Université de Montréal

Groupage et protection du trafic dynamique dans les réseaux WDM

par
Ammar Metnani

Département d'informatique et de recherche opérationnelle
Faculté des arts et des sciences

Thèse présentée à la Faculté des arts et des sciences
en vue de l'obtention du grade de Philosophiæ Doctor (Ph.D.)
en informatique

Mars, 2011

Université de Montréal
Faculté des arts et des sciences

Cette thèse intitulée:

Groupage et protection du trafic dynamique dans les réseaux WDM

présentée par:

Ammar Metnani

a été évaluée par un jury composé des personnes suivantes:

Jean-Yves Potvin,	président-rapporteur
Brigitte Jaumard,	directeur de recherche
Jacques Ferland,	membre du jury
Martin Maier,	examinateur externe
Jean-Yves Potvin,	représentant du doyen de la FAS

Thèse acceptée le: . . . . . .

RÉSUMÉ

Avec les nouvelles technologies des réseaux optiques, une quantité de données de plus en plus grande peut être transportée par une seule longueur d'onde. Cette quantité peut atteindre jusqu'à 40 gigabits par seconde (Gbps). Les flots de données individuels quant à eux demandent beaucoup moins de bande passante. Le groupage de trafic est une technique qui permet l'utilisation efficace de la bande passante offerte par une longueur d'onde. Elle consiste à assembler plusieurs flots de données de bas débit en une seule entité de données qui peut être transporté sur une longueur d'onde.

La technique de multiplexage en longueurs d'onde (*Wavelength Division Multiplexing WDM*) permet de transporter plusieurs longueurs d'onde sur une même fibre. L'utilisation des deux techniques : WDM et groupage de trafic, permet de transporter une quantité de données de l'ordre de terabits par seconde (Tbps) sur une même fibre optique. La protection du trafic dans les réseaux optiques devient alors une opération très vitale pour ces réseaux, puisqu'une seule panne peut perturber des milliers d'utilisateurs et engendre des pertes importantes jusqu'à plusieurs millions de dollars à l'opérateur et aux utilisateurs du réseau. La technique de protection consiste à réserver une capacité supplémentaire pour acheminer le trafic en cas de panne dans le réseau.

Cette thèse porte sur l'étude des techniques de groupage et de protection du trafic en utilisant les *p*-cycles dans les réseaux optiques dans un contexte de trafic dynamique. La majorité des travaux existants considère un trafic statique où l'état du réseau ainsi que le trafic sont donnés au début et ne changent pas. En plus, la majorité de ces travaux utilise des heuristiques ou des méthodes ayant de la difficulté à résoudre des instances de grande taille.

Dans le contexte de trafic dynamique, deux difficultés majeures s'ajoutent aux problèmes étudiés, à cause du changement continuel du trafic dans le réseau. La première est due au fait que la solution proposée à la période précédente, même si elle est optimisée, n'est plus nécessairement optimisée ou optimale pour la période courante, une nouvelle optimisation de la solution au problème est alors nécessaire. La deuxième difficulté est due au fait que la résolution du problème pour une période donnée est différente de sa

résolution pour la période initiale à cause des connexions en cours dans le réseau qui ne doivent pas être trop *dérangées* à chaque période de temps.

L'étude faite sur la technique de groupage de trafic dans un contexte de trafic dynamique consiste à proposer différents scénarios pour composer avec ce type de trafic, avec comme objectif la maximisation de la bande passante des connexions acceptées à chaque période de temps. Des formulations mathématiques des différents scénarios considérés pour le problème de groupage sont proposées.

Les travaux que nous avons réalisés sur le problème de la protection considèrent deux types de *p*-cycles, ceux protégeant les liens (*p*-cycles de base) et les FIPP *p*-cycles (*p*-cycles protégeant les chemins). Ces travaux ont consisté d'abord en la proposition de différents scénarios pour gérer les *p*-cycles de protection dans un contexte de trafic dynamique. Ensuite, une étude sur la stabilité des *p*-cycles dans un contexte de trafic dynamique a été faite. Des formulations de différents scénarios ont été proposées et les méthodes de résolution utilisées permettent d'aborder des problèmes de plus grande taille que ceux présentés dans la littérature. Nous nous appuyons sur la méthode de génération de colonnes pour énumérer implicitement les cycles les plus prometteurs.

Dans l'étude des *p*-cycles protégeant les chemins ou FIPP *p*-cycles, nous avons proposé des formulations pour le problème maître et le problème auxiliaire. Nous avons utilisé une méthode de décomposition hiérarchique du problème qui nous permet d'obtenir de meilleurs résultats dans un temps raisonnable. Comme pour les *p*-cycles de base, nous avons étudié la stabilité des FIPP *p*-cycles dans un contexte de trafic dynamique. Les travaux montrent que dépendamment du critère d'optimisation, les *p*-cycles de base (protégeant les liens) et les FIPP *p*-cycles (protégeant les chemins) peuvent être très stables.

Mots clés: Réseaux optiques, Groupage de trafic, Protection par p-cycles, Trafic Dynamique.

ABSTRACT

With new technologies in optical networking, an increasing quantity of data can be carried by a single wavelength. This amount of data can reach up to 40 gigabits per second (Gbps). Meanwhile, the individual data flows require much less bandwidth. The traffic grooming is a technique that allows the efficient use of the bandwidth offered by a wavelength. It consists of assembling several low-speed data streams into a single data entity that can be carried on a wavelength.

The wavelength division multiplexing (WDM) technique allows carrying multiple wavelengths on a single fiber. The use of the two techniques, WDM and traffic grooming, allows carrying a quantity of data in the order of terabits per second (Tbps) over a single optical fiber. Thus, the traffic protection in optical networks becomes an operation very vital for these networks, since a single failure can disrupt thousands of users and may result in several millions of dollars of lost revenue to the operator and the network users. The survivability techniques involve reserving additional capacity to carry traffic in case of a failure in the network.

This thesis concerns the study of the techniques of grooming and protection of traffic using *p*-cycles in optical networks in a context of dynamic traffic. Most existing work considers a static traffic where the network status and the traffic are given at the beginning and do not change. In addition, most of these works concerns heuristic algorithms or methods suffering from critical lack of scalability.

In the context of dynamic traffic, two major difficulties are added to the studied problems, because of the continuous change in network traffic. The first is due to the fact that the solution proposed in the previous period, even if optimal, does not necessarily remain optimal in the current period. Thus, a re-optimization of the solution to the problem is required. The second difficulty is due to the fact that the solution of the problem for a given period is different from its solution for the initial period because of the ongoing connections in the network that should not be too *disturbed* at each time period.

The study done on the traffic grooming technique in the context of dynamic traffic

consists of proposing different scenarios for dealing with this type of traffic, with the objective of maximizing the bandwidth of the new granted connections at each time period. Mathematical formulations of the different considered scenarios for the grooming problem are proposed.

The work we have done on the problem of protection considers two types of p-cycles, those protecting links and FIPP p-cycles (p-cycle protecting paths). This work consisted primarily on the proposition of different scenarios for managing protection p-cycles in a context of dynamic traffic. Then, a study on the stability of cycles in the context of dynamic traffic was done. Formulations of different scenarios have been proposed and the proposed solution methods allow the approach of larger problem instances than those reported in the literature. We rely on the method of column generation to implicitly enumerate promising cycles.

In the study of path protecting p-cycles or FIPP p-cycles, we proposed mathematical formulations for the master and the pricing problems. We used a hierarchical decomposition of the problem which allows us to obtain better results in a reasonable time. As for the basic p-cycles, we studied the stability of FIPP p-cycles in the context of dynamic traffic. The work shows that depending on the optimization criterion, the basic p-cycles (protecting the links) and FIPP p-cycles (protecting paths) can be very stable.

Keywords: Optical Networks, Traffic grooming, Protection by p-cycles, Dynamic Traffic.

TABLE DES MATIÈRES

LISTE DES FIGURES

À mes parents, ma femme, mes deux enfants et à toute ma famille.

REMERCIEMENTS

Au terme de cette thèse, je tiens à exprimer mes gratitudes et mes hautes considérations à toutes les personnes qui m'ont accordé le privilège de réaliser ce travail.

J'ai aussi le plaisir d'exprimer mes remerciements les plus sincères à Madame Brigitte Jaumard, Professeure à l'Université de Concordia, pour avoir supervisé cette thèse ainsi que pour ses très précieux conseils, pour tout l'intérêt et le suivi qu'elle n'a cessé de démontrer durant la réalisation de cette thèse.

Je tiens à remercier très sincèrement, Monsieur Jean-Yves Potvin et Monsieur Jacques Ferland, Professeurs, au Département d'Informatique et de Recherche Opérationnelle de l'Université de Montréal et Monsieur Martin Maier, Professeur aggrégé à l'Institut National de la Recherche scientifique affilié à l'Université du Québec à Montréal, pour m'avoir fait l'honneur de former le jury de ma thèse.

Une grande partie de recherche a été réalisée au Centre de recherche sur les Transports devenu récemment CIRRELT, Centre Interuniversitaire de Recherche sur les Réseaux d'Entreprise, la Logistique et le Transport. À cet égard, je voudrais souligner la qualité de l'assistance technique du Centre et que tout le personnel trouve ici mes remerciements les plus sincères pour leurs collaborations et leurs conseils.

J'ai une pensée émue pour le soutien émotionnel sans fin manifesté par ma femme tout au long de la réalisation de cette thèse. Je tiens à remercier également mes parents et toute ma famille pour leurs soutiens moraux. Enfin, je tiens à remercier mes deux enfants pour les séances de thérapie les plus efficaces contre le stress et les maux rencontrés pendant la réalisation de cette thèse.

CHAPITRE 1

INTRODUCTION

1.1 Contexte et motivation du projet de recherche

L'émergence de la technologie WDM (Wavelength Division Multiplexing) dans les réseaux de télécommunication a augmenté de façon considérable la capacité des fibres optiques, en permettant la transmission de plusieurs longueurs d'onde (canaux) sur une même fibre. Chaque longueur d'onde peut transmettre à un débit de 10 Gbps, voire 40 Gbps. Avec 80 longueurs d'onde, la fibre peut transporter une quantité de données de l'ordre de quelques terabits par seconde (Tbps). Avec cette quantité de données énorme à transporter, et dans un contexte de trafic dynamique, deux problèmes majeurs se posent :

1. Avec une possibilité de transmettre une quantité de données de 2.5, 10 ou 40 giga bits par secondes (Gbps), une longueur d'onde est sous-utilisée si on lui affecte une seule connexion, sachant que la majorité des connexions dans les réseaux sont des connexions de l'ordre d'OC-1 (51.84 Mbps), OC-3, OC-12 ou Oc-48 [131]. Si on a une requête pour une connexion de 51.84 Mbps avec une longueur d'onde qui peut transmettre jusque dans l'ordre de OC-192 (10 Gbps), alors presque toute la bande passante est gaspillée. Heureusement, il existe un moyen de remédier à ce problème, qui est de regrouper plusieurs connexions de petites granularités sur une seule longueur d'onde de telle sorte que la somme des bandes passantes de ces connexions ne dépasse pas la capacité de la longueur d'onde. Cette technique s'appelle le groupage de trafic (*traffic grooming*).
   En pratique, le problème du groupage de trafic consiste à résoudre de la façon la plus efficace possible les points suivants :

   - Parmi toutes les connexions, lesquelles devons-nous regrouper ensemble ?
   - Quel est la meilleure route pour acheminer ces connexions ?
   - Sur quelle longueur d'onde ces connexions seront transmises ?

Dans un contexte de trafic dynamique, on est continuellement en présence de nouvelles requêtes de connexions, et aussi, certaines connexions peuvent s'arrêter. Ce changement continuel dans le trafic du réseau rend le groupage initial de trafic inefficace. Une connexion qui s'arrête et qui est groupée sur un chemin optique laisse "*sa place*" vide et donc le groupage des connexions sur ce chemin n'est plus nécessairement efficace en termes d'utilisation de la bande passante. Une nouvelle requête de connexion peut être acceptée sur cet "espace" laissée par la connexion qui s'est arrêtée, si la bande passante totale ne dépasse pas celle de la longueur d'onde sur laquelle le chemin optique est routé. Cependant, l'origine et la destination de cette nouvelle requête peuvent être différentes et l'*insertion* de cette nouvelle connexion sur l'espace laissé par une connexion qui s'est arrêtée peut rendre le groupage encore moins efficace (ex. chemin plus long).

Si la nouvelle requête de connexion ne peut être acceptée sur aucun chemin optique, faute de bande passante, et si les équipements du réseau ne permettent pas d'établir de nouveaux chemins, cette connexion est bloquée. Qu'arrive-t il si cet espace peut se trouver sur deux chemins différents ? C'est à dire, si la somme des deux bandes passantes disponibles sur les deux chemins est supérieure à celle de la nouvelle requête ?

La bifurcation des flots de données est possible grâce aux protocoles VCAT/LCAS [7, 28], c'est ce qu'on appelle le multiplexage inversé [48]. Une requête de connexion qui ne peut être acceptée sur aucun chemin optique, peut être acceptée sur deux (ou plus) chemins optiques si on décide de répartir son trafic sur deux chemins différents.

Par contre, si on décide de ne pas accepter la répartition du trafic d'une seule connexion sur plus d'un chemin, le fait d'autoriser le reroutage de certaines connexions déjà établies peut aussi permettre l'acceptation de cette nouvelle connexion.

La première partie de la thèse étudie toutes ces possibilités pour minimiser le nombre de connexions bloquées, et montre la meilleure façon de faire pour maximiser le trafic accepté à chaque période de temps.

2. Une seule panne dans le réseau optique peut déranger des centaines de milliers d'utilisateurs et causer une perte de l'ordre de millions de dollars aux opérateurs et aux utilisateurs du réseau. Les pannes au niveau du lien (*link failures*) et au niveau du nœud (*node failure*) sont les deux pannes de base dans un réseau. La coupure de fibre optique est la panne la plus fréquente dans les réseaux optiques, elle peut être causée par un phénomène naturel ou par des erreurs humaines [125]. Il faut donc s'assurer de continuer de fournir le trafic aux destinataires en cas de panne dans le réseau. La protection du trafic en cours est une opération très importante pour la survie des réseaux optiques. Elle consiste à prévoir une certaine bande passante pour acheminer un trafic en cas de panne sur le chemin de routage initial.

   Dans [13], on rapporte qu'environ 160 coupures de fibre optique ont été recensées sur une période d'une année et que la durée moyenne de réparation d'une panne typique est d'environ 5 heures, ce qui rend la probabilité qu'une deuxième coupure de fibre se produise, pendant ce temps, très petite [29, 125]. C'est ce qui nous incite à considérer seulement les coupures d'une seule fibre (*single link failures*). Plusieurs méthodes de protection du trafic d'un réseau ont été proposées. Ces méthodes varient de la protection de tout le chemin de la connexion à la protection au niveau des liens en passant par la protection au niveau d'une partie du chemin de routage (segment). Une des principales méthodes de protection destinée aux réseaux optiques WDM est la protection par cycles préconfigurés et pré-cross-connectés ou p-cycles. Un des critères à prendre en considération lors d'un choix de schéma de protection, est le temps de réaction aux pannes. Les *p*-cycles ont un temps de reprise très rapide en cas de pannes, un critère très important pour tout mécanisme de protection. À l'origine, les *p*-cycles ont été introduits pour la protection des liens. Cependant, plusieurs extensions du concept des *p*-cycles ont été proposées par les chercheurs, pour la protection de segments ou de chemins. La majorité des études faites sur les *p*-cycles ont été faites dans un contexte de trafic statique. Parmi les études qui concernent un trafic dynamique, la majorité (encore une fois) a utilisé des heuristiques pour résoudre ce problème ou des méthodes qui ne sont pas efficaces d'un point de vue temps de calcul (*scalable*).

La deuxième partie de la thèse étudie la protection d'un trafic dynamique par *p*-cycles et propose plusieurs scénarios pour la mise à jour de la protection à chaque période de temps. En plus, elle étudie une extension des *p*-cycles, en l'occurrence les FIPP *p*-cycles. En plus, on propose des méthodes de résolution efficaces d'un point de vue temps de calcul qui permettent de résoudre des instances plus grandes. Ces méthodes donnent des solutions optimales ou quasi-optimales en pratique.

1.2 Contribution et organisation de la thèse

Plusieurs résultats ont été obtenus grâce a nos recherches sur les deux problèmes étudiés dans cette thèse. Ces résultats ont été soumis dans des revues ou des conférences internationales avec arbitrage. Nous avons inclus dans la thèse quatre des articles que nous avons produits. Chaque article est présenté dans un chapitre de la thèse. Puisque les articles ont été publiés indépendamment les uns des autres, quelques redondances seront observées notamment dans les sections d'introduction des différents articles.

Les chapitres 2 et 3 ont été ajoutés pour introduire des généralités sur les problèmes étudiés et les solutions proposées dans la littérature, respectivement.

