{cases} 1, \textit{if link e is used in the dimensioned network} \\ 0, \textit{otherwise} \end{cases}$
- $t_{n,s} \in \{0,1\} = \begin{cases} 1, \textit{if node n requires switch fabric dimensions corresponding to switch type s} \\ 0, \textit{otherwise} \end{cases}, n = 1..|V|, s = 1..S$

The *P* shortest paths that are candidates for serving demand *d* (*d*=1...*D*), as well as the *K* shortest paths that are candidates for protecting each of the *P* shortest paths for each demand d, are computed at a pre-processing step and the resulting path set is used as input to the ILP formulation. For convenience, we encode all computed paths using two indexed constants as follows:

$$\delta_{edp} = \begin{cases} 1, \textit{if link e is used by primary path p to serve demand d} \\ 0, \textit{otherwise} \end{cases}, e = 1..|E|, d = 1..D, p = 1..k$$

$$\beta_{edpk} = \begin{cases} 1, \textit{if link e is used by the kth backup path protecting path p that serves demand d} \\ 0, \textit{otherwise} \end{cases}, e = 1..|E|, d = 1..D, p = 1..P, k = 1..K$$

The ILP problem formulation that minimizes the cost of a network dimensioned to carry the input traffic matrix with 1+1 protection is given below:

$$\textit{Minimize} \quad \sum_{e=1}^{|E|}(u_e \cdot \alpha_e) + \sum_{e=1}^{|E|}(y_e \cdot \beta_e) + \sum_{e=1}^{|E|}\sum_{c=1}^{W}(w_{ce} \cdot \gamma) + \sum_{n=1}^{|V|}\sum_{s=1}^{S}(t_{n,s} \cdot \varphi_s \cdot \theta_s \cdot W)$$

$$\text{subject to} \quad \sum_{p=1}^{P}\sum_{c=1}^{W} x_{dpc} \geq h_d \,, d = 1..D \quad (1)$$

$$\sum_{k=1}^{K}\sum_{g=1}^{W} z_{dpckg} = x_{dpc} \,, d = 1..D, p = 1..P, c = 1..W \quad (2)$$

$$\sum_{d=1}^{D}\sum_{p=1}^{P}(\delta_{edp} \cdot x_{dpc}) + \sum_{d=1}^{D}\sum_{p=1}^{P}\sum_{k=1}^{K}\sum_{c'=1}^{W}(\beta_{edpk} \cdot z_{dpc'kc}) \leq w_{ce}, c = 1..W, e = 1..|E| \quad (3)$$

$$y_e \geq w_{ce} \,, c = 1..W, e = 1..|E| \quad (4)$$

$$y_e \leq u_e \cdot \max fibre \,, e = 1..|E| \quad (5)$$

$$\sum_{s=1}^{S} t_{n,s} = 1 \,, n = 1..|V| \quad (6)$$

$$\sum_{e=1}^{|E|}(2 \cdot \eta_{e,n} \cdot y_e) \leq \sum_{s=1}^{S} \theta_s \cdot t_{n,s} \,, n = 1..|V| \quad (7)$$

While the specification of the objective function follows naturally from the presented cost model, we further need to elaborate in the constraints of our ILP formulation. Constraint (1) guarantees that enough lightpaths will be installed to satisfy the number of lightpaths h_d posed by demand *d*. Constraint (2) makes sure that enough backup lightpaths will be installed such that each primary path is protected by exactly one (disjoint) backup path. Constraints (3) and (4) mandate that enough fibres are installed at each link *e* such that there is enough capacity to route the number of flows (primary and backup lightpaths) traversing the link. Constraint (5) makes sure that the number of fibres installed per link does not exceed the assumed maximum (*maxfibre*). Last, constraints (6) and (7) imply that the dimensions of wavelength switches installed at each node are large enough to accommodate the number of fibres incident to the node.

An optimal solution to the above ILP formulation yields a dimensioned network design (set of links, number of fibres per link, number of wavelengths per fibre and dimensions of switches), however without considering the fact that the signal carried by part of the installed paths may be impaired. We mitigate the potential side-effects of physical impairments through the installation of 3R regenerators at exactly these points of the networks such that the quality of the signal carried by any lightpath to be acceptable with regard to some predefined threshold. Although all-optical 3R regeneration offers transparency, this work focuses on optoelectronic regeneration due to the technological maturity and commercial availability of this technology. However, the methodology presented applies equally to networks employing all-optical regeneration techniques. It should be noted that we only consider regeneration location at OXCs, i.e. span-based regeneration is not taken into account

We use the bit error rate (BER) at the termination point of a lightpath as a measure for signal quality: BER is calculated and compared against a predefined threshold value ($B_{thresh}=10^{-15}$) to decide whether the lightpath is impaired or not. In case a candidate lightpath – either primary or backup – is impaired, the least number of regenerators required to rectify the lightpath is computed. The algorithm used for regenerator placement is the same as the one used for the

same purpose in [D5.7]: given a path *p* we consider all possible placements of *i* regenerators on *p.* Starting with *i=1,* the number of potential regenerators on *p* is increased by one until the first *i* is found that results in at least one placement of regeneration nodes with acceptable end-to-end signal quality across p. For this value of *i*, all possible placements of the *i* regenerators that rectify the signal on *p* are added to the list of candidate paths for serving the demand between the source and the destination of *p.*

The pre-processing phase is slightly complex with worst case complexity $D \cdot P \cdot K \cdot \sum_{i=0}^{d_G - 1} \binom{d_G}{i}$, where d_G is the diameter of the network graph, *D* the number of demands and *P* and *K* is the number of alternative primary and backup paths considered respectively. For *P* and *K* being small integers, the above complexity is $O(D \cdot 2^{dG})$, resulting theoretically to exponential preprocessing time. However, as already reasoned in [D5.7], the expected complexity of the preprocessing phase will hardly reach its theoretical upper bound and thus it is computationally feasible in most practical cases.

The result of the pre-processing phase described above is a set of at most $P \cdot K$ candidate paths for each demand *d*, together with a fixed position of *i* regenerators for each path *p* $(0 \le i \le n(p) - 1, n(p) : number\ of\ nodes\ of\ p)$. We incorporate the additional cost of regeneration to the objective function of the ILP presented above through the integer variable r_{dp} that denotes the number of regenerators for each path *p* serving demand *d*. Let also *μ* stand for the equipment cost (transponder cost) of regenerating a single wavelength. The updated objective function of the ILP incorporating the additional cost of selective regeneration is given below:

$$Minimize \sum_{e=1}^{|E|}(u_e \cdot \alpha_e) + \sum_{e=1}^{|E|}(y_e \cdot \beta_e) + \sum_{e=1}^{|E|}\sum_{c=1}^{W}(w_{ce} \cdot \gamma) + \sum_{n=1}^{|V|}\sum_{s=1}^{S}(t_{n,s} \cdot \varphi_s \cdot \theta_s \cdot W) + \sum_{d=1}^{D}\sum_{p=1}^{P}\sum_{c=1}^{W}(x_{dpc} \cdot r_{dp} \cdot \mu) +$$
$$+ \sum_{d=1}^{D}\sum_{p=1}^{P}\sum_{c=1}^{W}\sum_{k=1}^{K}\sum_{g=1}^{W}(z_{dpckg} \cdot r_{dp} \cdot \mu)$$

By incorporating the cost of regenerated paths to the objective function, the solution to the minimization problem will pick those lightpaths that – besides satisfying all network design constraints – also minimize the total cost of regeneration.

3 Resilient Traffic Engineering

While the previous section coped with efficiently designing resilient optical networks, this section assumes an already designed and capacitated network, where efficient traffic engineering must be applied. Traffic engineering in WDM optical networking is synonymous to providing a solution to the online RWA (Routing and Wavelength Assignment) problem. Previous studies [D5.3] in the Phosphorus project have elaborated in the problem, however without addressing the critical aspect of resilience.

In this regard, this section delves into the problem of connection provisioning in WDM optical networks, with the additional requirement of allocating enough resources that will be used in case of failure to recover potentially affected connections. Additionally, this has to be accomplished efficiently. Section 3.1 addresses the problem of double link failures and quantifies the extent, to which such incidents affect provisioned connections. Otherwise, the rest of this section is devoted to single failure, as their higher frequency of occurrence merit a more detailed study. Section 3.2 addresses traffic engineering in the presence of physical impairments, demonstrating the necessity of incorporating the awareness of physical-layer phenomena into the problem of traffic engineering. The last three sections, namely sections 3.3, 3.4 and 3.5 introduce priority classes into the problem, assuming diversification of resilience requirements among connection requests, and demonstrate that differentiated provision of resilience can significantly improve resource utilization.

3.1 Path Provisioning under Multiple Link Failures

As already stated in Section 1.1.2, the service offered over a lambda Grid infrastructure may not only be disrupted by single equipment failures, but also by concurrent multiple failures. The failures may be associated either with computational infrastructure (e.g. clusters, storage equipment, laboratory devices) or with the optical network used to interconnect computing sites. This subsection focuses on the network part. More precisely, we review existing approaches for mitigating dual link failures in optical WDM networks. Subsequently, we quantify the extent of the catastrophic consequences that dual link failures have on networks that have been engineered to sustain single link failures.

3.1.1 Related Work

In [Doucette03], SBPP (Shared Backup Path Protection) is presented, as an extension of 1+1 APS (Automatic Protection Switching), where the spare capacity of the backup paths is shared among failure disjoint primary paths. In this context, and after dual-failure considerations, the authors give a definition of dual-failure restorability of survivable network designs. They also quantify the consequences to a specific dual failure scenario and define dual-failure restorability of

the affected service paths. This definition results in the probability that a given path is down due to a dual failure, which is defined as the likelihood that a dual span failure state exists, times the likelihood that the particular path is not restorable. This expression is the direct practical interpretation of answering the question “on average dual failure combination, what fraction of the affected demands will experience a service disruption?” Then, a comparison of dual-failure restorability between SBPP and span-restoration follows, observing that about 70-80% of SBPP service paths withstand a dual-failure that affected them and that the restorability of dual span failures for SBPP is around 20% lower that for span restoration. Finally, the authors expand the SBPP approach and show that if they change the SBPP design in order to limit the maximum sharing relationships, then the dual-failure restorability can be improved, with a trade-off in the capacity required to serve the same demand volume.

In [Kim03], the authors, after classifying double link failures in two categories, fundamental (disconnection and capacity) and algorithmic, they present some measures in order to evaluate the performance of dynamic protection reconfiguration (since it is more capacity-efficient than pre-calculated) with four protection algorithms (Dedicated Path Protection, Shared Path Protection, Dedicated Link Protection and Shared Link Protection). The results show that all four algorithms achieve over 96% recovery ratio in the cost of only 4% additional capacity, showing that this may be a very promising technique in the way to more reliable networks under the assumption of double-link failures.

[Frederick04] evaluates and compares the performance of sub-graph routing and backup multiplexing techniques under double link failure scenarios. The authors assume periodic faults and compute restorability by successively having each link fail in the network, computing the coverage for a second failure and averaging under the consideration of all possible failure scenarios. Their results indicate high levels of dual failure restorability for both algorithms tested (60-95%). As in most of the related work presented here, the techniques studied are evaluated on networks designed to tolerate single failure scenarios, since complete dual failure restorability by design requires a lot of spare capacity.

3.1.2 Approach

As indicated in the previous subsection, shielding an optical Grid against dual link failures may be achieved to a satisfactory level by deploying specialized recovery (mostly restoration) techniques. While this is technically viable, it will at the same time dramatically increase the cost of the entire investment, mainly due to the increased spare capacity required and due to the increased implementation costs, questioning thus the viability of the entire solution due to economic factors. Particularly for the case of concurrent dual (and generally multiple) optical link failures, this cost overhead is not well justified, considering that the frequency of such incidents is considerably low compared to single failures [Bouillet07]. Motivated by this statement, this work quantifies the ability of 1+1 protection – i.e. protection against single-failures – to sustain dual link failures. By doing so, we intend to quantify the extent of failed lightpaths due to dual concurrent failures and through that

to rate the appropriateness of 1+1 protection to offer a certain degree of sustainability in the case of dual failures.

Towards this, we used discrete-event simulation to simulate path provisioning in the Phosphorus European test-bed topology (see Figure 8). The duration of each simulation instance was set to 30 time units, with traffic requests being generated according to a Poisson process. The rate of the Poisson arrivals together with the lifetime of each request formed the independent parameters of our experiment, yielding load values that ranged from 50 to 300 Erlang. The number of wavelengths per fibre was set to 16. We tested two scenarios for connection provisioning, both using 1+1 protection:

1. The first scenario uses shortest path routing for primary paths (using hop-count as routing cost) and least-congestion (load-balancing) routing for the backup paths. We used first-fit wavelength assignment for the primary and last-fit for the backup paths.

2. The second scenario uses impairment-aware routing (IAR) for routing primary paths and least-congestion (load-balancing) routing for the backup paths. The wavelength assignment scheme used was the same as in the first scenario.

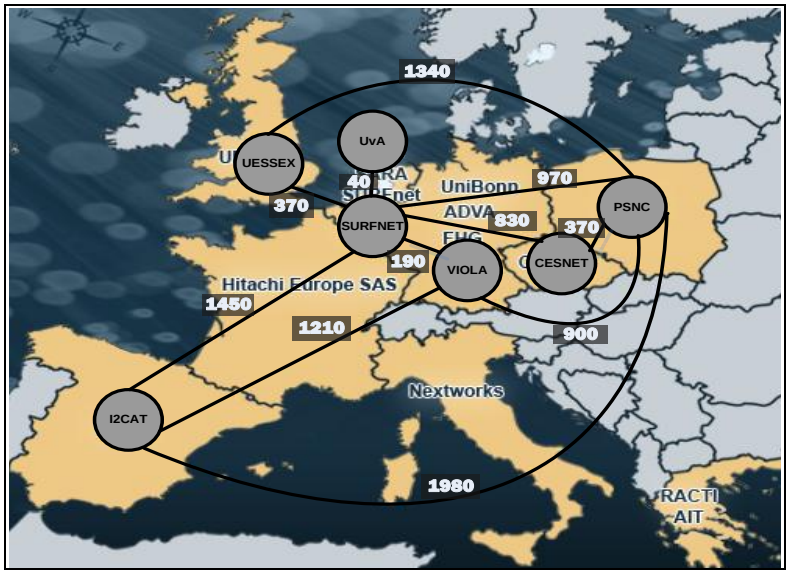


Figure 8: Phosphorus European test-bed topology (link labels correspond to link lengths in km).

3.1.3 Results

Figure 9 illustrates the results of the first scenario of our simulations, namely under the use of min-hop routing for routing primary paths. We first plot the blocking probability experienced by incoming requests due to unavailability of resources (spare wavelengths along shortest paths), both for routing the primary and the backup path. As manifested by the red and green curves, the

network experienced relatively low blocking that increased slightly as the load imposed to the network grew. Additionally to blocking due to resource unavailability, we also plot the average percentage of provisioned connections that are found to be impaired. The latter is based on an a-posteriori calculation of BER (Bit Error Rate) on each lightpath that is provisioned. Clearly, a constantly prohibitively large fraction of installed lightpaths (over 52% across all loads) – either primary or backup – has been found to exceed the BER threshold of 10^{-15}. Most importantly, we additionally plot (denoted by the magenta curve in Figure 9) the average fraction of lightpaths that have been affected by a concurrent failure of two random links. Given the two links that were taken down at some random time point *t* well-off after the start of simulation, the curve captures among all lightpaths provisioned at time *t,* the fraction of lightpaths that had either of the two links as part of their primary or backup path. As manifested by this curve, the fraction of affected lightpaths under random double-failure incidents was close to 19%. The fluctuation of the curve versus load is believed to be an artefact of our simulation and as also manifested by the results of the second scenario, we expect the fraction to remain almost constant independent of the load.

Figure 10 plots the blocking results obtained by simulating scenario 2, where IAR is used for routing primary paths. The considerably lower total blocking probability (which inevitably incorporates the inability to find a path with acceptable BER) compared to the respective curve shown in Figure 9 fully justifies the use of impairment-constraint routing, assuming that signal quality is a requirement of the Grid application. However, pertaining to the effect of double link failure incidents, the results are pretty similar as those of scenario 1, i.e. were close to 16% and remained constant across load.

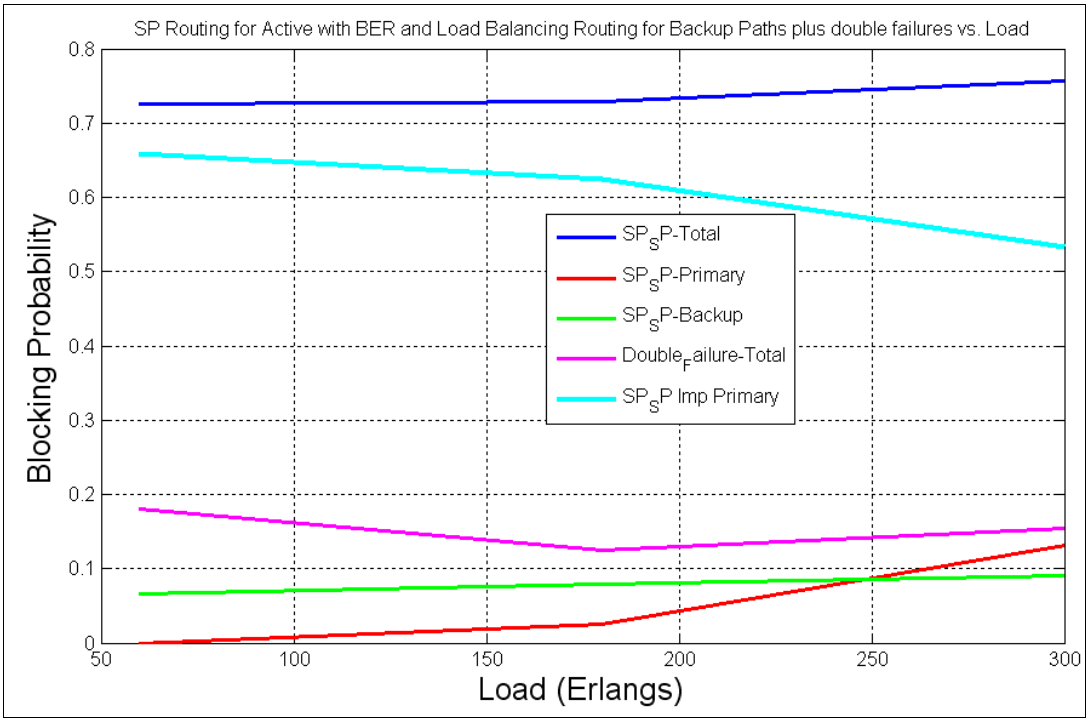


Figure 9: Blocking probability and failure probability due to double failures, when shortest path (min-hop) routing is used to route primary paths (1+1 protection scheme).

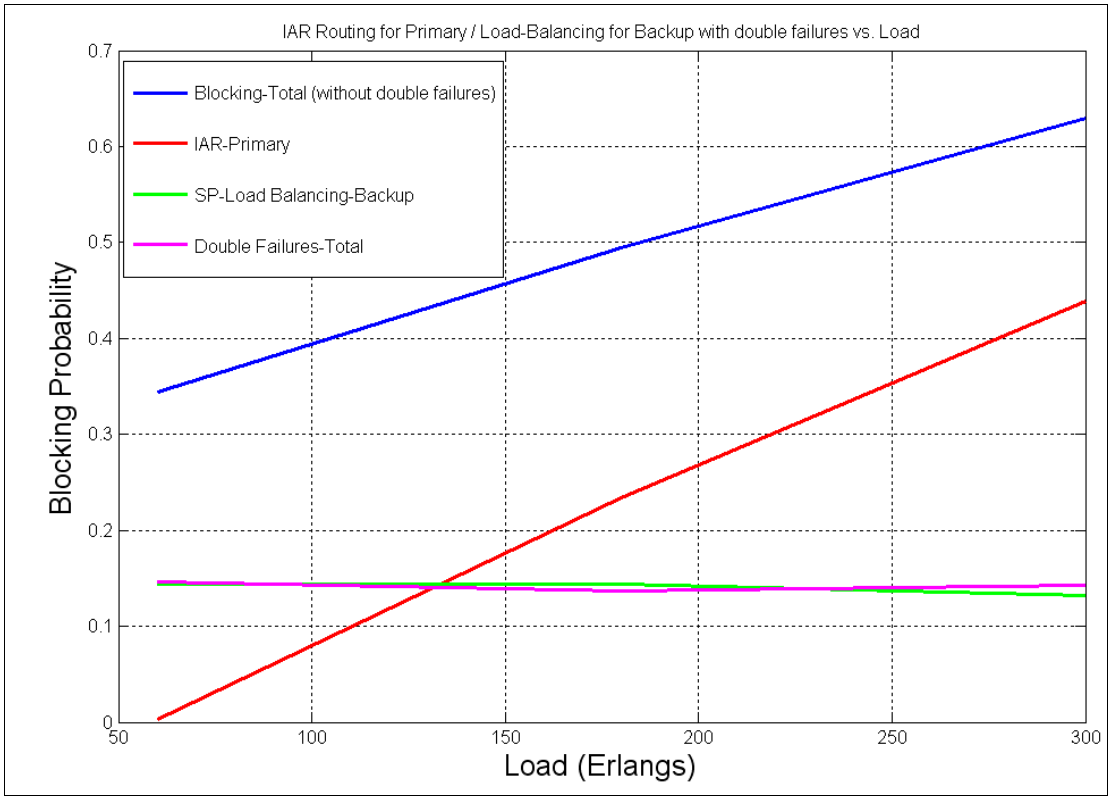


Figure 10: Blocking probability and failure probability due to double failures, when impairment-constraint based routing is used to route both primary paths (1+1 protection scheme).

3.1.4 Conclusions

Motivated by the relatively lower – compared to single link failure rate – frequency of concurrent double link failures, we quantified in this section the influence of double failures to single-backup path protected connection provisioning. Our evaluation results, obtained for the Phosphorus European test-bed topology, indicate that the fraction of affected lightpaths at the event of a double failure is lower than 20%, consistently for both routing schemes used and independent of the load values tested. This finding is to be interpreted by the Grid network infrastructure designer or service provider in combination with the expected cost and revenue, if double-failure recovery mechanisms are incorporated into routing. Specifically, the projected over time investment (in terms of additional spare capacity and operating costs) for implementing protection/restoration mechanisms against double failures has to be compared to the penalty of having 20% of provisioned connections fail, when a double-link failure occurs.

3.2 Physical Impairments Aware Resilient Routing

The already high complexity of the resilient path provisioning problem in WDM networks is only exacerbated when considering the unique characteristics of the optical medium. Physical impairments may degrade the quality of the signal carried by the optical network, leading to either excess overhead to higher layers or worst-case to practically unusable paths. In any case, the effect of physical impairments could be detrimental to a Grid application.

3.2.1 Motivation and Problem Statement

The effects of physical impairments have been well studied in the context of the Phosphorus in previous work [D.5.3] with regard to signal quality on the primary path. In the work presented here, we extend the focus to the paths provisioned or otherwise reserved for resilience purposes (backup paths). Since these paths will be actively used in the case of primary path failure, their signal quality is of equal importance as the quality of the corresponding primary paths, if seamless QoS provisioning is a requirement. Our previous work on resilience requirements of traffic requests in WDM networks has revealed that the protection paths are highly susceptible to physical layer impairments as they are commonly longer that the primary paths [Mar08]. This has a direct impact on the overall network performance in terms of blocking probability as a number of protection paths and therefore the corresponding primary paths are blocked due to unacceptable signal quality. To overcome this issue we propose for the first time to jointly address resilience and physical layer performance requirements through the design and implementation of a suitable RWA method. The performance of the proposed algorithm is compared with conventional routing approaches and evaluated through simulations exploring relevant trade-offs. Significant network performance improvement in terms of blocking probability reduction is shown for specific network conditions.

This section focuses on proactive protection: at the time that the primary path is discovered, one or more alternative paths—backup paths—are also identified and the relevant network resources are reserved for protection purposes in case of a failure. The specific protection method applied is path-based and employs shared protection known as backup multiplexing. According to the shared method and under the single link failure assumption, if two or more primary paths are link-disjoint their backup paths can share wavelength channels, to provide improved resource utilization, as introduced in [Han97]. This work, in addition to taking into consideration the protection requirements of the connection requests, jointly performs routing and wavelength assignment for both primary and protection paths considering their physical performance. More precisely, not only the availability of optical bandwidth is considered, before primary connections and their protection paths are established and reserved respectively, but also their quality in terms of bit error rate (BER).

3.2.2 Approach

The BER of primary and protection paths is calculated through the quality factor Q and compared against a predefined threshold value to decide whether they are of acceptable quality. Here, the BER threshold value is set to $B_{thresh}=10^{-15}$. The analytical model of Q-factor for the performance evaluation of a static unicast IA-RWA has been used to integrate different types of degradations [D5.3]. The impairments considered in the Q-factor evaluation include amplified spontaneous emission noise (ASE), cross-phase modulation (XPM) and four-wave mixing (FWM) assuming that they follow a Gaussian distribution. Also, optical filtering and the combined self-phase modulation/group velocity dispersion (SPM/GVD) effects were introduced.

To evaluate the effectiveness of the proposed solution two cases are studied: a) IA-RWA applied for both primary and protection paths and b) IA-RWA used for the primary path and minimum hop routing applied to the protection paths. Simulation results show substantial blocking probability reduction when IA-RWA is used for both primary and protection paths.

3.2.3 Algorithm Specification

Our work has concentrated on solving the online version of the RWA/resilience problem, i.e. traffic demands arrive and get served sequentially without knowledge of future incoming requests. This makes our contribution valid for use mostly in the context of traffic engineering. In addition, it is assumed that only a single link could fail at any instance of time and re-routing of already established connections is not allowed. The model assumes there is no wavelength conversion capability of the network and thus wavelength continuity across any path is a tight constraint in the problem definition.

We assume that all requests have a bandwidth demand of one wavelength unit and for each request a link disjoint backup path is required along with its primary path to provide guaranteed protection. The physical bandwidth of each link l can be divided into the following three parts: A_l, B_l, and R_l [Mar08]. A_l represents the total amount of reserved bandwidth dedicated to primary paths carried by link l and it is not allowed to be shared. B_l is the total bandwidth occupied by all backup paths on link l and unlike A_l it can be shared by backup paths, whose associated primary paths are link disjoint. The residual bandwidth R_l is the difference between the physical bandwidth on link l and the total consumed bandwidth (A_l + B_l). For any future primary path established on link l, R_l is the only available bandwidth that can be used. For setting up a backup path on link l for a new primary path a, the available bandwidth $S_{l,a}$ consists of two components: the residual bandwidth R_l and the portion of B_l that is able to be shared for carrying this backup path. To identify path costs, the relevant link weights are identified for both primary and backup paths. As primary paths do not share bandwidth their cost is the sum of the weight of each link they traverse. In the case of backup paths we give preference to wavelengths that have already been allocated as backup wavelengths by assigning to them a lower weight and therefore reinforce sharing.

The routing and wavelength assignment problems are solved in two separate steps. Routing is implemented based on the Dijkstra's algorithm to compute a primary and a backup path for a given demand. The wavelength assignment algorithm assigns wavelengths to the primary and backup paths allowing resource sharing as explained above between the current demand and the already established requests. Connection requests follow a Poisson arrival process with time duration that follows an exponential distribution. In the primary computation phase a primary lightpath is provisioned for each request. In this phase impairment aware routing (IAR) is performed by assigning the Q penalty as the link cost and the Dijkstra algorithm is deployed on the weighted graph to calculate the shortest path. If no path is found, the connection is blocked. If at least one path is found, a list of possible wavelengths that can be allocated is identified and the first wavelength is chosen applying the first fit (FF) wavelength assignment (WA) algorithm to form the primary lightpath. Furthermore a module that monitors the bit error rate (BER) of the provisioned primary path is involved that checks the path quality and decides whether the path satisfies the quality constraints against the predefined BER threshold. Subsequently, the backup computation phase starts with identifying the portion of the backup bandwidth that can be shared and with the exclusion of the links utilized by the primary path. This results in an auxiliary graph representing the current network state. In the case of protection paths we test two routing algorithms: minimum hop routing reinforcing sharing as described above and IAR. After the link costs are assigned if no lightpath is found for any wavelength, the connection is blocked due to backup path blocking. In case of discovery of multiple backup lightpaths, the algorithm allocates one wavelength, based on the last fit (LF) WA scheme. The LF WA algorithm has been used as it has been shown that when used in conjunction with the FF WA algorithm for the primary paths, it maximizes the backup path link reuse. As in the case of primary paths a module that monitors the BER of the selected protection path checks the path quality and decides whether the path satisfies the requirement of the predefined BER threshold.

3.2.4 Evaluation Results

The results presented in this section, are generated based on the Pan-European test network defined by the COST-239 action [Batchelor00] (see Figure 11), assuming 16 wavelengths/fibre. Links are considered bidirectional and if a link failure occurs, the traffic flow in both directions will be disrupted. The results shown in the following figures are the averages over 10 statistically independent repetitions of a provisioning experiment, and for various loads imposed on the network.

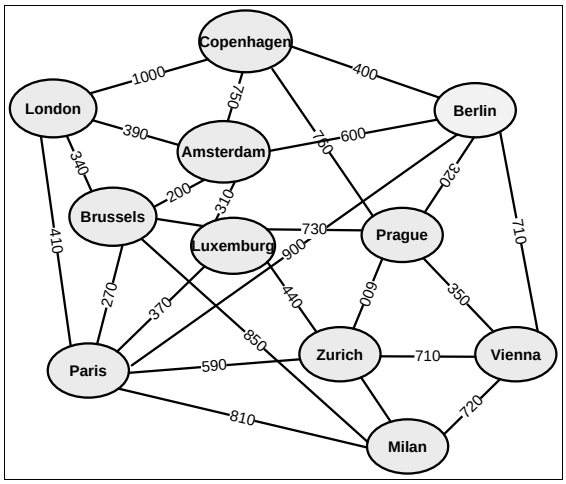


Figure 11: COST-239 European optical network topology.

Figure 12 illustrates how the network blocking probability varies with traffic load, concentrating on the blocking probability of protection paths and the total blocking probability of both primary and protection paths. The results shown were taken assuming that IAR has been used for the primary paths, while two different routing approaches have been used to discover the protection paths: minimum hop routing with reinforced wavelength sharing and IAR. These results clearly indicate that when evaluating network performance it is important to include protection requirements, as allocation of protection capacity has a significant contribution to the total blocking probability of the network. In addition, Figure 12 demonstrates that even if IAR is used as the routing approach for the primary paths it is important to consider the effect of the physical impairments also in the protection path. More specifically, in case of minimum hop routing for the protection paths, when BER monitoring is applied to ensure acceptable signal quality, the blocking probability of the protection paths becomes high. This is due to that a large number of protection paths do not satisfy the BER threshold criterion, resulting thus in blocked connections. It should be noted that in general protection paths are longer than primary paths, exhibiting higher probability to be impaired. This has a significant contribution to the total network blocking probability. An alternative approach that can improve the overall network performance is to apply IAR not only to the primary but also to the protection paths. As manifested by Figure 12, this approach offers blocking probability reduction of the order of 15% for low loading conditions, corresponding to a blocking probability decrease of about 42%. The benefit starts decreasing for higher loading conditions as in this case there is a smaller reserve of alternative paths that can be exploited; also this is due to the fact that in general minimum hop routing provides the ability to allow a form of load balancing in the network [Mar08]. Through this discussion and the results shown in Figure 12, it becomes clear that the use of IAR for the protection paths has a benefit in terms of blocking probability. However, this comes at the expense of network resource sharing. As depicted in Figure 13, minimum hop routing offers reduced total number of link wavelengths that are uniquely

allocated to protection paths and thus enables increased resource sharing, compared to using IAR routing for the protection paths.

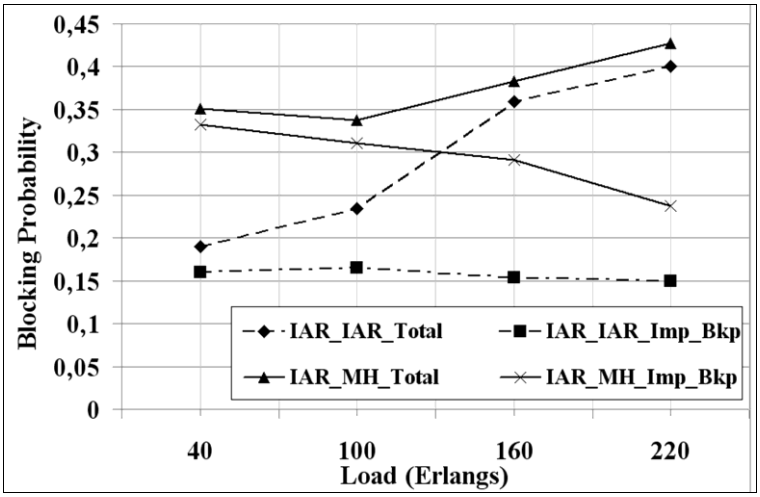


Figure 12: Blocking probability for the IAR-IAR approach (using IAR for both primary and backup paths) and the IAR-MH approach (using minimal hop for the backup path). The Bkp curves indicate the portion of total blocking due to unacceptable signal degradation on the backup.