Dans le chapitre 4, nous proposons quatre scénarios différents pour résoudre le problème de groupage de trafic dynamique dans un réseau maillé (*mesh network*) où la granularité des requêtes de connexion est hétérogène. Dans le premier scénario, on s'intéresse à maximiser le nombre de connexions acceptées, pour chaque période de temps, sans la modification du groupage existant, c'est-à-dire qu'on ne remet pas en question les connexions placées aux instants précédents. Dans le deuxième scénario proposé, on s'intéresse, comme dans le premier scénario, à maximiser le nombre de connexions acceptées, mais on autorise le dérangement de connexions déjà placées aux instants précédents. Ce dérangement doit être minimal. Le troisième scénario est similaire au deuxième, hormis le fait qu'on autorise la mise à jour de certains ports. Le quatrième scénario, quant à lui, est défini dans un contexte de réseau agile. À chaque période de temps, on redéfinit un schéma de groupage et tout ce qui a été défini aux périodes précédentes est remis en cause. Dans tous les scénarios proposés, on a deux modèles de

trafic :

- Le trafic est acheminé avec des flots non bifurqués, où tout le trafic d'une connexion doit emprunter, à partir de chaque nœud, une même longueur d'onde sur un même segment, et par conséquent une connexion ne peut être accommodée que si on a, sur une des longueurs d'onde de chacun des segments d'un chemin de la source à la destination, une bande passante disponible supérieure ou égale à la bande passante demandée par la connexion en question.

- Le trafic est acheminé avec des flots bifurqués, où le trafic d'une même connexion peut emprunter des longueurs d'onde différentes sur des chemins différents de la source à la destination.

On a proposé des modèles mathématiques aux scénarios proposés, soit un modèle pour chacun des scénarios considérés.

Dans le chapitre 5, nous proposons trois stratégies différentes pour la protection du trafic dynamique par les *p*-cycles. La différence entre les trois stratégies, à chaque période de temps, provient de la façon dont on traite les *p*-cycles établis dans les périodes précédentes. Dans la première stratégie, aucun *p*-cycle établi ne peut être dérangé ou démantelé s'il protège encore du trafic. Dans la deuxième stratégie, et pour améliorer la protection globale, on tolère le dérangement ou le démantèlement de *p*-cycles en place. Cependant, le nombre de cycles dérangés ne peut dépasser un certain seuil défini. La troisième stratégie, quant à elle, permet une ré-optimisation globale de la protection. Cela signifie trouver une meilleure protection possible pour le trafic en cours, à chaque période de temps. Cette protection peut inclure les anciens, comme les nouveaux *p*-cycles.

Le chapitre 6 présente une étude sur la stabilité des *p*-cycles dans un contexte de trafic dynamique. Nous proposons deux modèles mathématiques différents. Le premier modèle consiste à minimiser le coût de la bande passante totale des *p*-cycles de protection à chaque période de temps. Le deuxième modèle minimise le nombre de ports utilisés ou reconfigurés par les *p*-cycles, pour protéger le trafic à chaque période de temps. Nous proposons aussi une comparaison de la stabilité entre ces deux modèles.

Nous proposons, dans le chapitre 7, d'étudier le problème de protection par les FIPP *p*-cycles dans le contexte d'un trafic dynamique. Nous introduisons une nouvelle formulation mathématique et une résolution basée sur la technique de génération de colonnes. Nous étudions la stabilité des FIPP *p*-cycles dans un trafic dynamique et l'effet de l'évolution de trafic sur cette stabilité.

Le chapitre 8 conclut la thèse et propose quelques suggestions de recherches futures.

1.3 Articles rédigés durant la thèse

Les articles de revues et de conférence rédigés durant mes travaux de thèse sont énumérés ci-dessous. Ceux inclus comme chapitres dans la thèse sont indiqués avec un astérisque ($*$).

1$*$. A. Metnani et B. Jaumard "Directed *p*-cycle protection in dynamic WDM networks", *International Conference on Ultra Modern Telecommunications and Workshops, 2009. ICUMT'09*. Article publié [68].

2. B. Jaumard et A. Metnani "Stability of *p*-cycle under dynamic traffic", *IEEE Global Telecommunications Conference. GLOBECOM'10*. Article publié [54].

3$*$. A. Metnani et B. Jaumard "Dynamic Provisioning and Stability of p-Cycles in WDM Networks". Article accepté. À paraître dans *IEEE/OSA Journal of Optical Communications and Networking*.

4. A. Metnani et B. Jaumard "Dynamic Protection Provisioning with FIPP *p*-Cycles in WDM Networks". *International Conference on Optical Network Design and Modeling. ONDM 2011*. Article publié [69].

5$*$. A. Metnani et B. Jaumard "Stability of FIPP *p*-Cycles under Dynamic Traffic in WDM Networks". Article accepté. À paraître dans *IEEE/ACM Transactions on Networking*.

6$*$. A. Metnani, B. Jaumard et A. Houle "Dynamic Traffic Grooming in WDM Mesh Networks". Article soumis pour publication.

7. A. Metnani et B. Jaumard "Connection Rerouting in GRWA networks". Article soumis pour publication.

CHAPITRE 2

GÉNÉRALITÉS SUR LES RÉSEAUX OPTIQUES

2.1 Introduction

Dans une fibre optique, on a de plus en plus de longueurs d'onde. De plus, la quantité de données que peut transporter chacune des longueurs d'onde est de plus en plus grande. Bien que les applications sur le web soient de plus en plus gourmandes en bande passante, l'écart entre la capacité de la longueur d'onde et la quantité de la bande passante demandée par une requête ne cesse de s'accroître. Pour bien exploiter les capacités offertes par les longueurs d'onde, plusieurs requêtes doivent être groupées ensemble sur une même longueur d'onde. Avant d'introduire en détail le problème du groupage, introduisons quelques définitions.

2.2 Les réseaux WDM/DWDM

La fibre optique offre une capacité de transport énorme, et pour bien exploiter cette capacité, il faut faire appel à la technique de multiplexage.

2.2.1 Le multiplexage en longueurs d'onde WDM

Pour une meilleure exploitation de la très grande bande passante de la fibre optique, on procède au regroupement de plusieurs canaux sur cette fibre. Le multiplexage WDM consiste à regrouper plusieurs signaux optiques et les transmettre sur une seule fibre. À l'autre bout, les données sont retransformées par un démultiplexeur (figure 2.1). On a constaté qu'on peut insérer dans une fibre optique simultanément des signaux sur des longueurs d'ondes différentes sans que les signaux s'influencent mutuellement. Une exploitation de cette propriété nous permet donc d'augmenter la quantité de données transportée sur une fibre. Elle conduit au multiplexage des signaux sur une même fibre mais sur des longueurs d'onde différentes. La norme IUT G692 a défini un peigne de lon-

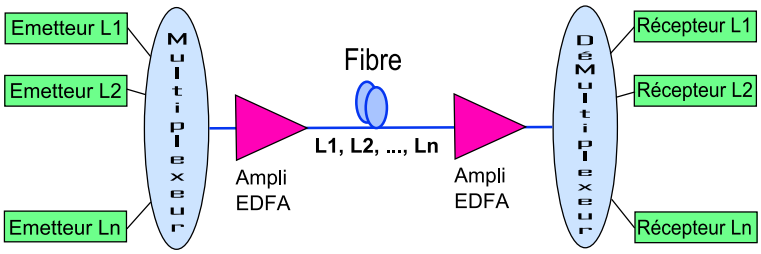


Figure 2.1 – Principe d'une liaison WDM/DWDM (extraite de www.telcite.fr)

gueurs d'onde autorisées dans la fenêtre de transmission 1530-1565 nm. Elle normalise l'espacement en nanomètre (nm) ou en gigahertz (GHz) entre deux longueurs d'onde permises de la fenêtre : 200 GHz ou 1,6 nm ou 100 GHz ou 0,8 nm. Si l'espacement spectral entre deux longueurs d'ondes est très réduit (égal ou inférieur à 100 GHz, i.e., 0,8 nm), on peut donc transmettre plusieurs longueurs d'ondes, on parle du DWDM (Dense WDM). Des systèmes à 50 GHz (0,4 nm) et à 25 GHz (0,2 nm) permettent d'obtenir respectivement 80 et 160 canaux optiques.

2.2.2 Le multiplexage temporel TDM

Chaque longueur d'onde peut transporter une quantité de trafic de l'ordre de 10 ou même 40 Gbps. Le trafic d'un seul client atteint rarement cette quantité de trafic, et pour

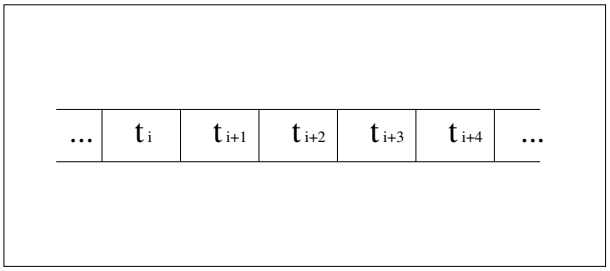


Figure 2.2 – Principe du TDM

cette raison on procède à un multiplexage temporel pour bien *remplir* chaque longueur d'onde. Le multiplexage temporel est donc une technique qui regroupe les données de

plusieurs canaux (des clients) sur un seul canal (longueur d'onde), dans des fenêtres temporelles t_i. Chaque client se voit attribuer des fenêtres temporelles pour transmettre ses données. Il peut arriver que, pendant plusieurs fenêtres, consécutives ou pas, aucune information ne soit transmise. Les liens d'entrée à la fibre ont un faible débit par rapport à la fibre de sortie. L'implantation du multiplexage temporel est définie par la norme SONET (Synchronous Optical NETworking) [103].

2.2.3 Le système SONET/SDH

SONET (Synchronous Optical NETworking) est un standard très important de multiplexage et de transmission dans les réseaux optiques. SONET est basé sur le multiplexage temporel TDM. Il est totalement synchrone, où toutes les horloges du réseau sont synchronisées à un horloge *maitre*. Une trame est émise, en permanence, périodiquement (même s'il n'y a pas de trafic à transmettre) toutes les 125 μ. Cette trame de base est un bloc de 810 octets, ce qui donne un débit de 51,84 Mbit/s = 1 STS-1/OC-1 (Synchronous Transport Signal-1/ Optical Carrier-1).

Signal SONET	Débit (Mbps)
STS-1	51.84
STS-3	155.52
STS-12	622.08
STS-24	1244.16
STS-48	2488.32
STS-192	9953.28
STS-768	39814.32

Tableau 2.I – Débits des signaux SONET

Les signaux de haut débit sont multiplexés à partir des signaux de plus bas débit par entrelacement octet par octet. Par exemple, pour former un signal STS-3 (un signal STS-n est égal à n signal STS-1), trois signaux STS-1 sont multiplexés de la façon suivante : on prend le premier octet du premier signal STS-1 puis le premier octet du deuxième signal STS-1 puis le premier octet du troisième signal STS-1 avant de revenir avec le deuxième octet du premier signal STS-1 et ainsi de suite. De la même façon, un signal

STS-12 peut être obtenu en multiplexant quatre signaux STS-3. La table 2.I montre les signaux SONET et leurs débits.

Les signaux non-SONET de débit plus petit qu'un signal STS-1 sont groupés dans des affluents virtuels VTs (Virtual Tributaries). Les VTs sont définis dans quatre formats : VT1.5 (1.5 Mbit/s), VT2 (2Mbps), VT3 (3Mbps) et VT6 (6Mbps). Par exemple, un utilisateur qui demande une bande passante T1 = 1.5 Mbit/s, son trafic est affecté à un VT1.5 alors qu'un autre utilisateur qui demande une bande passante E1 = 2.0 Mbit/s, son trafic est affecté à un VT2. Dans le niveau suivant de l'hiérarchie on trouve le groupe VT (VT groupe) qui contient soit quatre VT1.5 ou trois VT2 ou deux VT3 ou encore un seul VT6. Chaque sept groupes VT sont arrangés et entrelacés avant d'être insérés dans des trames SONET [80], [103].

2.2.4 Quelques éléments d'un réseau SONET/WDM

En plus de la fibre qui compose le réseau optique WDM, plusieurs autres équipements sont utilisés pour mettre en place la technologie WDM et bien exploiter le réseau. Parmi ces éléments [79][111], on trouve :

OLT (Optical Line Terminal) : ce type d'équipements est utilisé aux extrémités d'une fibre. Un OLT a trois fonctions principales. Premièrement il sert de transpondeur pour faire la conversion du signal d'un client à un signal optique prêt à être transporté sur un lien WDM et inversement d'un signal optique à un signal adapté au client. Deuxièmement, il fait le multiplexage/démultiplexage des longueurs d'onde, et finalement, il peut contenir un amplificateur optique pour amplifier le signal en cas de besoin.

OADM (Optical Add/Drop Multiplexer) : ces équipements utilisés dans un nœud servent, comme leur nom l'indique, à ajouter (en multiplexant) une longueur d'onde à une fibre de sortie et terminer (en démultiplexant) une longueur d'onde dans ce nœud à partir d'une fibre d'entrée. Un OADM dans un nœud permet aussi, à une longueur d'onde qui n'est ni originaire de ce nœud ni destinée à ce nœud, de *traverser* le nœud en restant dans le domaine optique sans être convertie. La figure 2.3 montre

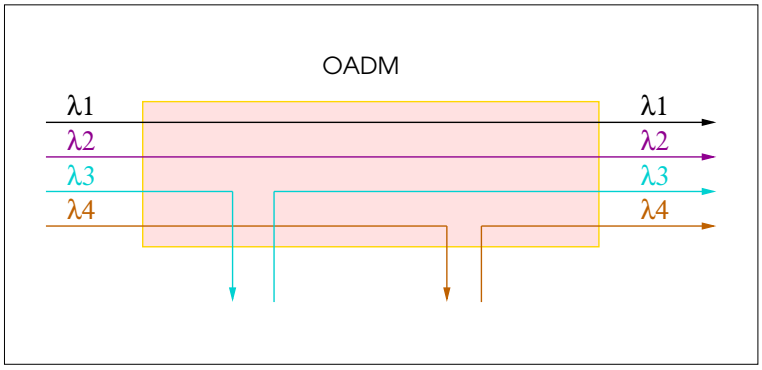


Figure 2.3 – Exemple d'un OADM

un OADM qui permet aux données de deux longueurs d'onde $\lambda 1$ et $\lambda 2$ de passer directement et qui retranche et ajoute des données aux deux autres longueurs d'onde $\lambda 3$ et $\lambda 4$.

SADM (SONET Add/Drop Multiplexer) : ces équipements électriques utilisés dans un nœud servent à ajouter et retrancher le trafic d'une longueur d'onde selon le multiplexage temporel. Elles permettent aussi de laisser passer le trafic qui n'est pas destiné au nœud courant. Dans la figure 2.4 on voit l'utilisation de l'équi-

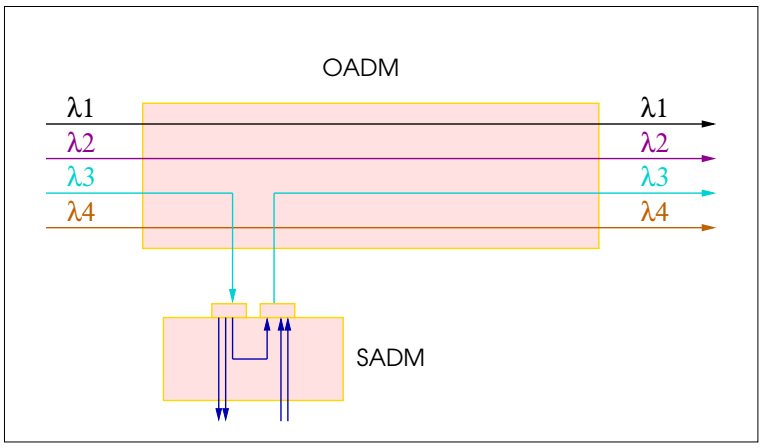


Figure 2.4 – Utilisation d'un SADM avec un OADM

pement SADM avec un OADM. La longueur d'onde $\lambda 3$ étant retranchée par le

OADM, le SADM y ajoute/retranche le trafic du nœud actuel et si nécessaire laisse passer le trafic qui est destiné à d'autres nœuds.

ROADM (Reconfigurable Optical Add/Drop Multiplexer) : c'est un OADM reconfigurable, comme son nom l'indique. Il offre, en plus des fonctions d'un OADM, la possibilité de commuter des signaux de transport optiques à distance au niveau de la couche de longueurs d'onde. Cela permet à une ou plusieurs longueurs d'onde transportant des signaux clients d'être ajouté ou retranché de la fibre optique sans la nécessité de convertir les signaux sur tous les canaux WDM en des signaux électriques et vice versa.

DXC (Digital Cross Connect) : c'est un équipement qui opère, comme le SADM, dans le domaine électrique, il peut ajouter ou retrancher le trafic d'une longueur d'onde selon le multiplexage temporel, mais aussi peut commuter le trafic qui arrive sur une longueur d'onde pour l'orienter sur une autre. On peut voir un DXC comme un ensemble de SADMs [111]. Dans la figure 2.5, le trafic de la longueur d'onde $\lambda 3$ peut être envoyé sur la longueur d'onde $\lambda 4$ et inversement.