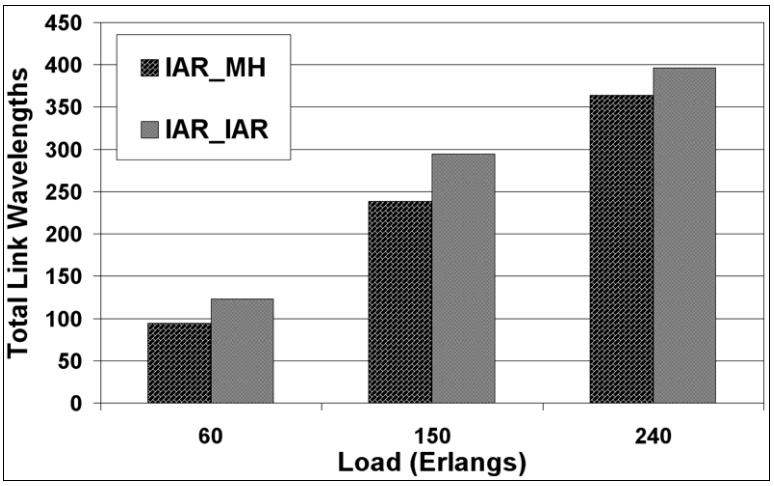


Figure 13: Total link wavelengths uniquely allocated to protection paths (not shared).

3.2.5 Conclusion

In this section, we tested the effect of using impairment-constrained routing not only for assigning routes to working paths, but also to protection paths. More precisely, the approach was applied in the

SBPP (Shared Backup Protection Path) protection scheme. The motivation for testing such an approach includes the same rationale common to impairment-aware routing in general; namely the optimization of dynamic path provisioning not only with regard to efficient resource sharing, but also regarding the peculiarities of the optical medium.

As manifested by our results, a cross-layer approach in backup path computation manages to reduce total blocking, as long as the network is operated at low to medium loads. As the load increases further, blocking increases due to unavailability of spare wavelengths. The latter is due to the fact that IA-routing does not reinforce sharing of resources between backup paths, whose primary paths do not share common risks. In any case, the impairment-aware approach did never yield worse blocking ratios than the min-hop backup routing approach throughout our simulation experiments.

3.3 Differentiated Survivability Services in WDM networks

Applying service differentiation to network services in the context of optical Grid systems (based on WDM technology), is an option towards reducing the cost of the required infrastructure resources, while satisfying the diverse availability and resiliency requirements of the applications. This work is based on the backup multiplexing technique in order to facilitate efficient resource sharing and investigates different routing and wavelength assignment schemes that considerably enhance the spare capacity utilization. A simple approach that can be used to assign different classes of service supporting varying restoration requirements is proposed and significant network performance improvement is demonstrated through relevant simulations.

3.3.1 Background

Optical networking employing wavelength division multiplexing (WDM) is capable of carrying tremendous amount of information and is expected to be extensively used to support the requirements of next generation networks and the future Internet. The deployment of WDM technology enables the routing of multiple lightpath connections utilizing different wavelength channels in an optical fibre. In WDM networks, a number of issues need to be addressed when provisioning lightpaths. One of these is the requirement that a lightpath must occupy the same wavelength across the selected path due to the immaturity of all-optical wavelength conversion technologies, which is known as the *wavelength continuity constraint*. Given a connection request from source to destination node in such a network the problem of computing a route and assigning a wavelength to the connection is the so-called *routing and wavelength assignment* (RWA) problem. This work considers mesh WDM networks with dynamic traffic conditions and without any wavelength conversion capabilities. The physical links are assumed to comprise a fixed number of wavelengths and when a new connection request arises, an appropriate route and an available wavelength are selected to form the lightpath. The primary objective of this approach is to identify efficient RWA schemes able to minimize the blocking of new connections due to bandwidth limitations.

WDM networks with dynamic traffic patterns can easily provide guaranteed timeliness using simple resource reservation schemes and dedicating the entire bandwidth of a lightpath to a certain application.

This partially satisfies real-time critical applications, which usually require not only strict timeliness but also fault-tolerance. Fault-tolerance is an essential requirement in high-speed networks since a single link failure causes loss of services that carry an enormous amount of information and thus may lead to significant revenue losses. Therefore it is indispensable for WDM networks to have resilience mechanisms in place to be able to reroute/restore the affected traffic upon a failure. These resilience mechanisms can be classified according to the different requirements requested by various applications supported by the network. Ideally it would be desirable to provide a 100% resilience guarantee to all types of traffic supported by existing and future networks, but this may be unnecessary and wasteful in terms of resource utilization resulting in cost inefficiencies. Thus, a more efficient resilience scheme suitable for a network supporting a variety of applications would provide different level of network survivability to different traffic types in accordance with the respective Service Level Specifications (SLS) maximizing the network utilization [Zhang02]. Therefore in a network environment such as the new global and business oriented internet, an important requirement will be to provide differentiated survivability services to different types of traffic enabling higher priority demands to exploit higher network availability [Fuma06] [Pandi06]. The first part of the work is focused on fault-tolerance of high priority, high resilience traffic through the backup multiplexing technique [Han97]**.** The use of the backup multiplexing technique is selected in order to facilitate efficient resource sharing. In this framework different routing and wavelength assignment schemes that considerably enhance the spare capacity utilization are investigated and proposed. Through our novel wavelength assignment scheme (that dedicates a consecutive number of wavelengths to protection lightpaths), a performance improvement of about 4% to 14% is observed compared to commonly used techniques. Moreover a simple method that can be used to assign different classes of service supporting varying restoration requirements is proposed and a significant network performance improvement is demonstrated through simulation. More specifically, our extensive simulations revealed a significant blocking improvement of up to 12% by assigning pre-emption authority to high-priority traffic over lower-priority traffic due to the accomplished efficient resource reuse.

3.3.2 Survivability Scheme

Network survivability can be defined as the capability of the network to provide continuous services in the presence of failures resulting in lightpath disruptions. Due to the large amount of traffic at the lightpath level, resilience mechanisms are highly critical and many schemes have been proposed to address this issue [Mohan00]. Survivability can be broadly classified in two types: dynamic restoration and pre-designed protection. In dynamic restoration [Rama99] [Iraschko00], the backup lightpath discovery procedure is initiated after a primary lightpath fails. This procedure might not result in backup lightpath identification due to lack of network spare capacity and therefore this method does not guarantee successful recovery. On the other hand, in pre-designed protection [Rama99-b] [Doshi99] [Kodialam00-a], a backup lightpath is computed and reserved at the time of establishing the primary lightpath. If a backup lightpath cannot be found under current network conditions, the connection request is blocked. A database of restoration paths for this method can be populated by dynamic restoration. Hence it is possible to implement a single restoration algorithm to be used “pre-emptively” before a failure occurs, as part of a pre-designed protection method and dynamically after the occurrence of a failure not previously considered. The advantages offered by the pre-designed protection method compared to dynamic restoration are the shorter restoration times and the 100% restoration guarantee.

A further classification of the pre-designed protection method is performed based on link or path protection schemes. In the link-based method, the failed link is replaced by a new path, which is then merged with the unaffected portion of the primary path, to constitute the backup path. This method constrains the choice of the backup paths and requires more spare resources than the path-based method [Iraschko98], which computes a complete end-to-end backup path from the source to the destination of the failed primary path. In the path-based method, wavelength channels on the backup path can be either dedicated or shared. In the dedicated case, the wavelength channels assigned to a specific backup path cannot be assigned to other backup paths, whereas in the shared method, backup paths can share wavelength channels under the single link failure assumption, if their primary paths are link-disjoint. This is known as *backup multiplexing* and provides improved resource utilization [Han97]. Specifically in [Rama01], it was shown that the total resource requirement for the dedicated backup method is 260-265% of the requirement without lightpath protection, and it can be reduced to 186-195% by considering backup multiplexing.

In this work, survivability is provided by implementing the backup multiplexing technique under dynamic traffic demands, where existing lightpaths cannot be rerouted and future lightpath requests are not known. We differentiate traffic demands to three classes of service in what concerns network recovery performance, and adopt the concept of resilience priority classes to maximize network resource utilization. We considered three types of lightpaths: 1) high priority protected lightpaths, 2) unprotected lightpaths and 3) low priority pre-empted lightpaths. A high priority protected lightpath has a working path and a diversely routed backup path. The wavelength channels on the working path of the high priority protected lightpath are dedicated to that lightpath and carry user traffic under normal operating conditions. Both the working and the backup lightpaths are identified before the provisioning of the working path. In this case the wavelength channels on the backup path are shared among different high priority lightpaths. Wavelength channels are shared to ensure that any single fibre link failure on the working path of any high priority lightpath can be restored. An unprotected lightpath is not protected with a backup path and upon any failure along the lightpath, a dynamic restoration mechanism is initiated to provide an alternative route without any guarantees. Finally low priority pre-empted lightpaths are unprotected lightpaths that allow pre-emption of their utilized resources in case of a high priority lightpath failure. Under this scheme, the wavelength channels allocated for the low priority lightpaths can be shared with the backup routes of the high priority lightpaths. Note that class 3 is actually a special instance of class 2 traffic: both classes are unprotected, and their working paths can either be shared with class' 1 backup paths (class 3), or not (class 2).

3.3.3 Algorithm Specification

In this section, we describe our routing and wavelength assignment algorithm that computes a primary and a backup lightpath (if required by a given traffic demand) and assigns wavelength channels to these paths.

Initially we introduce the main definitions and assumptions used by our algorithm. We assume that all requests have a bandwidth demand of one unit and can be classified to class 1 if a link disjoint backup path is required along with their primary path to provide guaranteed protection, or class 2 if just dynamic restoration is acceptable. The physical bandwidth of each link l can be divided into the following three parts: Al, Bl, and Rl [Li05]. Al represents the total amount of reserved bandwidth dedicated to primary paths carried by link l and is not allowed to be shared. Bl is the total bandwidth occupied by all backup paths on

link l and unlike Al it can be shared by some backup paths, provided that their associated primary paths are disjoint. Specifically if two primary paths share a common link (so they are not disjoint) they will be both affected by a single network fault on this link. Therefore, their backup paths cannot share any common bandwidth since it will be necessary for both paths to be activated simultaneously in case of their common primary link failure. Finally, the residual bandwidth Rl is the difference between the physical bandwidth on link l and the total consumed bandwidth (Al + Bl). For any future primary path established on link l, Rl is the only available bandwidth that can be used whereas for setting up a backup path on link l for a new primary path a, the available bandwidth Sl(a) consists of two components : the residual bandwidth Rl and the portion of Bl that is able to be shared for carrying this backup path. Since primary paths do not share bandwidth, their cost is the number of hops or links that they traverse. On the other hand, the cost of a backup path is the number of free wavelengths used on each link it traverses. If a wavelength is not free and is currently used by some primary lightpath (either of class 1 or class 2), it cannot be used by the backup path. If a wavelength is not free and it is currently used by a set of backup lightpaths S, it can be used by the new backup path with no extra cost (zero cost) if and only if its primary path is link-disjoint with the primary route of each and every backup lightpath in S. If a wavelength is free, it can be used by the backup path with a cost value equal to one. Unlike primary paths, the path cost of a longer backup path may cost less than that of a shorter one, because of bandwidth sharing. This cost function approach leaves a higher number of wavelengths available for use in future requests, thus improving the network performance. The main scope of our proposed RWA scheme combined with our resilience differentiation approach is to maximize resource reuse. Through extensive simulations we demonstrate how the restoration capacity increases the overall network performance.

The proposed algorithm solves the routing and wavelength assignment problems in two separate steps. Routing is implemented based on the Dijkstra's algorithm to compute a primary and a backup path for the given demand. The wavelength assignment algorithm assigns wavelength channels to the primary and backup paths favoring resource sharing among the current demand and the already established requests. We assume that the network nodes have no wavelength conversion capability, therefore a lightpath is not allowed to occupy different wavelength channels along its route.

In Figure 14 the flow chart of the algorithm is presented. After the initialization phase in which the algorithm collects network topology information (i.e. number of nodes, number of links, wavelengths per fibre, network connections, backup path wavelength assignment scheme) and constructs the required matrices to monitor the network state (Al, Bl and Rl), connection requests arrive for random source and destination pairs. First, independent of the request service requirement, a primary lightpath is established through the primary lightpath computation phase. This phase consults the Rl matrix and assigns costs to the network links based on the following approach. If a link has no free wavelengths, its cost is set to infinite and is not considered by the Dijkstra algorithm for the path computation. If available wavelengths exist on the link the cost is set to be inversely proportional to the number of these wavelengths, which results in a form of load balancing. After weights are assigned to the network links, the widest shortest path (WSP) routing algorithm takes place, calculating a number of shortest paths and selecting the first one that traverses the minimum number of hops and for which at least one common free wavelength exists on all its links. If no path is found, the connection is blocked, and the blocking probability due to primary path blocking is increased. If at least one path is calculated, a list of possible wavelengths that can be allocated for it is identified and the first wavelength is chosen (assuming that they are sorted in increasing order) to form the primary lightpath.

After the primary lightpath is ready to be established the Al and Rl matrixes are updated to reserve the appropriate wavelength and the algorithm proceeds to the examination of the demand service requirement. If the request belongs to class 2 traffic and pre-emption is enabled Bl matrix is also informed to allow sharing of the allocated wavelength from future backup paths of class 1 traffic that has the authority to pre-empt class 2 lightpaths.

If the established demand requires a backup path (class 1), the flow control moves to the backup computation phase. Here, the available bandwidth Sl(a) consisting of the residual bandwidth (Rl) and the portion of the backup bandwidth (Bl) that can be shared as described earlier, is first identified excluding the links utilized by the primary path. Then based on this available bandwidth (Sl(a)), for each wavelength an auxiliary graph is generated representing the current network state. For this new topology formulation, link costs are assigned based on the following strategy: On the links for which the wavelength under consideration belongs to Sl, a zero weight is assigned and if it belongs to Rl a unit cost is assumed. On the other hand, links on which the wavelength is already allocated (by primary lightpaths) are not considered in the auxiliary graph and cannot be used for the backup calculation. An attempt to find a lightpath for each wavelength follows and if no lightpath is found for any wavelength, the connection is blocked due to backup path blocking, requesting from the algorithm to roll back the updates of Al and Rl previously performed by the primary path computation phase. In case of multiple backup lightpaths computations the algorithm must allocate one, based on the selected wavelength assignment scheme. If the random pick (RP) wavelength assignment scheme is selected, the lightpath is chosen randomly from the set of the available lightpaths. For the last fit (LF) scheme, the lightpaths with minimum cost are identified and the last one (when sorted in increasing order) is selected, whereas for the first fit (FF) the first one from the minimum cost lightpaths is allocated. In the final step of the algorithm Bl and Rl are updated for the links where residual bandwidth is used.

3.3.4 Performance Evaluation

We performed simulations of dynamic provisioning on several representative backbone mesh topologies. The results presented here are generated based on the Pan-European test network (Figure 15) defined by COST 239 [Batchelor00] that has 11 nodes and 26 links and are representative of results for other mesh network topologies. Links are considered bidirectional and if a link failure occurs the traffic flow in both directions will be disrupted. Lightpaths comply with the wavelength continuity constraint and connections requests are equally likely to have any of the network nodes as its source or destination. Also we assume that calls arrive one by one and their holding time is long enough to consider that accepted calls do not leave (incremental traffic). A connection is blocked if neither a primary nor a backup path can be established. The results shown in the following figures are the average values over 20 independent experiments.

First we explore the behaviour of the three wavelength assignment schemes available to the backup lightpath establishment. First fit is the wavelength assignment scheme used for the primary path establishment through all simulation results presented. In Figure 16 the average blocking probabilities for Last Fit, First Fit and Random Pick are compared for uniform fibre capacities of C=8 and C=16 wavelengths. The LF wavelength assignment scheme provides improved network performance compared

with FF of around 4% and 2% for high network loads for 16 and 8 channels per fibre respectively. In addition, the LF significantly outperforms RP since it improves the blocking probability of 14% and 8% for the two different fibre capacity parameters.

These observations can be explained and validated if we examine the difference in the restoration capacity occurring from the various wavelength assignment schemes. In Figure 17 we present the link utilizations for LF and RP schemes. “Shared Links” refer to the number of links that are used more than once for path formulation, “Not shared Links” represent the links that are utilized by the primary lightpaths, “Able to share Links” correspond to the links that are able to be used for backup lightpaths and finally “Total Links” is the sum of “Not to share” and “Able to share Links”.

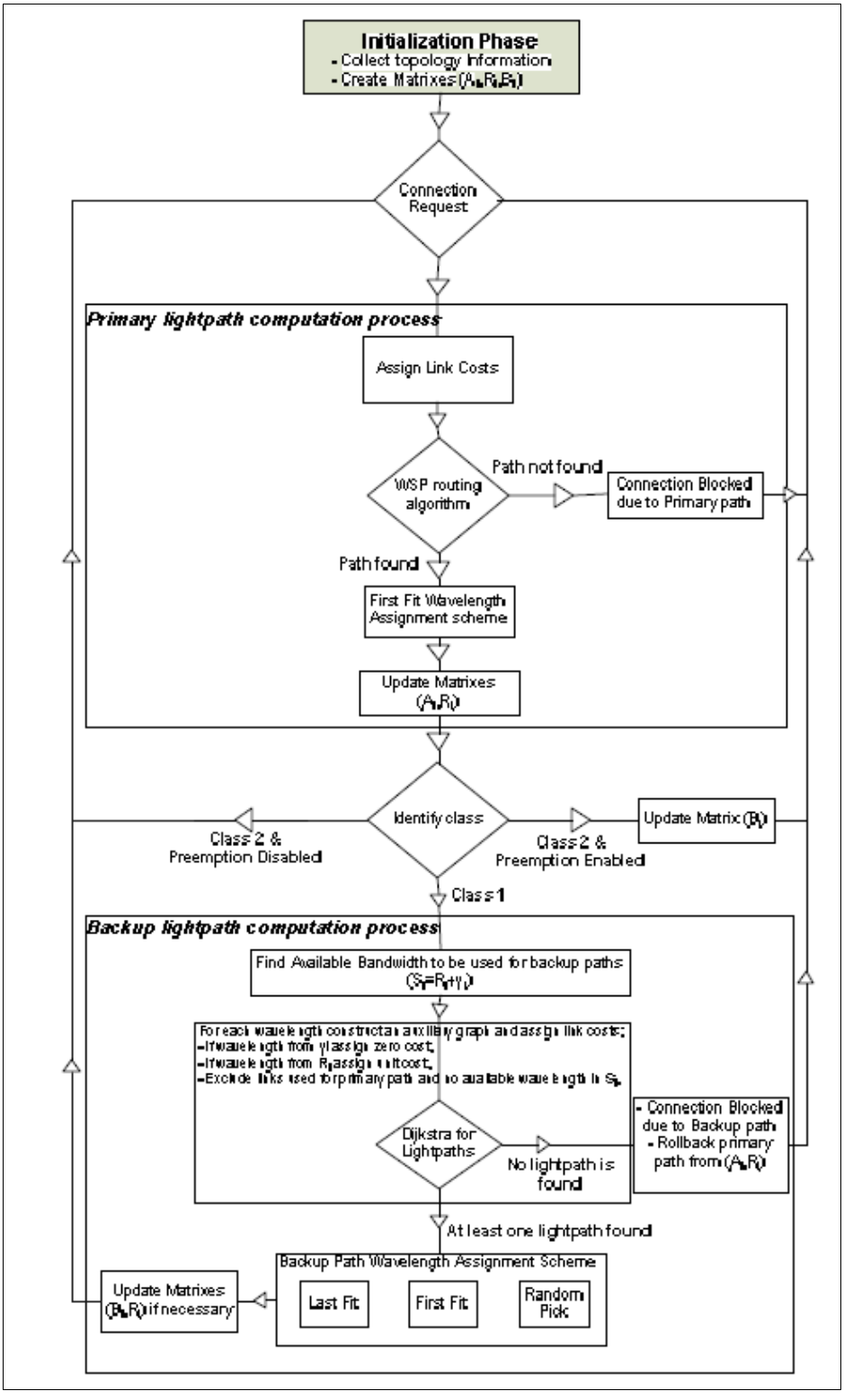


Figure 14: Flowchart of differentiated resilience scheme.

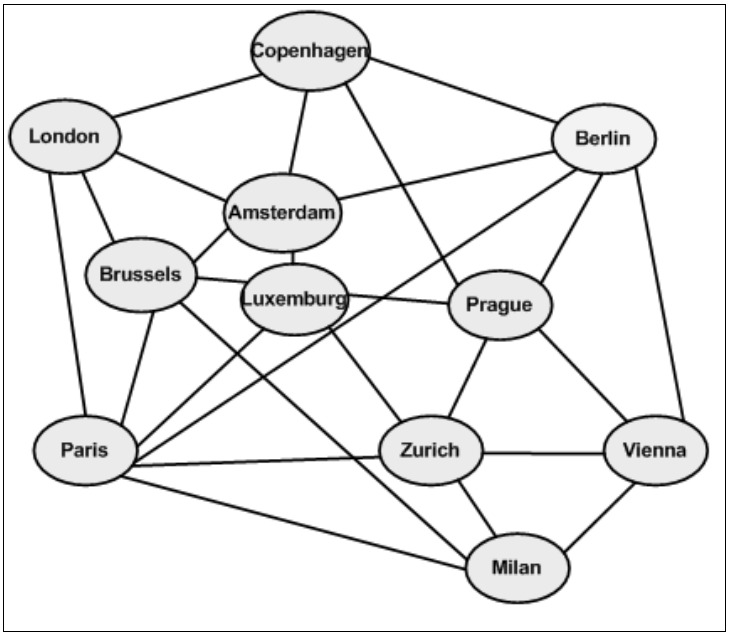


Figure 15: COST-239 Pan-European topology.

It is clear that the LF scheme maximizes the backup path link reuse (1100 compared to 700 of RF for 550 requests) although a small number of links are dedicated for backup paths (180 compared to 300). The increase in restoration capacity of the LF over the RP scheme is around 58% and constitutes the main reason of the lower blocking probability of the LF scheme. LF is a simple and fast wavelength assignment scheme able to considerably increase the backup link reuse by dedicating a small but consecutive portion of the wavelength band to backup paths, allowing a large amount of the precious residual bandwidth for the primary paths that are allocated based on a FF scheme.

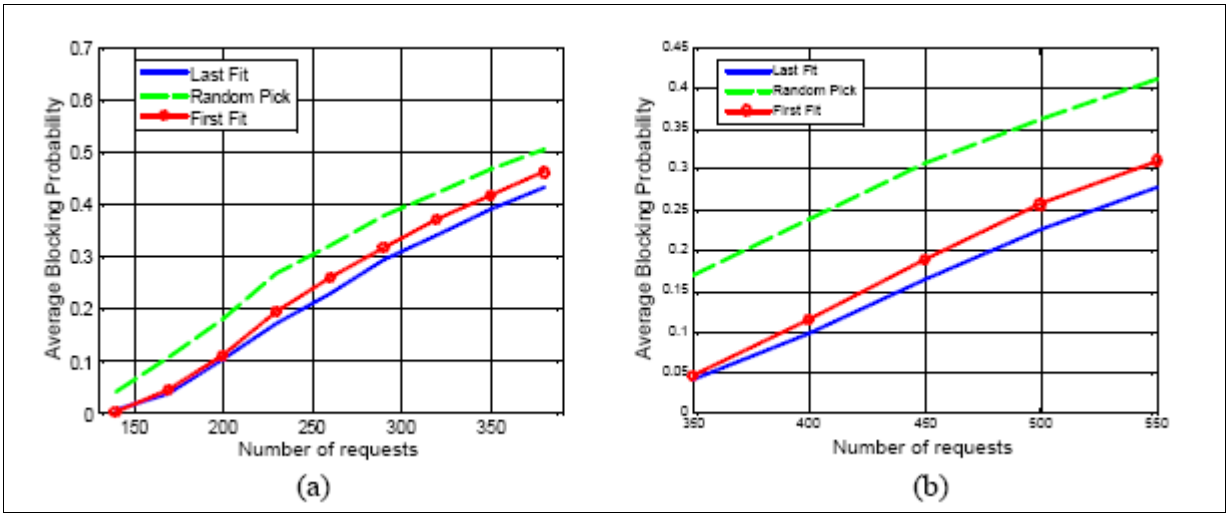


Figure 16: Network performance for the three backup path wavelength assignment schemes and for different fibre capacity (a) C=8, (b) C=16.

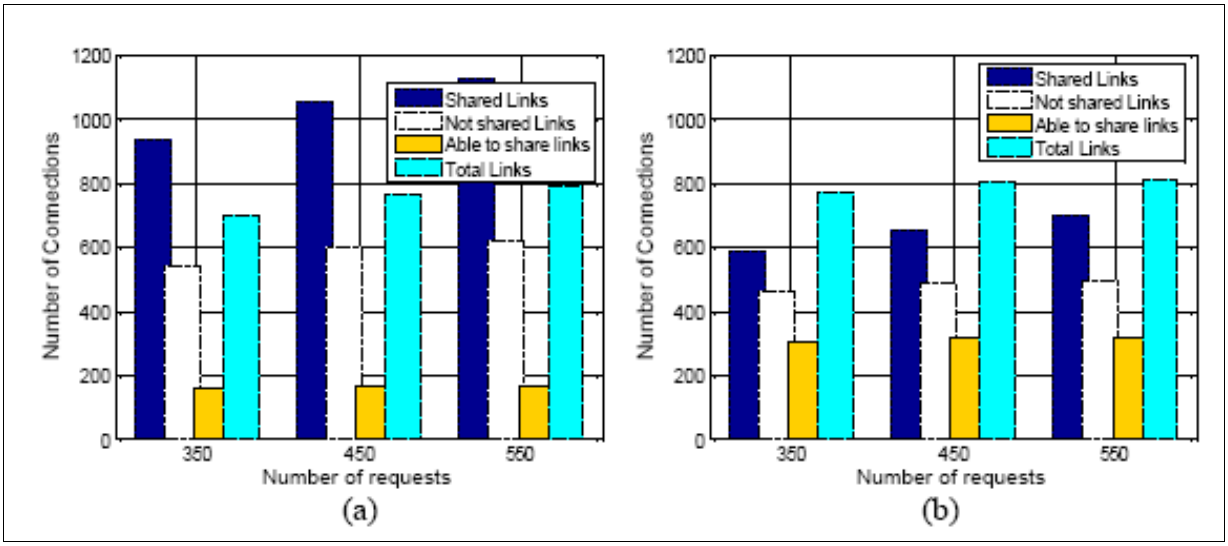


Figure 17: Link distribution flow charts for (a) LF and (b) RP for fibre capacity C=16.

In the next step of our simulations, we analyse the results obtained by considering the coexistence of both class 1 and class 2 traffic, with the pre-emption authority disabled and enabled. In Figure 18 we compare the average blocking probabilities when the class 1 traffic is 50% and 80% of the total requests with the case in which all the traffic is considered as class 1 traffic. The benefit offered by the pre-emption enabled scheme is up to 12% when half of the incoming traffic is assigned as class 1 and up to 8% when 80% is set us class 1. For the non pre-emptive scheme, the benefit reduces to 5% and 3% respectively indicating the superiority of the pre-emptive approach in terms of network performance. This improvement offered by the pre-emptive scheme is at the expense of the reliable provisioning of low priority traffic, which can be tolerated for many non-real time applications. The insights of the pre-emption and the non pre-emption cases are further explored in Figure 19. It can be observed that the pre-emptive scheme, although utilizing a smaller number of links compared to the non pre-emptive case, provides an increase in the link reuse

percentage since it allows the low priority class 2 traffic to be shared among the backup paths of the higher priority traffic. When no pre-emption is allowed, the number of possible shared paths is significantly reduced since only 50% of the total demands require backup paths resulting in inefficient backup resource utilization with considerable impact on the network performance. The increase of the restoration capacity as the network load increases (from 10% to 25%) implies that the benefit of the pre-emptive scheme continues to rise as indicated by the blocking probability curves in Figure 18 a.

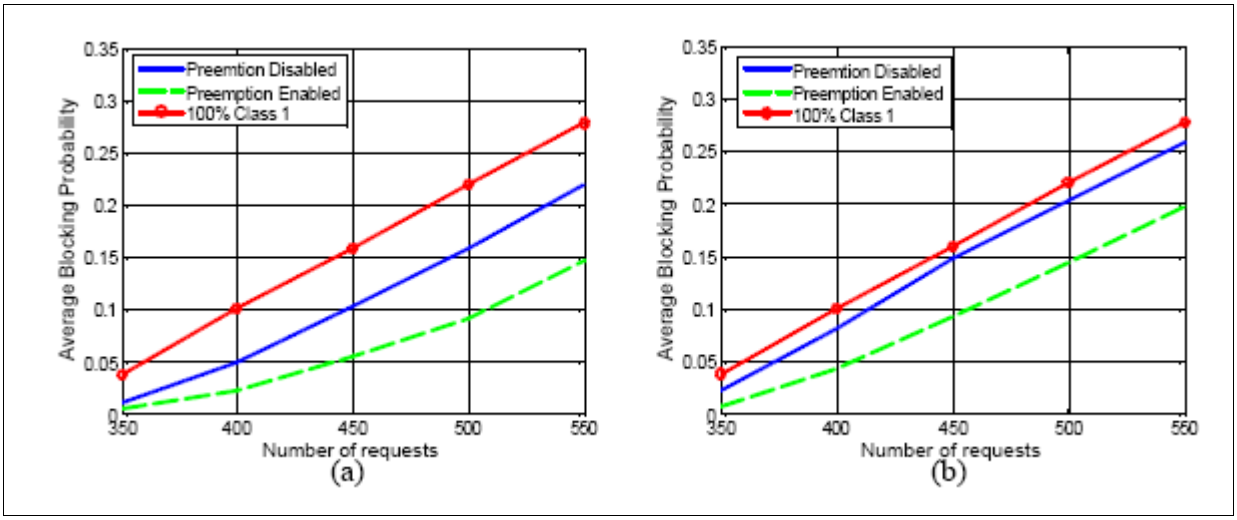


Figure 18: Average blocking probability when (a) 50% and (b) 80% of the requested connections are assigned as class 1 traffic and LF scheme is used for C=16.

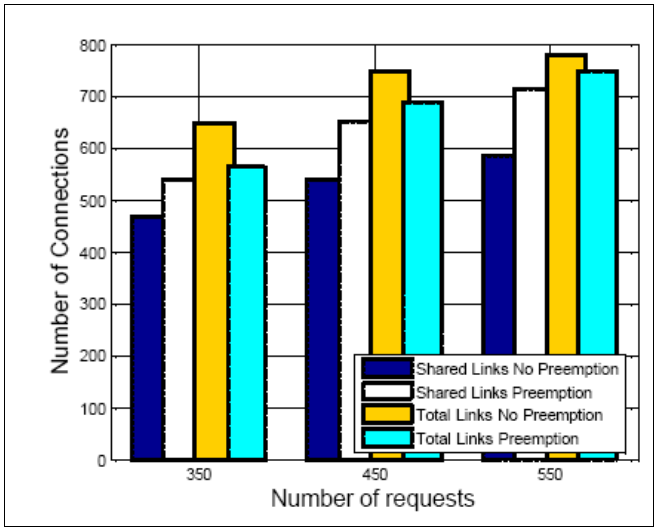


Figure 19: Shared and total link usage when preemption is allowed and not allowed for the case of 50% of class 1 traffic.

Last, we analyse in Figure 20 the blocking probabilities of the different classes coexisting in the network when preemption is allowed. In Figure 20 a, 80% of the total traffic is considered as class 1 and 20% as

class 2. The blocking probability of the class 1 traffic is high compared to the low priority traffic (a difference of about 10% is observed) although the overall blocking is reduced when considering this differentiation scheme. In Figure 20 b the same percentage of class 1 and class 2 demands is assumed and almost the same blocking probability is observed for the two classes, causing a higher reduction in the overall blocking probability. Also in this case the blocking probability of high priority traffic is reduced considerably at least for heavier network loadings (around 8%) whereas the blocking of the lower priority traffic is increased in a much smaller scale (about 4%).