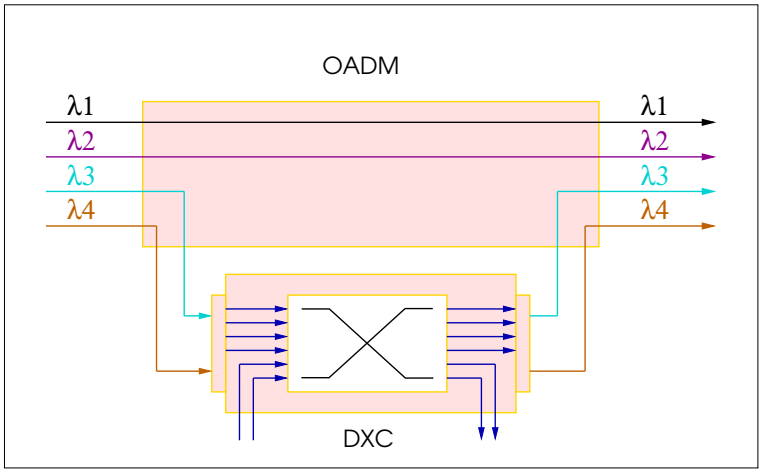


Figure 2.5 – Utilisation d'un DXC avec un OADM

OXC (Optical Cross Connect) : les OADMs ne sont plus suffisants quand on a plu-

sieurs fibres qui sont connectées à un nœud donné et que chaque fibre a plusieurs longueurs d'onde (comme dans les réseaux maillés) [79]. L'élément le mieux adapté dans ce cas est l'OXC. Un OXC simple permet de commuter les don-

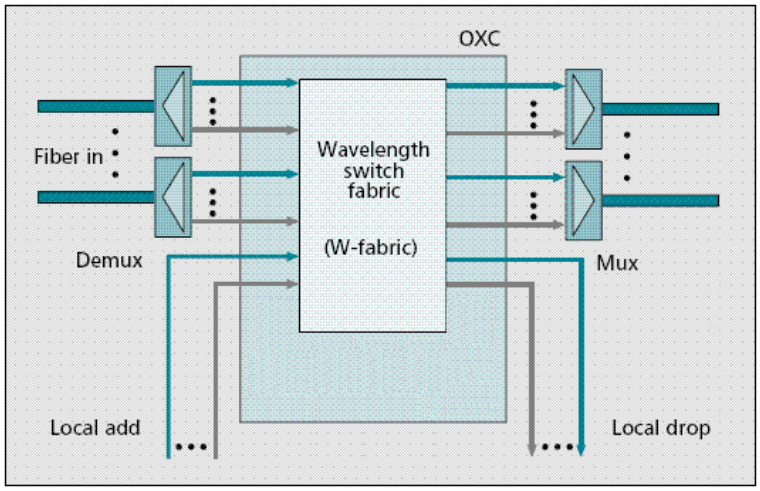


Figure 2.6 – Exemple d'un OXC simple

nées d'une longueur d'onde λ d'une fibre d'entrée vers la même longueur d'onde λ d'une fibre de sortie dans le domaine optique. L'OXC permet aussi de retrancher des données d'une longueur d'onde et d'ajouter des données à une longueur d'onde. L'OXC peut avoir la capacité de conversion de longueur d'onde, c'est-à-dire, commuter les données d'une longueur d'onde $\lambda 1$ d'une fibre d'entrée vers une autre longueur d'onde $\lambda 2$ d'une fibre de sortie. Un *grooming OXC* (Figure 2.7), est composé de deux parties, premièrement, le W-fabric (Wavelength Switch Fabric) qui peut commuter des longueurs d'onde entières, et deuxièmement, le G-fabric (Grooming Fabric) qui, quant à lui, peut commuter des parties d'une longueur d'onde, et faire le groupage de trafic.

MSPP (MultiService Provisioning Platform) : les MSPPs sont des systèmes très complexes, impliquant une variété de technologies matérielles et logicielles. Un MSPP combine plusieurs fonctionnalités offertes par les différents équipements qu'on a mentionnés plus haut. Il offre une multitude d'interfaces couvrant une large

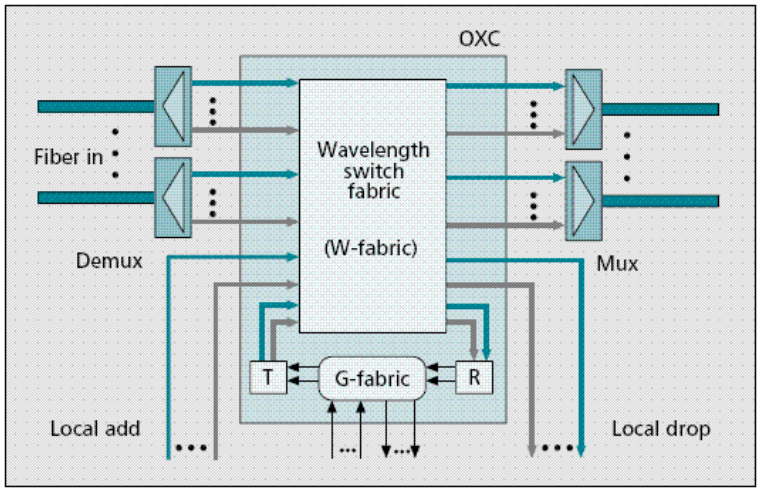


Figure 2.7 – Grooming OXC (extraite de [131])

gamme des signaux clients : des interfaces téléphoniques (DS-1, DS-3), des interfaces optiques (OC-3, OC-12), des interfaces Ethernet (10/100Base-T) et même des interfaces DSL et Gigabit Ethernet. Un MSPP peut connecter un protocole à n'importe quel port. Il permet la gestion du trafic, le routage, le groupage et la commutation, ainsi que la conversion et la régénération des signaux de transport.

2.3 La topologie virtuelle

Dans un réseau optique basé sur le WDM les liens sont des fibres optiques, chacune transporte (contient) W longueurs d'onde λ. Chaque longueur d'onde peut transporter une quantité d'information égale à 10 Gbps (OC-192), voire 40 Gbps (OC-768). Un chemin optique est un chemin établi entre deux nœuds (un nœud source et un nœud destination), qui transporte l'information sous forme lumineuse. Une topologie virtuelle est formée de tous les chemins optiques qui sont établis à un moment donné. Donc, l'ensemble des chemins optiques établis est vu par les protocoles des couches supérieures comme une topologie virtuelle au dessus de la topologie physique du réseau [85] comme indiqué sur la figure 2.8. En établissant un chemin optique entre deux nœuds, ces derniers deviennent voisins même s'ils ne sont pas reliés directement par une fibre optique.

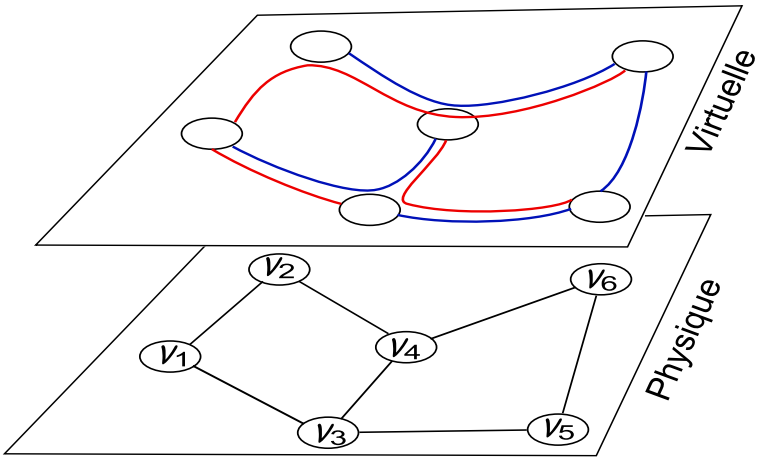


Figure 2.8 – Topologie virtuelle

Cependant, il est possible qu'il ne soit pas possible d'établir un chemin optique entre chaque paire de nœuds à cause d'un manque de ressources ou pour des raisons économiques [24], donc le trafic doit être commuté électroniquement d'un chemin optique à un autre dans les nœuds intermédiaires, c'est ce qu'on appelle *"Multihopping"* [24].

2.4 Le problème RWA

Le réseau optique est représenté comme un graphe G = (V, E) où V représente l'ensemble des nœuds du réseau et E représente l'ensemble des liens qui sont des fibres optiques qui relient ces nœuds. Soit $n = |V|$.
L'ensemble des requêtes ou demandes de connexion est représenté par une matrice $n \times n$ $D = (d_{ij})$ où d_{ij} représente le nombre de demandes (requêtes) de connexions de la source v_i vers la destination v_j. Soit k une requête de connexion de la source v_s vers la destination v_d, le problème RWA consiste à trouver pour la connexion de la requête k

- Une route p pour l'acheminer de la source v_s à la destination v_d.
- Une longueur d'onde λ, sur les liens de la route p, sur laquelle les données de la connexion seront portées.

Le but du problème RWA est de router les requêtes de connexions de la matrice D, soit en utilisant le minimum de longueurs d'onde possible, soit en minimisant le coût du réseau (en minimisant le nombre de ports, par exemple) ou encore en maximisant le nombre de requêtes acceptées si les ressources du réseau ne sont pas suffisantes pour router toutes les demandes. Le problème RWA dans sa forme initiale ne considère pas de conversions de longueurs d'onde, c'est-à-dire qu'une connexion doit utiliser la même longueur d'onde de la source à la destination. Deux connexions qui ont la même source et la même destination peuvent utiliser deux chemins différents. Soient deux requêtes k et k', de même source v_s et de même destination v_d. Les possibilités de définition de leur chemin optique sont :

- Le même chemin p pour les requêtes k et k'. Dans ce cas, la connexion de la requête k' doit impérativement utiliser une autre longueur d'onde que celle utilisée par k.

- Un chemin p pour la requête k et un autre chemin p' pour la requête k'. Deux situations peuvent se produire :

    1. Le chemin p' est complètement disjoint (par rapport aux liens) du chemin p, et dans ce cas la connexion de la requête k' peut utiliser la même longueur d'onde que la connexion de la requête k.

    2. Le chemin p' a au moins un lien en commun avec le chemin p. La connexion de la requête k' doit impérativement utiliser une autre longueur d'onde que celle de la requête k.

Dans le cas du problème RWA où on considère la conversion des longueurs d'onde au niveau des nœuds intermédiaires, plusieurs contraintes se simplifient : la chose la plus importante qu'on doit retenir est qu'au niveau de chaque lien, deux connexions doivent utiliser deux longueurs d'onde différentes.

Le problème RWA est reconnu être un problème NP-difficile dans le cas général, et il est généralement résolu en deux étapes, la première consiste à trouver les chemins sur

lesquels les connexions seront acheminées, et la deuxième consiste à affecter des longueurs d'onde à ces connexions. Une résolution exacte en une phase est aussi possible sous certaines conditions. Dans l'étude de Jaumard *et al.* [55], on a étudié différentes formulations sous forme de programmes linéaires en nombres entiers (PLNE) au problème RWA qui permettent de résoudre efficacement le problème RWA en une phase.

2.5 Le problème GRWA

Dans le cas où le nombre de connexions est trop grand, il est possible que le RWA ne peut satisfaire toutes les requêtes de connexions, et ce malgré que la somme des bandes passantes des connexions satisfaites est loin de remplir toute la bande passante offerte par le réseau. Cela est dû au fait que toute la bande passante d'une longueur d'onde est allouée à une connexion qui la remplit avec un petit pourcentage. Pour améliorer l'efficacité de l'utilisation de la bande passante offerte par le réseau optique, une méthode de groupage (grooming) des connexions est utilisée.

Le groupage de trafic (grooming) consiste à grouper des connexions de bas débit dans une même longueur d'onde de telle sorte que la somme des bandes passantes des connexions groupées ensemble ne dépasse pas la capacité de bande passante d'une longueur d'onde. Le problème GRWA peut être divisé en deux sous problèmes :

1. le problème de groupage (topologie virtuelle), qui consiste à regrouper les connexions dans une unité qui a la même capacité qu'une longueur d'onde, cette unité s'appelle un chemin optique. Cette partie consiste donc à trouver un ensemble de chemins optiques à satisfaire.

2. le problème RWA, qui consiste à router et affecter des longueurs d'onde aux chemins optiques définis dans le sous problème précédent.

Plusieurs façons sont généralement possibles pour grouper un trafic dans un réseau optique. Cependant, dépendamment de l'objectif à atteindre, il y a souvent une façon plus efficace qui résout le problème de groupage.

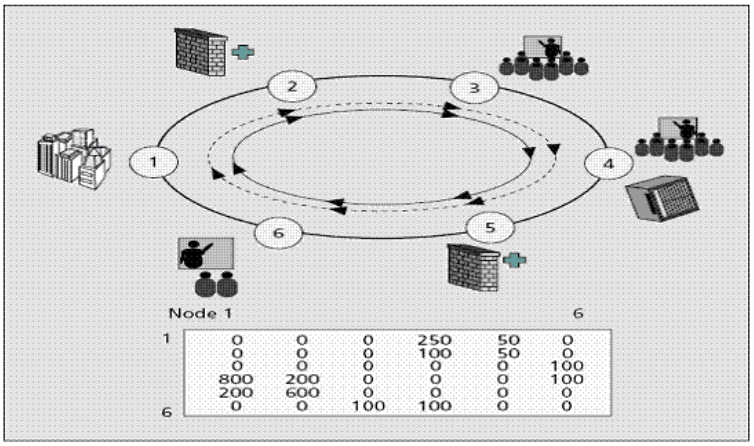


Figure 2.9 – Un anneau à six nœuds (extraite de [19])

Pour bien illustrer l'impact d'un GRWA efficace sur le nombre d'équipements et de longueurs d'onde utilisées, on se réfère à l'exemple cité dans [19]. Dans la figure 2.9, on a un réseau en anneau unidirectionnel de 6 nœuds et une matrice qui représente le trafic qui doit être transmis entre les nœuds du réseau. On considère que la capacité de chaque longueur d'onde est de 1 Gbps. En utilisant le groupage des connexions, on aura seulement besoin de 2 longueurs d'onde pour transporter les 12 connexions du réseau, ce qui représente le nombre minimum de longueurs d'onde qu'on peut utiliser.

Un réseau optique WDM consiste en des nœuds qui sont reliés par des fibres optiques. Ces nœuds qui traitent les données électroniquement doivent faire un travail énorme pour traiter toute cette quantité d'information qui est transmise par la fibre et qui est de l'ordre du Tbps. Généralement, la majorité du trafic qui arrive à un nœud n'est pas destinée à ce nœud même, et donc il n'est pas nécessaire de traiter ces données dans le domaine électrique, il suffit au nœud de laisser passer ces données sans les traiter, ce qui peut sauver du temps et de l'argent et améliore le rendement des réseaux. Dans l'exemple de la figure 2.9, chaque nœud est obligé de traiter électroniquement le trafic des deux longueurs d'onde. Une autre solution pour cet exemple est celle de la figure 2.10, où on a besoin de 6 longueurs d'onde pour router les 12 connexions et aucun nœud n'est obligé de traiter les données qui ne lui sont pas destinées.

Une bonne solution de routage de connexions doit être entre les deux solutions pré-

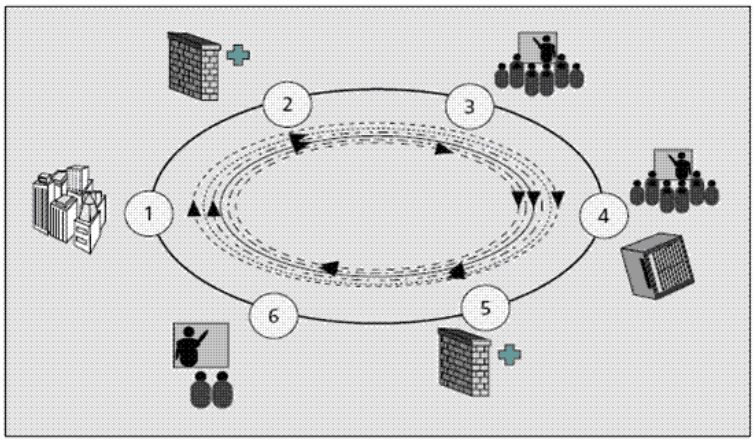


Figure 2.10 – Exemple de routage (extraite de [19])

cédemment citées. Cette solution doit utiliser le minimum de longueurs d'onde possible tout en minimisant le besoin de traitement électronique aux nœuds du réseau.

La figure 2.11 montre une meilleure solution pour l'exemple précédemment cité, où comme dans la première solution, on n'a besoin que de 2 longueurs d'onde. Par contre, on n'a pas besoin de traiter toutes les données électroniquement à chaque nœud. La longueur d'onde $\lambda 1$ (représentée en ligne continue) termine électroniquement à chaque nœud et transporte le trafic des connexions dont les nœuds sources sont 2, 3, 5 et 6 ainsi que la connexion (4, 6). La longueur d'onde $\lambda 2$ ne termine électroniquement que dans les nœuds 1 et 4, et transporte les connexions de la source 1 ainsi que les connexions (4, 1) et (4, 2) ; les connexions (1, 5) et (4, 2) sont commutés électroniquement, vers $\lambda 1$, aux nœuds 4 et 1 respectivement.

2.6 La protection de trafic dans les réseaux optiques

La protection de trafic est un mécanisme vital pour la survie des réseaux optiques. Il consiste à prévoir les ressources nécessaires pour réacheminer le trafic dérangé (pour qui la panne se produit sur son chemin de routage) en cas de panne dans le réseau. Le chemin de protection est utilisé pour recouvrir le trafic routé initialement sur le chemin

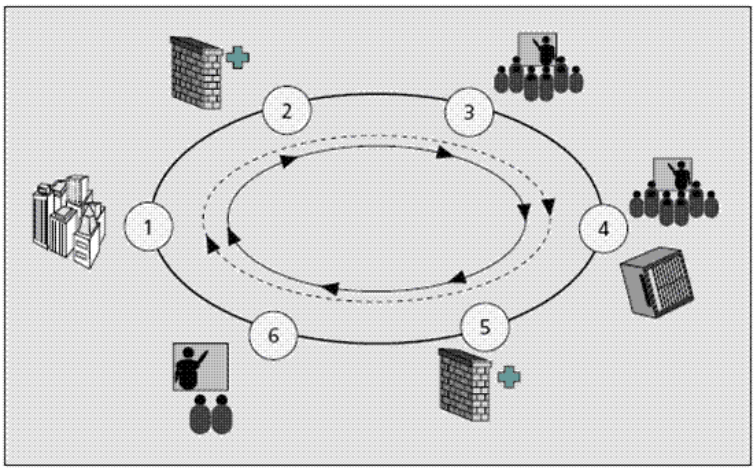


Figure 2.11 – Une meilleure solution (extraite de [19])

où la panne est survenue. Ce paradigme de protection nécessite donc plus de bande passante que ce que les connexions en cours demandent, contrairement au paradigme de restauration, qui quant à lui, réagit (en temps réel) uniquement si une panne se produit dans le réseau. Un chemin de restauration est alors recherché dans le réseau. Il n'est pas toujours garantie qu'il soit possible d'en trouver un dans les ressources libres du réseau à ce moment là. Le fait de ne pas réserver de bande passante en prévision d'une éventuelle panne rend ce paradigme plus efficace en termes de bande passante requise dans le réseau optique, cependant le temps de restauration (à ne pas confondre avec le paradigme du même nom) est plus long. Le paradigme de protection, quant à lui, offre un temps de restauration plus avantageux, et offre aussi une garantie quant à la disponibilité des ressources pour servir le trafic dérangé. Dans cette thèse on s'intéresse uniquement au paradigme de protection.

2.6.1 La protection dans les différentes couches du réseau

On peut trouver différentes techniques de protection qui opèrent au sein de différentes couches du modèle OSI (Open Systems Interconnection) d'un réseau :

- La Couche 1 (physique) : comme dans SONET/SDH, Optical Transport Network (OTN) et la couche optique (la couche WDM).

- La Couche 2 (liaison de données) : comme dans Ethernet, Risilient Packet Ring (RPR) et MPLS (qui est considéré comme appartenant à une couche entre les couches 2 et 3 du modèle OSI).

- La Couche 3 (réseau) : comme dans la couche IP.

Ceci est dû à différentes raisons. Par exemple, chaque couche peut protéger contre certains types de pannes mais probablement pas contre toutes les pannes de façon efficace [80].

Par défaut, les différents mécanismes de protection des différentes couches travaillent indépendamment les uns des autres. Lorsqu'une panne se produit, chaque couche essaie de restaurer le service simultanément, ce qui provoque un nombre important d'alarmes.

Il est alors souhaitable qu'une sorte de coordination existe entre les mécanismes de protection des différentes couches. Il est possible d'ajouter un mécanisme de priorité entre les couches, où une couche essaie de rétablir le service, avant que l'autre couche essaie à son tour. Cela peut être possible en s'assurant que le temps de restauration dans une couche est très petit comparativement à l'autre couche de telle sorte que la couche client ne s'aperçoit même pas qu'il y a eu une panne. Ceci pourrait être le cas dans un réseau IP sur WDM, dans la mesure où la couche IP peut prendre quelques secondes pour rétablir le service. Cependant, lorsqu'on a un réseau SONET en anneau sur WDM, ce mécanisme peut ne pas fonctionner dans la mesure où le service peut être rétabli très rapidement sur le réseau SONET en anneau [30]. L'autre façon de faire est d'ajouter un temps d'attente à la couche supérieure avant d'essayer de restaurer le service, mais ce temps d'attente pourrait augmenter le temps global de restauration de service [80].

2.6.2 Protection partagée et protection dédiée

Le paradigme de protection dans les réseaux optiques a deux architectures principales, la protection dédiée et la protection partagée. La protection dédiée consiste à réserver les ressources de protection nécessaires exclusivement pour chaque chemin de routage. Donc pour chaque flot de trafic, deux chemins sont requis ; un chemin de routage et un chemin de protection. On a deux protections dédiées typiques, la protection

1+1 où le trafic est initialement transmis sur deux chemins différents. Les deux signaux sont comparés au nœud destination et le meilleur est choisi [127]. L'avantage de la protection 1+1 est qu'elle a un temps de recouvrement très rapide. Dans la protection 1 :1, le chemin de protection n'est utilisé que lorsqu'une panne se produit sur le chemin de routage initial. L'avantage de la protection 1 :1 est, étant donné que le chemin de protection n'est pas utilisé, on peut router le trafic de basse priorité sur ce chemin, ce qui augmente le taux d'utilisation du réseau [125]. Ce taux est de moins de 50 % en cas de protection dédiée.

Dans la protection partagée, les ressources utilisées pour la protection peuvent être partagées par plusieurs chemins de protection, sous l'hypothèse que les chemins de routage des flots protégés ne tombent pas en panne en même temps. Cette caractéristique rend nécessaire la configuration des chemins de protection, en cas de panne, ce qui augmente le temps de recouvrement. La protection partagée, par conséquence, est inefficace en cas d'une panne multiple.

2.6.3 Protection dans les réseaux non-WDM

Dans les réseaux optiques non-WDM, deux systèmes de protection sont principalement utilisés, la commutation de protection automatique (Automatic Protection Switching APS) et l'anneau auto-réparateur (Self-Healing Ring SHR). La commutation de protection automatique est utilisée typiquement en cas de coupure de fibre, et a deux types de base : la protection 1+1 et la protection 1 :1. La protection 1 :1 peut être étendue à la protection 1 :N qui consiste à partager un lien de protection par N liens de routage primaire (working). En général, dans un système de protection M :N, N liens de routage primaire (working links) partagent M liens de protection. Dans le système d'anneau auto réparateur, comme son nom l'indique, le réseau est sous forme d'anneau. L'UPSR (Unidirectional Path-Switching Ring) et le BLSR (Bidirectional Line-Switching Ring) [32] sont les deux modèles du SHR les plus populaires. L'UPSR est basé sur une protection 1+1, deux anneaux sont utilisés et chaque flot de trafic est transmis sur les deux anneaux dans les deux directions opposées, et le nœud destinataire choisit le meilleur signal. C'est le système SHR le plus rapide, en cas de panne aucune commutation n'est

nécessaire. Dans le BLSR, deux architectures sont utilisées [127], le BLSR/2 avec deux fibres et le BLSR/4 à quatre fibres. Dans le BLSR/2, la moitié de la capacité de chaque fibre est réservée pour la protection alors que l'autre moitié est utilisée pour le routage primaire des flots de données. Lorsqu'une panne se produit, les deux nœuds adjacents à cette panne transfèrent le trafic sur les capacités réservées à la protection dans les deux fibres. Dans le BLSR/4 deux fibres sont utilisées entièrement pour le routage primaire de trafic alors que les deux autres sont réservées pour la protection. En cas de panne le nœud adjacent transfère le trafic vers les fibres de protection.

2.6.4 Protection dans les réseaux WDM

L'apparition de la couche optique (la couche WDM) est rendue possible grâce à l'utilisation des commutateurs optiques (OADM, ROADM, OXC) [67] et autres composants tout optiques dans les réseaux WDM, qui sont présentement très déployés. Cette couche supporte plusieurs services des couches supérieures telles que les connexions SONET, ATM (Asynchronous Transfer Mode) et le trafic IP [127]. Malgré que plusieurs de ces services assurent la survie de leur trafic, il est préférable d'assurer la protection dans la couche WDM, pour les raisons suivantes [14, 25, 26, 79] :

- Un temps de recouvrement plus rapide dans la couche optique que dans les couches supérieures [70].
- La couche optique assure les fonctions de survie que les couches supérieures ne peuvent assurer.
- La couche optique demande moins de coordination que les couches supérieures.
- La couche optique assure les fonctions de survie avec moins de coût.
- Un niveau supplémentaire de résilience peut être assuré par la couche optique (ex. contre les pannes multiples).
- La protection et la restauration des pannes est plus efficace dans la couche optique que dans les couches client (supérieures).

Malgré ces avantages, la couche optique ne peut pas assurer la protection contre les pannes ou les erreurs des couches supérieures, et donc une partie de la survie doit être assurée par les couches supérieures.

Les recherches se tournent de plus en plus vers les réseaux maillés (*mesh networks*) et la majorité des travaux considèrent la protection pour le cas de la panne sur un seul lien (*single-link failure*), qui est la panne la plus fréquente, comme on l'a déjà indiqué. En plus, une coupure au niveau d'une fibre entraîne l'arrêt de la transmission de données sur plusieurs longueurs d'onde en même temps, ce qui rend la protection contre les coupures des fibres très importante pour les réseaux WDM.

2.6.4.1 Protection par lien

Chaque lien du chemin d'opération (de routage) est protégé par un chemin/segment de protection (Figure 2.12(a)). En cas de panne sur ce lien, les deux nœuds d'extrémité de ce lien sont avertis de l'existence de cette panne et le trafic est alors détourné autour de ce lien sur le chemin de protection. Tous les autres liens du chemin d'opération sont encore utilisés pour router le trafic. Dans les réseaux WDM, chaque lien est composé de plusieurs longueurs d'onde. Chaque longueur d'onde d'un lien est protégée par une capacité d'une longueur d'onde sur un chemin. Les capacités utilisées pour protéger les longueurs d'onde d'un seul lien doivent être sur des chemins et/ou des longueurs d'onde différents.

2.6.4.2 Protection par chemin

Le chemin d'opération est protégé de bout-en-bout par un chemin de protection (Figure 2.12(b)). Lors d'une panne sur le chemin d'opération, les nœuds source et destination sont notifiés de la présence de cette panne et le chemin de protection est alors utilisé pour acheminer le trafic. Dans ce paradigme de protection, les chemins d'opération et de protection doivent être mutuellement disjoints. Dans le cas d'un panne simple de lien, le chemin de protection est utilisé pour protéger le chemin d'opération contre la panne de n'importe quel lien sur le chemin optique. L'identification de ce lien n'est pas néces-

saire pour lancer l'opération de recouvrement de trafic [127]. La protection par chemin partagé ou Shared Backup Path Protection (SBPP) est le mécanisme de protection par chemin le plus populaire et qui a reçu beaucoup d'attentions de la part des chercheurs sur la protection [58, 94, 100].

2.6.4.3 Protection par segment

La protection par lien offre généralement un temps de recouvrement plus rapide. Cependant, elle utilise plus de ressources à cause de la nécessité de protéger chaque lien. De plus, il faut noter que la protection par lien n'offre pas de protection contre les pannes d'un nœud du fait que le chemin de protection et le lien protégé partagent les mêmes nœuds aux extrémités. La protection par chemin, quant à elle offre cette possibilité sauf pour les nœuds source et destination, qui sont protégés par dédoublement d'équipements.

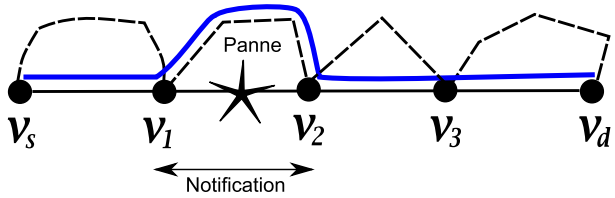
La protection par segment est une solution intermédiaire et un concept généralisé des deux premiers. Il consiste à diviser le chemin d'opération en plusieurs segments, qui se chevauchent ou non, et à protéger chaque segment séparément (Figure 2.12(c)). Le chevauchement des segments, introduite dans [39] et développée dans [43, 44, 98] permet de protéger tous les nœuds intermédiaires. En cas de panne, seulement le segment affecté utilise son chemin de protection pour acheminer le trafic, les autres segments routent le trafic sur leurs chemins de routage habituels (Figure 2.12(d)).

2.6.5 Protection par *p*-cycles

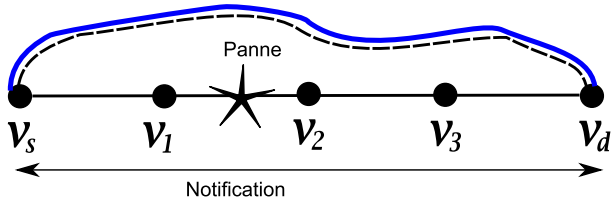
Un des principaux mécanismes de protection destiné aux réseaux optiques est la protection par cycles préconfigurés et pré-connectés ou *p*-cycles. Le concept de *p*-cycle a été introduit en 1998 par Grover et Stamatelakis [37].

2.6.5.1 Les *p*-cycles protégeant les liens

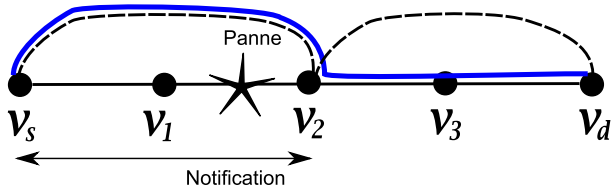
L'idée de base des *p*-cycles est inspirée de la protection en anneau, mais à la différence que les *p*-cycles ne protègent pas uniquement les liens constituant l'anneau, mais



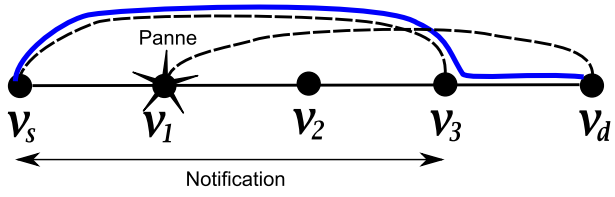
(a) Protection par lien



(b) Protection par chemin



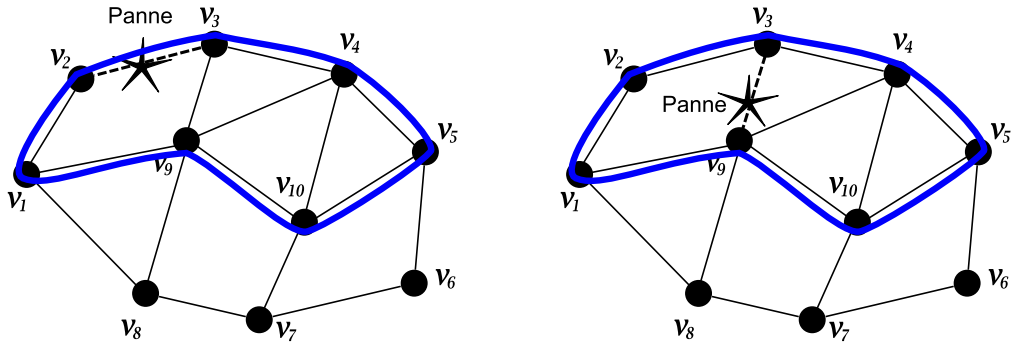
(c) Protection par segments



(d) Protection par segments se chevauchant

Figure 2.12 – Les modèles de la protection maillée

protègent aussi les liens cordes [1] (*straddling links*). Ce qui rend l'utilisation de la capacité de protection beaucoup plus efficace que les anneaux mais qui garde un temps de recouvrement similaire à celui des anneaux [38]. En cas de panne sur un lien, seulement les deux nœuds aux extrémités de ce lien sont reconfigurés. Aucune autre opération n'est nécessaire sur les autres nœuds du cycle.



(a) Panne sur un lien du cycles (b) Panne sur une corde

Figure 2.13 – Protection par *p*-cycle

La figure 2.13 décrit un exemple montrant le fonctionnement de la protection par *p*-cycle. Un même *p*-cycle est montré dans les figures 2.13(a) et 2.13(b) et dispose d'une capacité égale à une unité de bande passante. Dans la figure 2.13(a), un lien $v_2 - v_3$ tombe en panne. Le trafic est alors routé sur la partie encore fonctionnelle du cycle. Dans la figure 2.13(b), le lien $v_3 - v_9$ tombe en panne, le trafic est alors routé soit sur la partie $v_3 - v_4 - v_5 - v_{10} - v_9$, soit sur la partie $v_3 - v_2 - v_1 - v_9$. Chaque unité de capacité d'un *p*-cycle protège deux unités de capacité d'un lien corde de ce *p*-cycle.