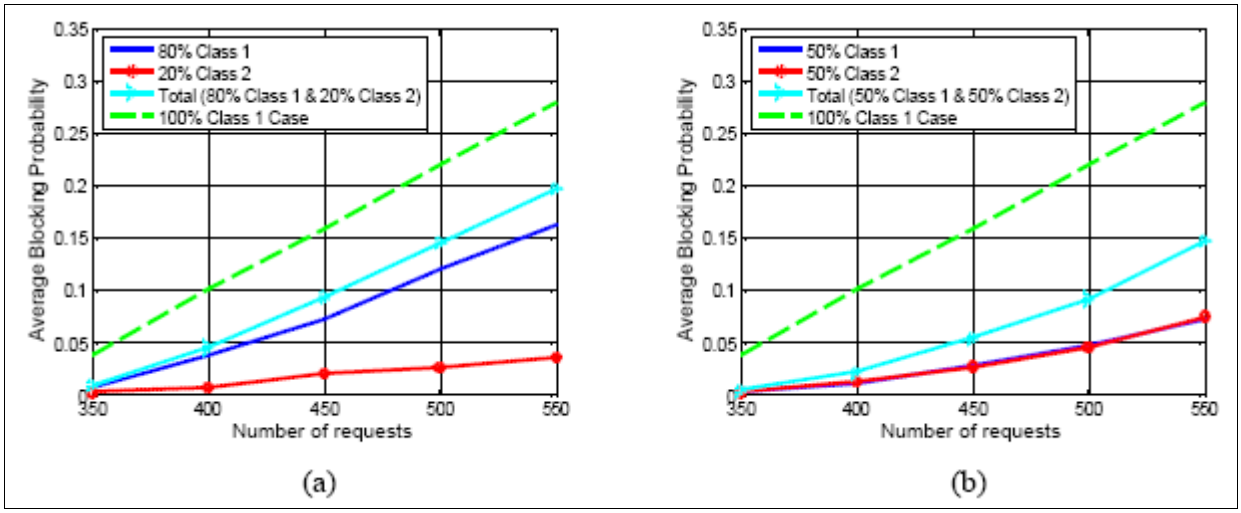


Figure 20: Analyzing the blocking probabilities of the different classes in the network when (a) 80% and (b) 50% of class1 traffic is requested. (pre-emption allowed).

3.3.5 Conclusions

In this section, we addressed the problem of efficiently provisioning lightpaths with different protection requirements in a dynamic WDM network environment. As a first step, various RWA schemes were investigated with the aim to enhance the backup resource utilization and improve the network performance. Our Last Fit wavelength assignment scheme applied on the backup lightpaths, used in parallel with the First Fit assignment method applied on the primary lightpaths, demonstrated considerable performance improvements compared to the commonly used Random Pick and First Fit assignment schemes. Specifically, LF provided a significant benefit of around 14% and 4% compared with the RP and FF cases respectively. We also showed that the improvement of the average blocking probabilities occurs due to the effective capacity reuse offered by the LF scheme over the other schemes used. In the next step of the analysis, the incoming traffic is differentiated to classes of service according to their survivability requirements, and the pre-emption of low priority traffic by higher priority demands in the event of a link failure is proposed. This technique enables backup paths to reuse the already assigned wavelengths of low priority traffic, increasing therefore the reuse of the available network resources. In this case, detailed simulation results demonstrate significant network performance improvements of up to 12% and considerable decrease in the blocking probability of the high priority traffic.

3.4 Differentiated Resilience with Dynamic Traffic Grooming for WDM Mesh Networks

As discussed in Section 1.1.6, the differentiation of resilience services becomes a major architectural and design issue. [Cholda07] presents a comprehensive survey of research efforts related to resilience differentiation in the Internet and telecommunications networks.

Most of the previously studied algorithms assume that every connection requires a full wavelength. While the wavelength transmission rate has reached OC-192 (10Gbps) and is expected to reach OC-768 (40Gbps) in the future, the network may be required to support traffic connections at rates lower than the full wavelength capacity. The traffic demand granularity varies a lot, possibly from OC-3 (155Mbps) to OC-192 (10Gbps). In addition, for networks of practical size, the number of available wavelengths is still a few orders of magnitude lower than the number of source-destination connections. The routing problem with the bandwidth gap between the low-rate connection and high-rate wavelengths is addressed by grooming traffic onto wavelengths so as to efficiently utilize the network resources. In WDM mesh networks, the traffic-grooming problem has mainly addressed static traffic where a traffic demand matrix is known a priori [Zhu02-a], [Zhu03]. Online approaches for traffic grooming in WDM mesh networks have been reported in [Thiagarajan01-a] [Zhu02-b] [Zhu02-c]. The work in [Thiagarajan01-a] proposed a call-admission-control algorithm to address the capacity-fairness issue, i.e. a connection request with higher bandwidth requirement is more likely to be blocked than a connection request with lower bandwidth requirement. The work in [Zhu02-b] proposed different grooming policies and route-computation algorithms for different network states. The work in [Zhu02-c] developed an algorithm for dynamically grooming low-speed connections to meet different traffic-engineering objectives based on the generic graph model proposed in [Zhu03].

Survivable traffic grooming, in which sub-wavelength-granularity connections need to be protected, is a less explored topic. A number of research papers in literature have considered survivable traffic grooming. [Xiang03] and [Thiagarajan01-b] proposed shared path protection algorithms considering traffic grooming. In [Yao04-a] two grooming algorithms were proposed to provision availability based on per-connection requirements. [Xiang04] considered a differentiated shared protection algorithm supporting traffic grooming in WDM mesh networks. Different levels of protection for single- or multi-hop lightpaths are provided according to the bandwidth and the reliability of low-rate connections.

In [OU03], three approaches for shared protection in the context of dynamic provisioning are proposed - protection at lightpath (PAL) level, mixed protection at connection (MPAC) level, and separate protection at connection (SPAC) level. These three schemes explore different ways of backup sharing, and make a tradeoff between wavelengths and grooming ports. Since the existence of a solution to the problem of provisioning one connection request with shared protection is NP-complete, effective heuristics were proposed. However differentiated levels of protection were not taken into consideration in [OU03]. In [Yao04-b], rerouting is employed to improve network throughput under a dynamic traffic model. Two rerouting schemes were proposed, rerouting at lightpath (RRAL) level and rerouting at connection (RRAC) level. Simulation results show that rerouting significantly reduces the connection blocking probability.

In this work we consider different classes of recovery to meet different needs of traffic classes. There are mainly two levels of recovery mechanisms: restoration and protection. While restoration is defined as the real-time establishment of appropriate resources to recover affected traffic, protection involves the establishment of pre-calculated replacement resources. In the latter scheme, the pre-calculated backup paths can be either shared or dedicated. A detailed discussion of these recovery mechanisms was given in Section 1.1. How to efficiently groom low-speed connections while satisfying their resilience requirements is the main focus of this section. To address the problem of differentiated resilience based on the bandwidth assigned to users in a traffic grooming WDM mesh architecture we propose and evaluate two schemes that perform resilience at the lightpath level and at the connection level.

3.4.1 Problem Statement

A connection request is represented as C(*s*, *d*, *B, CR*), where *s* is the source node, *d* is the destination node, *B* is the traffic bandwidth requirement, and CR represents the class of resilience required by the connection. Two types of resource constraints are considered when provisioning a connection request — wavelengths and grooming ports. Typically, as the number of wavelengths in the network increases, the number of grooming ports a node requires decreases, and *vice versa*.

Given the current network state, including the network topology, existing lightpath/connection information, wavelength and grooming-port utilization, new arriving connections should be routed according to their bandwidth and resilience requirements while minimizing the total cost of the working and backup paths.

In [OU03] and [OU04] dedicated and shared protection have been proven to be NP-complete. Therefore, the problem of differentiated resilience with traffic grooming considered in this study is also assumed to be NP-complete.

3.4.2 Proposed Schemes

In this study, connections have different resilience requirements. We assume three different classes of resilience based on the bandwidth assigned to connections. The three classes of resilience are shown in Table 2. Class 1 is fully protected by 1+1 or 1:1 dedicated protection. Class 2 is recovered by shared protection. Class 3 assures restoration using the spare capacity left after recovery of Class 1 and Class 2. If necessary, connections of the Class 3 can be rerouted (meaning that they can be tear down and setup on a different lightpath) to establish Class 1 and Class 2 connections.

In the following we propose two schemes for provisioning survivable paths to connection requests with different resilience classes: (i) Differentiated Resilience at Lightpath (DRAL) level scheme and (ii) Differentiated Resilience at Connection (DRAC) level scheme. These schemes examine different ways of provisioning backup paths and the tradeoff between bandwidth efficiency and the number of required grooming ports.

In the rest of this section, we present these schemes and illustrate them via examples. The initial network configuration used in the illustrative examples is shown in Figure 21. Edges in the figure correspond to bidirectional fibers and each fiber has 4 wavelengths. The wavelength capacity is STS-192. Every node has 4 grooming ports where T and R represent the number of available grooming-add and grooming-drop ports, respectively.

Class Number	Recovery Plan
Class 1	Dedicated protection
Class 2	Shared protection
Class 3	Restoration

Table 2: Classes of the differentiated resilience scheme with dynamic traffic.

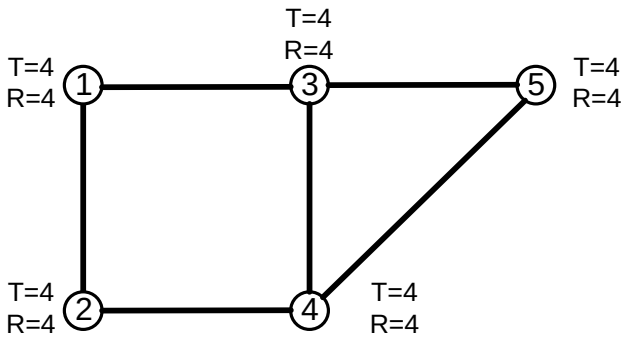


Figure 21: Initial network configuration.

3.4.2.1 *Differentiated Resilience at Lightpath (DRAL) Level*

DRAL provides differentiated end-to-end resilience with respect to lightpaths. For Class 1 and Class 2 where protection is required, a connection is routed through a sequence of protected lightpaths (P-lightpaths). In addition to the normal working path traversing a sequence of lightpaths where grooming-ports are used at the source and destination nodes, each P–lightpath has a link-disjoint path serving as backup path. However, the two classes differ in the way the backup path is provisioned. During normal operation for traffic of Class 1 with dedicated protection, the backup path of this class is set as a sequence of lightpaths using additional grooming ports at the source and destination nodes. On the other hand, the wavelengths used by the backup path of Class 2 with shared protection are only reserved but they are not set up, and therefore, no additional grooming ports are required. In case of working path failure, the backup path is set up as a lightpath by utilizing the grooming ports previously used by the working path. Under shared

protection, two P-lightpaths can share wavelengths along common backup links if their working paths are link-disjoint. For Class 3, where protection is not required, the work path is provisioned as a sequence of lightpaths. In case of working path failure, the spare capacity in the network is used to restore Class 3 connections in a sequence of lightpaths utilizing the grooming ports previously used by the working path.

As mentioned before the lightpaths carrying Class 3 connections can be rerouted to establish new lightpaths to carry otherwise blocked connections of Class 1 and Class 2. The rerouting algorithm is only executed when the normal routing algorithms fails to establish a path for an arriving connection of Class 1 or Class 2. The basic idea of rerouting under DRAL is to reroute some of the existing lightpaths of Class 3 so that otherwise blocked connection requests of Class 1 and Class 2 can be established. Obviously this implies that Class 3 connections should be routed on separate lightpaths from Class 1 and Class 2 as under DRAL the whole lightpath will be rerouted.

Under both DRAL and DRAC (discussed in the next subsection), to reduce the complexity of the rerouting algorithms and the amount of traffic affected by the rerouting operation, the rerouting operation is restricted to one lightpath or connection for a connection request.

DRAL is illustrated through the following example:

Assume the empty network of Figure 21 and the arrival of the first connection request C_1 (3, 4, STS-48c, Class 3). One way of provisioning C_1 under DRAL is shown in Figure 22 (a). Connection *C1* is routed via the lightpath L_1. The lightpath consumes a grooming-add port at the source node (node 3) and a grooming-drop port at the destination node (node 1).The remaining capacity on L_1 is STS-144.

Suppose that C_1 remains in the network when the second connection request C_2 (1, 2, STS-12c, Class 2) arrives. Based on the current network state, one way of provisioning C_2 under DRAL is shown in Figure 22 (b). Connection C_1 is routed via the P-lightpath L_2 which has lightpath L_{2w} (1,2) as a working path and path (1,3,4,2) as a backup path. L_{2w} consumes a grooming-add port at the source node (node 1) and a grooming-drop port at the destination node (node 2). The remaining capacity on L_{2w} is STS-180. Two wavelengths need to be reserved along links (1,3), (3,4) and (4,2).

Suppose that C_1 and C_2 remain in the network when C_3 (1, 4, STS-48c, Class 1) arrives. To establish C_3 under the current network state C_1 (which is of Class 3) should be rerouted to L_3 as shown in Figure 22 (c). Connection C_3 is routed via the P-lightpath L_4, which consists of two link-disjoint lightpaths L_{4w} (working) and L_{4b} (backup). Both lightpaths L_{4w} and L_{4b} consume a grooming-add port at node 1 and a grooming-drop port at node 4. The remaining capacity on P-lightpath L_4 is STS-144.

While all the previous connections remain in the network suppose that C_4 (5, 4, STS-48, Class 2) arrives. One way of provisioning C_4 under DRAL is shown in Figure 22 (d). Connection C_4 is routed via the P-lightpath L_5 which has lightpath L_{5w} (5,4) as a working path and path (5,3,4) as a backup path. Please note that the backup paths of P-lightpaths L_3 and L_5 share the wavelength reserved along link (3,4).

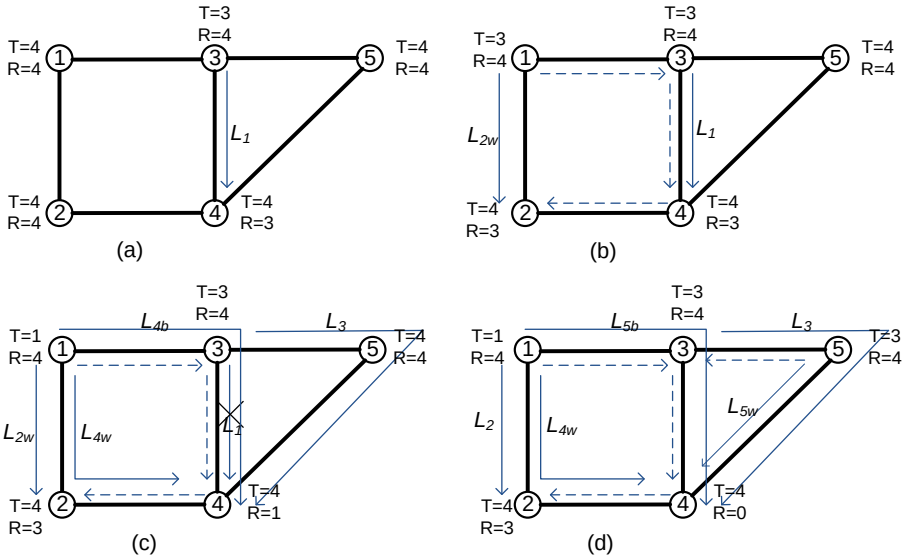


Figure 22: Example illustrating provisioning connections under DRAL

3.4.2.2 *Differentiated Resilience at Connection (DRAC) Level*

DRAC provides differentiated end-to-end resilience with respect to lightpaths. For Class 1 and Class 2 where protection is required a connection is routed via link-disjoint working and backup paths, each of which traverses a sequence of lightpaths.

Under DRAC the rerouting algorithm is executed to reroute some of the existing connections of Class 3 so that otherwise blocked connection requests of Class 1 and Class 2 can be established. The constrain that Class 3 connections should be routed on separate lightpaths from Class 1 and Class 2 does not apply to grooming at the connection level as rerouting is implemented at connection level. As mentioned before, the rerouting operation is restricted to one connection for a connection request.

DRAC is illustrated through the following example:

Assume the empty network of Figure 21 and the arrival of the first connection request C_1 (3, 4, STS-48c, Class 3). One way of provisioning C_1 under DRAC is shown in Figure 23 (a). Connection C_1 is routed via the lightpath L_1. The lightpath consumes a grooming-add port at the source node (node 3) and a grooming-drop port at the destination node (node 1).The remaining capacity on L_1 is STS-144.

Suppose that C_1 remains in the network when the second connection request C_2 (1, 2, STS-12c, Class 2) arrives. Based on the current network state, one way of provisioning C_2 under DRAC is shown in Figure 23 (b). Connection C_1 is routed via two link-disjoint paths- lightpath L_2 considered as the working path, and the three-lightpath sequence (L_3, L_1, L_4) considered as the backup path. The free capacity of lightpaths L_2, L_3

and L_4 is STS-180. The free capacity of lightpath L_1 is STS-132 as it is shared by both C_1 and C_2. Each lightpath of L_2, L_3 and L_4 uses a grooming-add port at its source node and a grooming-drop port at its destination node.

Suppose that C_1 and C_2 remain in the network when C_3 (1, 4, STS-48c, Class 1) arrives. As shown in Figure 23 (c) connection C_3 is routed via two link-disjoint paths- lightpath L_5 and the two-lightpath sequence (L_3, L_1). The remaining capacity on the lightpaths is as follows: STS-84 of L_1, STS-132 of L_3, and STS-144 of L_5.

While all the previous connections remain in the network suppose that C_4 (5, 4, STS-192c, Class 2) arrives. According to the current network state, C_4 has to be blocked as it is not possible to provision both a working and a backup path. However, C_4 can be provisioned by applying rerouting at the connection level. C_1, which is a Class 3 connection, is rerouted so enough capacity becomes available at L_1. As shown in Figure 23 (d), C_1 is rerouted via the two-lightpath sequence (L_6, L_7) to allow C_4 to be routed via two link-disjoint paths- lightpath L_7 considered as the working path, and the two-lightpath sequence (L_8, L_1) considered as the backup path. Please note that the C_2 and C_4 share the backup capacity on L_1 as their working paths are link-disjoint (the larger capacity STS-192 is reserved for both of them).

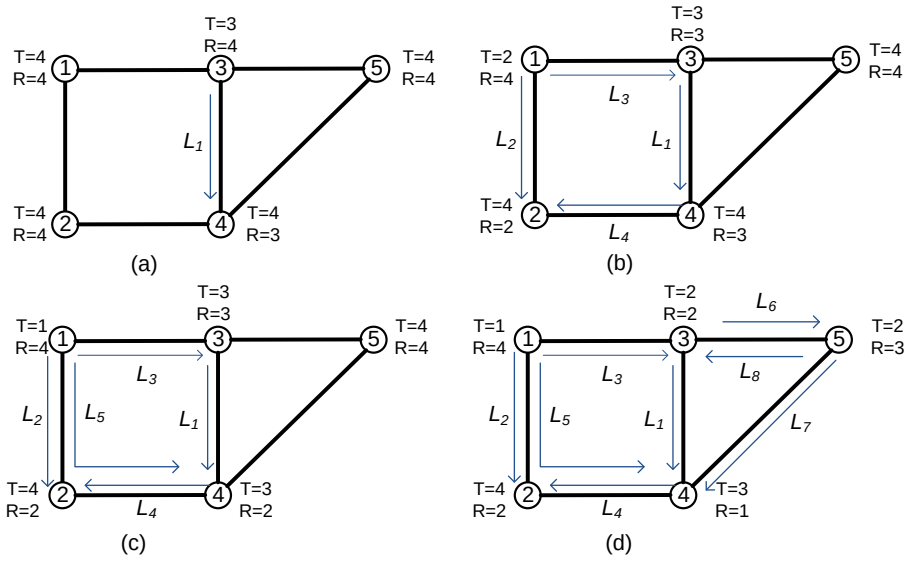


Figure 23: Example illustrating provisioning connections under DRAC

3.4.2.3 *Comparison between DRAL and DRAC*

From the discussion and illustrative examples in the above sections, it is clear that DRAL and DRAC perform differently. In this section we compare their characteristics in terms of grooming ports consumption, backup sharing, rerouting and complexity.

Under both DRAL and DRAC a connection with dedicated protection requirements (Class 1) is provisioned a sequence of lightpaths as its working and backup paths, however the two schemes differ in the utilization

of these lightpaths. Under DRAL the two lightpaths of a P-lightpath are considered as an integrated unit and cannot be utilized individually. This constrain does not apply to lightpaths under DRAC. Therefore under DRAC Class 1 connections are more likely to be groomed onto new lightpaths and more grooming ports are more likely to be consumed.

Connections with shared protection (Class 2) under DRAC are provisioned backup paths which traverse a sequence of lightpaths. However, under DRAL wavelengths used by the backup path are only reserved but they are not set up, and therefore, no additional grooming ports are required.

For above it is clear that DRAL trades the bandwidth efficiency in routing each connection request for the savings in grooming ports usage.

An important aspect of comparison between DRAL and DRAC is backup sharing for connections with shared protection requirements (Class 2). DRAC allows working paths and backup paths (of different connections) to be groomed onto the same lightpath. Also the difference between backup paths of Class 2 under the two schemes creates a difference in the way they share the backup capacity. As a lightpath may traverse multiple links, the reserved backup capacity on a lightpath, in case of DRAC, is less likely to be shared among multiple connections compared to the backup capacity reserved on a link in case of DRAL. Under DRAL the reserved wavelengths on a link act like a “pool” for all the failure scenarios and wavelength converters facilitate backup capacity sharing among different wavelengths on a link. However, under DRAC, sharing the backup capacity among different wavelengths on a link is not possible as the backup capacity is in the form of lightpaths and multiple lightpaths cannot share their reserved backup capacity. From above it is clear that for Class 2 connections DRAC provides flexibility in terms of grooming as it allows grooming working paths and backup paths (of different connections) onto the same lightpath, however DRAL is more flexible in terms of sharing backup capacity.

Regarding rerouting, as discussed before, DRAL performs rerouting at an aggregate (lightpath) level, while DRAC performs rerouting at a per-flow (connection) level. Therefore DRAL affects more traffic than DRAC during the rerouting process as DRAL disconnects all the connections on the rerouted lightpath, while DRAC only disconnects the rerouted connection. Rerouting at connection level results in more flexibility in terms of selecting rerouted connections and their new paths and in terms of allowing connections from different classes to be rerouted through the same lightpaths. Also, rerouting at connection level preserves the quality of service (QoS) and the traffic engineering (TE) constraints of the rerouted connection.

From implementation point of view, DRAL is relatively simple compared to DRAC as it only needs the global information of the lightpaths in the network to compute working and backup paths for a new arriving connection request. On the other hand, DRAC requires, in addition to the global information of all the lightpaths, the detailed routing information of all the existing connections in order to compute working and backup paths for a new arriving connection request. Therefore, DRAC results in a larger complexity. From control point of view, DRAL has lower signaling overhead. Assume that a lightpath can carry up to *C* connections. In case of a link failure, *W* lightpaths can be disrupted in the worst case. In DRAL, at most *L* protection-switching processes are needed. However, in DRAC, up to *W*x*C* (*W* the number of wavelengths, *C* the number of connections) protection-switching processes are required in the worst case. As protection-

switching processes typically require signaling, DRAL demands lower control bandwidth and involves lower signaling complexity compared to DRAC.

3.4.3 Performance Evaluation

The proposed differentiated resilience scheme is verified through simulation. We conduct our simulations on the Italian network, illustrated in Figure 24, as an example of a real world network. The Italian network consists of 21 nodes and 36 bidirectional links. For the simulation scenario a ratio of 10% Class 1, 30% Class 2, and 60% Class 3 iss assumed. Connections are assumed to be uniformly distributed among all the node pairs. The connection-arrival process is assumed to follow a Poisson distribution and the connection-holding time is assumed to be exponential distributed. Connections bandwidth requirements are assumed to be as follows: Class 1 and Class 2 requirements are assumed to be uniformly distributed in the range of (4-10) units. Requirements of Class 3 are assumed to be uniformly distributed in the range of (0-4) units. 100,000 connections are simulated in each simulation scenario. Each fibre supports 16 wavelengths and full wavelength conversion capability is assumed. The number of grooming ports at each node is set as the number of wavelengths times its nodal degree times a variable σ ($0 \leq \sigma \leq 1$) [OU03]. σ=1 implies that any incoming wavelength can be groomed. We assume that there are equal number of grooming-add and grooming-drop ports. Failures are generated randomly. The inter-arrival time and holding time of failures are assumed to be exponentially distribution. Links are affected by failures according to a uniform distribution. Simulation results compare the performance of each traffic class under DRAL and DRAC with a large number of grooming ports (σ=1) and a smaller number of grooming ports (σ=0.5). The connection requests blocking probability is used as the comparison metrics.

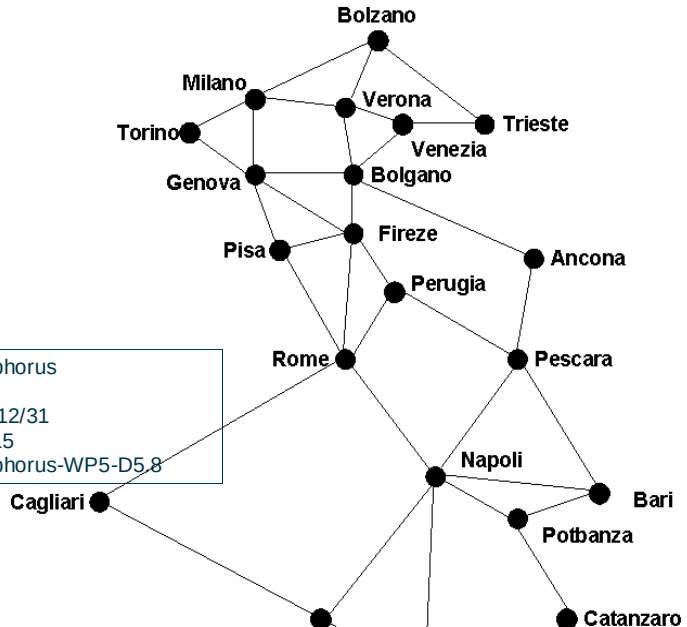


Figure 24: The Italian mesh network

Figure 25 illustrates the blocking probability for Class 1 (dedicated protection). With a large number of grooming ports (σ=1). DRAC has much lower blocking probability than DRAL. This is a result of the bandwidth efficiency in grooming at the connection level and the fact that DRAC allows working paths of different traffic classes to be groomed onto the same lightpath. However, when the number of grooming ports is small (σ = 0.5), DRAL has much lower blocking probability than DRAC. DRAL is not very sensitive to the changes in the number of grooming ports as it utilizes wavelengths more quickly than grooming ports. However, DRAC utilizes grooming ports more aggressively as it trades grooming ports for bandwidth efficiency in routing and grooming.

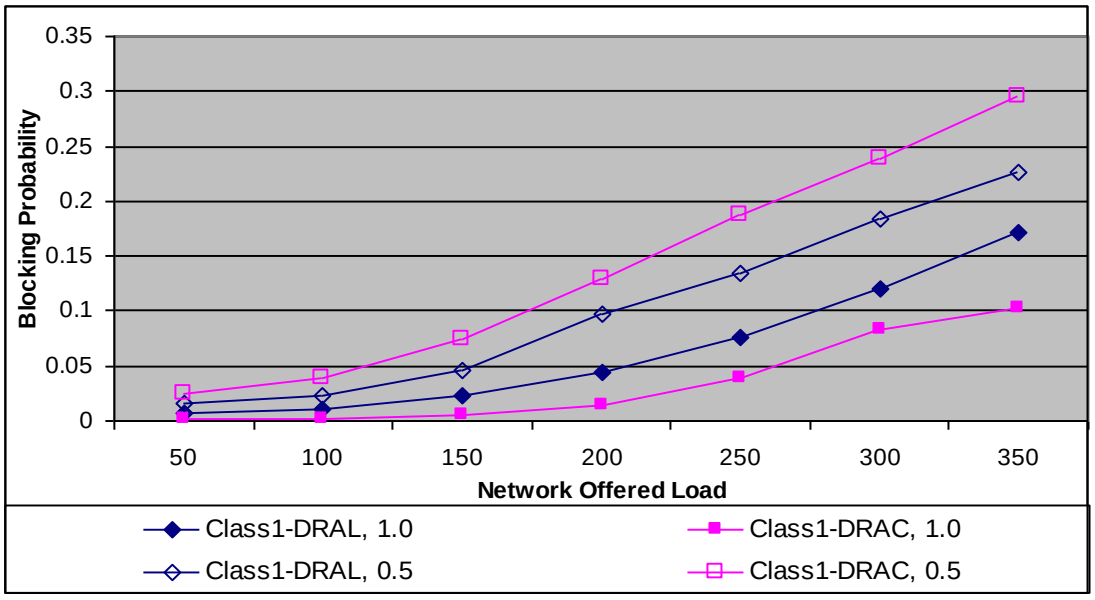


Figure 25: Blocking probability of Class 1 traffic.

Results of Class 2 (shared protection) traffic are shown in Figure 26. For Class 2 DRAC has the advantages of allowing working paths of different classes to be groomed onto the same lightpaths. However, the approach of provisioning shared backup capacity under DRAL gives it advantage over DRAC in term of sharing backup capacity and grooming ports consumption. Figure 26 shows that DRAL slightly outperforms DRAC when the number of grooming ports is large (σ =1). However under a smaller number of grooming ports (σ=0.5) DRAL blocking probability is much lower than that of the DRAC as shared protection under DRAC is very sensitive to the changes in the number of grooming ports.

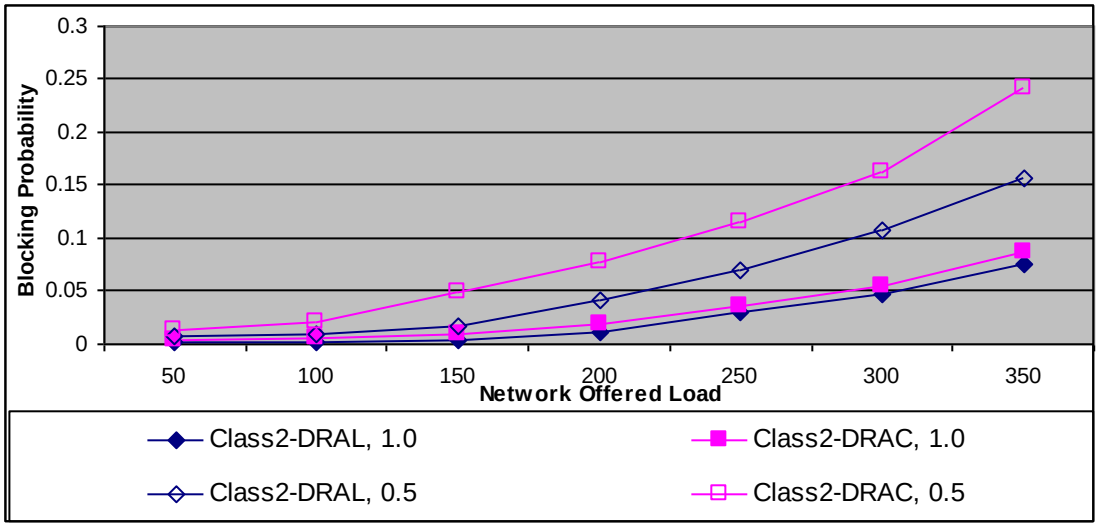


Figure 26: Blocking probability of Class 2 traffic.

The blocking probability experienced by Class 3 connections during the attempt to provision them working paths (Class3-WP) and the blocking probability experienced by Class 3 connections during the attempt to provision them backup paths in case of failure (Class3-BP) are shown in Figure 27 and Figure 28, respectively. When the number of grooming ports is large (σ=1), DRAC outperforms DRAL as DRAL allows Class 3 connections to be groomed onto the same lightpaths with other Classes. Due to DRAC high sensitivity to change in the number of the grooming ports, DRAL outperforms DRAC when the number of grooming ports is smaller (σ=0.5).

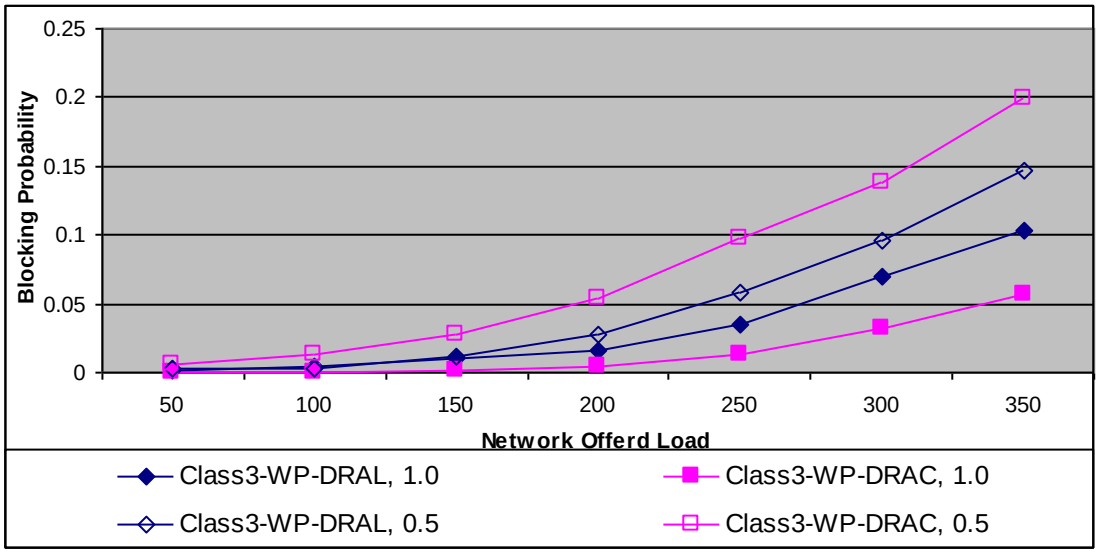


Figure 27: Blocking probability of working paths of Class 3 traffic.

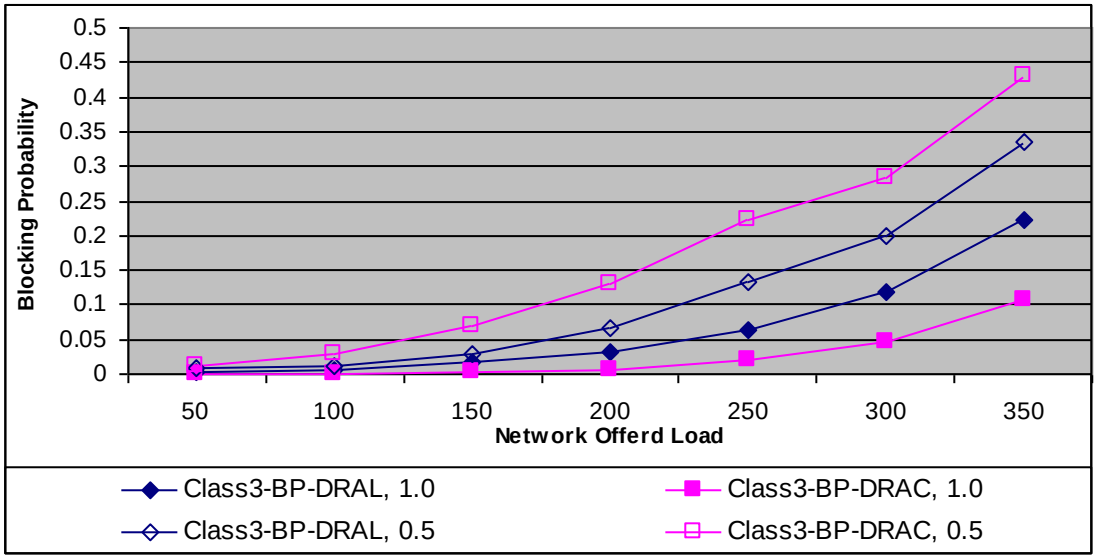


Figure 28: Blocking probability of backup paths of Class 3 traffic.

Figure 29 also shows the rerouting probability experienced by Class 3 to allow Class 1 and Class 2 to establish their connections. As discussed before, DRAL affects more traffic than DRAC during the rerouting process as DRAL disconnects all the connections on the rerouted lightpath, while DRAC only disconnects the rerouted connection. Therefore, the rerouting probability of Class 3 is higher under DRAL. The same trend is noticed when the number of grooming ports is smaller (σ=0.5).

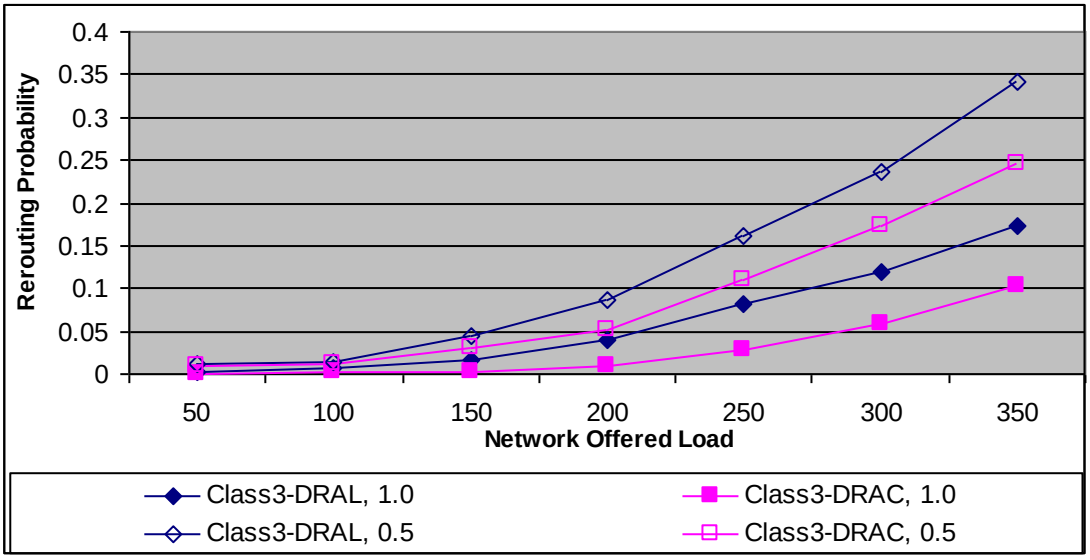


Figure 29: Rerouting probability of working paths of Class 3 traffic.

3.4.4 Conclusion

In this section survivable traffic grooming is investigated under a differentiated resilience scheme. Two schemes of provisioning survivable paths to connection requests with different resilience classes are proposed, Differentiated Resilience at Lightpath (DRAL) level and Differentiated Resilience at Connection (DRAC) level. These schemes examine different ways of provisioning backup paths. Performance of different resilience classes (dedicated protection, shared protection and restoration) under DRAL and DRAC are compared through simulation.

Simulation results show that traffic with dedicated protection requirements obtains a better blocking probability under DRAC when the number of grooming ports is large. However, when the number of grooming ports is small, DRAL results in a lower blocking probability due to the high sensitivity of DRAC to the change in the number of grooming ports. The blocking probabilities experienced by traffic with shared protection requirements under DRAL and DRAC are very close under a larger number of grooming ports. With a small number of grooming ports DRAL outperforms DRAC.

Both the blocking probabilities experienced by connections with restoration requirements during the attempt to provision them working paths and during the attempt to provision them backup paths in case of failure are lower under DRAC when the number of grooming ports is large. With a small number of grooming ports the opposite trend is observed. The rerouting probability experienced by connections with restoration requirements to allow higher resilience classes to establish their connections are found to be higher under DRAL.

3.5 Differentiated Resilience for Anycast Flows in MPLS Networks

Anycasting is a service that allows communication between a single sender and at least one, and preferably only one, of a group of several receivers. Anycasting is implemented in applications where a client wants to send data to any one of several possible servers offering a particular service but it does not really matter which server provides the service. Therefore, the client can select one of these servers according to some criteria. Using anycasting may considerably simplify some applications. In Grid networks, the destination of the handling resource is of less significance as different resources can process the job. Therefore users can submit their jobs without assigning a specific destination. In previous studies [DeLeenheer05] [DeLeenheer06], researchers have shown that anycasting can efficiently support many emerging high-performance grid applications in OBS networks. Anycasting provides a number of advantages to the network. It reduces network traffic and avoids congestion causing big delays in data delivery [Walkowiak07]. Also anycasying facilitates survivability by allowing users to select another server offering the same service in case of failure.

Most of previous research on survivable dynamic routing has focused on unicast traffic flows [Bagula04] [Kar00] [Kar03] [Kodialam00-b] [Kodialam02] [Kodialam03]. However, anycast traffic flows have recently gained much attention. Most studies on anycast routing have explored IP networks using connection-less transmission modeled as bifurcated multi-commodity flows [Doi04] [Hao02] [Lin04]. MPLS adds a number of traffic engineering capability and QoS performance mechanisms which are not supported by pure IP networks. A limited number of papers have considered anycast routing in MPLS networks. In [Walkowiak07] the author examines the performance of several constraint-based algorithms for anycast server selection and QoS routing algorithms in connection oriented MPLS network.

In this work, we propose a differentiated resilience scheme that allows anycast flows to survive any link or server failure. In this scheme three classes of survivability are implemented. Class 1 and Class 2 connections are recovered through dedicated and shared protection to an alternative server, respectively. A Class 3 connection assures restoration by assigning it to another server using the spare capacity left after recovery of the other two classes.

3.5.1 Multi-Protocol Label Switching (MPLS)

3.5.1.1 *MPLS Architecture*

Multi-protocol label switching (MPLS) [Rosen01] is an extension to the existing Internet Protocol (IP) architecture intended to support traffic engineering capabilities and QoS performance. MPLS enables the implementation of a simpler, high-performance packet-forwarding engine by forwarding packets according to labels attached to them. The main components of MPLS architecture are illustrated in Figure 30. Arriving packets are labeled at the ingress side of the MPLS domain by Label Edge Routers (LERs) according to the Forwarding Equivalence Class (FEC) they map to. Internal routers in the MPLS domain known as Label Switching Routers (LSRs), use a label forwarding table to switch and re-label incoming packets. The label sequence determines the Label Switched Path (LSP) that the packet follows through the network. To achieve efficient traffic engineering, FECs can also incorporate QoS, security, or other considerations. This concept is known as policy routing. LSPs are setup via signalling protocols such as the Label Distribution Protocol (LDP) with its extensions for Constraint-based Routing (CR-LDP), and the Resource Reservation Protocol (RSVP) with its Traffic Engineering extension (RSVP-TE).

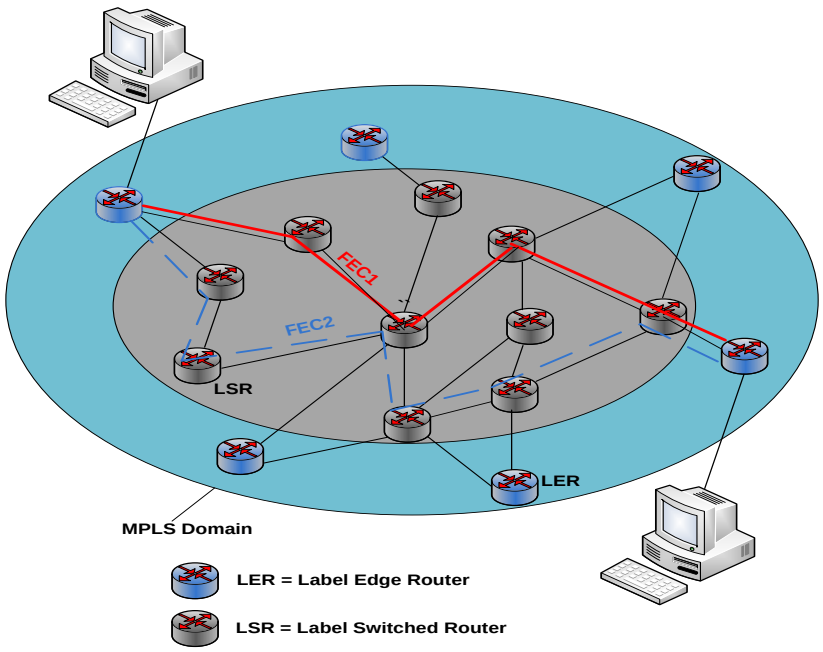


Figure 30: MPLS architecture.

3.5.1.2 *MPLS Routing*

The selection of an LSP for a particular FEC could be based on one of two options: hop-by-hop routing and explicit routing [Rosen01]. Under *hop-by-hop routing*, each LSR independently decides the next hop for each FEC using an ordinary routing protocol, such as Shortest Path First (SPF). *Hop-by-hop routing* provides a number of advantages, such as rapid switching by labels, label stacking, and differential treatment of packets from different FECs following the same route. However, hop-by-hop routing attaches network resources such as link and switch capacity to packet flows too late to support traffic engineering or policy routing. Pre-calculated LSP, supported in explicit routing, facilitates early attachment of resources to the flows and hence traffic engineering and policy routing. Explicit routing is also implemented to improve path controllability, eliminate loop paths. Explicit routing can be implemented either statically or dynamically. Dynamic explicit routing provides the best scope for traffic engineering, although it needs topology and QoS-related information about the MPLS domain.

The most common routing algorithm for MPLS is SPF algorithm. SPF is based on an administrative weight (metric) such as the number of hops. The path's length is calculated as the sum of the cost of all links along a path. SPF is simple and requires a relatively short execution time. However, as packets are routed through the same set of shortest paths until they are saturated, SPF leads to congestion in some areas of the network.

Constraint Shortest Path First (CSPF) algorithm [Crawley98] was proposed as an enhancement of the SPF algorithm to solve the problem of load balancing in SPF by introducing the residual network approach. The residual network reflects the current resource availability for a new call, and consists of all routers and feasible links. The residual capacity of a link is defined as the difference between the capacity of the link and the current flow on the link, which is calculated as a sum of the LSPs' bandwidths that are routed on that link. Thus, routing in the residual network guarantees that allocation of a new call will not violate the capacity constraint. Note that if the ingress and the egress routers are disconnected in the residual network then there is no path and the LSP request is rejected. However both SPF and CSPF are poorly equipped to support traffic engineering under heavy loads [Widjaja02].

Another dynamic routing algorithm proposed in the context of MPLS networks is the minimum interference routing algorithm (MIRA), proposed in [Kar00]. MIRA prevents the creation of bottlenecks by avoiding the selection of critical links that results in maximizing the minimum available capacity on potential future paths. The major limitation of MIRA is its computation complexity which is not necessarily translated into equivalent performance gains. Another limitation of MIRA is that it may lead to unbalanced network utilization.

In [Bouaba02] a routing algorithm known as dynamic online routing algorithm (DORA) was proposed. DORA effectively utilizes existing network resources and minimizes network congestions by carefully

mapping paths across the network. It was shown in [Bouaba02] that DORA requires fewer paths to be rerouted and obtains a higher successful rerouting percentage than both SPF and MIRA. Also, DORA reduces computational complexity compared to MIRA, and its runtime is equal to that of SPF. However its main limitation compared to SPF is the computational complexity of its first stage.

The authors of [Bagula04] proposed another routing algorithm known as the Least Interference Optimization Algorithm (LIOA). LIOA is based on a combination of the SPF method and the residual network approach. It reduces the interference among competitive flows by balancing the number of flows carried by a link. Results show that LIOA outperforms MIRA in terms of rejection ratio, and can perform successful rerouting in case of single link failures.

In [Kodialam00-b] [Szeto02], issues related to dynamic routing algorithms in MPLS networks, such as routing constraints, online/offline routing, computational complexity and reliability, are discussed.

3.5.1.3 *Resilience in MPLS Architecture*

Although conventional IP routing algorithms support resilience by automatical rerouting of packets around a failure through routing table updates. However, as the convergence time of IP routing protocols is on the order of several seconds to several minutes, IP rerouting is considered to be very slow for some real-time and mission critical traffic. MPLS appears to be a promising solution as it separates packet forwarding from routing. MPLS facilitates provisioning various routing services independent of the packet forwarding paradigm. MPLS can support faster recovery than traditional IP rerouting, and additionally provides best-effort IP networks with QoS and traffic engineering capabilities.

MPLS resilience techniques can be classified into two methods: protection switching and restoration (or MPLS rerouting). In case of protection switching, an alternative pre-established backup LSP is used to recover traffic in case of failure of the working LSP [Haskin00] [Makam99]. The backup LSP is pre-provisioned to realize the shortest disruption of the traffic in case of a failure. Traffic may either be forwarded simultaneously on the working and the backup LSPs (known as 1+1 protection), or it can be forwarded on the predefined backup LSP only after the detection of a network failure (1:1 protection).

In case of restoration (or MPLS rerouting), backup LSPs are established after the failure detection. The backup LSP is found using network routing policies and signalling protocols. As calculating, signalling and reserving the new LSP are time-consuming operations, restoration recovery is considerably slower than protection recovery. However, restoration is less expensive as no additional resources are reserved during normal operation. In [Yoon01], one way of implementing restoration is presented.

Protection and restoration may also be used together. For example, protection may be used to provide a fast recovery to a failed path. Rerouting may be applied after that to determine a new optimal path in an offline manner [Sharma03] [Grover04].

Recovery methods in MPLS networks can be further classified as local recovery, global recovery and reverse backup. The LSR responsible for switching the working LSP to the alternative LSP is called Path

Switch LSR (PSL). The router responsible for merging the alternative and the working LSP is called Path Merge LSR (PML). Local recovery protects the working path against a link or neighbor node failure and does not need any end-to-end failure notification and signalling as the PSL and PML are adjacent to the failed link or node. These can then detect the failure and immediately trigger the appropriate recovery actions the LSP. Global repair protects the working path against any link or node fault on a path, except the failures occurring at the ingress egress node of the path. Finally, in the reverse backup approach, the traffic on a failed link is reversed from the point of the failure, i.e. the ingress node of the failed link, back to the ingress node of the protected path where traffic will be rerouted to an alternative recovery path.

3.5.2 Survivable Routing of Anycast Flows in MPLS Networks

A unicast demand is defined by its origin node, destination node and bandwidth requirements. Establishing a unicast flow involves calculating a route connecting origin and destination nodes and satisfying the bandwidth requirements. Anycasting introduces further complexity as an anycast flow is only defined by its origin node, and upstream/downstream bandwidth requirements. To establish an anycast flow to one of multiple destinations, a destination is selected and then both upstream and downstream routes connecting the origin node and the selected server are calculated.

For server selection the residual network approach introduced in the context of QoS unicast routing algorithm is used. The residual network for a new anycast demand is built with nodes with sufficient resources (e.g. memory, processor, etc.) and links with a residual capacity exceeding the request bandwidth requirement. Bandwidth requirements of both upstream and downstream routes are considered, and as such only reachable servers with enough resources are taken into account in the selection.

Servers can be selected according to two metrics: hop distance between the user and the server and the residual capacity of the server. The hop distance ensures that the nearest server is selected, which may however result in congesting the network links leading to the server node. In contrast, selecting the server according to the residual capacity of the server helps balancing the anycast flows among various servers in the network. The residual capacity of the server node is calculated as the residual capacity of all links leaving the considered node. In [Walkowiak05], three server selection algorithms were presented: Hop Number Server (HNS), Residual Capacity Server (RCS) and Hop Number Widest Server (HNSW). The HNS algorithm is based on the hop distance metric; in case there is more than one nearest server, the residual capacity metric is applied. The RCS algorithm uses the node residual capacity as the selection criterion. The HNWS algorithm chooses the server with the smallest value of the quotient of distance and residual capacity. If the algorithm fails, the considered anycast request is rejected. In addition to the information required for QoS unicast routing (i.e. network topology, link capacity, total flow on link), anycast routing also needs detailed information about which network nodes host replica servers.

After successful selection of the server according to one of the three algorithms presented above, the anycast request can be considered as a conventional unicast request and dynamic routing algorithms can be applied to find routes connecting the server node and the user node. Due to the nature of the server selection algorithm, it is guaranteed that if a server is successfully selected, two routes can be established for the downstream and upstream connections.

Another issue to be considered when studying anycasting in MPLS architectures is survivability. An anycast traffic flow is affected by a failure if one of its upstream or downstream connections is broken. Similar approaches described in Section 3.5.1.3 for MPLS unicasting recovery can be applied for anycast flow recovery. Under these approaches, referred to as *backup path approaches,* the broken anycast connection (downstream and/or upstream) is recovered to the same server. Any of the dynamic routing protocols described in Section 3.5.1.2 can be applied to calculate the backup path, in which a path is computed in the residual network after removal of all broken connections. However, the fact that an anycast request can be served by any suitable replica server allows an alternative approach to recover anycast flows. Under this approach, referred to as backup server, routes for both upstream and downstream connections to a backup server(s) are calculated; if the working server fails, then the backup server takes responsibility. [Walkowiak07] compared the backup server method against the backup path method, and found that the former method restores more connections than the latter method. This is because after the failure, the area close to the failure can become congested. If the failure occurs close to a server node, all broken anycast connections connected to this server can not be restored due to limited availability of residual resources.

3.5.3 Proposed Differentiated Resilience Scheme for Anycast Flows

Providing a high degree of resiliency for all traffic flows in a network is expensive and tends not to scale well. A well-designed resilience scheme has to consider the different resiliency requirements of traffic flows, ultimately resulting in a more cost-effective network design and traffic engineering.

According to [Grover04], a single link cut is the most probable failure in modern networks, although other failure scenarios (node failures, multiple failures) could also be considered. In this work, we propose a differentiated resilience scheme that allows anycast flows to survive any link or server failure. Three classes of survivability schemes are investigated. In Class 1, as the anycast flow is established, the two connections (upstream and downstream) of the anycast request receive dedicated backup LSPs to another server, which are selecteded according to one of the previously discussed methods. LSPs to the backup server should be link disjoint with the LSPs to the working server to guarantee resiliency against server and link failures. Class 2 LSPs are recovered by sharing LSPs to another backup server. LSPs sharing a LSP should be link and server disjoint, in order to guarantee resiliency against server and link failures. However, reservation of extra spare capacity for each LSP is not always economically, as some LSPs carry traffic of low importance. These LSPs are classified as Class 3. In case of failure of the server or a link in the path to the server, Class 3 connections are assured restoration by assigning them to another server using the spare capacity left after recovery of the other two classes. Therefore we accept that some of Class 3 LSPs may not be recovered in case of failure due to limited availability of spare capacity.

Explicit routing is implemented for the LSPs selection, and, as discussed in 3.5.1.2, dynamic routing is implemented to calculate routes (upstream and downstream connections) for both the working and backup server. In this work, we compare the performance of the differentiated resilience scheme under the CSPF and LIOA algorithms.

Link state information required for routing includes link capacity, total link flow and the network topology. While the network topology is supported by classical routing protocols such as SPF, the link capacity and

the total flow capacity on links require extensions to traffic engineering as supported by an extended version of SPF [Katz03] [Smit04]. In addition to this information, anycast flows also requires information on the location of the replica servers. Routing protocols proposed in [Katz03] [Smit04] can easily be extended to include this information. An extension to the existing QoS architectures, known as Resilience-Differentiated QoS (RD-QoS), is proposed in [Autenrieth02] to integrate the signalling of resilience requirements with traditional QoS signalling. In this model, applications signal their resilience requirements in addition to their QoS requirements to the network edge, where they are considered for both resource management and traffic handling. In [He06] both the RSVP-TE and the CR-LDP are extended to include resilience classes in the signalling message. For simplicity, in our architecture it is assumed that all the above information is known administratively.

Due to their high resource requirements, Class 1 and Class 2 traffic flows are expected to experience high blocking probabilities. The blocking probability will be higher for Class 1 traffic, as it requires dedicated backup resources for each flow. To reduce the blocking probability, rerouting is implemented, whereby Class 3 traffic flows are rerouted to allow otherwise blocked requests of Class 1 and Class 2. Note that in this scheme, rerouting of Class 3 involves the rerouting of both upstream and downstream connections.

3.5.4 Performance Evaluation

The effectiveness of the proposed differentiated resilience scheme is verified through simulation. The proposed scheme is evaluated with the different server selection algorithms discussed in Section 3.5.2 and under the CSPF and LIOA routing algorithms. All combinations of both types of algorithms are evaluated.

We conduct our simulations on the Italian mesh network, illustrated in Figure 24, as an example of a real world network. Each node can be used as an ingress and egress node. Four nodes were randomly selected to function as servers. Results obtained are averaged out over various cases of server locations.

It is assumed that each server has unlimited resources (e.g. storage, processor etc) which, makes implies that link bandwidth is the only potential bottleneck for the anycast flows. The link capacity is assumed to be 10 units, and we consider static paths that resemble long-lived MPLS tunnels, once established, these are assumed to stay in the network for a long time.

For the simulation scenario, we assume the total traffic consists of 10% Class 1, 30% Class 2, and 60% Class 3 traffic. As the anycast flow is asymmetrical, bandwidth requirements of the upstream connection of all traffic classes are assumed to be 1 bandwidth unit. The downstream connections requirements are assumed to be as follows: Class 1 and Class 2 requirements are assumed to be uniformly distributed in the range of (4-10) units. Requirements of Class 3 are assumed to be uniformly distributed in the range of (0-4) units.

Failures are generated randomly, with both the inter-arrival times and holding times of failures assumed to be exponentially distributed. Links and servers are affected by failures according to a uniform distribution.

In the simulation scenario, we compared the performance of the different classes of the differentiated resilience scheme under different combinations of the implemented routing and server selection algorithms. The connection blocking probability is used as the comparison metrics. An anycast request of Class 1 and Class 2 is considered to be blocked if neither the primary nor the backup path of either its upstream or downstream connections can not be established. The blocking probability of a certain traffic class is calculated as the ratio of the number of blocked requests of that class to the total number of requests of that traffic class in the network.

Figure 31 illustrates the average blocking probability experienced by different traffic classes under all the considered combinations of routing and server selection algorithms. For traffic Class 3 we show the blocking probability of requests as they arrive to the network where routes to primary servers need to be established (Class3-WP). Also we plot the blocking probability in case of failure where routes to an alternative server need to be established (Class3-BP). All simulations are run under the same network load (200 Erlang). We can easily notice from Figure 31 that the best performance of different traffic classes is obtained under the combination of HNWS and LIOA algorithms. Compared to other traffic classes, Class 1 experiences the highest blocking probability as it has the highest resource requirements.

As mentioned previously, rerouting is introduced to reduce the blocking probability of Class 1 and Class 2. Figure 32 compares the blocking probabilities of all traffic classes with and without rerouting under the best combination of algorithms (HNWS-LIOA). It is clear in the figure that rerouting has significantly reduces the blocking probability of Class 1 and Class 2. It can be seen that Class3-WP and Class3-BP blocking probabilities increase in the case where rerouting applied. This is a direct result of the decrease in the probability of blocking of the other classes i.e. less bandwidth is available to Class 3.

Figure 32 also shows the rerouting probability experienced by Class 3 to allow Class 1 and Class 2 to establish their connections. Under a load of 350 Erlang the rerouting probability is about 0.33 while the blocking probabilities has decreased by about 40% of Class 1 and 44% for Class 2.

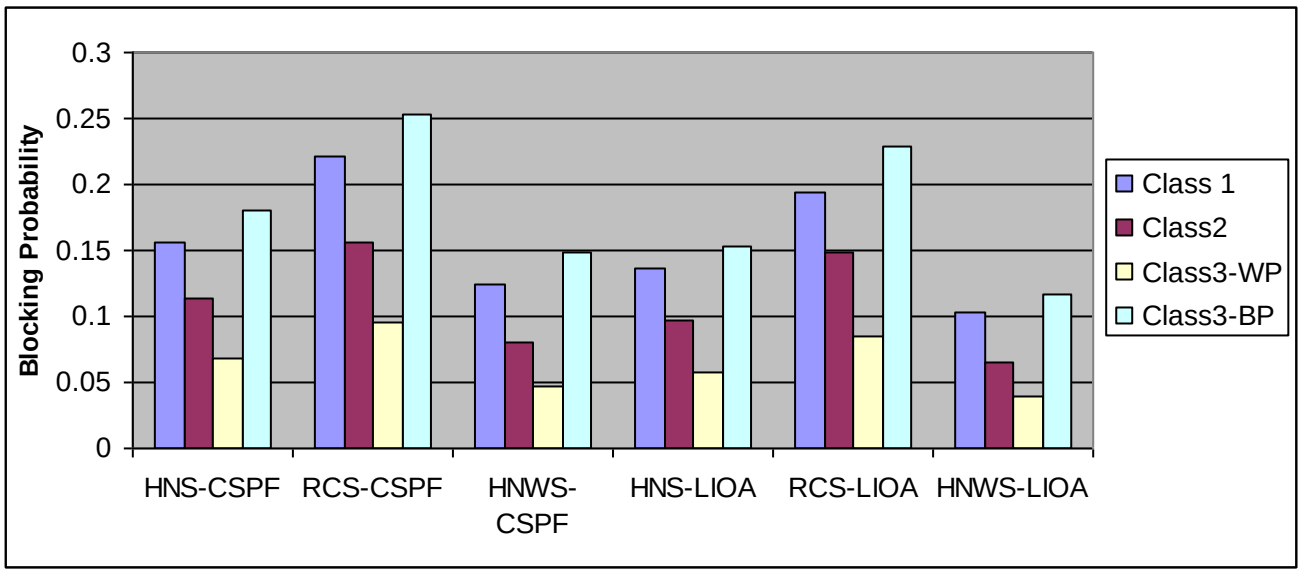


Figure 31: Average blocking probability of different traffic classes under different combinations of server selection and routing algorithms.

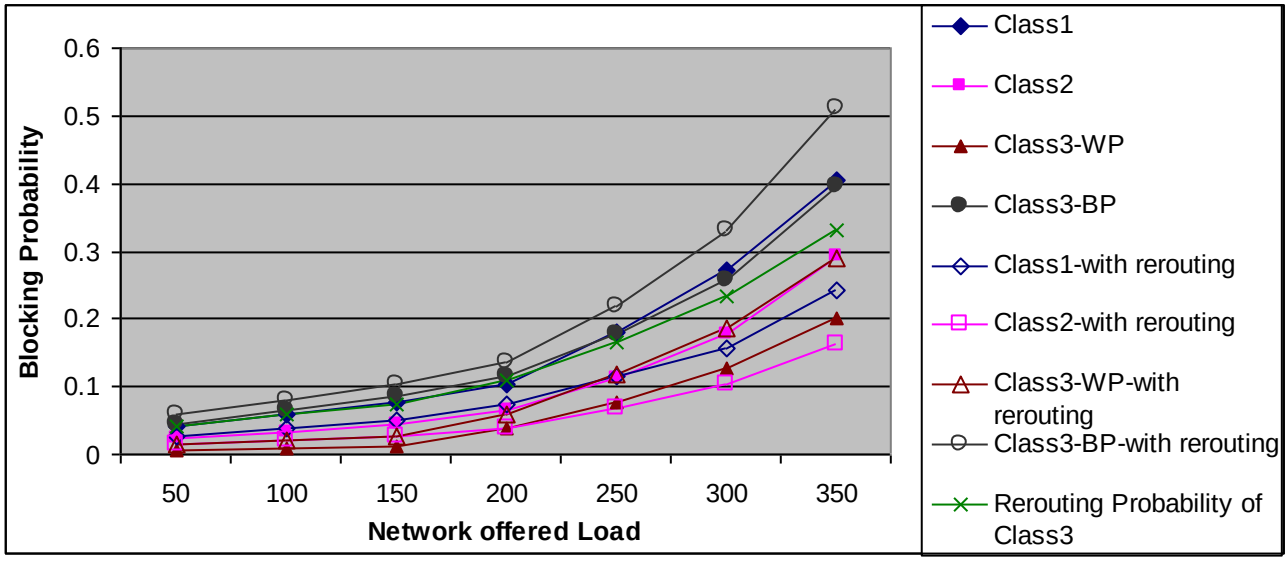


Figure 32: Average blocking probability of different traffic classes with and without rerouting.

Figure 33 we examine the effect of deploying different number of servers in the network on the blocking probability under a load of 200 Erlang. Remember that servers are assumed to have unlimited resources. Therefore we are not interested on the extra computational resources added by these servers. We just want to examine their effect on balancing the traffic flows through the network. It is clear from the figure that increasing the number of servers decreases the blocking probability is a direct result of balancing traffic flows as they are destined toward more servers. The decrease in blocking probability tends to get smaller as the number of server increase.

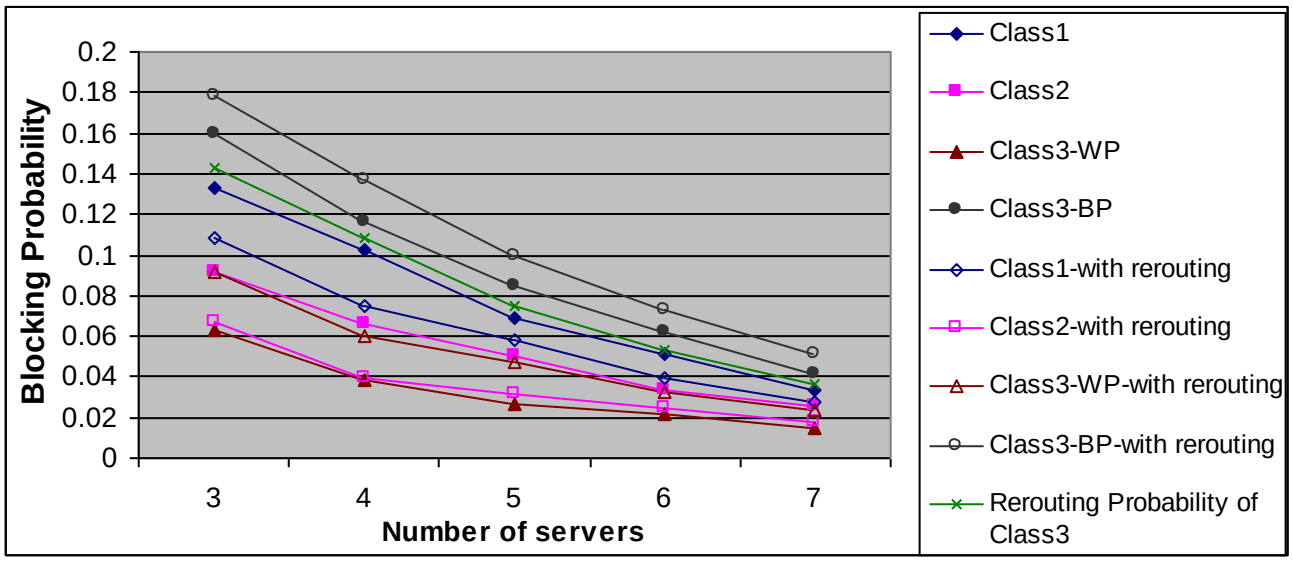


Figure 33: Average blocking probability of different traffic classes under different number of servers

3.5.5 Conclusion

In this study a differentiated resilience scheme is investigated in the context of an MPLS architecture where anycasting principle is implemented as the communication paradigm. The proposed resilience scheme allows anycast flows to survive any link or server failure. Three classes of survivability are considered based on backup capacity allocation. Rerouting of the lower traffic class is implemented to reduce the blocking probability experienced by higher traffic classes. Simulation is carried out to examine the performance of the different traffic classes. Performance is compared with and without the rerouting algorithm. It is shown that rerouting has significantly reduced the blocking probability of Class 1 and Class 2. The effect of deploying different number of servers is also examined. It is shown that increasing the number of servers decreases the blocking probability; however, this decrease tends to get smaller as the number of server increase.