Les *p*-cycles de base qu'on vient de décrire protègent les liens contre les coupures de fibre. Le concept de *p*-cycles a été généralisé pour d'autres formes de protection (voir [32, 33]). Dans les deux prochaines sections on décrit deux de ces techniques, les *p*-cycles protégeant les segments (*Flow p*-cycles) et les *p*-cycles protégeant le trafic de bout-en-bout (FIPP *p*-cycles). Dans le reste de la thèse, le terme "*protection par p-cycles*" sera utilisé pour désigner la protection par *p*-cycles protégeant les liens.

[1] une corde est un lien qui n'appartient pas au cycle et qui relie deux nœuds de ce cycle

2.6.5.2 Protection par *p*-cycles segmentés

Les *p*-cycles segmentés (Flow *p*-cycles) ont été introduits par Grover et Shen [36, 93] pour étendre la protection offerte par les *p*-cycles aux segments de type cordes (*straddling flows*). Un segment est défini comme une seule portion contigu d'un chemin d'opération entre deux nœuds. Ce qui fait que chaque lien sur le chemin d'opération, une séquence de liens sur le chemin d'opération et le chemin d'opération en entier peuvent être considéré comme segment. Dans l'exemple de la figure 2.14, on considère un che-

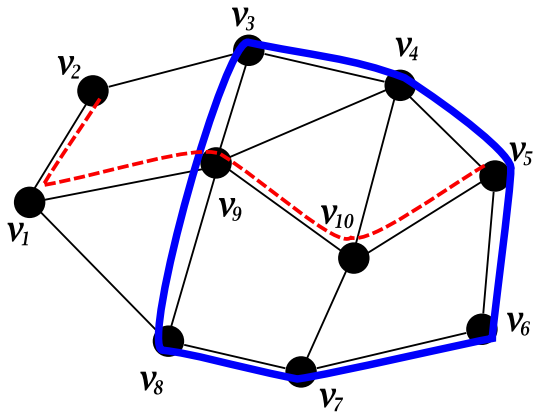


Figure 2.14 – Exemple d'un *p*-cycle segmenté

min d'opération $v_2 - v_1 - v_9 - v_{10} - v_5$ et un cycle. Dans le cas d'une panne sur les liens $v_9 - v_{10}$ et $v_{10} - v_5$, le trafic ne peut être protégé par le cycle si c'est un *p*-cycle de base. Cependant, si le cycle est considéré comme *p*-cycle segmenté, le segment $v_9 - v_{10} - v_5$ peut être protégé par deux chemins alternatifs $v_9 - v_3 - v_4 - v_9 - v_5$ et $v_9 - v_8 - v_7 - v_6 - v_5$. En plus, en cas de panne sur le nœud v_{10}, le trafic transitant par ce nœud peut être protégé par ce cycle. À noter cependant que n'importe quel trafic qui est ajouté ou retranché par le nœud en question v_{10} ne peut être protégé par ce cycle. Voir ex. [53, 93] pour plus de détails.

2.6.5.3 Protection par FIPP *p*-cycles

Le concept de FIPP *p*-cycles a été introduit (par Kodian et Grover [59]) pour assurer la protection d'un chemin d'opération de bout-en-bout. Les deux nœuds du chemin

protégé doivent appartenir au cycle de protection. Le cycle de protection peut être utilisé pour protéger plusieurs chemins d'opération si :

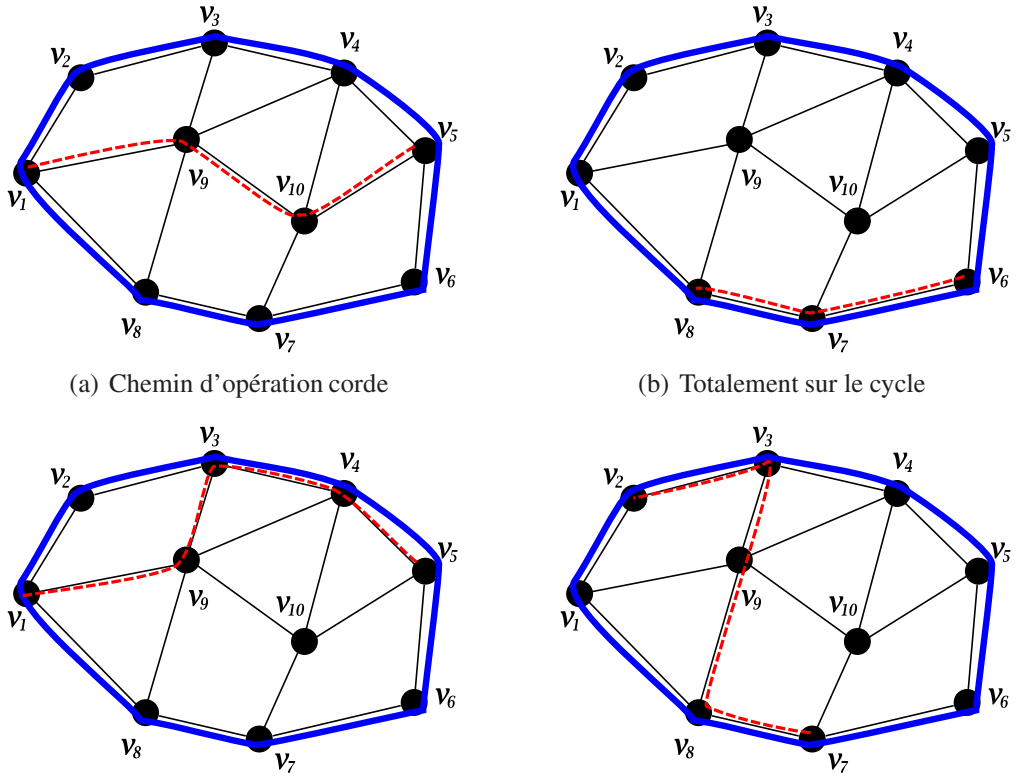
1. Ces chemins d'opération sont mutuellement disjoints, ou sinon
2. Leurs chemins de protection sont mutuellement disjoints.

Notons ici que si les chemins sont disjoints par rapport aux liens, les chemins sont protégés contre les pannes simples de liens. Si les chemins sont disjoints par les nœuds, les chemins sont protégés contre les pannes simples de nœuds.

Les FIPP *p*-cycles protègent les chemins de routage qui s'appuient sur le cycle, c'est-à-dire tous les liens ou une partie des liens du chemin sont sur le cycle. Ils protègent également les chemins de routage qui sont des chemins cordes, c'est-à-dire tels que leurs nœuds d'extrémités sont sur le cycle mais aucun des liens constituant le chemin n'appartient au cycle. Les propriétés des FIPP *p*-cycles sont présentées dans [59] et peuvent être résumées dans les points suivants :

- ✓ Seulement les deux nœuds aux extrémités du chemin sont manipulés en temps réel pour commuter le trafic lors d'une panne sur le chemin d'opération.
- ✓ Les chemins de protection sont cross-connectés à l'avance, ce qui assure leur fonctionnement en cas de panne.
- ✓ La commutation lors de la protection est contrôlée entièrement par le nœud à l'extrémité et elle est totalement indépendante de l'endroit où la panne s'est produite sur le chemin d'opération.
- ✓ Les chemins qui sont des cordes du *p*-cycle peuvent supporter deux chemins d'opérations qui sont protégés par une seule unité de capacité de ce *p*-cycle.
- ✓ Les chemins de protection sont connus à l'avance. On peut limiter la longueur de ces chemins par la limitation de la taille des cycles de protection.
- ✓ La protection des nœuds est faisable si les chemins d'opération sont disjoints en termes de nœuds (et donc de liens). On peut relaxer cette condition si on veut seulement protéger les chemins contre les pannes de liens.

Les relations de protection qui existent entre un FIPP p-cycle et un chemin d'opération protégé par ce cycle peuvent être classées dans deux catégories différentes. Dans la première, le chemin d'opération est un chemin corde par rapport au cycle comme dans la figure 2.15(a). Dans ce cas, deux chemins de protection sont disponibles, et par conséquent, deux chemins d'opération sur le chemin $v_1 - v_9 - v_{10} - v_5$ peuvent être protégés par ce cycle, et à chacun, un chemin de protection est alors désigné. Lorsqu'une panne se produit sur le chemin d'opération, les nœuds aux extrémités commutent le trafic de chaque chemin d'opération sur le chemin de protection qui lui a été affecté.



(a) Chemin d'opération corde

(b) Totalement sur le cycle

(c) Partiellement sur le cycle

(d) Non disjoint du chemin de protection

Figure 2.15 – Les relations du chemin d'opération avec un FIPP p-cycle

Dans la deuxième catégorie on trouve différentes situations, la première est mentionnée dans la figure 2.15(b). Le chemin d'opération $v_6 - v_7 - v_8$ est totalement sur le cycle et son chemin de protection est le complément de ce chemin sur le cycle, à savoir

$v_8 - v_1 - v_2 - v_3 - v_4 - v_5 - v_6$. Dans la deuxième situation, le chemin d'opération est partiellement sur le cycle. Dans ce cas, deux types de relations entre le chemin d'opération et le chemin de protection peuvent exister. Le premier type est représenté dans la figure 2.15(c), le chemin d'opération $v_1 - v_9 - v_3 - v_4 - v_5$ est totalement disjoint de son chemin de protection $v_1 - v_8 - v_7 - v_6 - v_5$. Dans le deuxième type représenté dans la figure 2.15(d), il n'existe aucun chemin de protection possible qui est complètement disjoint du chemin d'opération. Dans un tel cas, les deux chemins de protection doivent être considérés. Il est évident que si la panne est sur un lien de la partie $v_3 - v_9 - v_8$, les deux chemins de protection $v_2 - v_1 - v_8 - v_7$ et $v_2 - v_3 - v_4 - v_5 - v_6 - v_7$ peuvent être utilisés. Par contre si le lien qui tombe en panne est $v_2 - v_3$, le chemin de protection doit être $v_2 - v_1 - v_8 - v_7$, sinon si c'est $v_7 - v_8$ le chemin de protection doit être $v_2 - v_3 - v_4 - v_5 - v_6 - v_7$. Le choix du chemin de protection dans ce dernier type (figure 2.15(d)), peut être résumé de la façon suivante : un chemin de protection est désigné comme chemin de protection par défaut (par exemple $v_2 - v_1 - v_8 - v_7$). Une règle est ensuite appliquée dans les nœuds aux extrémités, cette règle dit que si le lien qui tombe en panne est sur le chemin de protection par défaut (le lien $v_7 - v_8$ dans ce cas), l'autre chemin de protection est utilisé ($v_2 - v_3 - v_4 - v_5 - v_6 - v_7$ dans ce cas).

CHAPITRE 3

REVUE DE LA LITTÉRATURE

Ce chapitre présente une revue de la littérature sur les deux thèmes de cette thèse. Dans la première partie, on présente un aperçu des travaux sur le problème de groupage dans les réseaux optiques et particulièrement dans les réseaux maillés. La section 3.1.1 présente un aperçu des travaux réalisés dans un contexte de trafic statique alors que dans la section 3.1.2 on trouve ceux réalisés dans un contexte de trafic dynamique. Dans la section 3.1.3, on présente une récapitulation des principaux travaux sur le problème de groupage de trafic.

La deuxième partie présente un aperçu des techniques de protection en utilisant les *p*-cycles. La section 3.2.1, présente les travaux sur les *p*-cycles protégeant les liens, alors que dans la section 3.2.2, on trouve un aperçu des recherches faites sur le concept de FIPP *p*-cycles.

Pour des connaissances plus approfondies sur le problème de groupage de trafic, le lecteur peut se référer à [18, 71], et à [32, 35] pour plus de détails sur la survie des réseaux optiques en général et le concept de *p*-cycles en particulier.

3.1 Groupage de trafic

La plupart des recherches effectués sur le problème de groupage de trafic dans les réseaux optiques, considèrent un type particulier de réseaux qui est le réseau en anneau où chaque nœud est relié à deux autres nœuds du réseau. Le modèle de trafic considéré dans la majorité des cas est le modèle statique et l'objectif à atteindre dans ce type de réseau est généralement de minimiser le nombre d'équipements de type ADM [128]. Parmi ces recherches nous pouvons citer :

- Wang *et al.* [104] ont formulé le problème du groupage de trafic dans un réseau en anneau (ring) à l'aide d'un modèle de programmation linéaire en nombres entiers (PLNE), où la fonction objective est de minimiser le nombre des équipements

ADM. L'étude considère que le trafic est statique et explore les deux types d'anneaux, à savoir unidirectionnels et bidirectionnels. Les auteurs ont proposé une relaxation de quelques contraintes pour obtenir des résultats sur un réseau réel. Des heuristiques sont utilisées pour résoudre le problème sur un grand réseau.

- Gerstel *et al.* [27] ont fait une étude de design du réseau optique en anneau en utilisant des ADM optiques (OADM). Le but est de minimiser le coût total du réseau qui inclut le coût des équipements de transmission et de traitement électronique des données et le nombre des longueurs d'onde utilisées. Une comparaison entre différents types d'architectures est faite.

- Dans l'étude de Xin *et al.* [111], on propose un design du réseau en anneau, où le problème de groupage est résolu en plusieurs étapes, chacune ayant une fonction objective différente. Les requêtes de connexions sont groupées dans des anneaux virtuels, avec certains objectifs tels que minimiser le nombre des équipements qui font le traitement électronique des données (ex. DXC : Digital Cross Connect), de telle sorte que chaque anneau virtuel occupe une longueur d'onde. La topologie virtuelle est donc composée des anneaux virtuels résultants. En supposant que le prix des DXCs est le prix dominant dans le réseau, le premier objectif est de minimiser le nombre total de ces équipements. Le problème de la reconfiguration des anneaux virtuels est aussi considéré dans cette étude.

Quelques études ont été faites sur un trafic dynamique. L'étude de Berry et Modiano [8] propose plusieurs méthodes pour minimiser le nombre total des équipements de type ADM utilisés dans le réseau tout en acceptant tout le trafic injecté. Les auteurs ont considéré un cas particulier de trafic où on a une limite sur le nombre de connexions qui partent de chaque nœud. Une borne inférieure sur le nombre des ADMs nécessaires et quelques conditions, pour supporter ce type de trafic, sont aussi données. Des algorithmes pour affecter les circuits et le placement des équipements de type ADM sont proposés.

Dans les réseaux maillés, pour résoudre le problème du GRWA, qui peut être divisé en deux sous problèmes comme on a vu dans la section 2.5, on a deux possiblités :

1. Trouver la solution optimale pour chaque sous problème, ce qui rend le problème GRWA plus facile à résoudre, surtout lorsqu'on sait que plusieurs travaux ont été faits pour résoudre chacun des sous problèmes. Malheureusement, cette méthode ne donne que très rarement la solution optimale [128], même si la solution de chaque sous problème est optimale, car les sous problèmes ne sont pas nécessairement indépendants.

2. Consider les sous problèmes comme un seul problème, mais le problème devient évidemment très grand. Une solution optimale peut être trouvée pour un certain type de trafic (statique) et pour un petit réseau [128].

Au cours des cinq dernières années, de plus en plus de chercheurs se sont tournés vers les réseaux maillés (mesh), où les nœuds sont reliés entre eux de façon quelconque (un nœud peut être relié à plusieurs nœuds du réseau). Dans les deux sections suivantes 3.1.1 et 3.1.2, on présentera une sélection de recherches faites sur le groupage de trafic dans les réseaux maillés.

3.1.1 Groupage avec un trafic statique

Plusieurs chercheurs ont travaillé sur le problème de groupage en présence d'un trafic statique. Fang et Somani [22] ont proposé une formulation en PLNE avec pour fonction objective la minimisation du nombre des émetteurs et récepteurs dans le réseau (équivalent à minimiser le nombre de chemins optiques). La complexité de cette formulation est réduite par l'ajout de quelques contraintes comme la longueur des routes (c'est-à-dire, le nombre de routes possibles est réduit). À cause de la complexité de la formulation, on ne peut trouver une solution optimale dans des réseaux réels, des heuristiques ont été développées pour trouver des bonnes solutions au problème.

Chen *et al.*[10] ont considéré le problème de groupage de trafic dans un réseau maillé avec un trafic statique. Ils ont proposé un cadre hiérarchique pour le groupage avec l'objectif de minimiser le nombre de ports dans les nœuds. Premièrement, le réseau est divisé en classes (clusters en anglais) où chaque classe est un ensemble de nœuds voisins. Chaque classe est vue comme un réseau étoile virtuel. À chaque classe on désigne un

nœud *hub* pour grouper le trafic. Deuxièmement, les nœuds *hubs* désignés dans le niveau 1 forment à leur tour une classe, et là encore un nœud *hub* est désigné pour ce deuxième niveau. L'idée derrière cette stratégie est de résoudre le sous problème 1 du problème de groupage de trafic (section 2.5) en deux étapes. Dans la première étape, on applique un algorithme sur chaque classe et le résultat est un ensemble de chemins optiques dans chaque classe. Dans la deuxième étape, on applique le même algorithme sur les *hubs* pour déterminer les chemins optiques entre les classes. Finalement, on termine avec la résolution du sous problème 2 (RWA) sur les chemins optiques trouvés lors des deux étapes précédentes. Les auteurs ont utilisé un réseau de 32 nœuds pour tester cette approche, et ont remarqué que lorsque le nombre de classes augmente, le nombre total de chemins optiques dans le réseau augmente, par contre le nombre de longueurs d'onde nécessaires diminue ainsi que la longueur moyenne des chemins optiques.