4 Resource Resilience

Compared to other distributed environments, such as clusters, complexity of grids mainly originates from decentralized management and resource heterogeneity. The latter refers to hardware as well as to foreseen utilization. These characteristics often lead to strong variations in grid availability, which in particular depends on resource and network failure rates, administrative policies and fluctuations in system load. Apparently, run-time changes in system availability can significantly affect application (job) execution. Since for a large group of time-critical or time-consuming jobs, delay and loss are not acceptable, fault-tolerance should be taken into account.

Providing fault-tolerance in a distributed environment, while optimizing resource utilization and job execution times, is a challenging task. To accomplish it, two techniques are often applied: job checkpointing and job

replication. In the following, we show that neither of these techniques in their pure form is able to cope with unexpected load and failure conditions in a Grid environment. In response, we propose several solutions that dynamically adapt the checkpointing frequency and the number of replicas as a reaction on changing system properties (number of active resources and resource failure frequency). Furthermore, a novel hybrid scheduling approach is introduced that switches at run-time between checkpointing and replication depending on the system load.

4.1.1 Adaptive Checkpointing Heuristics

We propose several algorithms [Chtepen09] that differentiate the initial checkpointing interval based on history statistics and current status of a particular job and its execution environment, with as objective the reduction of the checkpointing run-time overhead. The algorithms will on the one hand try to eliminate unnecessary checkpointing and on the other hand they will introduce extra job state savings where the danger of failure is considered to be severe.

4.1.1.1 *Last Failure Dependent Checkpointing (LastFailureCP)*

The main disadvantage of unconditional periodic job checkpointing (PeriodicCP) is that it performs identically whether the job is executed on a volatile or on a stable resource. The goal of LastFailureCP is to reduce the overhead introduced by excessive checkpointing in relatively stable distributed environments, *i.e.*, the algorithm omits unnecessary checkpoints of a job based on its estimated total execution time and the failure frequency of the resource, to which the job is assigned.

For each resource the algorithm keeps a timestamp of its last detected failure. Furthermore, each checkpointing request generated by a running job is evaluated and the checkpointing is performed only if time elapsed since the last resource failure is smaller than the expected remaining job execution time. To prevent excessively long checkpoint suspension, a maximum number of omissions can be limited.

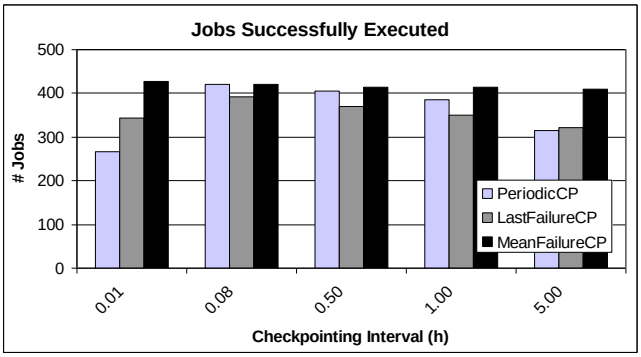
4.1.1.2 *Mean Failure Dependent Checkpointing (MeanFailureCP)*

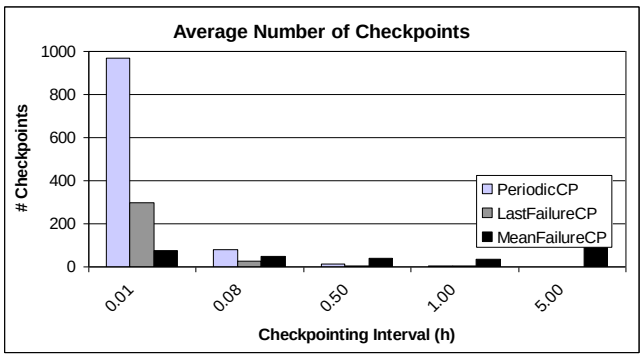
Contrary to LastFailureCP that only considers checkpoint omissions, MeanFailureCP dynamically modifies the initially specified checkpointing frequency to deal with inappropriate checkpointing intervals. The algorithm modifies the checkpointing interval based on the run-time information on the remaining job execution time and the average failure interval of the resource where the job is assigned, which results in a customized checkpointing interval.

While PeriodicCP and LastFailureCP are first run after the expiration of the predefined checkpointing interval, the MeanFailureCP activates checkpointing within a fixed and preferably short time period after the beginning of a job execution. The latter approach opens the possibility to modify the checkpointing frequency at the early stage of job processing. Each time the checkpointing is performed, the checkpointing interval is adapted as follows: if remaining job execution time is smaller than the average failure interval, the

frequency of checkpointing will be reduced by increasing the current checkpointing interval; on the other hand, when the above-mentioned condition is not satisfied, it seems to be desirable to decrease the checkpointing interval and thus to perform checkpointing more frequently. Depending on the stability of the execution environment at hand, different values for the increase / decrease of the checkpointing interval can be chosen, *i.e.*, a certain percentage of the initial checkpointing interval or a percentage of the total job execution time. Experiments have shown that gradual incrementation by a percentage of the initial checkpointing interval ensures rapid achievement of the optimal checkpointing frequency in most distributed environments. However, in case of rather reliable systems, the calibration of the checkpointing interval can be accelerated by considering a desirable percentage of the job execution time.

4.1.1.3 *Simulation Results*





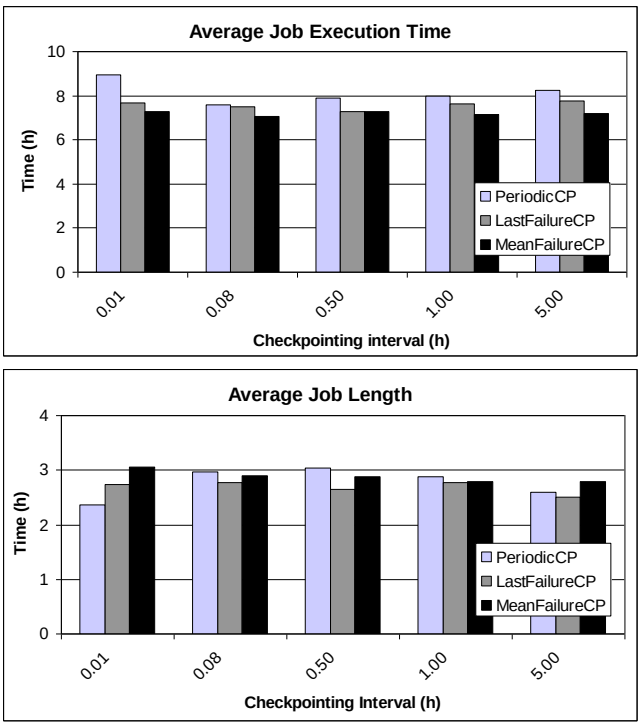


Figure 34: Checkpointing heuristics performance for varying initial checkpointing interval.

Figure 34 shows a comparison between the performance of the proposed dynamically adapting heuristics and the PeriodicCP approach for a randomly varying initial checkpointing interval. From the figures it is clear that the efficiency of PeriodicCP strongly depends on the chosen value of the interval, which remains constant during the simulation. For instance, overly frequent as well as scarce checkpointing can result in up to 40% decrease in number of processed jobs, compared to the best achieved situation, and significantly increase the average job execution time. Furthermore, from the figure can be observed that at high checkpointing frequencies the average job length significantly decreases. This relates to the fact that exaggerated checkpointing substantially prolongs job execution and therefore only short jobs finish within the observed time interval. However, when the checkpointing interval decreases and longer jobs can get processed, an increase in job run-time is in effect.

The results achieved with PeriodicCP are partially improved by LastFailureCP due to omittion of redundant checkpoints. Apparently, the technique provides the best results for short checkpointing intervals. Since the algorithm does not consider checkpoint insertion, it performs slightly worse than PeriodicCP for large checkpointing interval values. However, in the latter case the effectiveness of LastFailureCP strongly depends on failure periodically. In the best case when failures occur quite periodically and thus can easily be predicted by the algorithm, LastFailureCP will perform similar to PeriodicCP.

Finally, the fully dynamic scheme of MeanFailureCP proves to be the most effective. Starting from a random checkpointing frequency it results in a number of executed jobs and average job run-time that are close to the results achieved by PeriodicCP with the best performing checkpointing interval. As can be seen from

the simulation results, this selective increase in checkpointing keeps the number of processed jobs and the average execution time of MeanFailureCP more or less constant, while in the case of the PeriodicCP and LastFailureCP algorithms the performance drops considerably.

4.1.2 Replication-based Heuristics

4.1.2.1 *Load Dependent Replication (LoadDependentRep)*

Providing fault-tolerance in distributed environments through replication has as an advantage that otherwise idle resources can be utilized to run job copies without significantly delaying the execution of the original job. Obviously, the more job copies are running on the grid, the larger is the chance that one of them will execute successfully. On the other hand, running additional replicas on a distributed environment with an insufficient number of free resources can considerably reduce throughput and prolong job execution. To deal with this dilemma, the proposed heuristic [Chtepen09] considers the system load and postpones or reduces replication during peak hours. The algorithm requires a number of parameters to be provided in advance, *i.e.*, the minimum and maximum number of job copies, and the CPU limit. In each iteration, the system status is observed and based on this information jobs are scheduled as follows: if there are more free CPU's than the predefined CPU limit, additional job replica's are started, until the maximum number of copies is reached for each job or until the amount of free CPU's drops below the CPU limit; when there is an insufficient number of free CPU's, additional replicas are started only for jobs with less copies than the predefined minimum number. Even if the grid system is heavily loaded it can be desirable to initialise the minimum number of job replicas to more than 1, since the failure rate of resources in distributed environments increases with the intensity of the workload running on them. When one of the job duplicates finishes, other replicas are automatically cancelled. Important to notice is that the algorithm assigns each new job replica to a distributed site with the smallest number of identical replicas, since spreading replicas over different sites increases the probability that one of them will be successfully executed. Furthermore, inside the chosen site the job will be submitted to the fastest available resource with no identical job replicas. Distribution of similar replicas to a single CR is avoided because it is assumed that CPUs inside a single node have more chance to fail simultaneously in case of the resource malfunction.

4.1.2.2 *Failure Detection and Load Dependent Replication (FailureDependentRep)*

To increase fault-tolerance of the previously discussed LoadDependentRep heuristic, the approach was combined with a failure-detection technique [Chtepen09]. The principle of failure detection is straightforward: as soon as a resource failure is discovered, all jobs submitted to the failed resource are redistributed. The algorithm proceeds as LoadDependentRep, except that in each scheduling round not only newly arrived jobs are considered for submission, but also all jobs distributed to failed nodes.

The algorithm makes use of the dynamic information on status of computational resources, however this information is most likely to be available with certain delay. This means that although the method offers a higher level of fault-tolerance compared to solely replication-based strategies, it does not ensure job execution.

4.1.2.3 *Adaptive Checkpoint and Replication-Based Fault-Tolerance (CombinedFT)*

In this section a combined checkpointing and adaptive replication-based scheduling approach is considered [Chtepen09] that dynamically switches between both techniques based on run-time information on system load. The algorithm can be particularly advantageous for distributed environments with frequent or unpredictable alternations between peak hours and idle periods. In the first case, replication overhead can be avoided by switching to checkpointing, while in the second case the checkpointing overhead is reduced by using low-cost replication.

When the CPU availability is low the algorithm is in checkpointing mode. In this mode, CombinedFT rolls back, if necessary, the earlier distributed active job replicas and starts job checkpointing. When processing the next job, the following situations can occur:

- **One or several replicas of this job are already running**: start checkpointing the most advanced active replica, cancel execution of other replicas.
- **No active replicas of this job are running but there are some free CPUs available**: start the job on the least loaded available resource within the least loaded site.
- **No active replicas of this job are running and there are no free CPUs but there exists another job with several active replicas:** select a random replicated job, start checkpointing its most advanced active replica, cancel execution of its other replicas, use the released resources to schedule the idle job.
- **None of the above described cases are applicable:** skip the scheduling round.

The algorithm switches to replication mode when either the system load decreases or enough resources restore from failure. In replication mode all jobs with less than the maximum number of replicas are considered for submission to the available resources. When a job is selected, it is assigned to the fastest resource (with no similar job replicas) connected to a grid site with the maximum number of free CPUs and the smallest number of identical replicas. If the job was previously in checkpointing mode and the replication completed successfully, the checkpointing is switched off.

4.1.2.4 *Simulation results*

The performance of the replication-based and hybrid approaches was compared against the performance of the best checkpointing heuristic (MeanFailureCP). The comparison is performed within grid systems with varying load and availability. Four replication algorithms are considered: UnconditionalRep(2), unconditional job replication with 2 job copies; UnconditionalRep(3), unconditional job replication with 3 copies; LoadDependentRL(1,3,40) adaptive replication with the minimum and maximum number of job replicas set to respectively 1 and 3, and the free CPU limit initialized to 40 (approximately 1/3 of the total grid capacity); FailureDependentRep(1,3,40), failure detection and adaptive replication based algorithm with the same parameters as LoadDependentRep. Also the performance of First Come First Served (or UnconditionalRep(1)) was observed to serve as a reference for comparison with the other algorithms. The combined approach (CombinedFT) is initialized with the same replication parameters as FailureDependentRep and switches in the checkpointing mode to the MeanFailureCP approach. The chosen parameter values for the replication-based heuristics are not necessarily optimal but they are

believed to be reasonable for the case at hand. The term “unconditional job replication algorithm” refers to an algorithm that sequentially processes jobs arriving to the grid scheduler, based on the timestamp of their arrival. Independent of the current system load, the algorithm creates for each job a predetermined number of replicas that are assigned to different available resources, until all resources are filled.

The distributed environment considered consists of 128 computational resources, equally spread over 4 sites. Sites are connected by WAN links with an equal bandwidth of 100 Mbit/s and latency varying from 3 to 10 ms. Computational resources inside the sites communicate through LAN networks arranged into a star topology, with a link bandwidth of 100 Mbit/s and latency of 1ms. The algorithms are evaluated for varying grid availability, where the latter is achieved by differentiating the frequency of resource failures, while keeping constant restore times (approximately 30 min).

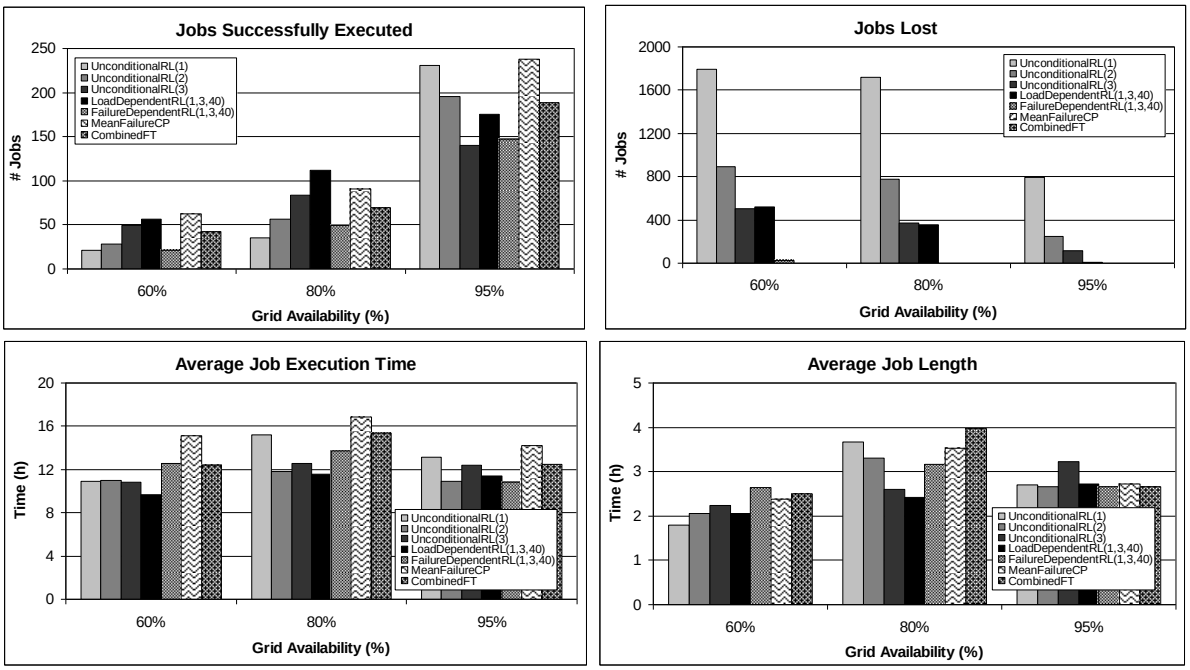


Figure 35: Performance of replication-based, checkpointing-based and hybrid algorithms on heavily loaded grids with varying availability: number of successfully executed jobs, number of jobs lost, average job execution time and average job length.

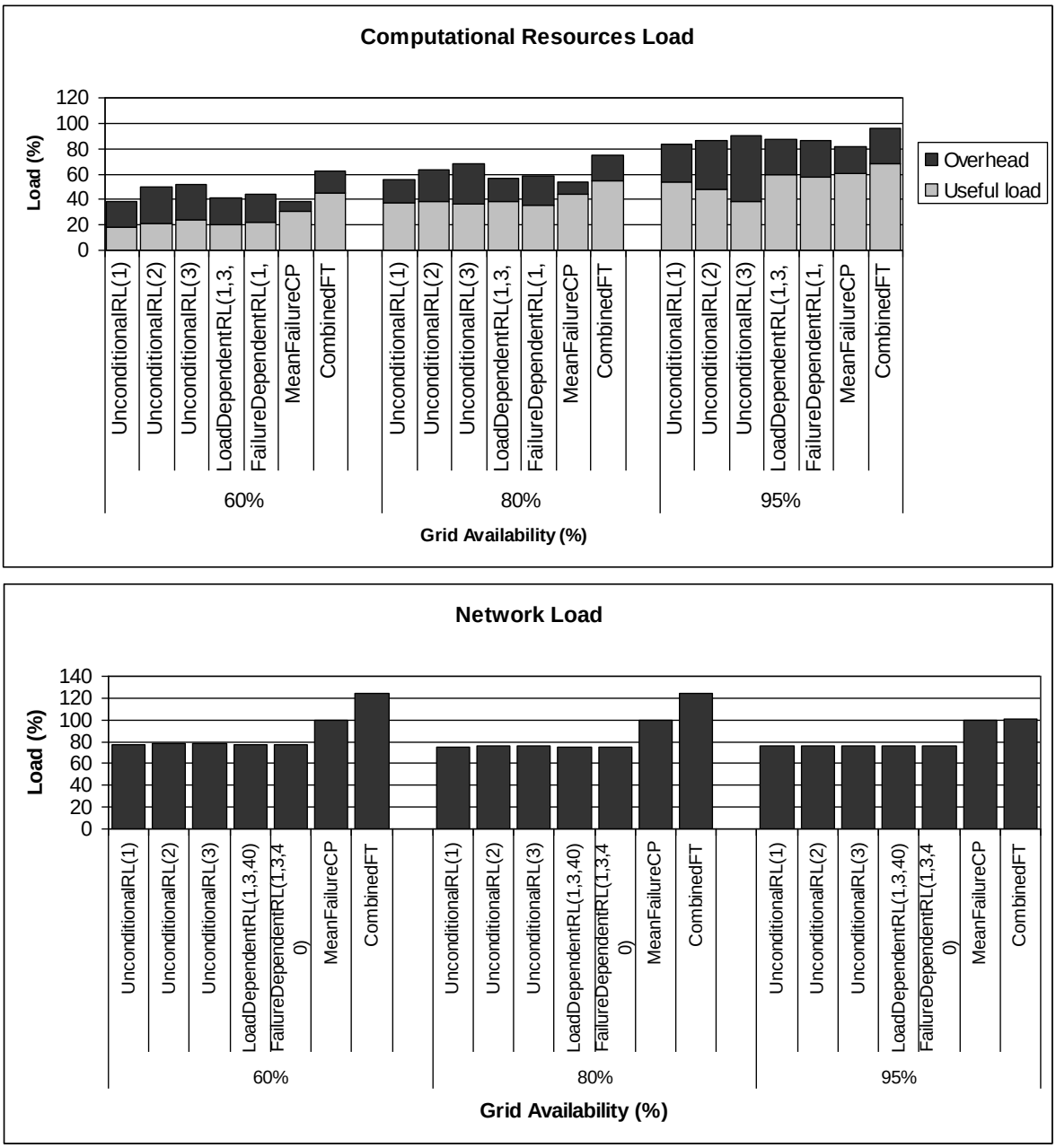


Figure 36: Performance of replication-based, checkpointing-based and hybrid algorithms on heavily loaded grids with varying availability: computational resources and network load.

Two job submission scenarios are considered. In the fist scenario about 7,000 jobs arrive into the system during the observation period of 24 hours (simulated time), which leads to heavy loaded grid with long periods of system overload alternating with relatively short “idle” time-intervals. In the second scenario jobs are generated occasionally (about 700 during 24 hours of simulated time), resulting into lightly loaded grid system where most of the time a large part of the resources remains idle. To warrant low system utilization, also the average job length is reduced from 2.5 hours in the first scenario to 0.3 hours in the second one.

The size of job input and output data in both simulation scenarios is set to 10 GB, to yield large data volumes often generated by real-world computationally intensive applications. Average size of generated checkpoints amounts to several GB.

It is important to mention that workload parameters, as well as resource failure patterns, are derived from logs that were collected from several large scale parallel production systems.

Figure 35 and Figure 36 visualize the evaluated scheduling methods' performance on a highly loaded grid, while Figure 37 and Figure 38 summarize the results for low grid load. The following system parameters are observed: number of successfully executed / lost jobs, average job execution time, average job length, system and network load. Important to notice is that a replicated job is assumed to be lost when all its replicas were started and afterwards failed.

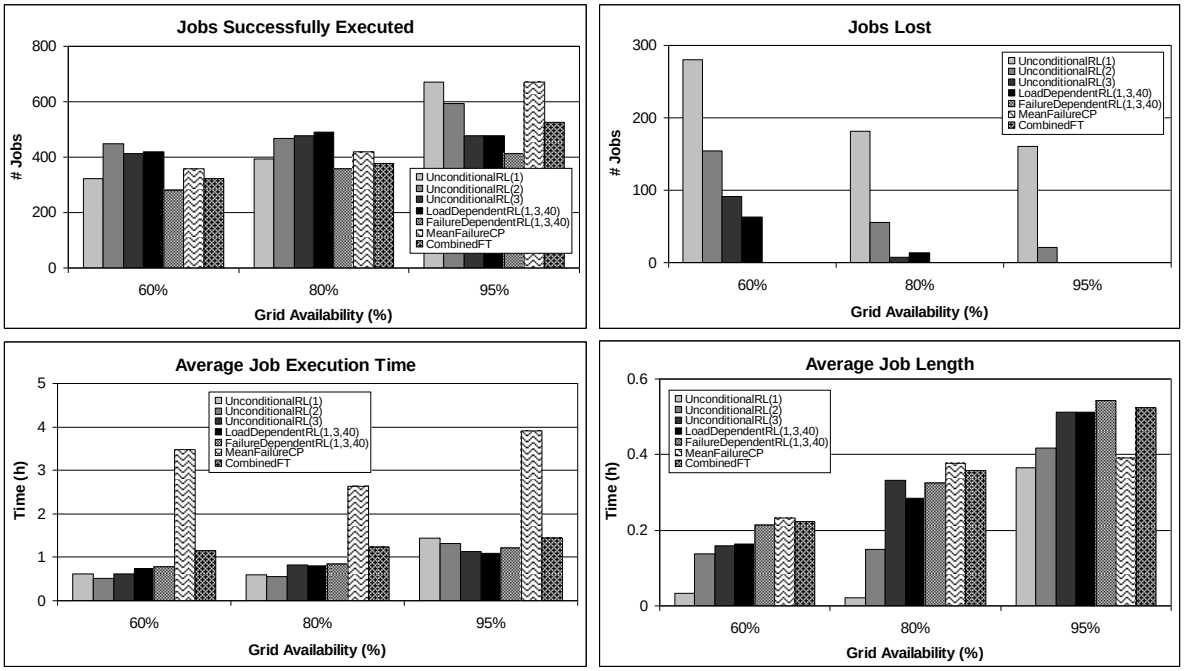


Figure 37: Performance of replication-based, checkpointing-based and hybrid algorithms on grids with low load: number of successfully executed jobs, number of jobs lost, average job execution time and average job length.

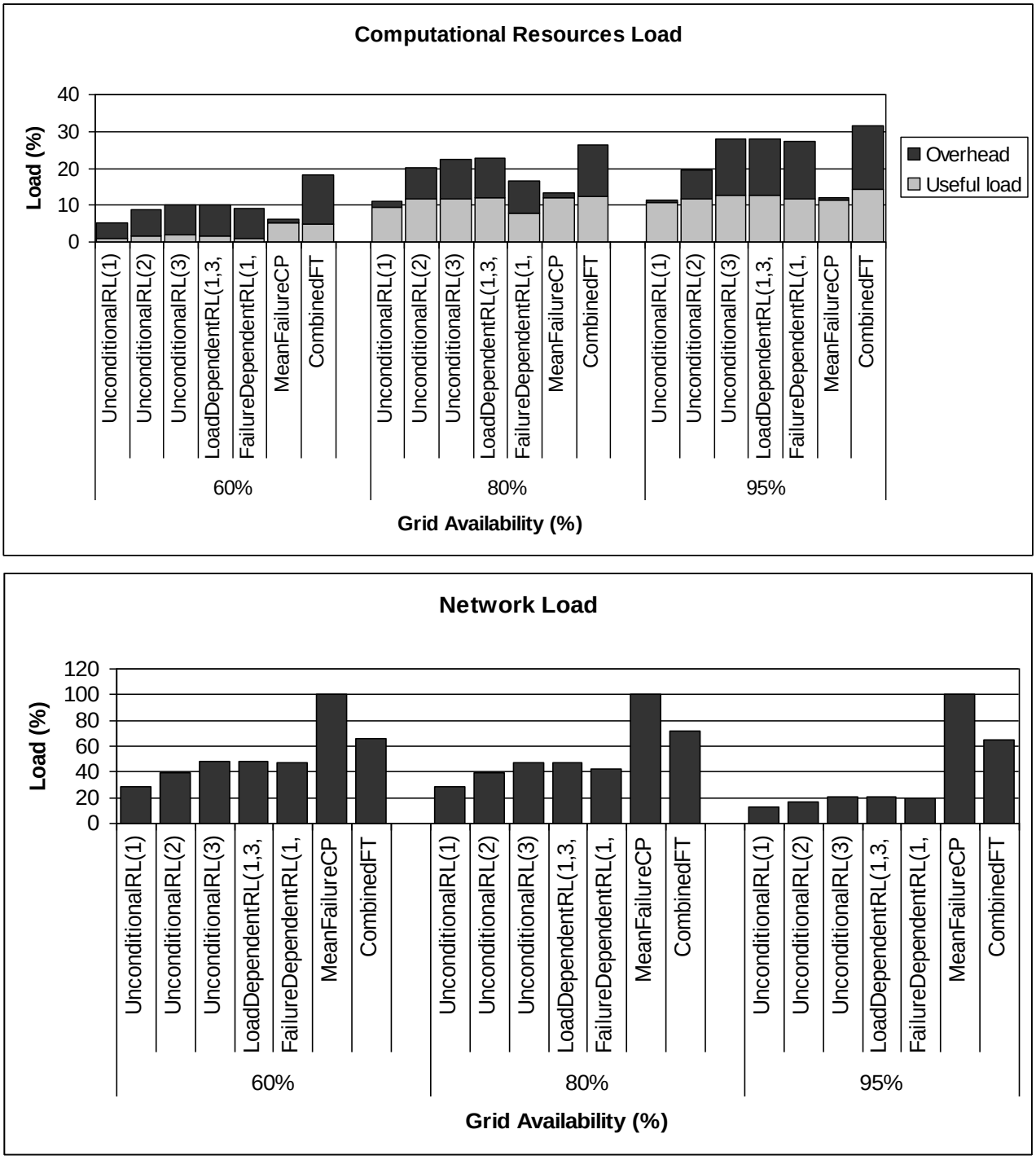


Figure 38: Performance of replication-based, checkpointing-based and hybrid algorithms on heavily loaded grids with varying availability: computational resources and network load.

The figures show that for heavily as well as lightly loaded grids with relatively low availability, additional replication clearly provides better system performance and lower job loss rate. This is the consequence of the fact that replication-related overhead is compensated by increased grid reliability and consequently by a higher ratio of successfully executed jobs. However, as the grid availability improves (95%), additional replication provided by UnconditionalRep(2) and UnconditionalRep(3) leads to system throughput reduction. This reduction is getting more significant as the grid load increases and resources become more scarce. The results for LoadDependentRep show that the performance of unconditional replication can be improved by postponing the execution of additional replicas during the peak hours. The higher the system load, the more gain can be achieved from the postponement. On the other hand, for slightly loaded systems, LoadDependentRep performs only slightly better than UnconditionalRep(2), since replication almost never has to be delayed. The main advantage of FailureDependentRep is certainly its high reliability in absence of sophisticated mechanisms for providing fault-tolerance. Implementation of the algorithm requires only a replica counter and a simple job monitoring facility. Another benefit is that failure-sensitive long jobs have a higher chance to finally get processed due to the restart mechanism. The disadvantage of FailureDependentRep is the slower grid performance (larger average execution times) as a result of the postponed replication in combination with the run-time overhead related to repetitive restart of failed jobs. However, lightly loaded systems are less sensitive to this last disadvantage since most of the time enough computational resources are available and thus multiple job restarts do not penalize the execution of other jobs. In the condition of high load, the fully fault-tolerant MeanFailureCP results in the best system throughput compared to the other considered heuristics. This is the consequence of the considerable overhead introduced by the execution of additional replicas in an overloaded grid system. On the other hand, the average job execution time in case of the checkpointing approach is always relatively high, which leads to the algorithm performance reduction in the lightly loaded grid, where replication provides for almost costless fault-tolerance. However, it is important to notice that the exact relation between the performance of the checkpointing and the replication-based solutions is largely determined not only by the system load, but also by the run-time cost of checkpointing and the size of job input and output data. Finally, the throughput and average job execution times generated by CombinedFT for both types of system load are located between respectively the throughputs and average job execution times of FailureDependentRep and MeanFailureCP. This is the logical consequence of the fact that job submissions are clustered in time and that the heuristic performs some calibrations, after each variation in the system load, before achieving its “optimal” state. Regarding the other observed performance parameters, CombinedFT is almost fully fault-tolerant and results in one of the best average job lengths among the considered algorithms.

As was mentioned earlier, the load introduced on computational and network resources was also measured during the simulation experiments. For instance, Figure 36 and Figure 38 show the proportions of the total active system time (resource failure times not included) that was occupied by performing “useful” and “redundant” work, for systems with respectively high and low load. As “useful work” we consider the execution of the workload belonging to a successfully terminated job, while “redundant work” pertains to interrupted replica’s, checkpointing overhead, failed jobs, *etc*. In the same figures, the network load, resulting from the different heuristics execution, is shown as a proportion of the network load generated by the MeanFailureCP heuristic.

From the observations can be concluded that under all circumstances replication leads to significant computational overhead. Apparently, the amount of “redundant” workload grows (compared to the amount

of “useful” work) as the degree of replication increases. This overhead can partially by reduced by applying more intelligent replication-based heurists, such as LoadDependentRL and FailureDependentRL. Concerning network load, replication gives relatively good results, compared to checkpointing-based heuristics. In the worst scenario of a heavily loaded grid, the network load of replication algorithms lies at least 20% lower than the load imposed by MeanFailureCP. Notable is that in the latter scenario, all replication heuristics perform similar with respect to this parameter. This can be explained by the fact that almost all available resources are permanently occupied (independent of the number of redundant replicas) and thus the network has to transfer more or less the same amount of data.

On the other hand, the checkpointing-based algorithms result in low computational overhead but at the same time they make extensive use of network resources, as they require regular transfers of checkpoints from computational resources to checkpointing servers.

Finally, the computational and network overhead of the combined approach strongly depends on variations in the system load. When the system load is high the hybrid approach mainly operates in the checkpointing mode. However, the occasional switching to the replication mode adds some additional computational and network overhead. On slightly loaded grids, the combined approach results in lower overhead than the solely checkpointing-based solutions but at the same time it introduces redundant computations due to replica execution.

4.1.3 Conclusion

Fault-tolerance forms an important problem in the scope of distributed computing environments. To deal with this issue several adaptive heuristics, based on job checkpointing, replication and the combination of both techniques, were designed. The heuristics where evaluated under varying system load and availability. The results have shown that the run-time overhead characteristic to periodic checkpointing can significantly be reduced when the checkpointing frequency is dynamically adapted in function of resource stability and remaining job execution time. Furthermore, adaptive replication-based solutions can provide for even lower cost fault-tolerance in systems with low and variable load, by postponing replication in function of system parameters. Finally, the advantages of both techniques are combined in the hybrid approach that can best be applied when the distributed system properties are not known in advance.

4.2 Data Consolidation and Resiliency

4.2.1 General

In this section we examine Data Consolidation (DC) and resiliency. DC applies to data-intensive applications that need several pieces of data for their execution. The DC problem consists of three interrelated sub-

problems: (i) the selection of the replica of each dataset (i.e., the data repository site from which to obtain the dataset) that will be used by the task, (ii) the selection of the site where these pieces of data will be gathered and the task will be executed and (iii) the selection of the paths the datasets will follow to arrive at the data consolidating site. It is evident that the site were the datasets consolidate becomes a critical point for the operation of the Grid Network. In case this site fails in any way, then the DC operation needs to be repeated, increasing the task delay and the network load.

In [D5.7] we proposed a number of simple Data Consolidation techniques and examined how they are affected by the network design (e.g., number of sites). In this deliverable we add to the proposed DC techniques, resilience features in order to provide fault-tolerance to the DC operation. Furthermore, we propose new DC techniques in order to cope with the increase in the network load. This way we show that it is possible to perform DC efficiently with increased fault-tolerance.

4.2.2 Proposed Techniques

4.2.2.1 *Data Consolidation Techniques*

In [D5.7] we proposed a number of DC schemes that considered only the data consolidation (*Consolidation-Cost - ConsCost algorithm*) or only the computation (*Execution-Cost - ExecCost algorithm*) or both kinds (*Total-Cost - TotalCost algorithm*) of task requirements. In this deliverable we also propose the *Total-Cost Queuing (TotalCost-Q) algorithm:* This algorithm is similar to the *Total-Cost* algorithm with the addition that we take into account the communication queuing delays at the links. Generally, scheduling and routing can be improved if link queue length information is available. However, in practice, it is quite difficult to keep such information up-to-date at all the sites. In Grid networks there are appropriate mechanisms for gathering and communicating queue length information, such as the Network Weather Service (NWS) [Wolski99]. In our simulations, for the Total-Cost Queuing algorithm we assume that each site has instantly valid and updated queue length information for every network link. One of our goals is to investigate the benefits induced to the DC problem, when such information is used.

Moreover, in this deliverable regarding the third sub-problem of the DC, that is the selection of the paths the datasets will follow to arrive at the data consolidating site, we investigate a number of tree-based DC schemes. Intuitively this seems like the right thing to do, since in DC we have many repository sites (the leaves, or intermediate nodes of a tree) whose data are transferred to a DC site (the root). Since we want these transfers to occur concurrently and efficiently, we try to select the tree using as optimization criterion either the time for transferring the data over a link or the load of a link, as measured by the size of data queued or under transmission at it. In the algorithms proposed previously we used the tree constructed by Dijkstra's shortest path algorithm for routing. In the following algorithms we use Minimum Spanning Trees (MST) obtained by Kruskal's algorithm. Other tree algorithms can also be used, such as the Essau-Williams MST algorithm [Esau66], or algorithms for solving the Steiner tree problem.

Total-Cost - Minimim Spanning Tree (TotalCost-MST) algorithm: In this algorithm we first select the data replica holding sites and the DC sites using the TotalCost algorithm. Next, we assign to each link a weight, which is equal to the size of the queued data plus the size of the data that will pass through this link based on the

decisions made using the TotalCost algorithm. Finally, we apply a Minimum Spanning Tree algorithm (Kruskal) to construct the paths the datasets will follow.

The TotalCost-MST algorithm tries to improve the routing paths selected by the TotalCost algorithm, between the data holding sites and the data consolidation site. Figure 39 a shows an example of the paths initially selected by TotalCost algorithm, for the datasets A, B and C. Using the TotalCost (or the ConsCost) algorithm routing paths like the ones depicted in Figure 39a will be selected, where many datasets (A, B, C) cross the same network link(s). By applying the MST approach, the paths selected are improved by spreading the network traffic more evenly across the network (Figure 39 b).

Minimim Spanning Tree Cost algorithm (MST-Cost) algorithm: In this algorithm, we alter the order in which the three sub-problems of DC are handled. Specifically, when a new task requests service, we assign to each link (*i*, *j*) a weight based on the delay required for transmitting over this link, which includes both the queuing and the propagation delay. Next, we apply a MST algorithm (Kruskal) to construct the paths the datasets should follow, independently of the data replica holding and DC sites that will be chosen in the next phases. Following that, we apply the TotalCost algorithm in order to find these sites.

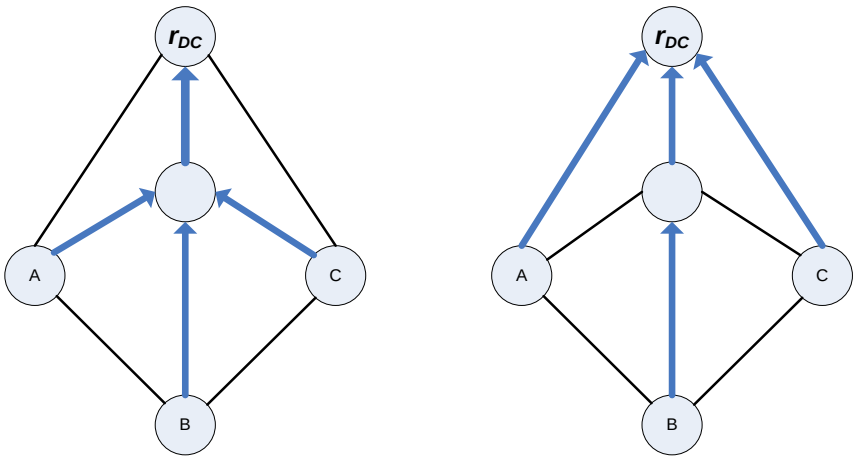


Figure 39: The paths selected for transferring datasets A, B and C to the r_{DC} site: a) using the original TotalCost algorithm, b) after the application of the MST approach in the TotalCost-MST algorithm.

4.2.2.2 *Resiliency Techniques*

We combine the proposed DC schemes with two relative simple resiliency techniques. In the first technique, called Double Site, with select two Data Consolidation sites, the first and the second “best” according to the corresponding DC technique used and we transfer data to both sites. The task is transfer only in the first site, while the other is used as a backup in case the first site fails. In the second technique, called Half Data, we again select in the same way two DC sites, however in the second-“best” site we transfer only half of the data needed by the task. This way we reduce the load induced to the network, but we also reduce the resiliency efficiency, since in a case of a failure in the first DC site, the rest of the data need to be transferred to the second DC site.

4.2.2.3 *Complexity*

In general the Data Consolidation problem is an NP-hard problem. A DC scheme has to select the *L* data holding sites and the DC site, choosing from *N* total sites. In the worst case, $\binom{N}{L+1}$ sets of sites must be examined to select the optimal one (with respect to time, overhead, cost, or any other criteria) for performing a DC operation. The number of possible sets of sites searchable by a DC scheme increases exponentially with the number of nodes/sites *N*. In order to solve the DC problem efficiently, schemes of polynomial complexity are required, which however will be sub-optimal.

Regarding the complexity of the proposed DC schemes, the Rand algorithm is polynomial, as it randomly chooses the *L+1* sites used in the DC operation. Similarly, the ExecCost algorithm is polynomial, since it randomly selects the *L* data holding sites, while the r_{DC} site is chosen among the *N* sites of the grid network based on the task execution time criterion. All the other proposed algorithm are based on the ConsCost algorithm. The complexity of these algorithms is determined by the complexity of the ConsCost algorithm, since any additional operations performed by these algorithms (for example the construction of the Minimum Spanning Tree) require polynomial time. In the ConsCost algorithm, for each candidate DC site we find the *L* sites that minimize the data consolidation time of the corresponding datasets. So, the execution of a shortest path algorithm, with polynomial complexity, is required for each candidate DC site $r_{DC.}$. At the end of the ConsCost algorithm, the r_{DC} site and the corresponding *L* sites with the minimum-maximum data consolidation time are selected. So, the total complexity of the ConsCost algorithm is polynomial.

Finally, we should note that the application of the mentioned resiliency techniques to the Data Consolidation algorithms does not increase their complexity.

4.2.3 Performance Evaluation

4.2.3.1 *Simulation Environment*

We use the same simulation environment as in [D5.7]. Specifically, we use the topology presented in Figure 40, which is very similar to the Phosphorus European testbed topology presented in [D5.2]. Our network consists of 11 nodes and 16 links, of capacities equal to 10Gbps. In our experiments we assume a P2P (opaque) network; the delay for transmitting between two nodes includes the propagation, queuing and transmission delays at intermediate nodes. Only one transmission is possible at a time over a link, so a queue exists at every node to hold the data waiting for transmission.

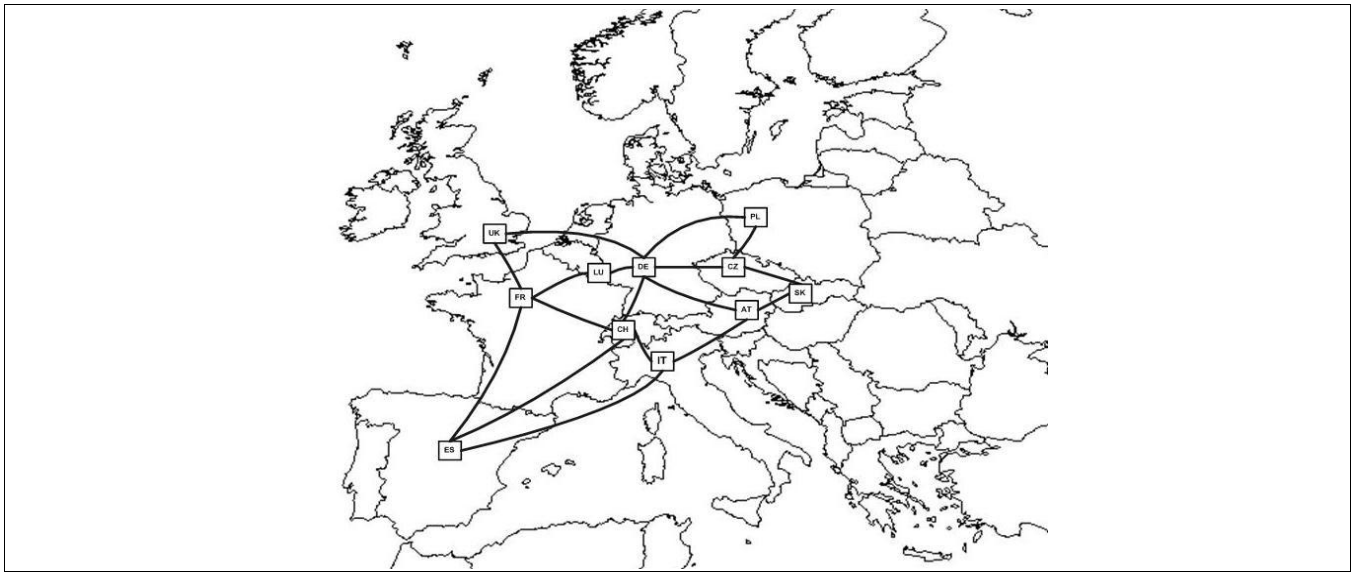


Figure 40: The topology used in our simulations.

We assume that 5 sites are equipped with computation and storage resources, while the others act as simple routers. We also assume that there is Tier 0 site in the network, which holds all the datasets, but does not have any computational capability. Each experimental scenario was run 5 times, using an independent random seed. In every repetition, the placement in the network of the 5 sites and the Tier 0 was random. Furthermore, experiments were performed using more than 5 sites equipped with computation and storage resources.

The size of each dataset is given by an exponential distribution with average I (GB). At the beginning of the simulation a given number of datasets are generated and two copies of each dataset are distributed in the network; the first is distributed among the sites and the second is placed at Tier 0 site. The storage capacity of each storage resource is 50% of the total size of all the datasets. Since the storage capacity is bounded, there are cases where a site does not have the capacity required to store a needed dataset. In such a case, one or more randomly chosen, unused datasets are deleted until the new dataset can be stored at the resource.

In each experiment, users generate a total of 10.000 tasks, with exponential interarrival times of average value $1/\lambda$. Unless stated otherwise, we assume $1/\lambda$=0.01 sec. In all our experiments we keep constant the average total data size S that the tasks require:

$$S = L \cdot I,$$

where L is the number of datasets a task requests and I is the average size of each dataset. We use average total data size S equal to 600 GB and 800 GB and examine the (L, I) pair values. In each experiment the total number of available datasets changes in order for their total size to remain the same: 15 TB and 20 TB, respectively.

4.2.3.2 *Performance Metrics*

We use the following metrics to measure the performance of the algorithms examined:

- The average task delay, which is the time that elapses between the creation of a task and the time its execution is completed at a site.

- Task success ratio: This is the ratio of the tasks that where successfully scheduled, over all the tasks generated. When a large number of tasks are queued or under execution, then it may be difficult for the scheduler to find a resource with sufficient large free storage space, where a new task's datasets can consolidate. In this case the task cannot be scheduled and it fails.

- The average load per task imposed to the network, which is the product of the size of datasets transferred and the number of hops these datasets traverse.

- The DC probability, which is the probability that the selected DC site will not have all the datasets required by a task and as a results DC will be necessary.

The first metric characterizes the performance of the DC strategy in executing a single task, while the second and third express the overhead the DC strategy induces to the network and storage resources. The fourth metric gives information on the way the DC site is selected, with respect to the datasets that are located (or not) at this DC site.

4.2.3.3 *Simulation Results*

Figure 41 shows the task success ratio of the Rand, ConsCost, ExecCost and TotalCost DC algorithms with and without resiliency techniques, when tasks requests different number of datasets, *L*, for their execution. The average total data size per task is *S*=800 GB. When a resiliency technique is applied then the storage resources are overused, causing many task failures. Moreover, the number of data transfers in the network increases, causing network congestion and larger task delays. As a result the average time a task reserves the storage resources also increases. An efficient DC scheme, that better handles the network congestion can achieve smaller task delay, smaller storage resource reservation time per task and as a result larger task success ratios.

The Double Site resiliency technique in Figure 41 induces more network load and leads to the reservation of larger total storage space than the Half Data technique; and as a result the task success ratio is lower. When no resilience technique is applied, then the network load and the storage space reserved are smaller and the success ratio larger. Also, in [D5.7] we observed that the TotalCost algorithm performed the best results in comparison to the other DC algorithms (Rand, ConsCost, ExecCost). However, this is not the case when the TotalCost algorithm is combined with a resilience technique. The TotalCost_Double algorithm produces the worst results, especially when the number of datasets requested (*L*) is larger than 4. These results indicate that the TotalCost algorithm does not efficiently handle the increase in the network load. On the other hand the TotalCost_HalfData algorithm, seems to produce better results than the other HalfData algorithms.

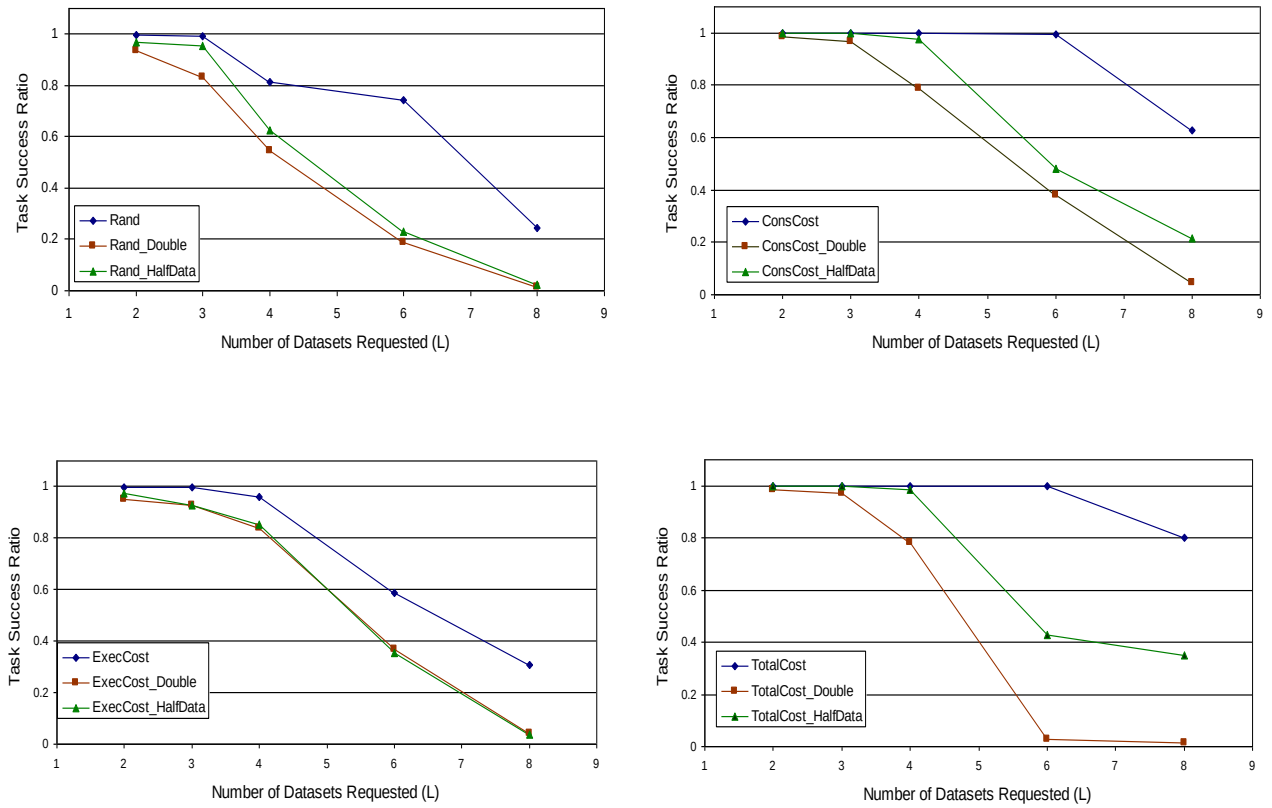


Figure 41: The task success ratio of the Rand, ConsCost, ExecCost and TotalCost DC algorithms with (Double Site, Half Data) and without resiliency techniques, when tasks requests different number of datasets, *L*, for their execution. The average total data size per task is *S*=800 GB.

Based on the above results and observations we proposed the TotalCost-Q, TotalCost-MST and MST-Cost DC algorithms, trying to improve the behaviour of the DC and resiliency algorithms. First we performed a number of simulations with the TotalCost-Q, TotalCost-MST and MST-Cost algorithms without resiliency features, and compared them with the TotalCost algorithm. Our results show that tree-based algorithms and especially Minimum Spanning Tree algorithms, fit quite well to the DC problem and can considerably improve performance. Moreover, we show that taking into account the queuing delays is not so beneficial, and does not justify the communication and processing overhead required for communicating such information across the network. Furthermore, it seems that the order in which the DC operations are performed and in particular the order in which the corresponding DC sub-problems are solved, significantly affects the efficiency of the DC schemes.

In Figure 42 we observe that the TotalCost-MST algorithm performs better than the TotalCost, the TotalCost-Q and the MST-Cost algorithms. The TotalCost-MST algorithm builds more efficient paths than the Dijkstra algorithm, resulting in smaller network load and average task delay. Furthermore, we observe that the TotalCost-Q algorithm produces only slightly better results than the TotalCost algorithm, though the former takes queuing delays into account. Generally, the sizes of the queue lengths must be large, in order for the

queuing delays to have a noticeable effect. So, in most practical cases, the knowledge of the links queuing delay is not necessary for making efficient DC related decisions. Finally, we observe that the MST-Cost algorithm produces worse results than the TotalCost, TotalCost-Q and TotalCost-MST algorithms. This occurs because by applying a MST algorithm first, we limit the number of links and as a result the number of possible decisions one can make regarding the data repository and the DC sites. This has a significant negative effect on the final results.

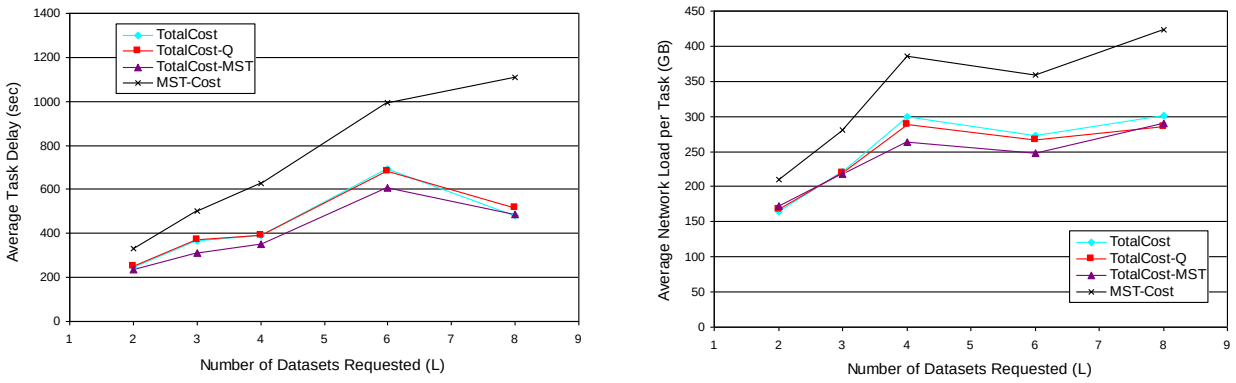


Figure 42: (a) The average task delay (in sec) and (b) the average network load per task (in GB), for the TotalCost-Q, MST-Cost and TotalCost-MST DC algorithms, when tasks requests different number of datasets, *L*, for their execution. The average total data size per task is *S*=800 GB.

Figure 43 shows the task success ratio of the TotalCost-Q, MST-Cost and TotalCost-MST DC algorithms with (Double Site, Half Data) and without resiliency techniques, when tasks request different number of datasets, *L*, for their execution. The average total data size per task is *S*=800 GB. We observe that the TotalCost-Q_HalfData, TotalCost-Q_Double, MST-Cost_HalfData, MST-Cost_Double produce similar results to that of the algorithms in Figure 41. However, the TotalCost-MST_HalfData and TotalCost-MST_Double algorithms produce the best task success ratio results, while both algorithms have similar values. This is due to the fact that TotalCost-MST algorithm, during the construction of the MST takes into account the data transfers needed for transferring data to both sites (first-"best" and second-"best" site). As a result the data transfers are performed more efficiently, reducing the congestion and the task delay, and increasing the task success ratios.

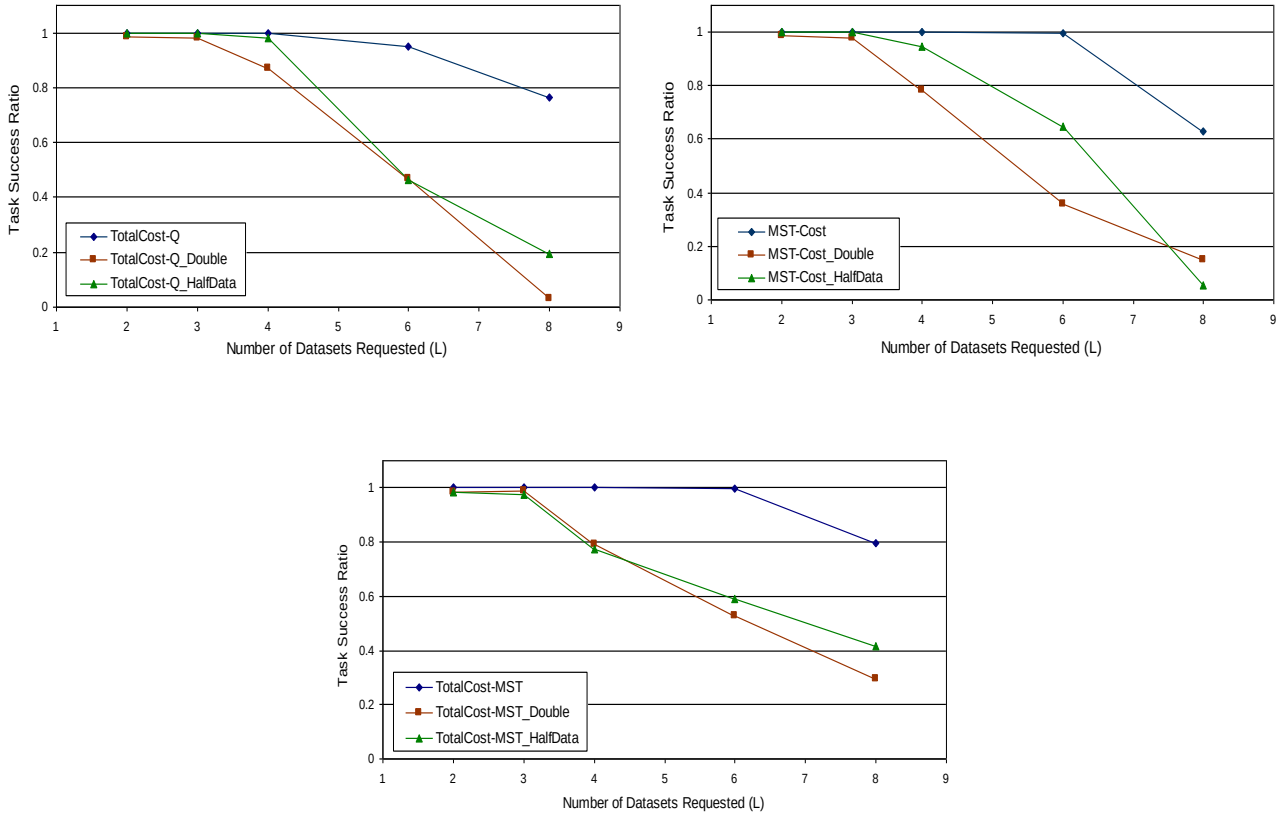


Figure 43: The task success ratio of the TotalCost-Q, MST-Cost and TotalCost-MST DC algorithms with (Double Site, Half Data) and without resiliency techniques, when tasks requests different number of datasets, *L*, for their execution. The average total data size per task is *S*=800 GB.

4.2.4 Conclusions

In this section we applied resiliency techniques to the Data Consolidation (DC) operation. Our main metric of interest is the task success ratio, defined as the ratio of the tasks that where successfully scheduled, over all the tasks generated. When a large number of tasks are queued or under execution, then it may be difficult for the scheduler to find a resource with sufficient large free storage space, where a new task's datasets can consolidate. In this case the task cannot be scheduled and it fails. We showed that the efficiency with which the DC schemes handle the network congestion, caused by the applied resiliency techniques, strongly affects the task success ratio achieved. For this reason the DC schemes that use tree based techniques for the routing of the datasets, achieve larger task success ratio than the other DC schemes examined.

5 Network Protection vs. Job Relocation

As described in the previous chapters, resilient network design is about reserving the necessary network resources in order to successfully route a predefined demand matrix and making sure that in case of a network failure a connection that is affected by the failure will be able to be recovered. There are two paradigms on how to provide for network recovery: a) network protection schemes, which reserve an either dedicated or backup path that is used in case of a link/node failure to route failed connections and b) restoration schemes which strive to dynamically find an alternate route at the event of a failure to recover failed lightpaths. In this section, we incorporate grid-specific computational resource resilience schemes into the problem of network resilience and see evaluate its potential benefits. As an example, consider the Grid instance illustrated in Figure 44. The blue lines represent the primary path from the client to Server 1, whereas the red lines represent the associated back up path. In a typical protection situation, the backup path would be taken in case of a failure. Instead, if we relocate the job to Server 2, only need the path consisting of 4 links would be needed instead of the 5-link backup path. This relocation scheme can be reinforced by the anycast principle, which states that the exact location of job execution should be indifferent to the user and instead only the timely delivery of the job results is of significance.

In what follows, we conduct an offline ILP study, aiming at quantifying the bandwidth and the routing paths required, given:

1. A plain network: simply solving a simple RWA problem.

2. A Network with dedicated protection

3. A Network with shared protection

4. A Network, with the additional flexibility of relocating jobs or using the shared protection scheme.

The objective of the algorithms presented below is to jointly minimize the number of wavelengths used by primary and protection paths. In the case, where relocation is possible, we also study the bounds on the necessary extra computing power required by involved resources to carry out relocation seamlessly.

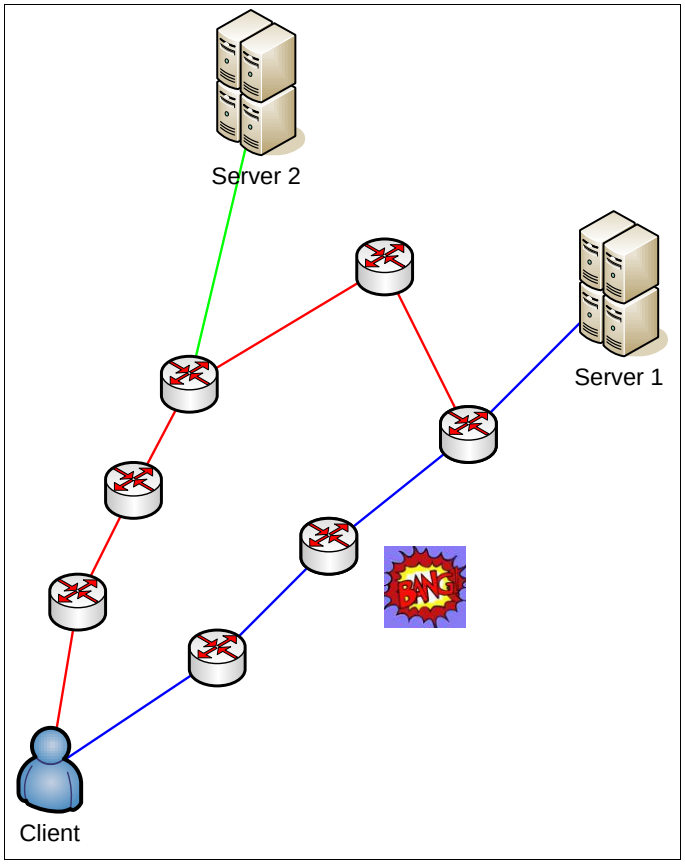


Figure 44: Instead of protecting the path from the client to Server 1, we can alternatively relocate the job to Server 2 , economizing thus in required network resources.

5.1 Network Protection without Relocation

5.1.1 Plain Network without Wavelength Conversion

The ILP which we describe here is based on the one described in [Hui Zang]. In the plain network case, we just create a network, which is dimensioned such that every connection request is satisfied. This gives us an idea on how much bandwidth must be reserved due to the protection scheme.

First we give an overview of the used notation and parameters. The network is modelled as an undirected graph $G = \{V, E\}$ where V is the set of vertices and E is the set of edges. Depending on the problem instance we intend to solve, a vertex may be equipped with wavelength conversion capabilities. The

maximum number of wavelengths a link can carry is W . Let also the number of nodes in the network denoted by $N = |V|$ and $L = |E|$ stand for the number of links in the network.

A list of the variables and constants used throughout our ILP formulations is shown in below.