Dans l'étude de Jaumard *et al.* [51], on a proposé des heuristiques pour résoudre le problème de groupage avec un trafic statique. Le but est de minimiser le coût des équipements MSPP (Multi-Service Provisioning Platform). Un MSPP est un équipement utilisé pour grouper le trafic des clients à bas débit (e.g. OC-3 et OC-12) pour le transformer en des signaux de haut débit (e.g. OC-48 et OC-192) en utilisant le multiplexage temporel TDM. L'heuristique proposée peut être appliquée sur un réseau en anneau ou sur un réseau maillé. Des expériences ont été réalisées sur un réseau en anneau ainsi que sur les réseaux NSFNET et EONET.

Resendo *et al.* [81] ont proposé des formulations sous forme de PLNE au problème de groupage de trafic dans un réseau maillé avec un trafic statique. La première formulation a comme fonction objective de minimiser le nombre total des émetteurs/récepteurs dans le réseau. Une relaxation en Min-Max de la première fonction objective donne une deuxième fonction objective. Cette deuxième fonction objective minimise le nombre des émetteurs/récepteurs dans le nœud qui nécessite le plus grand nombre de ces équipements. Cette relaxation permet de trouver une solution au problème de groupage de trafic dans un réseau de 16 nœuds (ce qui n'est pas le cas pour la première formulation), mais cette solution est très coûteuse en termes de nombre des équipements d'émission/réception (et qui devrait être optimisé !).

3.1.2 Groupage avec un trafic dynamique

Parmi les travaux qui ont été faits sur le groupage de trafic (grooming) dans un réseau général (maillé), on trouve le travail de Zhu *et al.* [131]. Les auteurs ont proposé quelques éléments de *traffic engineering* qui doivent être pris en considération dans ce type de réseaux. Les auteurs ont proposé une extension, pour ce type de réseaux, de leur modèle de graphe, proposé dans [128].

Xin et Bang [109] ont formulé le problème de groupage de trafic dans un réseau maillé avec un trafic dynamique sous forme d'un PLNE. Cependant, pour minimiser la complexité du problème, on ne considère que le *single-hop grooming*, c'est-à-dire qu'une connexion entre une source v_s et une destination v_d ne doit parcourir qu'un seul chemin optique entre v_s et v_d. Quatre formulations sont données. Dans la première, la fonction objective est de minimiser le nombre de ports (émetteurs et récepteurs) dans le réseau. Les trois autres formulations consistent à minimiser la somme des nombres de longueurs d'onde dans chaque lien du réseau, en considérant (*i*) qu'on a pas de conversion de longueurs d'onde, (*ii*) une conversion totale des longueurs d'onde et (*iii*) une conversion partielle des longueurs d'onde (quelques nœuds ont la capacité de conversion totale de longueurs d'onde). Des heuristiques ont été proposées pour les grands réseaux.

Xin *et al.* [110] ont proposé une nouvelle façon de résoudre le problème de groupage de trafic dans un réseau maillé avec un trafic dynamique. Au lieu d'établir et d'enlever des chemins optiques selon les besoins du trafic dynamique, ce qui peut être très fréquent, on fait un design du réseau pour un trafic statique connu *a priori* et après on route les connexions qui arrivent au fur et à mesure. Deux problèmes sont alors considérés, le premier est de minimiser l'utilisation des ressources en tenant compte du blocage des connexions (ex. le taux de blocage des connexions entre une source v_s et une destination v_d ne doit pas dépasser une certaine valeur), et le deuxième problème est celui de maximiser les performances du réseau (acceptation des connexions) en tenant compte des ressources disponibles. Ces problèmes sont formulés sous forme de PLNE. Les résultats obtenus montrent que l'utilisation des ressources diminue considérablement lorsque

la contrainte de blocage est relaxée et les performances de groupage augmentent doucement quand on ajoute plus de ressources. En plus, le nombre de ports (récepteurs) dans les nœuds destinations (clients) a plus d'impact sur le groupage que le nombre de longueurs d'onde.

Yao *et al.* [115] ont étudié le problème de groupage de trafic dans un réseau maillé (mesh) où seulement quelques nœuds ont la capacité de grouper le trafic (appelés nœuds G). Les auteurs ont proposé un algorithme pour supporter un trafic dynamique dans un tel réseau en sachant qu'on a quatre types de chemins optiques selon la disponibilité ou non, aux extrémités du chemin optique, d'un nœud ayant la capacité de grouper le trafic (nœud G). Le but de l'algorithme est de minimiser le nombre de connexions non satisfaites. Des simulations ont été faites sur le réseau NSFNet avec un placement aléatoire des nœuds G. Parmi les résultats montrés, il faut mentionner l'effet du nombre des nœuds G sur le nombre de connexions non satisfaites. À part le nombre de nœuds G, d'autres facteurs peuvent influencer le taux de blocage tels que la topologie du réseau et le placement des nœuds G eux mêmes.

Sivakumar *et al.* [95] ont travaillé sur le problème de groupage de trafic dans un réseau maillé où le groupage du trafic peut se faire à chaque nœud, mais il y a seulement quelques ports de chaque nœud qui ont la capacité de grouper le trafic, c'est-à-dire, le groupage ne peut se faire que sur quelques longueurs d'onde parmi celles disponibles à partir de chacun des nœuds. Quelques architectures et politiques de groupage dans les nœuds ont été présentées. Les auteurs ont étudié l'effet de certains facteurs tels que la granularité des connexions, le nombre de ports dans chaque nœud et la conversion de longueurs d'onde. Les résultats obtenus montrent que les performances données par ce type de réseau peuvent être comparables à celle des réseaux où le groupage peut se faire dans n'importe quel nœud et sur n'importe quel port, et avec moins de coût.

Yao *et al.* [114] considèrent une technique de *reroutage* des connexions en cours. Cette technique essaie de trouver un autre chemin pour des connexions en cours, pour pouvoir router de nouvelles connexions. Ils proposent deux versions différentes : (*i*) RRAL qui consiste à *rerouter* tout un chemin optique sur une autre route ; (*ii*) RRAC qui permet de *rerouter* seulement une connexion parmi celles en cours. Des simulations

sur deux réseaux différents montrent qu'on peut réduire le taux de blocage en utilisant la technique considérée.

3.1.3 Récapitulation des travaux sur le groupage de trafic

Nous résumons les travaux effectués sur le problème de groupage de trafic dans les réseaux optiques dans le tableau 3.I. Plusieurs travaux ont été réalisés sur ce problème dans les réseaux en anneau (ring). Dans les dernières années, de plus en plus de chercheurs se tournent vers les réseaux maillés dans les deux cas de trafic, statique et dynamique.

Trafic / **Réseau**	***Statique***			***Dynamique***		
Anneau	Wang *et al.* [104] Gerstel *et al.* [27] Xin *et al.* [111]			Berry et Modiano [8]		
Maillé	**Auteurs**	PLNE	Heur.	Auteurs	PLNE	Heur.
	Fang et Somani [22]	✓	✓	Zhu *et al.* [131]	-	✓
	Chen *et al.* [10]	-	✓	Xin *et al.* [109]	✓	✓
	Resendo *et al.* [81]	✓	-	Xin *et al.* [110]	✓	✓
	Jaumard *et al.* [51]	-	✓	Yao *et al.* [114, 115]	-	✓
Fonction objective						
Nombre de ports	Xin et Bang [109], Fang et Somani [22], Xin *et al.* [110] Resendo *et al.* [81], Konda *et al.* [61]					
Taux de blocage	Yao *et al.* [115]					

Tableau 3.I – Tableau récapitulatif des travaux sur le groupage de trafic

3.2 Protection en utilisant les *p*-cycles

La majorité des recherches sur la protection en utilisant les *p*-cycles considère un trafic statique, où l'état du réseau et le routage du trafic sont connus. Le but est de trouver le meilleur ensemble de *p*-cycles qui protège le trafic du réseau, dépendamment de l'objectif à atteindre. La plupart des travaux sur les *p*-cycles, dans un contexte de trafic statique, proposent l'énumération des *p*-cycles candidats [15, 21, 34, 65, 74, 93, 120, 124]

et appliquent ensuite une méthode (PLNE ou autre) pour sélectionner les *p*-cycles de protection parmi ces *p*-cycles candidats. Certains travaux énumérent tous les *p*-cycles candidats comme [34, 65, 93, 124] et passent ensuite par une phase de pré-selection de *p*-cycles ou utilisent directement une stratégie ou une métrique pour trouver les *p*-cycles de protection parmi les *p*-cycles candidats. L'utilisation d'un PLNE pour trouver les *p*-cycles de protection parmi les *p*-cycles candidats a aussi été proposée dans [74]. D'autres solutions ont été proposées pour trouver un ensemble de *p*-cycles candidats sans être obligé d'énumérer tous les *p*-cycles possibles en utilisant des algorithmes de génération de *p*-cycles [15, 120].

On trouve aussi dans la littératures des travaux qui utilisent des stratégies qui considèrent implicitement tous les cycles possibles afin d'éviter les inconvénients de l'énumération explicite des cycles, comme dans [3, 87, 97, 107].

Quelques recherches ont été faites dans un contexte de trafic dynamique où on est en présence d'un changement continuel du trafic du réseau. Dans les deux sections suivantes 3.2.1 et 3.2.2, on présentera une sélection de recherches faites avec un trafic dynamique, d'abord sur la protection par *p*-cycles, puis sur la protection par FIPP *p*-cycles. Une section qui résume de ces travaux, sera présenté à la fin de cette partie.

3.2.1 Protection par *p*-cycles

Comme pour le trafic statique, deux approches peuvent être utilisées pour résoudre le problème de protection par *p*-cycles, dans un contexte de trafic dynamique. La première approche consiste à énumérer les *p*-cycles candidats au préalable. L'énumération préalable des *p*-cycles de protection consiste à trouver soit tous les *p*-cycles possibles dans un réseau, ou un ensemble de cycles prometteurs. Le nombre de *p*-cycles possibles dans un réseau augmente de façon exponentielle avec la taille du réseau, ce qui représente l'inconvénient majeur de cette stratégie.

Dans le travail de He *et al.* [41], un PLNE avec une fonction objective qui minimise la longueur totale des *p*-cycles est utilisé pour sélectionner un ensemble de *p*-cycles de telle sorte que chaque lien du réseau soit, au moins, sur un *p*-cycle ou sur la corde d'un *p*-cycle. La moitié de la capacité de chaque lien sur les *p*-cycles est réservée pour la

protection alors que le reste peut être utilisée pour le routage des connexions. Chaque nouvelle requête de connexion est routée sur un chemin qu'on doit calculer de telle sorte que la connexion soit protégée par les *p*-cycles trouvés au début. Trois stratégies ont été proposées pour router les nouvelles connexions. La première consiste à router la connexion sur le plus court chemin de la source à la destination. La deuxième stratégie consiste à utiliser le chemin le moins occupé parmi trois chemins entre la source et la destination et la troisième stratégie consiste à choisir le chemin le *plus libre*. Les performances des trois stratégies de routage ont été comparées par simulation.

Eshoul et Mouftah [20] ont proposé deux stratégies de routage et d'affectation de longueurs d'onde pour les nouvelles requêtes de connexion. La première stratégie (*OPP*) consiste à trouver un plus court chemin sur chaque longueur d'onde de la source à la destination. Pour chaque chemin possible, il faut trouver les *p*-cycles qui le protègent. La combinaison (chemin de routage, *p*-cycles de protection) qui a le coût le plus petit est considéré. Dans la deuxième stratégie (*BARWA*), le chemin de routage d'une nouvelle requête est calculé après avoir *libéré* la capacité de protection. Ceci permet de trouver un chemin pour une requête qui peut être bloquée autrement. Cependant, lors de cette opération, on doit s'assurer que le fait de router cette requête n'empêche pas la protection des connexions déjà en cours. Ces deux stratégies de routage on été utilisées avec deux stratégies de protection. La première (*OLPC*) consiste à utiliser le même PLNE que dans [41] pour trouver un ensemble de *p*-cycles pour protéger tous les liens du réseau, au préalable, alors que la deuxième (*RRR*) consiste à réoptimiser les *p*-cycles de protection à chaque fois qu'une connexion est bloquée. La simulation montre que l'utilisation du *BARWA* avec *RRR* donne les meilleurs résultats en terme de blocage de connexions. Cependant l'utilisation du *OLPC* est intéressante dans un trafic hautement dynamique.

Les chercheurs dans [126] utilisent une adaptation au trafic dynamique de la méthode heuristique *ER-Based Unity-p-cycle* [124] proposée pour un trafic statique. Pour chaque requête de connexion qui n'est pas protégée par les *p*-cycles déjà existants, on calcule tous les *p*-cycles candidats en se basant sur l'algorithme en [49]. On sélectionne parmi ces *p*-cycles candidats, celui dont la valeur du facteur ER (*Efficiency Ratio*) est la plus grande et on répète cette étape jusqu'à ce que la connexion soit protégée.

Dans l'article de Ruan et Tang [84], les auteurs calculent au début un ensemble de *p*-cycles (*offline*) de tel sorte que chaque lien du réseau soit protégé par au moins un *p*-cycle. Ils désignent ensuite, pour chaque lien du réseau, un *p*-cycle "primaire" *pimary p-cycle* qui le protège parmi l'ensemble de *p*-cycles calculé précédemment. Le choix du chemin de routage de la connexion quant à lui, se fait en temps réel (*online*) au moment où la connexion arrive, pour exploiter le fait que les liens de ce chemin peuvent déjà être protégés par des *p*-cycles existants. Si un lien du chemin de routage n'est pas protégé, on créé une copie du *p*-cycle "primaire" qu'on a désigné pour le protéger.

L'inconvénient majeur de cette approche est que le nombre de *p*-cycles possibles dans un réseau est très grand et seul le problème de protection dans des réseaux de petite taille peut être résolu de façon optimale. Si par contre, seulement un ensemble de *p*-cycles prometteurs est considéré, cela compromettra la qualité de la solution de façon significative surtout dans un contexte de trafic dynamique.

La deuxième approche pour résoudre le problème de protection par *p*-cycles, dans un contexte de trafic dynamique, consiste à générer un ou quelques *p*-cycles au besoin, c'est à dire, lorsqu'une nouvelle connexion est routé sur un chemin et que certains liens de ce chemin ne sont pas (suffisamment) protégés. À notre connaissance, dans un contexte de trafic dynamique, aucun travail utilisant cette approche n'est disponible dans la littérature à ce jour.

3.2.2 Protection par FIPP *p*-cycles

Généralement deux approches sont considérés pour le design des FIPP *p*-cycles. La première consiste à définir des ensembles de chemins d'opération disjoints et ensuite à définir un FIPP *p*-cycle pour protéger chaque ensemble de telle sorte que chaque connexion est protégée par au moins un *p*-cycle. La deuxième approche quant à elle consiste à trouver d'abord un ensemble S de *p*-cycles candidats, et ensuite à trouver, pour chaque *p*-cycle de S, un ensemble de chemins d'opération qui peuvent être protégés par ce cycle.

Dans [59], Kodian et Grover ont proposé une formulation linéaire en nombres entiers (PLNE) du problème. Le modèle considéré prend en entrée les chemins d'opération des

connexions dans le réseau et un ensemble de cycles candidats. Avec le nombre très élevé de variables et de contraintes du modèle, seulement un petit ensemble de cycles candidats doit être considéré pour pouvoir obtenir une solution.

L'article [60] est basé sur la première approche pour le design des FIPP *p*-cycles citée plus haut. Un algorithme trouve les ensembles de chemins d'opération disjoints parmi les chemins d'opération des connexions. Une limite est considérée sur le nombre de chemins qui composent chaque ensemble et on s'assure que chaque chemin d'opération est inclus dans au moins un nombre donné de ces ensembles (dix ensembles dans cet article). Dix cycles admissibles sont trouvés pour protéger chaque ensemble. Ensuite, un modèle de PLNE, qui prend comme données les ensembles de chemins disjoints trouvés et l'ensemble de cycles pour chaque ensemble, est considéré pour résoudre le problème de design des FIPP *p*-cycles.

Le travail dans [4] peut être considéré comme une extension de la méthode présenté dans [60]. On considère l'optimisation conjointe du routage et de la protection. Les N plus courts chemins d'opération sont calculés pour chaque connexion, et les ensembles des chemins disjoints sont ensuite trouvés. Un modèle de PLNE est utilisé pour résoudre le problème, comme dans [60].

Une optimisation conjointe est aussi proposée dans [23]. La méthode est basée sur l'énumération d'un ensemble de cycles candidats et pour chaque cycle candidat on trouve un ou plusieurs ensembles de chemins d'opération disjoints qui peuvent être protégés par ce cycle. Une métrique est utilisée pour choisir les meilleurs combinaisons de cycles et de chemins protégés.

Zhang et Zhong [119] ont proposé une nouvelle heuristique pour le design des FIPP *p*-cycles. Elle consiste à utiliser une métrique pour sélectionner des cycles parmi une liste déjà calculée. À chaque cycle sélectionné, on associe ensuite un ensemble de chemins disjoints qui peuvent être protégés. L'opération se termine lorsque tous les chemins d'opération sont protégés.

Les travaux cités plus hauts ont trouvé que la réduction de capacité de protection par rapport à la protection par *p*-cycles de base, est de 6 % dans [60] et varie entre 20 % et 100 % dans [119]. Cependant, cette capacité n'est pas très loin de celle utilisée par

la protection par chemin partagé SBPP. Elle est plus grande de 10 % à 18 % dans [60] et de 47 % dans le pire des cas dans [59]. Somani affirme dans le livre [18] chapitre 10, que la protection par chemins partagés (SBPP) utilisent moins de capacité pour la protection que dans les protections utilisant les *p*-cycles, dans les réseaux avec un petit degré de connectivité, mais que ces techniques sont comparables en termes de capacité de protection dans les réseaux de grande connectivité.

Les travaux sur les FIPP *p*-cycles qu'on trouve dans la littérature considèrent uniquement le trafic statique. À notre connaissance, aucune étude dans le contexte d'un trafic dynamique n'a été faite, à ce jour.

3.2.3 Récapitulation des travaux sur les *p*-cycles

La majorité des travaux de la littérature concernant la protection en utilisant les *p*-cycles considère un trafic statique. Des travaux qui considèrent un trafic dynamique ont été faits sur la protection par *p*-cycles de base. Ces travaux utilisent l'énumération préalable de tous les cycles ou d'une partie des cycles du réseau ou encore la génération d'un ensemble de cycles candidats. En temps réel, si un lien du réseau est non (suffisamment) protégé, un cycle est choisi dans cet ensemble calculé préalablement pour le protéger.

Dans le cas des FIPP *p*-cycles, on trouve seulement des études considérant un trafic statique. Les méthodes considérées sont basées soit (*i*) sur l'énumération des cycles candidats suivie de la recherche d'ensembles de chemins disjoints pour chaque cycle, ou (*ii*) la définition des ensembles de chemins disjoints et ensuite la définition des FIPP *p*-cycles pour protéger ces ensembles de telle sorte que tous les chemins d'opération soient protégés.

Comme les *p*-cycles attirent de plus en plus l'attention des chercheurs, on peut trouver dans la littérature des recherches traitant d'autres sujets que ceux mentionnés plus haut. Les *p*-cycles segmentés ont été traités dans [36, 52, 64, 75, 93]. Les enveloppes (*Protected Working Capacity Envelope PWCE*) ont été traitées dans [33, 89–92]. L'article [57] présente une vue d'ensemble sur les *p*-cycles et leurs designs, la reconfiguration des *p*-cycles, les *p*-cycles dans les réseaux multicast et l'utilisation de la technique de génération de colonnes dans la protection par *p*-cycles. L'article [2] présente un aperçu des

principaux travaux faits sur les *p*-cycles, les caractéristiques et les différents types de protection offerte par les *p*-cycles ainsi qu'une discussion sur certaines extensions ou des améliorations possibles de la notion de *p*-cycle.

3.3 Combinaison de groupage et de protection

Un petit nombre de recherches ont proposé l'étude du problème combiné de groupage et protection de trafic dans les réseaux. Dans [99], pour chaque connexion, on calcule deux chemins disjoints, un pour le routage et l'autre pour la protection. Les chemins de protection des différentes connexions sont aussi multiplexés. Les auteurs ont proposé deux méthodes de groupage de trafic, *Mixed primary-backup Grooming Policy MGP* et *Segregated primary-backup Grooming Policy SGP*. Dans SGP, une longueur d'onde est utilisée exclusivement pour les chemins d'opération ou pour les chemins de protection. Dans MGP, une longueur d'onde peut être utilisée par des chemins d'opération et de protection en même temps. Les résultats obtenus montrent que MGP est meilleur dans les réseaux SONET en anneau et que SGP est mieux adaptée aux réseaux maillés.

Dans [77], les auteurs ont proposé trois approches différentes pour grouper le trafic, *protection-at-lightpath level PAL*, *mixed protection-at-connection level MPAC* et *separate protection-at-connection level SPAC*. Les résultats montrent, comme dans [99], qu'il est préférable de séparer les chemins de protection des chemins d'opération (c'est-à-dire, la capacité totale des liens des chemins est utilisée soit pour la protection ou pour l'opération), spécialement lorsqu'on est en présence d'un nombre plus grand de connexions qui demandent des petites bandes passantes.

Onguetou et Grover [73] ont proposé une formulation mathématique au problème qui intègre le groupage de trafic dans le design de protection par *p*-cycles. Ils ont montré que la façon dont le trafic est groupé a un impact sur le design des *p*-cycles de protection.

Il est connu que le bon découpage d'un problème en sous problèmes facilite le traitement du problème global. Généralement le résultat total obtenu est meilleur lorsqu'on traite le problème en entier sans le découper. Cependant, le fait de résoudre les sous problèmes séparément de façon plus efficace peut donner un résultat meilleur que lorsque le

problème initial est traité sans découpage avec une méthode qui n'est pas suffisamment efficace. Le problème combiné de groupage et protection de trafic obeit à cette règle. Seul, le problème de groupage de trafic a été démontré comme NP-complet [78, 129], voir aussi le livre [18] chapitre 5 pour la complexité de calcul de ce problème. L'étude du problème combinant le groupage et la protection reste possible, en utilisant des heuristiques, avec des petits jeux de données de trafic et dans de petits réseaux. Cependant, dans un réseau réel avec un trafic dynamique, combiner les deux problèmes et les résoudre efficacement est non réaliste à notre avis, dans la mesure où chacun des problèmes est suffisamment compliqué à résoudre séparément.

CHAPITRE 4

DYNAMIC TRAFFIC GROOMING IN WDM MESH NETWORKS

Ammar Metnani, Brigitte Jaumard and Alain Houle

Préambule : Dans le contexte d'un trafic dynamique, à chaque période de temps, certaines connexions en cours s'arrêtent et de nouvelles connexions arrivent. Ces connexions sont généralement de granularités beaucoup plus petites que la capacité de transport qu'offre une longueur d'onde. Le but est de grouper le traffic du réseau de façon à maximiser la bande passante totale des connexions acceptées à chaque période de temps.

Nous avons proposé quatre scénarios différents pour le problème de groupage de traffic. Nous avons proposé une formulation mathématique pour chaque scénario proposé. Nous avons considéré deux modèles de flots de données pour chacun des scénarios. Les résultats montrent que le rapport entre la bande passante des connexions acceptées et la bande passante des connexions qui se présentent à chaque période de temps varie suivant les scénarios et que le scénario permettant la modification des capacités des ports offre le meilleur compromis.

Abstract-Traffic grooming in WDM (Wavelength Division Multiplexing) optical networks has increasingly gained interest due to the disparity between the user requirements and wavelength transport capacities. It consists in packing low rate traffic streams onto high-capacity optical channels in order to maximize the bandwidth usage and minimize the network cost. In this paper, we investigate the challenging question of dynamic grooming traffic, with the objective of maximizing the throughput, which is somewhat equivalent to minimizing the number of blocked connections.

We explore four different scenarios, with increasing rerouting or equipment resetting options in order to better accommodate the incoming requests, while attempting to maximize the resource usage (bandwidth and optical ports). In a small batch dynamic provisioning context over a set of time periods, we propose original scalable mathematical models for each of the four scenarios, subject to either bifurcated (VCAT/LCAS context) or non bifurcated flows.

Numerical experiments have been conducted on different traffic instances using the EONET network topology. Results show that : *(i)* the proposed mathematical models are highly scalable, and *(ii)* allowing limited rerouting (around 10 to 15%) is enough to maximize, in practice, the resource usage (bandwidth usage and optimized usage of available OC-48/OC-192 ports) in the context of dynamic traffic grooming.

4.1 Introduction

The emergence of Wavelength Division Multiplexing WDM technology in telecommunication networks increased drastically the transmission capacity of a single fibre. Not only a large number of non overlapping channels (wavelengths) can be transmitted without any interference, but a huge amount of data can also be transmitted over a single wavelength. The fact that the transmission capacity of a single wavelength has increased from 2.5 Gbps (OC-48) to 10 Gbps (OC-192) and soon up to 40 Gbps (OC-768), creates a large discrepancy between the wavelength transmission capacity and the granularities of the network client data flows. Consequently, traffic grooming technique has been introduced to improve the bandwidth usage in the optical networks. It consists in

the grooming of low bandwidth flows onto the same wavelength in an efficient manner. It comes today with the reconfigurable optical add-drop multiplexers (ROADM), which are becoming commercially available at an affordable cost. This means that the concerns of the granularity discrepancies and the traffic can be dynamically adjusted. In other words, traffic grooming combined with ROADMs is an opportunity to improve network performance by dynamically reacting to dynamically varying traffic.

In the context of static traffic and of network planning or design, the goal is usually to identify the required resources in order for the network to be able to carry all traffic demands while minimizing the network cost (or more recently the energy consumption [122]), considering that the traffic is known in advance, assuming a reasonable forecast of long-term traffic can be done. The cost is often evaluated through the number of required add/drop ports, with possibly different transport capacities, on (R)OADM nodes given a certain set of demand requests with heterogeneous granularities.

In the context of dynamic traffic, the physical network resources (nodal equipment) are known and the goal is usually to minimize the blocking rate of the new connection requests, or to maximize the network throughput. Under dynamic traffic grooming, those two objectives are not necessarily equivalent, as there are requests with different granularities, with the observation that small granularity requests are much more numerous than large granularity ones, see, e.g., [132]. When the traffic is dynamic, there are continuous changes in the network flows with some added and dropped connection requests. Hence, the current traffic grooming may not be the most efficient one for the incoming requests, since some bandwidth capacity may be missing between the source and the destination for an incoming request while there may exist enough available bandwidth capacity assuming we modify some existing network routes.

In this study, we decided to consider the objective of maximizing the throughput and consequently only indirectly minimizing the connection blocking rate. We propose to compare different scenarios for provisioning new incoming requests, starting with a very conservative scenario where no rerouting and no equipment resetting is allowed, and then setting three other scenarios with increasing rerouting or equipment setting flexibilities.

The paper is organized as follows. In Section 4.2, we review the most recent studies on dynamic traffic grooming, and on the rerouting strategies and algorithms. Section 4.3 is devoted to the details of the dynamic traffic grooming problem statement that we study in this paper. Mathematical models for the four proposed scenarios are described in Section 4.4. Numerical results are discussed in Section 4.5. Conclusions are drawn in the last section.

4.2 Literature Review

4.2.1 Dynamic Traffic Grooming

A lot of papers and even some books have been devoted to traffic grooming, mainly in the static case, see, e.g., [11, 19, 47, 102]. While the very first studies focused on ring topologies, the recent ones addressed traffic grooming in mesh networks. Very few studies deal with the definition of the optical hops and the best decomposition of a routing path into a set of segments to minimize the number of optical ports or to maximize the bandwidth usage of the established lightpaths. Very often, authors assume wavelength converters at every node while converters are still quite expensive. We will focus on the works on dynamic traffic grooming in this section.

Quite recently, Tornatore *et al.* [101] introduced a new algorithm to improve the blocking rate in WDM networks, based on the so-called holding-Time-Aware Provisioning Algorithm (HTA), which assumes that connection duration is known for all connection requests. Numerical results showed a reduction of the blocking rates compared to other previously proposed approaches which are holding-time unaware.

Xin and Wang [109, 110] formulated the problem of traffic grooming with an ILP model. However, to alleviate the complexity of the problem, they considered only *single-hop grooming*, i.e., each connection between a source v_s and a destination v_d is routed only on a single hop lightpath. The authors proposed four different formulations, the first one minimizes the number of ports (transmitters/receivers). The three other formulations consist in minimizing the sum of used wavelengths on each link considering that there is no wavelength conversion, full wavelength conversion and a partial wavelength

conversion (some nodes have full conversion capability) respectively. In order to overcome the lack of scalability of the ILP models, heuristics are proposed for network and traffic instances with realistic sizes. Results showed that resource usage dramatically decreases when the blocking requirement is relaxed, and the grooming performance slowly increases when given more resources.

Yao *et al.* [115] studied dynamic traffic grooming in sparse grooming mesh networks, i.e., in networks where only some of the nodes, called G-nodes, have grooming capabilities. Simulations were conducted on NSF network with a random method to select G-nodes among the network nodes. Numerical experiments are done in order to study the effect of the number of G-nodes on the connection blocking rate.

In [95, 96], contrarily to Yao *et al.* [115], Sivakumar *et al.* considered that the grooming can be done at every node by using electronic SONET add-drop multiplexers (SADMs), but with only a subset of the ports and wavelengths, at each node, equipped with SADMs. Authors concluded that limited grooming at each node is sufficient to obtain the performance obtained with full grooming, especially when connections occupy a small fraction of the wavelength capacity. Also, the authors allow the distribution of a single connection over different flows, therefore implicitly assuming the use of the VCAT/LCAS protocols. Under such an assumption, they showed that the connection granularities, the grooming policy and the number of wavelengths used per link for provisioning a given connection request all have a significant effect on the performance in terms of the blocking probability.

Ho and Lee [46] proposed the zone-based with neighbor expansion (ZWNE) algorithm. It relies on the use of an auxiliary graph which represents a region (zone) around the shortest physical route from the source to the destination of a new incoming request, rather than the whole network. Indeed, the auxiliary graph includes the nodes of the shortest path between the source and destination of a connection request, with edges between these nodes representing existing or allocable lightpaths. The proposed heuristic reduces the computational complexity required for computing a new lightpath when a connection request comes in since it computes the lightpath only within a zone. As will be seen in Section 4.3.1, we use a similar idea, except that instead of limiting the so-called auxi-

liary graph around a single shortest path, we restrict it around the 2 or 3 shortest paths (and consequently avoid the need to frequent neighbor expansion to overcome blocking without hampering scalability).

In [16, 17], Drummond and da Fonseca proposed a new heuristic algorithm, called the Alternative Routing with Virtual Topology Expansion (ARVTE) algorithm. The ARVTE algorithm uses the idea proposed in Ho and Lee [46] and consists in searching for a lightpath, between a source and a destination, in a reduced network region (zone) rather than in the whole network, in order to avoid the formation of bottlenecks (i.e., links crossed by multiple shortest paths) produced by the use of shortest path algorithms. The ARVTE algorithm introduces an off-line alternative routing procedure. The results show that the ARVTE algorithm seems to be efficient for taking care of dynamic traffic traffic grooming, based on comparisons made with several variants of their heuristic and the ZWNE heuristic of [46].

Xin [108] developed an analytical model for dynamic traffic grooming in WDM networks. The developed model is used to analyze the blocking performance of dynamic traffic, as well as to evaluate the performance of a given network topology or a grooming algorithm. The numerical results show that the analytical model matches the simulation results.

In [112], Yang and Silvalingam investigated the usage of the VCAT mechanism [7] in optical grooming networks. The VCAT technique can split requested connection bandwidth into a set of traffic sub-streams and route them on multiple paths. To select those paths, Yang and Silvalingam proposed four methods. The numerical results confirmed the benefits of taking advantage of the VCAT mechanism. The authors made comparison with a provisioning algorithm (SPSW) where only one route, of the shortest paths, is possible between every pair of nodes. They observed that their algorithm outperformed SPSW by 53.4 % to 95.7%, 93% to 99%, and 98.5% to 99.9% respectively in terms of blocking performance (but not much details are given on how they obtained those numbers). The study showed that the differential delay caused by VCAT technique increases with the number of link-disjoint paths used for routing sub-streams.

4.2.2 Rerouting

Rerouting of some ongoing connection requests is a commonly used technique ([1, 105, 106, 129, 130]) in DWDM optical networks with/without traffic grooming capability, in order to reduce the blocking rate in the context of a dynamic traffic, see, e.g., [116]. Although the first studies on rerouting have been published in the early 90s, few of them have dealt with traffic grooming. In addition, very few papers have appeared on rerouting with some concerns on quality of service, see, e.g., Saradhi and Murthy [86].