Notation	Explanation
$\phi_{i,j}$	The number of connections requests originating at source node i and being terminated at node j. This variable takes its value from the respective position in the demand matrix, which serves as input for the ILP problems.
$p_{i,j}$	The number of working paths (primary) carried by the link connecting node i with node j.
$P_{i,j}^{s,d,w}$	Boolean variable which signifies whether wavelength w on the link from node i to node j is used to serve a demand between source node s and destination node d.
$\lambda_{s,d}^{w}$	Variable stating that there is a light path starting at source node s to destination node d on wavelength w.
$R_{(i,j),(k,l)}^{s,d,w}$	Boolean variable which signifies whether the wavelength w is used on link (i,j) to protect a primary path from source node s to destination node d on link (k,j). This variable is used when describing the shared protecting ILP.
$r_{i,j}^{w}$	Boolean variable that signifies whether wavelength w on link (i,j) is used by a restoration route.
$s_{i,j}$	The number of wavelengths on link (i,j) which are used by backup paths.
$R_{(i,j),(k,l)}^{s,d,w,\delta}$	Boolean variable that signifies whether wavelength w on link (i,j) protects a path from s to d which runs on (k,l). The resource which receives the job in case of link failure is called δ . It is possible that $\delta = d$ in which case we speak of the normal shared network protection.
Δ	The set of all resources. This is the network without the pure client nodes.

This problem can be modeled as a multi-commodity flow problem which is bound by three constraints:

$$Minimize \sum_{(i,j)\in E} p_{(i,j)}$$

1. The capacity constraint.

$$\sum_{1 \le s,d \le N} P_{i,j}^{s,d,w} \le 1 \quad \forall (i,j) \in E, \qquad 1 \le w \le W$$

2. Flow conservation

$$\sum_{i:(i,j)\in links} P_{i,j}^{s,d,w} - \sum_{k:(j,k)\in links} P_{j,k}^{s,d,w} = \begin{cases} -\lambda_{s,d}^{w} & if\ j = s \\ \lambda_{s,d}^{w} & if\ j = d \\ 0 & otherwise \end{cases} \quad \forall 1 \le j \le N, \qquad \forall 1 \le w \le W, \ \forall s,d \in \phi$$

3. Demand satisfaction

$$\sum_{w=1}^{W} \lambda_{s,d}^{w} = \phi_{s,d} \qquad 1 \le s,d \le N$$

4. The resulting equations, giving us main variables.

$$p_{(i,j)} = \sum_{w=1}^{W} \sum_{1 \le s,d \le N} P_{i,j}^{s,d,w}, \qquad \forall (i,j) \in E$$

Solving this ILP yields a baseline routing and wavelength solution for a given network.

5.1.2 Plain Network, with Wavelength Conversion

When we consider a network where a connection can consist out a set of different links occupying a different number of wavelengths, we have to change the flow conservation constraint and the demand satisfaction which are now combined into one constraint.

$$Minimize \sum_{(i,j)\in E} p_{(i,j)}$$

1. Capacity constraint

$$\sum_{1 \le s,d \le N} P_{i,j}^{s,d,w} \le 1 \quad \forall (i,j) \in links, \qquad 1 \le w \le W$$

2. Flow conservation and demand satisfaction.

$$\sum_{w=1}^{W} \sum_{i:(i,j)\epsilon\ links} P_{i,j}^{s,d,w} - \sum_{w=1}^{W} \sum_{k:(j,k)\epsilon\ links} P_{j,k}^{s,d,w} = \begin{cases} -\phi_{s,d} & if\ j = s \\ \phi_{s,d} & if\ j = d \\ 0\ otherwise \end{cases} \quad \forall\, 1 \le j \le N,\ \forall\, s,d \in \phi$$

3. Resulting equations

$$p_{(i,j)} = \sum_{w=1}^{W} \sum_{1 \le s,d \le N} P_{i,j}^{s,d,w}, \qquad \forall (i,j) \in E$$

5.1.3 Network with Dedicated Protection, without Wavelength Conversion

For this type of protection, we use the base ILP's from above and change the flow conservation so that the demand capacity is doubled. If we then ensure that when a link fails the number of affected paths is less than the actual demand, then every primary path has a dedicated protection path which is link-disjoint.

$$Minimize \sum_{(i,j)\in E} p_{(i,j)}$$

1. The capacity constraint.

$$\sum_{1 \le s,d \le N} P_{i,j}^{s,d,w} \le 1 \quad \forall (i,j) \in E, \qquad 1 \le w \le W$$

2. Flow conservation

$$\sum_{i:(i,j)\epsilon\ links} P_{i,j}^{s,d,w} - \sum_{k:(j,k)\epsilon\ links} P_{j,k}^{s,d,w} = 0, \qquad \forall\, 1 \le j \ne s,d \le N, \qquad \forall\, 1 \le w \le W, \forall\, s,d \in \phi$$

3. Demand satisfaction

$$\sum_{w=1}^{W} \sum_{(s,i)\epsilon\ E} P_{s,i}^{s,d,w} = 2 * \phi_{s,d} \qquad 1 \le s,d \le N$$

$$\sum_{w=1}^{W} \sum_{(i,d)\epsilon\ E} P_{i,d}^{s,d,w} = 2 * \phi_{s,d} \qquad 1 \le s,d \le N$$

4. When a link fails, the number of affected paths which fail because of that link failure cannot exceed the actual demand.

$$\sum_{w=1}^{W} P_{i,j}^{s,d} \leq \phi_{s,d}, \qquad \forall (i,j) \in E, \forall (s,d) \in \phi$$

5. Resulting equations.

$$p_{(i,j)} = \sum_{w=1}^{W} \sum_{1 \leq s,d \leq N} P_{i,j}^{s,d,w}, \qquad \forall (i,j) \in E$$

5.1.4 Network with Dedicated Protection with Wavelength Conversion

$$Minimize \sum_{(i,j) \in E} p_{(i,j)}$$

1. Capacity constraint

$$\sum_{1 \leq s,d \leq N} P_{i,j}^{s,d,w} \leq 1, \qquad \forall (i,j) \in links, \qquad 1 \leq w \leq W$$

2. Flow conservation.

$$\sum_{w=1}^{W} \sum_{i:(i,j) \in links} P_{i,j}^{s,d,w} - \sum_{w=1}^{W} \sum_{k:(j,k) \in links} P_{j,k}^{s,d,w} = \begin{cases} -2 * \phi_{s,d} & if\ j = s \\ 2 * \phi_{s,d} & if\ j = d \\ 0 & otherwise \end{cases} \quad \forall\, 1 \leq j \neq s,d \leq N, \qquad \forall (s,d) \in \phi$$

3. When a link fails, the number of affected paths which fail because of that link cannot exceed the actual demand.

$$\sum_{w=1}^{W} P_{i,j}^{s,d} \leq \phi_{s,d}, \qquad \forall (i,j) \in E, \qquad \forall (s,d) \in \phi$$

4. Resulting equations.

$$p_{(i,j)} = \sum_{w=1}^{W} \sum_{1 \le s,d \le N} P_{i,j}^{s,d,w}, \qquad \forall (i,j) \in E$$

5.1.5 Network with Shared Protection without Wavelength Conversion

$$Minimize \sum_{i,j \in E} p_{i,j} + s_{i,j}$$

1. Demand satisfaction

$$\sum_{w=1}^{W} \lambda_{s,d}^{w} = \phi_{s,d} \qquad 1 \le s,d \le N$$

2. Flow conservation on the primary paths

$$\sum_{i:(i,j) \in links} P_{i,j}^{s,d,w} - \sum_{k:(j,k) \in links} P_{j,k}^{s,d,w} = \begin{cases} -\lambda_{s,d}^{w} & if\ j = s \\ \lambda_{s,d}^{w} & if\ j = d \\ 0 & otherwise \end{cases} \quad \forall 1 \le j \ne s,d \le N, \qquad \forall 1 \le w \le W, \forall s,d \in \phi$$

3. A wavelengths on a link (i,j) can only be used by either a primary path or backup path (Capacity constraint)

$$r_{i,j}^{w} + \sum_{1 \le s,d \le n} P_{i,j}^{s,d,w} \le 1, \qquad \forall (i,j) links, \qquad 1 \le w \le W$$

4. When a link fails in the network there must be a back up path which starts at the source node. (Demand satisfaction for the backup paths).

$$\sum_{w=1}^{W} P_{i,j}^{s,d,w} = \sum_{w=1}^{W} \sum_{e:(s,e) \in links} R_{(s,e),(i,j)}^{s,d,w} \quad \forall (s,d) \in \phi, \forall (i,j) \in E$$

$$\sum_{w=1}^{W} P_{i,j}^{s,d,w} = \sum_{w=1}^{W} \sum_{e:(e,d) \in links} R_{(e,d),(i,j)}^{s,d,w} \quad \forall (s,d) \in \phi, \forall (i,j) \in E$$

5. Flow conservation for the backup paths.

$$\sum_{i:(i,j) \in links} R_{(i,j),(k,l)}^{s,d,w} - \sum_{y:(j,y) \in links} R_{(j,y),(k,l)}^{s,d,w} = 0, \qquad \forall 1 \le j \ne s,d \le N, \qquad \forall (k,l) \in E, \qquad 1 \le w \le W, \forall s,d \in \phi$$

6. A link (i,j) can only protect one (s,d) pair on link (k,l)

$$\sum_{1\le s,d\le N} R^{s,d,w}_{(i,j),(k,l)} \le 1, \qquad \forall (i,j),(k,l) \in E, 1 \le w \le W$$

7. A link (i,j) cannot protect itself.

$$R^{s,d,w}_{(i,j),(i,j)} = 0, \qquad \forall (s,d) \in \phi, \forall\ (i,j) \in E, 1 \le w \le W$$

8. Constraints on $r^w_{i,j}$.

$$r^w_{i,j} \le \sum_{1\le s,d\le N} \sum_{(k,l)\in links} R^{s,d,w}_{(i,j),(k,l)}, \qquad \forall (i,j) \epsilon\ E, 1 \le w \le W$$

$$N.N.r^w_{i,j} \ge \sum_{1\le s,d\le N} \sum_{(k,l)\in links} R^{s,d,w}_{(i,j),(k,l)}, \qquad \forall (i,j) \epsilon\ E, 1 \le w \le W$$

9. Resulting equations

$$p_{i,j} = \sum_{1\le s,d\le N} \sum_{w=1}^{W} P^{s,d,w}_{i,j}, \qquad \forall\ (i,j) \in \phi$$

$$s_{i,j} = \sum_{w=1}^{W} r^w_{i,j}, \qquad \forall\ (i,j) \in \phi$$

5.1.6 Network with Shared Protection with Wavelength Conversion

$$Minimize \sum_{i,j\in links} p_{i,j} + s_{i,j}$$

1. Flow conservation on the primary paths and demand satisfaction.

$$\sum_{w=1}^{W} \sum_{i:(i,j)\in links} P^{s,d,w}_{i,j} - \sum_{w=1}^{W} \sum_{k:(j,k)\in links} P^{s,d,w}_{j,k} = \begin{cases} -\phi_{s,d} & if\ j = s \\ \phi_{s,d} & if\ j = d \\ 0 & otherwise \end{cases} \forall\ \mathbf{1} \le j \ne s,d \le N, \qquad \forall\ s,d\ \epsilon\ \phi$$

2. A wavelengths on a link (i,j) can only be used by either a primary path or backup path (Capacity constraint)

$$r_{i,j}^{w} + \sum_{1 \le s,d \le n} P_{i,j}^{s,d,w} \le 1, \qquad \forall (i,j) \in E, \qquad 1 \le w \le W$$

3. When a link fails in the network there must be a back up path which starts at the source node. (Demand satisfaction for the backup paths).

$$\sum_{w=1}^{W} P_{i,j}^{s,d,w} = \sum_{w=1}^{W} \sum_{e:(s,e) \in links} R_{(s,e),(i,j)}^{s,d,w} \quad \forall (s,d) \in \phi, \forall (i,j) \in E$$

$$\sum_{w=1}^{W} P_{i,j}^{s,d,w} = \sum_{w=1}^{W} \sum_{e:(e,d) \in links} R_{(e,d),(i,j)}^{s,d,w} \quad \forall (s,d) \in \phi, \forall (i,j) \in E$$

4. Flow conservation for the backup paths.

$$\sum_{w=1}^{W} \sum_{i:(i,j) \in links} R_{(i,j),(k,l)}^{s,d,w} - \sum_{w=1}^{W} \sum_{y:(j,y) \in links} R_{(j,y),(k,l)}^{s,d,w} = 0, \qquad \forall 1 \le j \ne s,d \le N, \qquad \forall (k,l) \in E, \forall s,d \in \phi$$

5. A link (i,j) can only protect one (s,d) pair on link (k,l)

$$\sum_{1 \le s,d \le N} R_{(i,j),(k,l)}^{s,d,w} \le 1, \qquad \forall (i,j),(k,l) \in \phi, \qquad 1 \le w \le W$$

6. A link (i,j) cannot protect itself.

$$R_{(i,j),(i,j)}^{s,d,w} = 0, \qquad \forall (s,d) \in \phi, \qquad \forall (i,j) \in E, \qquad 1 \le w \le W$$

7. Constraints on $r_{i,j}^{w}$.

$$r_{i,j}^{w} \le \sum_{1 \le s,d \le N} \sum_{(k,l) \in links} R_{(i,j),(k,l)}^{s,d,w}, \qquad \forall (i,j) \in E, \qquad 1 \le w \le W$$

$$N.(N-1).L.r_{i,j}^{w} \ge \sum_{1 \le s,d \le N} \sum_{(k,l) \in links} R_{(i,j),(k,l)}^{s,d,w}, \qquad \forall (i,j) \in E, \qquad 1 \le w \le W$$

8. Resulting equations

$$p_{i,j} = \sum_{1 \le s,d \le N} \sum_{w=1}^{W} P_{i,j}^{s,d,w}, \qquad \forall (i,j) \in \phi$$

$$s_{i,j} = \sum_{w=1}^{W} r_{i,j}^{w}, \qquad \forall (i,j) \in \phi$$

5.2 Network with Relocation Possibility

5.2.1 Network with Shared Relocation Possibility, without Wavelength Conversion

$$Minimize \sum_{(i,j) \in links} p_{i,j} + s_{i,j}$$

1. Demand satisfaction

$$\sum_{w=1}^{W} \lambda_{s,d}^{w} = \phi_{s,d} \qquad 1 \le s,d \le N$$

2. Flow conservation on the primary paths

$$\sum_{i:(i,j) \in links} P_{i,j}^{s,d,w} - \sum_{k:(j,k) \in links} P_{j,k}^{s,d,w} = \begin{cases} -\lambda_{s,d}^{w} & if\ j = s \\ \lambda_{s,d}^{w} & if\ j = d \\ 0 & otherwise \end{cases}, \qquad 1 \le w \le W, \qquad \forall s,d \in \phi, \qquad \forall j \in V$$

3. A wavelengths on a link (i,j) can only be used by either a primary path or backup path (Capacity constraint)

$$r_{i,j}^{w} + \sum_{1 \le s,d \le n} P_{i,j}^{s,d,w} \le 1, \qquad \forall (i,j) \in E, \qquad 1 \le w \le W$$

4. Demand constraints for the back up paths.

$$\sum_{w=1}^{W} P_{i,j}^{s,d,w} = \sum_{\delta \in \Delta} \sum_{w=1}^{W} \sum_{e:(s,e) \in links} R_{(s,e),(i,j)}^{s,d,w,\delta}, \qquad \forall (i,j) \in E, \qquad \forall (s,d) \in \phi$$

$$\sum_{w=1}^{W} P_{i,j}^{s,d,w} = \sum_{\delta \in \Delta} \sum_{w=1}^{W} \sum_{e:(e:\delta)\in links} R_{(e:\delta),(i,j)}^{s,d,w,\delta}, \qquad \forall (i,j) \in E, \qquad \forall (s,d) \in \phi$$

5. Flow constraints for the back up paths.

$$\sum_{w=1}^{W} \sum_{i:(i,j)\in links} R_{(i,j),(k,l)}^{s,d,w,\delta} - \sum_{w=1}^{W} \sum_{y:(j,y)\in links} R_{(j,y),(k,l)}^{s,d,w,\delta} = 0, \qquad \forall j \neq s,d \in V, \forall (k,l) \in E, \forall \delta \in \Delta \quad , j \neq \delta$$

6. A wavelength on link (i,j) cannot protect two different (s,d) pairs on the same link (k,l).

$$\sum_{s,d} R_{(i,j),(k,l)}^{s,d,w,\delta} \leq 1, \qquad \forall (i,j),(k,l) \in E, 1 \leq w \leq, W, \qquad \forall \delta \epsilon \Delta$$

7. A link cannot protect itself.

$$R_{(i,j),(i,j)}^{s,d,w,\delta} = 0, \qquad \forall (i,j) \in E, \forall s,d,w,\delta$$

8. Constraints on $r_{i,j}^{w}$.

$$r_{i,j}^{w} \leq \sum_{1\leq s,d\leq N} \sum_{(k,l)\in links} \sum_{\delta\in\Delta} R_{(i,j),(k,l)}^{s,d,w,\delta}, \qquad \forall (i,j)\epsilon E, \qquad 1 \leq w \leq W$$

$$N.(N-1).N.|\Delta|.r_{i,j}^{w} \geq \sum_{1\leq s,d\leq N} \sum_{(k,l)\in links} \sum_{\delta\in\Delta} R_{(i,j),(k,l)}^{s,d,w,\delta}, \qquad \forall (i,j)\epsilon E, \qquad 1 \leq w \leq W$$

9. Resulting equations

$$f_{i,j} = \sum_{1\leq s,d\leq N} \sum_{w=1}^{W} P_{i,j}^{s,d,w}, \qquad \forall (i,j) \in \phi$$

$$s_{i,j} = \sum_{w=1}^{W} r_{i,j}^{w}, \qquad \forall (i,j) \in \phi$$

5.2.2 Network with Shared Relocation Possibility and Wavelength Conversion

$$Minimize \sum_{(i,j)\in links} p_{i,j} + s_{i,j}$$

1. Flow conservation on the primary paths

$$\sum_{w=1}^{W}\sum_{i:(i,j)\in links} P_{i,j}^{s,d,w} - \sum_{w=1}^{W}\sum_{k:(j,k)\in links} P_{j,k}^{s,d,w} = \begin{cases} -\phi_{s,d} & if\ j = s \\ \phi_{s,d} & if\ j = d \\ 0 & otherwise \end{cases}, \qquad \forall\, s,d \in \phi, \qquad \forall\, j \in V$$

2. A wavelengths on a link (i,j) can only be used by either a primary path or backup path (Capacity constraint)

$$r_{i,j}^{w} + \sum_{1\le s,d\le n} P_{i,j}^{s,d,w} \le 1, \qquad \forall (i,j) \in E, \qquad 1 \le w \le W$$

3. Demand constraints for the back up paths.

$$\sum_{w=1}^{W} P_{i,j}^{s,d,w} = \sum_{\delta\in\Delta}\sum_{w=1}^{W}\sum_{e:(s,e)\in links} R_{(s,e),(i,j)}^{s,d,w,\delta}, \qquad \forall (i,j) \in E, \qquad \forall (s,d) \in \phi$$

$$\sum_{w=1}^{W} P_{i,j}^{s,d,w} = \sum_{\delta\in\Delta}\sum_{w=1}^{W}\sum_{e:(e:\delta)\in links} R_{(e:\delta),(i,j)}^{s,d,w,\delta}, \qquad \forall (i,j) \in E, \qquad \forall (s,d) \in \phi$$

4. Flow constraints for the back up paths.

$$\sum_{w=1}^{W}\sum_{i:(i,j)\in links} R_{(i,j),(k,l)}^{s,d,w,\delta} - \sum_{w=1}^{W}\sum_{y:(j,y)\in links} R_{(j,y),(k,l)}^{s,d,w,\delta} = 0, \qquad \forall\, j \ne s,d \in V, \qquad \forall (k,l) \in E, \qquad \forall \delta \in \Delta\ , j \ne \delta$$

5. A wavelength on link (i,j) cannot protect two different (s,d) pairs on the same link (k,l).

$$\sum_{s,d} R_{(i,j),(k,l)}^{s,d,w,\delta} \le 1, \qquad \forall (i,j),(k,l) \in links, 1 \le w \le, W, \qquad \forall\, \delta\epsilon\Delta$$

6. A link cannot protect itself.

$$R_{(i,j),(i,j)}^{s,d,w,\delta} = 0, \qquad \forall (i,j)\ \in links, \forall s,d,w,\delta$$

7. Constraints on $r_{i,j}^{w}$.

$$r_{i,j}^{w} \le \sum_{1\le s,d\le N}\sum_{(k,l)\in links}\sum_{\delta\in\Delta} R_{(i,j),(k,l)}^{s,d,w,\delta}$$

$$N.(N-1).L.|\Delta|.r_{i,j}^{w} \ge \sum_{1\le s,d\le N}\sum_{(k,l)\in links}\sum_{\delta\in\Delta} R_{(i,j),(k,l)}^{s,d,w,\delta}$$

8. Resulting equations

$$f_{i,j} = \sum_{1 \le s,d \le N} \sum_{w=1}^{W} P_{i,j}^{s,d,w}, \qquad \forall (i,j) \in \phi$$

$$s_{i,j} = \sum_{w=1}^{W} r_{i,j}^{w}, \qquad \forall (i,j) \in \phi$$

5.3 Discussion of the ILP Models

An important factor when formulating the ILP's is the number of variables which are necessary. The next table summarizes these data.

ILP Model	*Number of variables*
Plain network	$O(L * \lvert\phi\rvert * W)$
Dedicated protection	$O(L * \lvert\phi\rvert * W)$
Shared protection	$O(L * \lvert\phi\rvert * W + L^2 * \lvert\phi\rvert * W)$
Shared protection with relocation	$O(L * \lvert\phi\rvert * W + L^2 * \lvert\phi\rvert * W * \lvert\Delta\rvert)$

As it can be seen in the table the number of variables grows with the number of links the input networks contains, the number of wavelengths each link supports, the size of the demand matrix and - for the relocation case - also with the available resource sites that are candidate for receiving relocated jobs.

We implemented the above ILP's using Java and the ILOG Cplex library on an AMD Athlon 64 X2 Dual Core processor 2.11 GHz system with 3.25 GB of RAM memory. In order to reduce the memory use of the ILP program we have excluded some special variables based on the following assumptions:

1. When the demand matrix $\phi_{s,d}$ contains a 0 zero value, all the variables which are associated with that (s,d) pair are definitely zero.

2. The following variables are excluded from the model as portrayed in Figure 45 :

   a. The $R_{(i,s)}^{(s,d),w}$ is definitely 0 because if were 1, the following link in the path could be taken as the first link of that path.

b. The $R_{(d,i)}^{(s,d),w}$ is definitely 0 because if were 1 the path already has reached the destination and tries to reach it a second time.

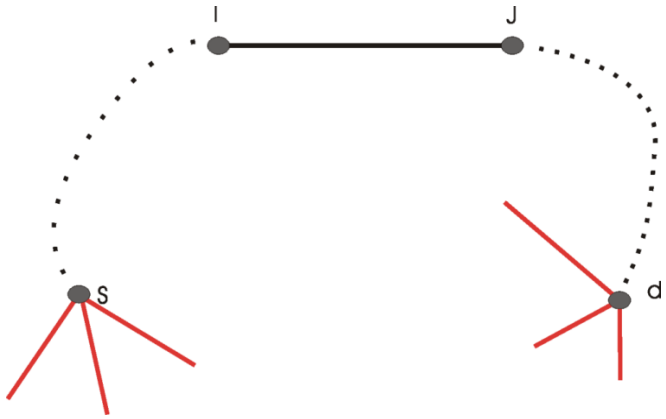


Figure 45: Special cases which make it possible to exclude some variables from the ILP's.

5.4 Results

Despite of these memory enhancements, the ILP's become infeasible for the computer to solve for larger networks with a large demand matrix. In order to prototype the ILP's which were formulated we have run several tests on a relative smaller network. The results which we derived will give us a good idea of the effectiveness of the resiliency schemes and are representative, because the proposed algorithms are expected to scale with the network dimensions

The network considered is derived from a part of the Geant2 core network where every link supports the same amount of wavelengths and is shown in Figure 46.

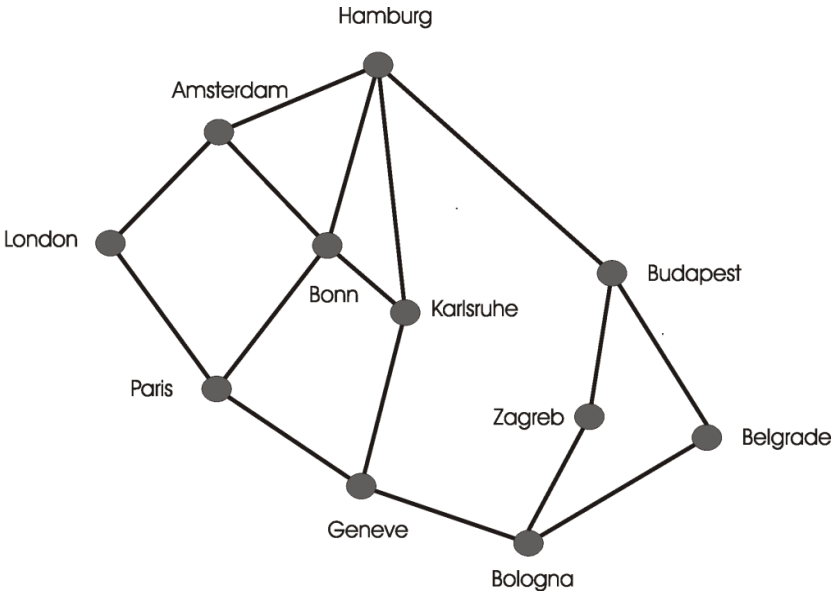


Figure 46: The considered network topology.

The ILPs which are compared to each other are the Plain Network, Dedicated Network, Shared Network and the Shared Relocation Network with wavelength conversion.

5.4.1 Specific Test Case

In order to visualise the ILP where relocation is possible we have performed a first test case using a demand matrix consisting out of the following connection demands:

1. Belgrade → Bonn
2. Amsterdam → Bologna
3. Amsterdam → Belgrade
4. Hamburg → Budapest

While in the dedicated- and shared protection scheme every connection is protected by another connection that is terminated at same resource site, the relocation algorithm enables the protecting of this connection by reserving wavelengths Figure 47 where the blue lines represent primary paths and a dotted red line represents a shared wavelength. We observe that every connection can be protected by relocating the respective job to a nearer resource site. In the connection between Belgrade and Bonn for example, every link on the primary path can be protected by a single wavelength going from Belgrade to Bologna.

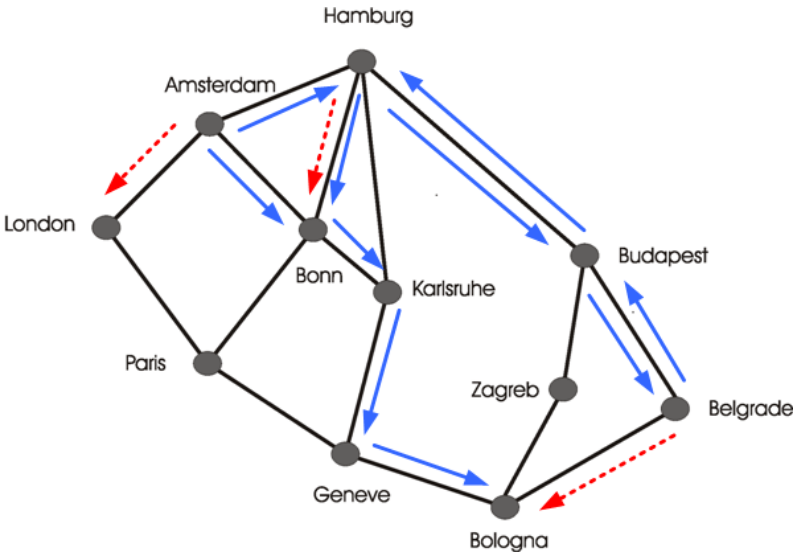


Figure 47: Wavelength assignment for the network with the considered demand matrix.

When we consider the connections from Amsterdam to Belgrade and Bologna, we can remark that they both share the same protection path and they also share the same back-up resource, thus making two back up paths obsolete. Both primary connections are link disjoint which makes the possibility of two network failures on the two distinct paths very small, which in turn lessens the probability both connections making use of their back up path.

5.4.2 Test Case

We have run the Plain Network -, the Dedicated Network -, the Shared Network - and the Shared Relocation network ILP with the same demand matrix ten times, each time randomizing the demand matrix. The four possible sites to send to are London, Bonn, Bologna and Budapest which are well connected into the network.

The results are shown in Figure 48 As we observe, the Shared Relocation scheme uses the least amount of bandwidth in terms of reserved wavelengths. Compared to the traditional shared protection scheme there is a network usage decrease of about 8 percent and in comparison to dedicated protection it is double as good. The relative small difference between the shared relocation - and the shared network protection scheme can be contributed to the rather small network size, the limited amount of connection demands and the reduced number of wavelengths used thanks due to the particularly exponential character of the ILP's.

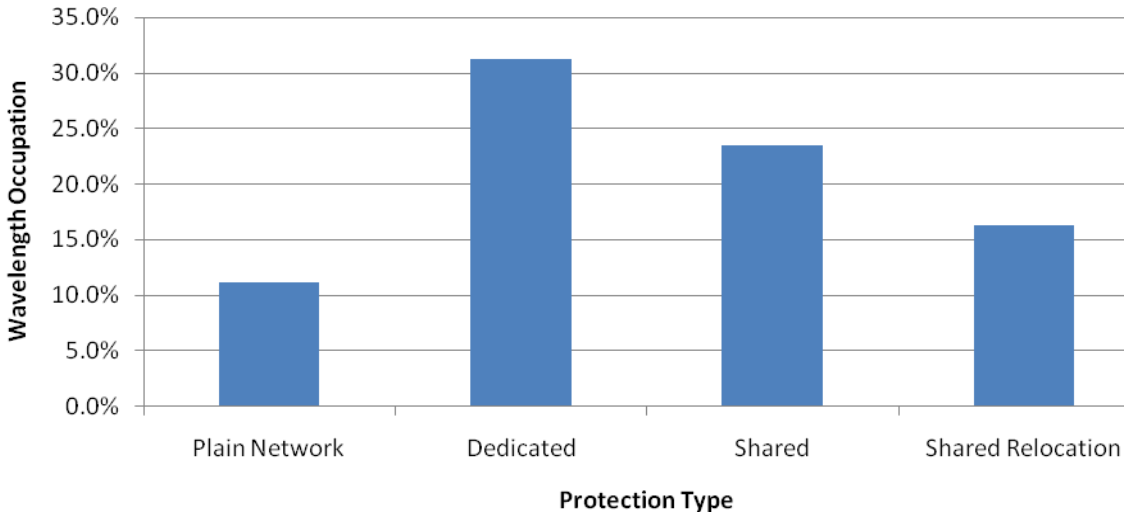


Figure 48: The number of wavelengths needed, to the total number of wavelengths the network comprises, for every resiliency algorithm.

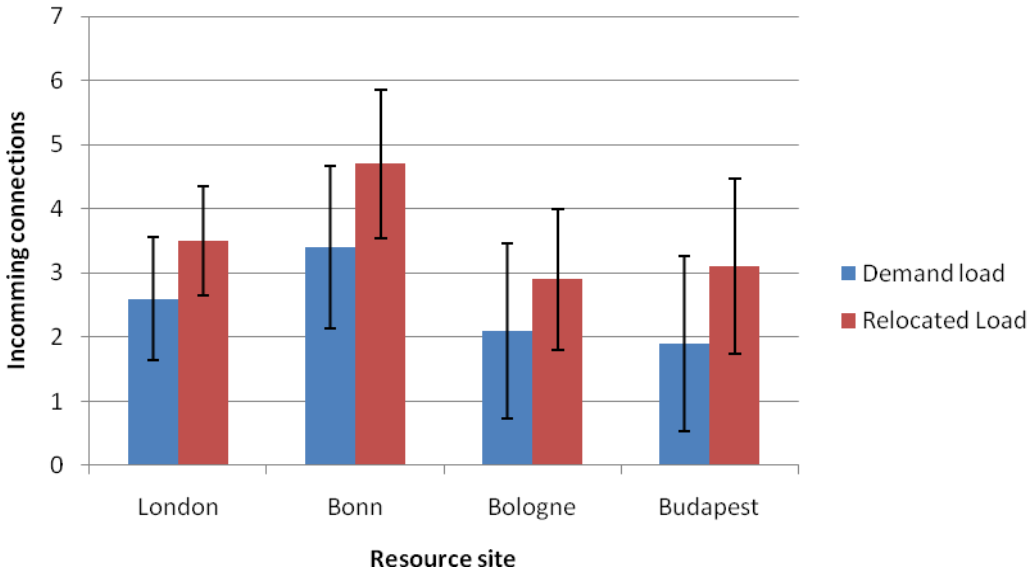


Figure 49: The maximum spare capacity a resource needs in case of a single link failure. The blue bars represent the number of connections a resource site receives in a fault-free scenario while the red bars represent the maximum number of connections a resource site receives in case of single link failure.

A reduction in network resources brings a change in the Grid resource site usage. By relocating a part of the jobs to other sites the resource load is redistributed.

Figure 49 depicts the spare capacity a resource needs, in according to the demand matrix in a fault-free circumstances and the maximum resource capacity in case of single link failure. That is the fraction of connections this link was supporting as part of a primary path which are conveyed to the resource in case of this particular link failure. We see that if every resource has reserves, about 42 % of the demand as spare capacity, every single link failure can be caught by relocation to another resource site. This seems much, but is in fact attributed by the actuality that we only performed a study on a small network with a relative small demand matrix. We expect this percentage to go down when larger calculation are possible. Therefore, the entity responsible for the resiliency in the network could check whether the back-up resource has enough spare capacity and based on that information could choose to either protect using the relocation strategy or to use another classical network protection scheme.

5.5 Conclusions

In this section we have conducted a study on the impact of Grid resource resiliency to the network resources. What is the impact of the relocation of a job to another, closer, resource site on the wavelength assignments compared to the other classical network resiliency strategies? We have constructed and implemented four different resiliency approaches namely plain network, dedicated network, shared network and shared relocation network. Along our solution line, the common objective was to minimize the wavelengths used by the primary and the secondary paths. Although all presented ILPs were exact and optimal,, they were very memory consuming and therefore tests with larger networks were not possible. We conducted a study on a smaller network, with the outcome that relocation is favourable compared to the other resilience schemes if every resource has a spare capacity of about 20 percent.

6 Conclusion

In this deliverable various problems associated with providing resilience in the context of Grid networks are addressed. In Section 2, offline RWA algorithms are proposed to support resilient network design. Two LP-relaxation formulations are proposed for the problem of finding link-disjoint primary/backup lightpath pairs, assuming 1+1 protection, assuming that connection requests known in advance. The proposed LP formulations differ in the set of candidate paths that they use in order to choose the primary and the backup lightpaths. Although small scale experiments are performed, the proposed algorithms are expected to scale and have acceptable execution time performance for large networks and high loads. Also an analytical solution is presented for the physical impairment-aware resilient network design based on an integer linear programming formulation.

Resilient traffic engineering is investigated in Section 3 by introducing online RWA algorithms. The influence of double failures to single-backup path protected connection provisioning is quantified. Evaluation results, obtained for the Phosphorus European test-bed topology, indicate that the fraction of affected lightpaths at the event of a double failure is lower than 20%, consistently for both routing schemes used and independent of the load values tested. This finding is to be interpreted by the Grid network infrastructure designer or service provider in combination with the expected cost and revenue, if double-failure recovery mechanisms are considered as an option. Also the effect of using impairment-constrained routing for assigning routes to working and protection paths is examined. More precisely, the approach is applied in the shared backup protection path protection scheme. As manifested by simulation results, a cross-layer approach in backup path computation manages to reduce total blocking, as long as the network is operated at low to medium loads. As the load increases further, blocking increases due to unavailability of free wavelengths. In any case, the impairment-aware approach did not yield worse blocking ratios than the min-hop backup routing approach throughout our simulation experiments.

Several studies on differentiation resilience are presented in Section 3. The problem of efficiently provisioning lightpaths with disparate protection requirements in a dynamically provisioned WDM network is addressed. In this context, various RWA schemes are investigated with the aim to enhance the backup resource utilization and improve the network performance. The Last Fit (LF) wavelength assignment scheme is applied on the backup lightpaths in parallel with the First Fit (FF) assignment method, applied on primary lightpaths. These schemes demonstrate considerable improvement compared to the commonly

used Random Pick (RP) and FF assignment schemes. Results show that the improvement on blocking probability occurs due to efficient capacity reuse offered by the LF scheme compared to other schemes used. Also the incoming traffic is differentiated to classes of service according to survivability requirements and the pre-emption of low priority traffic by higher priority demands in the event of a link failure is proposed. This technique enables the reuse of already assigned low-priority wavelengths by backup paths, reinforcing thus the reuse of available network resources. Survivable traffic grooming is investigated under a differentiated resilience scheme. Two schemes of provisioning survivable paths to connection requests with different resilience classes (dedicated protection, shared protection, and restoration) are proposed, Differentiated Resilience at Lightpath (DRAL) level and Differentiated Resilience at Connection (DRAC) level. These schemes examine different ways of provisioning backup paths. Simulation results show that (i) traffic with dedicated protection requirements obtains a better blocking probability under DRAC when the number of grooming ports is large, (ii) when the number of grooming ports is small DRAL results in a lower blocking probability due to the high sensitivity of DRAC to the change in the number of grooming ports. (iii) the blocking probabilities experienced by traffic with shared protection requirements under DRAL and DRAC are very close under a larger number of grooming ports, (iv) with a small number of grooming ports DRAL outperforms DRAC, (v) the blocking probability experienced by connections with restoration requirements is lower under DRAC when the number of grooming ports is large, with a small number of grooming ports the opposite trend is observed, (vi) the rerouting probability suffered by connections with restoration requirements to allow higher resilience classes to establish their connections are found to be higher under DRAL. A differentiated resilience, that allows anycast flows to survive any link or server failure, is investigated in the context of an MPLS architecture where the anycasting principle is implemented as the communication paradigm. Rerouting of the lower traffic class is implemented to reduce the blocking probability experienced by higher traffic classes. Results show that (i) rerouting has significantly reduced the blocking probability of higher classes, (ii) increasing the number of servers decreases the blocking probability; however, this decrease tends to get smaller as the number of server increase.

In Section 4, several adaptive heuristics, based on job checkpointing, job replication and the combination of these techniques, are applied to address Grid-resource fault-tolerance. The heuristics are evaluated under varying system load and availability conditions. Results show that the run-time overhead due to periodic checkpointing can significantly be reduced when the checkpointing frequency is dynamically adapted as a function of resource stability and remaining job execution time. Furthermore, adaptive replication-based solutions can reduce replication as a function of various system parameters, leading to even lower cost fault-tolerance in systems with low and variable load. The advantages of both techniques are combined in the hybrid approach that can best be applied when the distributed system properties are not known in advance.

Resiliency techniques are also applied to the Data Consolidation (DC) operation in Section 4. DC arises when a task requests datasets that are stored in more than one storage sites. Resilience features are added to previously proposed DC operation and novel DC techniques are presented to cope with the increase in the network load. Simulations are carried out to evaluate the performance of these techniques. It is shown that the efficiency achieved by DC schemes in terms of network congestion brought by the applied resiliency techniques,manage to considerably improve the measured task success ratio. For this reason, the DC schemes that use tree based techniques for dataset routing achieve higher task success ratio than the rest of the DC schemes tested.

A study of the impact of Grid-resource resiliency to the network resources is presented in Section 5. Four distinct resiliency approaches are proposed and evaluated, namely plain network, dedicated network, shared network and shared relocation network. Four different ILP programs are formulated whose main objective is to minimize the wavelengths used by primary and backup paths. The outcome of the study conducted shows that relocation is favourable compared to the other resilience schemes if every resource has a spare capacity of about 20 percent.

References

[A.Rahman06] M. S. A.Rahman, S. Shaari, “OXADM restoration scheme: Approach to optical ring network protection”, IEEE International. Conference on Networks, 2006.

[Aneja07] Y. Aneja, A. Jaekel, S. Bandyopadhyay, “Some studies on path protection in WDM networks”, Photonic Network Communication, vol. 14, pp.165-176, 2007.

[Autenrieth02] A. Autenrieth, A. Kirstädter: "Engineering End-to-end IP Resilience using Resilience- Differentiated QoS", IEEE Communications Magazine, Vol. 40, No. 1, January 2002.

[Avresky05] S. Frechette, D. R. Avresky, “Method for Task Migration in Grid Environments”, Intl. Symposium on Network Computing and Applications, 2005.

[Bagula04] B.A. Bagula, M. Botha, A.E. Krzesinski, “Online traffic engineering: the Least Interference Optimization Algorithm,” In Proceedings of IEEE International Conference on Communications ICC 2004, Paris, 2004.

[Batchelor00] P. Batchelor, et al. “Study on the Implementation of Optical Transparent Transport Networks in the European Environment—Results of the Research Project COST 239”, J. Phot. Network. Comm., Vol. 2, pp.15-32, 2000.

[Bhandari99] R. Bhandari, “Survivable Networks: Algorithms for Diverse Routing”, Kluer Academic Publishers, 1999.

[Bouaba02] W. Szeto, R. Boutaba, Y. Iraqi, “Dynamic online routing algorithm for MPLS traffic engineering,” Lectures Notes in Computer Science, LNCS 2345, 2002, pp. 936-946.

[Bouillet02] E. Bouillet, J. Labourdette, G. Ellinas, R. Ramamurthy, and S. Chaudhuri, “Stochastic approaches to compute shared mesh restored lightpaths in optical network architectures,” in Proc. IEEE INFOCOM, June 2002, pp. 801–807.

[Bouillet07] E. Bouillet, G. Ellinas, J.F. Labourdette and R. Ramamurthy, “Path Routing in Mesh Optical Networks”, John Wiley & Sons Ltd., 2007.

[Caenegem98] B. Van Caenegem et al., "Dimensioning of Survivable WDM Networks", IEEE Journal on Selected Areas in Communications, Vol. 16, No. 7, September 1998.

[Cholda07] P. Cholda et al., “A Survey of Resilience Differentiation Frameworks in Communication networks” *IEEE Communications Surveys and Tutorials*. Vol. 9, No. 4, IEEE Com Soc, Oct, 2007.

[Christodou08] K. Christodoulopoulos, K. Manousakis, E. Varvarigos, “Comparison of Routing and Wavelength Assignment Algorithms in WDM Networks”, presented in IEEE GLOBECOM 2008.

[Chtepen09] Maria Chtepen, Filip H.A. Claeys, Bart Dhoedt, Filip De Turck, Piet Demeester, Peter A. Vanrolleghem, "Adaptive Task Checkpointing and Replication: Toward Efficient Fault-Tolerant Grids," *IEEE Transactions on Parallel and Distributed Systems*, vol. 20, no. 2, pp. 180-190, Feb. 2009.

[Crawley98] E. Crawley, R. Nair, B. Jajagopalan, H. Sandick. “A Framework for QoS-based Routing in the Internet”, *RFC, http://www.ietf.org/rfc/rfc2386.txt*, August 1998.

[D5.3] Phosphorus WP5, “Grid Job Routing Algorithms”, Deliverable D.5.3, June 2007.

[D5.7] Phosphorus WP5, “Grid Network Design,” Deliverable D.5.7, September 2008.

[D3.1] Phosphorus WP3, “Use-cases, Requirements and Design of Changes and Extensions of the Applications and Middleware,” Deliverable D.3.1, December 2006.

[DeLeenheer05] M. DeLeenheer et al., “Anycast routing in optical burst switched grid networks ECOC 2005. 25-29 Sept. 2005, pp 699- 700 vol.3.

[DeLeenheer06] M. DeLeenheer et al., “Anycast algorithms supporting optical burst switched grid networks,” in *Proc. International Conference on Networking and Services (ICNS 2006)*, Sillicon Valley, CA USA, July 2006.

[Develder08] C. Develder et al., “Scalable impairment-aware anycast routing in multi-domain optical Grid networks”, in Proceedings of 10th Anniversary International Conference on Transparent Optical Networks (ICTON), pp. 150-153, 2008.

[Doi04] S. Doi, S. Ata, K. Kitamura, M. Murata, “IPv6 anycast for simple and effective service-oriented communications”, IEEE Comm. Mag. 5, 2004, pp. 163- 171.

[Doshi99] B. T. Doshi., “Optical Network Design and Restoration,” Bell Labs Tech. J., pp. 58-84,1999.

[Doshi00] B. T. Doshi., “Optical Network Design and Restoration,” Bell Labs Tech. J. 58-84,1999.

[Doucette03] J. Doucette, M. Clouqueur and W. Grover, “On the availability and capacity requirements of shared backup path-protected mesh networks“,Optical Networks Magazine Special Issue on Engineering the Next Generation Optical Interne, pp. 29-44, November/December 2003.

[Doverspike03] G. Li, R. Doverspike, and C. Kalmanek, “Fibre span failure protection in optical networks,” Opt. Networks Mag., vol. 3, pp. 21–31, May/June 2003.

[Du07] Cong Du, Xian-He Sun, Ming Wu: Dynamic Scheduling with Process Migration. CCGRID 2007: 92-99.

[Dunn94] D. Dunn, W. Grover, and M. MacGregor, “Comparison of k-shortest paths and maximum flow routing for network facility restoration,” IEEE J. Select. Areas Commun., vol. 2, pp. 88–89, Jan. 1994.

[Esau66] L. Esau and K. Williams, “On teleprocessing system design”, IBM Systems Journal, vol. 5, pp. 142-147, 1966.

[Frederick04] M. Frederick, P. Datta and A. Somani, "Evaluating dual-failure restorability in mesh-restorable WDM optical networks", in Proceedings of 13th International Conference on Computer Communications and Networks, pp. 309 - 314, Oct. 2004.

[Fuma06] A. Fumagalli and M. Tacca, “Differentiated Reliability (DiR) in Wavelength Division Multiplexing Rings”, IEEE/ACM Transactions on Networking, vol. 14, pp. 159-168, 2006.

[Grover04] W. Grover, “Mesh-based Survivable Transport Networks: Options and Strategies for Optical, MPLS, SONET and ATM Networking,” Prentice Hall PTR, Upper Saddle River, New Jersey, 2004.

[Gumaste05] A. Gumaste and S. Zheng, “Protection and restoration scheme for light-trail WDM ring networks,” in 9th Conference on Optical Network Design and Modelling (ONDM), February 7-9 2005, Milan, Italy.

[Gumaste05] A. Gumaste and S. Zheng, “Protection and restoration scheme for light-trail WDM ring networks,” in 9th Conference on Optical Network Design and Modelling (ONDM), February 7-9 2005, Milan, Italy.

[Han97] S. Han and K. G. Shin, “Efficient spare resource allocation for fast restoration of real-time channels from network component failures,” in Proceeding of IEEE Symposium on Real-Time Systems, pp. 99-108, 1997.

[Hao02] F. Hao, E. Zegura, M. Ammar, “QoS routing for anycast communications: motivation and an architecture for DiffServ networks”, IEEE Comm. Mag., 6, 2002, pp. 48-56.

[Harsha04] Harsha V. Madhyastha and C. Siva Ram Murthy, “Efficient Dynamic Traffic Grooming in Service-differentiated WDM Mesh Networks", Computer Networks, Elsevier Science, Vol. 45, No. 2, pp. 221{235, Jun. 2004.

[Haskin00] D. Haskin and R. Krishnan, “A Method for Setting an Alternative Label Switched Paths to Handle Fast Reroute,” Internet-Draft (draft-haskin-mpls-fast-reroute-05.txt), November 2000.

[He06] Huang He, Wang Jin, Yang Bo “Multi-Class MPLS Resilience Mechanism Supporting Traffic Engineering” Proceedings of the Seventh International

Conference on Parallel and Distributed Computing,Applications and Technologies (PDCAT'06), 2006.

[Ho04] PH Ho, J Tapolcai, T Cinkler, “Segment Shared Protection in Mesh Communications Networks with Bandwidth Guaranteed Tunnels” IEEE/ACM Transactions on Networking (TON), 2004.

[Ho06] T. Ho, D. Abramson; **“**A Unified Data Grid Replication Framework**”,** Second IEEE International Conference on e-Science and Grid Computing”, 2006 Dec. 2006 Page(s):52 – 52.

[Iraschko98] R. Iraschko, M. MacGregor, and W. Grover, “Optimal Capacity Placement for Path Restoration in STM or ATM Mesh-Survivable Networks,” IEEE/ACM Trans. Netw., Vol. 6, pp.326- 336, 1998.

[Iraschko00] R. Iraschko, W. Grover, “A Highly Efficient Path-Restoration Protocol for Management of Optical Network Transport Integrity,” IEEE J. Sel. Areas Comm., Vol. 18, pp. 779-794, 2000.

[Kar00-a] K. Kar, M. Kodialam, T.V. Lakshman, “Minimum interference routing of bandwidth guaranteed tunnels with MPLS traffic engineering applications,” IEEE JSAC 12, 2000, pp. 2566-2579.

[Kar00-b] K. Kar, M. Kodialam, T.V. Lakshman. “Minimum Interference Routing with Application to MPLS Traffic Engineering”, *Proc. IEEE INFOCOM, Vol 2, pp 884–893, Tel-Aviv, Israel*, March 2000.

[Kar03] K. Kar, M. Kodialam, T.V. Lakshman, “Routing restorable bandwidth guaranteed connections using maximum 2-route flows,” IEEE/ACM Transactions on Networking 5, 2003, pp. 772- 781.

[Katz03] D. Katz, K. Kompella, D. Yeung, “Traffic engineering (TE) extensionsto OSPF version 2,” RFC 3630, 2003.

[Kim03] S. Kim and S. Lumetta, "Evaluation of protection reconfiguration for multiple failures in WDM mesh networks", in Proceedings of OSA Optical Fibre Communications Conference,pp. 210-211, 2003.

[Kodialam00-a] M. Kodialam, T. V. Lakshma n, “Dynamic Routing of Bandwidth Guaranteed Tunnels with Restoration,” in Proceeding of IEEE conference on Computer Communication, pp. 902-911, 2000.

[Kodialam00-b] M. Kodialam, T.V. Lakshman, “Minimum interference routing with applications to MPLS traffic engineering,” In Proceedings of INFOCOM 2000, pp. 884-893.

[Kodialam02] M. Kodialam, T.V. Lakshman, “Restorable dynamic quality of service routing,” IEEE Comm. Mag. 6, 2002, pp. 72-80.

[Kodialam03] M. Kodialam, T.V. Lakshman, “Dynamic routing of restorable bandwidth-guaranteed tunnels using aggregated network resource usage information,” IEEE/ACM Transactions on Networking 3, 2003, pp. 399-410.

[Kvalbein04] A. Kvalbein, S. Gjessing, “Analysis and improved performance of RPR protection”. In Proceeding of ICON 2004, vol. 1, pp. 119–124 (2004).

[Li04] T. Li, “ISIS extensions for traffic engineering,” RFC3784, 2004.

[Li05] J. Li, K. L. Yeung , “A Novel Two-Step Approach to Restorable Dynamic QoS Routing”, IEEE/OSA J. Lightw. Tech., Vol. 23, pp. 3663-3670, 2005.

[Limal99] Emmanuel Limal and Kristian E. Stubkjaer “An algorithm for link restoration of wavelength routing optical networks”. IEEE International Conference on Communications 1999 (ICC '99).

[Lin04] C. Lin, J. Lo, S. Kuo, „Load-Balanced Anycast Routing”, In Proceedings of the Parallel and Distributed Systems, Tenth international Conference on (Icpads'04), Washington 2004.

[Liu01] Y. Liu, D. Tipper, and P. Siripongwutikorn, “Approximating optimal space capacity allocation by successive survivable routing,” in Proc. IEEE INFOCOM, 2001, pp. 699–708.

[Makam99] S. Makam, V.Sharma, K.Owens, and C.Huang, “Protection/Restoration of MPLS Networks,” Internet-Draft (draft-makam-mpls-protection-00.txt), October 1999.

[Mar08] G. Markidis and A. Tzanakaki, "Routing and wavelength assignment algorithms in survivable WDM networks under physical layer constraints", in Proceedings of the 3rd International GOPS Workshop, Broadnets (IEEE 2008).

[Mehnert-Spahn08] John Mehnert-Spahn, Michael Schöttner, Christine Morin, “Checkpointing Process Groups in a Grid Environment**”,** Ninth International Conference on Parallel and Distributed Computing, Applications and Technologies, PDCAT 2008, 1-4 Dec. 2008 Page(s):243 – 251.

[Moh00] G. Mohan and C. Murthy, “Lightpath Restoration in WDM Optical Networks,” IEEE Network, vol. 14, pp. 24-32, 2000.

[Ou03] C. Ou, K. Zhu, H. Zang, L. H. Sahasrabuddhe, and B. Mukherjee, "Traffic grooming for survivable WDM networks--shared protection," IEEE J. Sel. Areas Commun. vol. 21, no. 9, pp. 1367-1383, Nov. 2003.

[OU04] C. (Sam) Ou, K. Zhu, H. Zang, H. Zhu, L. H. Sahasrabuddhe, and B. Mukherjee, ``Traffic Grooming for Survivable WDM Networks—-Dedicated Protection [Invited],'' *OSA Journal of Optical Networking*, vol. 3, no. 1, pp. 50-74, Jan. 2004.

[Ou05] C. Ou (Author), B. Mukherjee, “Survivable Optical WDM Networks”, Springer (1st edition), 2005.

[Ozdaglar03] A. E. Ozdaglar, D. P. Bertsekas, “Routing and Wavelength Assignment in Optical Networks”, IEEE/ACM Transactions on Networking, vol. 11, no. 2, pp. 259-272, 2003.

[Pandi06] Z. Pandi, M. Tacca, A. Fumagalli, and L. Wosinska, “Dynamic Provisioning of Availability-Constrained Optical Circuits in the Presence of Optical Node Failures”, IEEE/OSA J. of Lightw. Tech. 24, pp. 3268-3279, 2006.

[Phung07] V.Phung, D. Habibi and H. Nguyen “On Diverse Routing for Shared Risk Link Groups (SRLGs) in Optical Mesh Networks”, ICON, pp. 235-239, 2007.

[Qiao02] C. Qiao, Y. Xiong, and D. Xu, “Novel Models For Efficient Shared-Path Protection,” Proc. OFC, Mar., pp. 546–47, 2002.

[Ram99-a] S. Ramamurthy and B. Mukherjee, “Survivable WDM Mesh Networks, Part II-Restoration,” in Proceeding of IEEE Conference on Communications, pp. 2023-2030, 1999.

[Rama99-b] S. Ramamurthy and B. Mukherjee, “Survivable WDM Mesh Networks, Part I - Protection,” in Proceedings of IEEE Conference on Computer and Communication Societies,pp. 744-751, 1999.

[Rama01] R. Ramamurthy, Z. Bogdanowicz, S. Samieian, D. Saha, B. Rajagopalan, S. Sengupta, S. Chaudhuri, K.. Bala, “Capacity Performance of Dynamic Provisioning in Optical Networks”, J. Lightw. Tech., vol. 19, pp. 40-48, 2001.

[Ramamurthy99-a] S. Ramamurthy, B. Mukherjee, Survivable WDM mesh networks, part I — protection, Proc. of IEEE INFOCOM"99 vol. 2, (New York, NY, USA, March 1999), pp. 744-751.

[Ramamurthy99-b] S. Ramamurthy and B. Mukherjee, “Survivable WDM mesh networks, part I— protection,” *Proc. IEEE INFOCOM '99,* vol. 2, (New York, NY), pp. 744-751, March 1999.

[Ramamurthy02] S. Ramamurthy, B. Mukherjee, “Fixed-Alternate Routing and Wavelength conversion in Wavelength-Routed Optical Networks”, IEEE/ACM Transactions on Networking, 10(3), pp. 351-367, 2002.

[Rosen01] E. Rosen, A. Viswanathan, R. Callon, “Multiprotocol label switching architecture,” RFC 3031, 2001.

[Sahasrabuddhe02] L. Sahasrabuddhe, S. Ramamurthy, B. Mukherjee, “Fault management in IP-over-WDM networks: WDM protection versus IP restoration,” IEEE Journal on Selected Areas in Communications 20 (1) (2002) 21-33.

[Sahin00] G. Sahin and M. Azizoglu, “Optical layer survivability: Single service class case,” in Proc. SPIE Opticomm, vol. 4233, Richardson, TX, pp. 267–278, 2000.

[Sharma03] V. Sharma, F. Hellstrand, “Framework for MPLS-based recovery,” RFC 3469, 2003.

[Shenai05] Ramakrishna Shenai and Krishna Sivalingam, "Hybrid Survivability Approaches for Optical WDM Mesh Networks," J. Lightwave Technol. 23, 3046- (2005).

[Smit04] H. Smit, T. Li, “ISIS extensions for traffic engineering,” RFC3784, 2004.

[Staessens06] D. Staessens et. al, “A Quantitative Comparison of Some Resilience Mechanisms in a Multidomain IP-over-Optical Network Environment”, in Proceedings of the IEEE International Conference on Communications, Vol.6 , pp. 2512-2517, June 2006.

[Subramani02] V. Subramani, R. Kettimuthu, S. Srinivasan, P. Sadayappan, “Distributed job scheduling on computational grids using multiple simultaneous requests”, HPDC, 2002.

[Suurballe74] J. Suurballe, “Disjoint paths in a network”, Networks, vol.14, pp. 125-145, 1974.

[Szeto02] W. Szeto, R. Boutaba, Y. Iraqi, “Dynamic online routing algorithm for MPLS traffic engineering,” Lectures Notes in Computer Science, LNCS 2345, 2002, pp. 936-946.

[Tapolcai08] J. Tapolcai, Pin-Han Ho, D. Verchere, T. Cinkler, A. Haque, *“A new shared segment protection method for survivable networks with guaranteed recovery time”*, IEEE Transaction on Reliability, Vol. 57, No. 2, June 2008, pp. 272-282

[Thiagarajan01-a] S. Thiagarajan and A. K. Somani, “Capacity fairness of WDM networks with grooming capabilities,” *SPIE Opt. Networks Mag.*, vol. 2, pp. 24–32, May/June 2001.

[Thiagarajan01-b] S. Thiagarajan, A.K. Somani, Traffic grooming for survivable WDM mesh networks, in: Proc. OptiCom'01, 2001, pp. 54-65.

[Tsuboi03] T.Tsuboi, M.Natori and S.Mitachi :Single-fibre optical protection ring architecture suitable for asymmetric traffic, Proc. IEEE GLOBECOM 2003, pp.4044-4048, San Francisco, Dec. 2003.

[Valcarenghi08] Luca Valcarenghi, Filippo Cugini, Francesco Paolucci, Piero Castoldi, "Quality-of-service-aware fault-tolerance for grid-enabled applications" Journal of Optical Switching and Networking, Vol: 5, Issue: 2-3,2008, p. 150-158.

[Walkowiak 05] K. Walkowiak, "QoS Dynamic Routing in Content Delivery Network," Lecture Notes in Computer Science, Vol. 3462, 2005 pp. 1120-1132.

[Walkowiak07] K. Walkowiak "Survivable Routing of Unicast and Anycast Flows in MPLS Networks" Conference on Next Generation Internet Networks, 3rd EuroNGI 2007.

[Wang02] J. Wang, L. Sahasrabuddhe & B. Mukherjee, "Path vs. Sub-Path vs. Link Restoration for Fault Management in IP-over-WDM Networks: Performance Comparisons Using GMPLS Control Signalling," IEEE communication magazine, Nov. 2002.

[Widjaja02] I.W. Widjaja, I. Saniee, A. Elwalid, D. Mitra. "Online Traffic Engineering with Design-Based Routing", *15th ITC Specialist Seminar on Internet Traffic Engineering and Traffic Management, W¨urzburg, Germany*, July 2002.

[Wolski99] R.Wolski, N. Spring, J. Hayes, "The Network Weather Service: A Distributed Resource Performance Forecasting Service for Metacomputing", Journal of Future Generation Computing Systems, vol. 14, pp. 757-768, 1999.

[Xiang03] B. Xiang, S. Wang and L.M. Li. "A traffic grooming based on shared protection in WDM mesh networks", in *pmc.IEEE* PDCAT'O3, ChcngDu. China, pp.254-258. August 2003.

[Xiang04] Bing Xiang et al., "A differentiated shared protection algorithm supporting traffic grooming in WDM mesh networks," International Conference on Communications, Circuits and Systems, 2004. ICCCAS 2004. 2004.

[Xin04] Y. Xin et al., "Fault management with fast restoration for optical burst switched networks. Broadband Netw. 34–42 (2004).

[Yao04-a] W. Yao and B. Ramamurthy, "Survivable traffic grooming with differentiated end-to-end availability guarantees in WDM mesh networks," in Local and Metropolitan Area Networks, 2004. LANMAN 2004. The 13th IEEE Workshop on, pp. 87–90, 2004.

[Yao04-b] W. Yao and B. Ramamurthy, "Rerouting schemes for dynamic traffic grooming in optical WDM mesh networks," in Global Telecommunications Conference, 2004. GLOBECOM '04. IEEE, vol. 3, pp. 1793–1797, 2004.

[Yoon01] Sangsik Yoon, Hyunseok Lee, Deokjai Choi, Youngcheol Kim, Gueesang Lee and Lee M, "An efficient recovery mechanism for MPLS-based protection LSP," IEEE International Conference, 2001.

[Zang02] H. Zang, J. P. Jue, and B. Mukherjee, "A review of routing and wavelength assignment approaches for wavelength-routed optical WDM networks," Opt. Netw. Mag., vol. 1, pp. 47–63, Jan. 2000.

[Zang03] H. Zang, C. Ou and Biswanath Mukherjee, "Path-Protection Routing and Wavelength Assignment (RWA) in WDM Mesh Networks Under Duct-Layer Constraints" IEEE/ACM Transactions on Networking, vol. 11, no. 2, April 2003.

[Zhang01] Yi Zhang; Jianping Hu; “Checkpointing and process migration in network computing environment” ICII 2001 - Beijing. Volume 3, 29 Oct.-1 Nov. 2001 Page(s):179 - 184 vol.3.

[Zhang02] H. Zhang, A.Durresi, “Differentiated Multi-layer Survivability in IP/WDM Networks”, in Proceedings of IEEE/IFIP Symposium on Network Operations and Management, pp. 681–694, 2002.

[Zhang03] J. Zhang et al., “On The Study Of Routing And Wavelength-Assignment Approaches for Survivable Wavelength-routed WDM Mesh Networks,” SPIE Opti. Nets. Mag., 2003.

[Zhu02-a] K. Zhu and B. Mukherjee, “Traffic grooming in an optical WDM mesh network,” *IEEE J. Select. Areas Commun.*, vol. 20, pp. 122–133, Jan. 2002.

[Zhu02-b] K. Zhu and B. Mukherjee, “On-line approaches for provisioning connections of different bandwidth granularities in WDM mesh networks,” in *Proc. OFC*, Mar. 2002, p. ThW5.

[Zhu02-c] H. Zhu, H. Zang, K. Zhu, and B. Mukherjee, “Dynamic traffic grooming in WDM mesh networks using a novel graph model,” *Proc. IEEE GLOBECOM*, pp. 2681–2685, Nov. 2002.

[Zhu03] H. Zhu, H. Zang, K. Zhu, and B. Mukherjee, “A novel, generic graph model for traffic grooming in heterogeneous WDM mesh networks,” *IEEE/ACM Trans. Networking*, vol. 11, pp. 285–299, Apr. 2003.

[Zini02] W. Bell, D. Cameron, L. Capozza, A. Millar, K. Stockinger, F. Zini, “Simulation of Dynamic Grid Replication Strategies”, OptorSim, LNCS, Vol. 2536 , pp. 46-57, 2002.

Acronyms

[APS]	Automatic Protection Switching
[ASE]	Amplified Spontaneous Emission (noise)
[BER]	Bit Error Rate
[CR-LDP]	LDP extensions for Constraint-based Routing
[CSPF]	Constraint Shortest Path First
[DC]	Data Consolidation
[DORA]	Dynamic Online Routing Algorithm
[DRAL]	Differentiated Resilience at Lightpath
[DRAC]	Differentiated Resilience at Connection
[FEC]	Forwarding Equivalence Class
[FF]	First-Fit
[FWM]	Four Wave Mixing
[GVD]	Group Velocity Dispersion
[HNS]	Hop Number Server
[HNSW]	Hop Number Widest Server
[IAR]	Impairment-Aware Routing
[IA-RWA]	Impairment-Aware Routing and Wavelength Assignment
[ILP]	Integer Linear Program
[LF]	Last Fit
[LIOA]	Least Interference Optimization Algorithm
[LP]	Linear Program
[MIRA]	Minimum Iinterference Routing Algorithm
[MPAC]	Mixed Protection at Connection
[MPLS]	MultiProtocol Label Switching
[MST]	Minimum Spanning Tree
[NP]	Non-deterministic Polynomial time
[PAL]	Protection at Lightpath
[RRAC]	Rerouting at Connection
[RRAL]	Rerouting at Lightpath
[SPAC]	Separate Protection at Connection
[LDP]	Label Distribution Protocol
[LERS]	Label Edge Routers
[LSRs]	Label Switching Routers

[LSP]	Label Switched Path
[MEMS]	Micro Electro Mechanical Systems
[OXC]	Optical Cross-connect
[PML]	Path Merge LSR
[PSL]	Path Switch LSR
[RCS]	Residual Capacity Server
[RD-QoS]	Resilience-Differentiated QoS (RD-QoS)
[RSVP]	Resource Reservation Protocol
[RSVP-TE]	RSVP Traffic Engineering extension
[RWA]	Routing and Wavelength Assignment
[S-RWA]	Survivable Routing and Wavelength Assignment
[SBPP]	Shared Backup Path Protection
[SLS]	Service Level Specifications
[SPF]	shortest path first
[SPM]	Self-Phase Modulation
[WA]	Wavelength Assignment
[WDM]	Wavelength Division Multiplexing
[WSP]	Widest Shortest Path
[XPM]	Cross Phase Modulation