Two types of rerouting can be found in the literature : Reactive and pro-active rerouting. In reactive rerouting, rerouting is only considered when the routing of a new incoming connection request cannot be done unless we reroute some ongoing (active) connections. In [116, 117], the authors proposed two heuristics. The first one allows for lightpath rerouting, and the second one for connection rerouting. However, for the connection reroutings, the heuristics are limited to the cases where no more than 2 ligh-paths need to be perturbed. In proactive rerouting, rerouting is regularly conducted, either at regular time intervals, or from time to time according to some network measurements (which are attempting to measure whether there is room for improving resource provisioning). In [45], the authors suggested to perform rerouting as soon one or more connection requests are dropped (or by associating a timer to each lightpath based on its average duration). In such a case, the source node of the torn down connection sends rerouting solicit messages to k candidates nodes, and each candidate node selects p candidate ongoing connections of which it is the source, and attempts to reroute them on their respective optimal (resource usage) path. Then, there is the issue (not addressed in this paper) that network resources may become fragmented because of the network dynamism, as time passes. Under these circumstances it may be highly beneficial to re-optimize (i.e., de-fragment) the existing lightpath configurations at some specific time instances to improve the network resource utilization and reduce the risk that future connection requests will be blocked, see, e.g., [1]. Another issue, which has been investigated by Huang *et al.* [48], is inverse multiplexing where, in order to increase the grade of service, we allow the splitting of a sub-wavelength connection into sub-connections,

thanks to VCAT.

In this study, we also propose to consider rerouting, but with some differences with previous studies, as it will be explained in the next section.

4.3 Problem Statement

4.3.1 Dynamic Traffic Grooming

We consider a provisioned optical network represented by a directed graph $G = (V, L)$ where V is the set of nodes indexed by v, and L is the set of directed fiber links, indexed by ℓ. We denote by Λ the set of available wavelengths (the same set on each fiber link), indexed by λ. Let $W = |\Lambda|$ be the number of available wavelengths. The provisioning of the network is made of a set of segments or optical hops. Let S be the set of segments, indexed by s. Each optical hop is characterized by a transport capacity (either OC-48 or OC-192) and a set of requests of various low granularities groomed on it. Optical hops are established thanks to a set of OXC ports, with either OC-48 or OC-192 capacities, and to a set of optical bypasses. Each optical hop is associated with the grooming of some requests, and its transport capacity is not necessarily fully utilized : let u_s^{R} be its residual capacity. An example of a provisioned network is depicted in Figure 4.1, with two wavelengths, and 12 optical hops, three with an OC-192 transport capacity, and the remaining ones with an OC-48 transport capacity.

In order to provision a new request from a given source (v_s) to a given destination (v_d), one has first to find a lightpath from v_s to v_d either using the same wavelength on all the traversed links (it is then called a single-hop lightpath, or a Wavelength Path (WP)), or using different wavelengths on different optical hops (it is then called a multi-hop lightpath or a Virtual Wavelength Path (VWP)). In order to find those lightpaths, we assume that, for each pair of source and destination nodes (v_s, v_d), a list of k shortest physical paths is available. In order to get such a list, we compute $k' > k$ shortest physical paths (where the length is evaluated with the number of links or the geographical distances of the links), and extract the k most link disjoint paths out of the k' paths. For instance, in the example of Figure 4.1, for $v_s = v_1$ et $v_d = v_6$ we can get $k' = 4$ paths : $\pi_1 = (v_1, v_2, v_3, v_6)$,

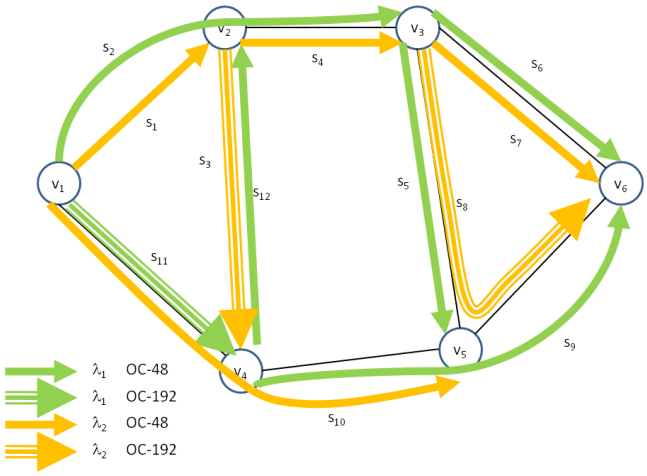


Figure 4.1 – A Provisioned Network

$\pi_2 = (v_1, v_2, v_3, v_5, v_6)$, $\pi_3 = (v_1, v_4, v_5, v_3, v_6)$, and $\pi_4 = (v_1, v_4, v_5, v_6)$. We then extract the $k = 2$ most disjoint, i.e., π_1 and π_4. Along those two physical paths, we then look for possible single or multi-hop paths. Along π_1, there are four possible lightpaths : $p_1 = (s_2, s_6)$, $p_2 = (s_2, s_7)$, $p_3 = (s_1, s_4, s_6)$ and $p_4 = (s_1, s_4, s_7)$, while there is a single 2-hop lightpath along π_2, i.e., $p_5 = (s_{11}, s_9)$.

We also assume that the provisioned network is such that each node contains active, spare and bypass ports where :

- an active port is a port which is in use and tunned on a certain wavelength, with grooming/de-grooming capability,
- a bypass port is such that the incoming traffic of the node is forwarded to a given adjacent link on the same wavelength.

The proposed study is within the context of a multi-period system, where T is the overall set of time periods for the network planning, with generic index t for a given time period. We denote by K_t the set of ongoing granted connections during period $t \in T$, and by K_t^{ADD}, K_t^{DROP} et K_t^{RER} the sets of new incoming, torn down and rerouted connections at period t, respectively. It follows that : $K_{t+1} = (K_t \setminus K_t^{\text{DROP}}) \cup K_t^{\text{ADD}}$. Connections are indexed by k. For a given k, we denote by b_k its requested bandwidth where b_k belongs

to a set of discrete granularities, e.g., $\{\text{OC}-1, \text{OC}-3, \text{OC}-12, \text{OC}-24, \text{OC}-48\}$.

At the outset of each time period, we are given a set of new incoming connection requests to be routed, as well as a number of terminating requests. The objective is to grant requests to maximize the throughput during each period, while possibly allowing the rerouting of some ongoing requests, depending on the scenario under consideration.

4.3.2 Four different Scenarios

We propose to investigate and compare four scenarios.

Scenario 1. The goal is to maximize the network throughput, while deciding which connections to grant, at each time period, without modifying the network equipment settings (without adding or upgrading ports), and without disturbing the already established connections. In other words, no change in the GRWA (Grooming, Routing and Wavelength Assignment) which was established during the previous time periods, is allowed for the still on-going connection requests.

Scenario 2. Therein, we allow the rerouting of a limited number of already established connections in order to increase the number of new granted connection requests and maximize the network throughput. The objective is then to establish the critical number (percentage) of reroutings in order to get a significant increase of the throughput. One does not want to reroute all the on-going connections in order to get a meaningless increase of the throughput.

Scenario 3. It is similar to Scenario 2 except that we allow some physical operations on the node ports, i.e., upgrading or down-grading the capacity of an OXC port. While OC-48 ports are cheaper than OC-192 ports (i.e., an OC-192 transport blade is about 2.5 times the price of an OC-48 for four times more bandwidth capacity), the difference justifies an upgrade (or a downgrade) depending of the bandwidth usage in the optical hops (segments).

Scenario 4. It is defined in the context of an agile network. At each time period, we redefine completely the GRWA considering the already granted connections and the new ones, in order to maximize the network throughput. All previously granted requests need to be provisioned while possibly only a fraction of the new incoming requests will be

granted. It can be viewed as a kind of holistic scenario, where GRWA is completely re-optimized at each time period. Even if such a scenario may appear unrealistic in practice, it serves the purpose of measuring how much we loose by refraining the modifications of the already established GRWA.

The goal in our study is to define an objective function that can be the best compromise between the cost ($), the connection bandwidth acceptance and the connection reroutings. Note that this compromise could be different depending on whether it is on the the network user side or the network operator side.

4.3.3 Bifurcated / Non Bifurcated Flows

For the flow of a connection, two models are considered.

Non bifurcated flows. The entire flow of each connection k must be routed on the same lightpath from source to destination.

Bifurcated flows. The data of the same connection can be split and routed on different routes and/or different wavelengths. The VCAT/LCAS (Virtual conCATenation/ Link Capacity Adjustment Scheme) [7] [71] allows the *extraction*, from the network, of the bandwidth requested by a connection. Indeed, let us consider a connection request k, from the source v_s to the destination v_d, which requests a bandwidth b_k. Enough bandwidth may be available in the network to meet the request k but not on the same route from v_s to v_d. Physically, the flow of the connection k is bifurcated (split and routed) on different lightpaths from source v_s to destination v_d.

4.3.4 Definitions and Notations

Let $\mathscr{P}$ be the set of established lightpaths. Each lightpath is made of one or several optical hops, where each hop is defined by a route (set of links) and a wavelength. We denote by $\mathscr{P}_{sd}$ the subset of $\mathscr{P}$ made of the lightpaths from source v_s to destination v_d. Let G be the set of allowed granularities for the set of requests, indexed by g.

When receiving a new incoming request, the objective is to reuse existing lightpaths, and use the bandwidth which has been freed by the requests which ended. All lightpaths

are created at the beginning and no new lightpath can be added at the other time periods.

At the outset of each time period, we are given a set of new incoming connection requests to be routed, as well as a number of terminating requests.

The following sets of variables are common to all models and scenarios :

$x_p^k \in \{0,1\}$ is equal to 1 if request k is granted on lightpath p, and 0 otherwise.

$z_k \in \{0,1\}$ is equal to 1 if request k is granted, 0 otherwise.

$x_p \in \mathrm{Z}^+$ represents the bandwidth amount carried out by lightpath p.

We also need the following parameters : $a_{sp} \in \{0,1\}$ is equal to 1 if s is an optical hop belonging to lightpath p, 0 otherwise. Parameters u_s and u_s^{R} represent the transport capacity of optical hop (segment) s and the residual capacity of s respectively. $\overline{y}$ is the maximum number of connections that can be disturbed.

4.4 Mathematical Models

We describe below the mathematical models associated with each of the four scenarios, each time with two variants, the first one with non bifurcated flows, the second one with bifurcated flows. Note that the models described in this section are solved for each time period. The t index is omitted in order to simplify the exposition, unless there might be confusion about the time period under concern.

4.4.1 Scenario 1 : Same set of ports and no GRWA perturbation

The objective, which is to maximize the additional network throughput with respect to the new granted requests, can be written as follows :

$$f_{\mathrm{OBJ}}^1(z) = \sum_{k \in K_t^{\mathrm{ADD}}} b_k z_k. \tag{4.1}$$

The set of constraints decomposes into three subsets, which can be described as follows.

Grooming. Lightpath p is used for the provisioning of possibly several requests, and we compute the additional amount of bandwidth it carries :

$$x_p = \sum_{k \in K_t^{\mathrm{ADD}}} b_k \, x_p^k \qquad\qquad p \in P. \tag{4.2}$$

Capacity constraints. The overall bandwidth of all lightpaths going through segment s should not exceed its residual capacity :

$$\sum_{p \in \mathscr{P}} a_{sp} \, x_p \leq u_s^{\mathrm{R}} \qquad\qquad s \in S. \tag{4.3}$$

All or nothing. If incoming request $k \in K_t^{\mathrm{ADD}}$, going from v_s to v_d, is granted on lightpath p ($x_p^k = 1$), all the bandwidth of the request k must be carried out on the same lightpath p :

$$\sum_{p \in \mathscr{P}_{sd}} x_p^k = z_k \qquad\qquad k \in K^{\mathrm{ADD}}. \tag{4.4}$$

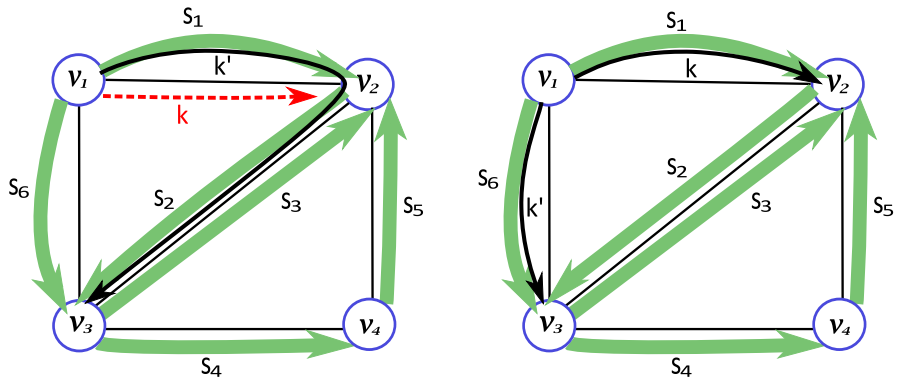
Constraints (4.4) ensure that we cannot only grant a fraction of the bandwidth requirement of request k : none or all the bandwidth requirement must be granted, but it can be distributed over multiple lightpaths between v_s and v_d.

Note that for bifurcated flows, $x_p^k \in [0,1]$ represents the traffic fraction of request k, from v_s to v_d, going through lightpath p.

4.4.2 Scenario 2 : Limit on the number of disturbed connections

In Scenario 2, we allow the rerouting of a limited number of ongoing connections in order to maximize the overall bandwidth.

Let us consider the example illustrated in Figure 4.2 where the network is composed of 4 nodes and 5 links. Connected pairs of nodes are joined by two links, one in each direction. Assume that each link has a fiber capacity of one single wavelength. The network provisioning is depicted in Figure 4.2(a) where we have 5 full capacitated optical hops $s_1, s_2, \ldots, s_5$ (i.e, , with one bandwidth unit capacity) and a free optical hop s_6 and a new connection request k from v_1 to v_2 with bandwidth requirement equal to 0.5 times



(a) Request k cannot be routed (b) Rerouting of $k' \rightarrow k$ can be routed

Figure 4.2 – Rerouting of connexion to serve a request

the transport wavelength capacity. Request k cannot be granted, because for all possible lightpaths, at least one segment is at full capacity (routing of the connection requests on the segments $s_1, s_2, \ldots, s_5$ have been made in a previous time period). Let us assume that there is a request k' from v_1 to v_3 with a bandwidth requirement equal to 0.6 the transport capacity of a wavelength. If we reroute the connection request k' on link $\{v_1, v_3\}$, connection request k can then be provisioned and routed on $\{v_1, v_2\}$, as shown in Figure 4.2(b).

We need to introduce one more set of variables :

$y_k \in \{0,1\}$ is equal to 1 if request k is rerouted, 0 otherwise.

Objective function In Scenario 2, the objective is still to maximize the throughput with respect to the new incoming requests, i.e., to grant requests to maximize the amount of carried bandwidth but, in addition, we are allowed to reroute some requests while minimizing the number of rerouted requests. The compromise between the two components of the objective depends on a weight parameter C^{PENAL} whose value will be discussed in Section 4.5 on numerical results :

$$f^2_{\text{OBJ}}(y,z) = \sum_{k \in K_t^{\text{ADD}}} b_k z_k - C^{\text{PENAL}} \sum_{k \in K_{t-1}} b_k y_k. \tag{4.5}$$

Note that, in order to model the possibility for a request already provisioned, say k,

to be rerouted, we introduce a "clone" of k, denoted by $\hat{k}$: some constraints (see below (4.12)) ensure that either the original request (k) or its clone ($\hat{k}$) is taken into account. If it is the original (k), it is the previous routing, if it is its clone ($\hat{k}$), it is the rerouting.
Capacity constraints. They are similar to the capacity constraints in Scenario 1, except that the right-hand side is equal to u_s because the bandwidth of all requests, new and already provisioned are considered in the left hand-side term, see (4.7).

$$\sum_{p \in P} a_{sp} x_p \leq u_s \qquad s \in S \tag{4.6}$$

Grooming. The overall number of flows going through path p is composed of the new incoming connection requests, the previously established and still ongoing requests, and some previously established and rerouted requests :

$$x_p = \sum_{k \in K^t \cup K_t^{\mathrm{RER}}} b_k x_p^k \qquad p \in \mathscr{P}, \tag{4.7}$$

where K_t^{RER} represents the set of rerouted requests of K_t.
No bifurcated flows. If incoming request $k \in K_t^{\mathrm{ADD}}$, going from v_s to v_d, is granted on lightpath p ($x_p^k = 1$), all the bandwidth of request k must be served along the same lightpath p. If the connection is rerouted ($y_k = 1$), the connection must be granted on another path :

$$\sum_{p \in \mathscr{P}_{sd}} x_p^k = z_k \qquad k \in K_t^{\mathrm{ADD}}; \tag{4.8}$$

$$\sum_{p \in \mathscr{P}_{sd}} x_p^{\hat{k}} = y_k \qquad k \in K, \hat{k} \in K_t^{\mathrm{RER}}. \tag{4.9}$$

Limit on the number of rerouted connection requests

$$\sum_{k \in K} y_k \leq \overline{y}. \tag{4.10}$$

Rerouting. If k is routed, it means that k has not been perturbed and is kept routed on the same path ; if $\hat{k}$ is routed, it means that k has been perturbed and has been rerouted on a

new path. The following sets of constraints translate in equations the above explanations :

$$\sum_{p \in P_{sd}} x_p^{\hat{k}} = y_k \qquad k \in K, \hat{k} \in K_t^{\mathrm{RER}} \tag{4.11}$$

$$\sum_{p \in P_{sd}} x_p^{k} = 1 - y_k \qquad k \in K. \tag{4.12}$$

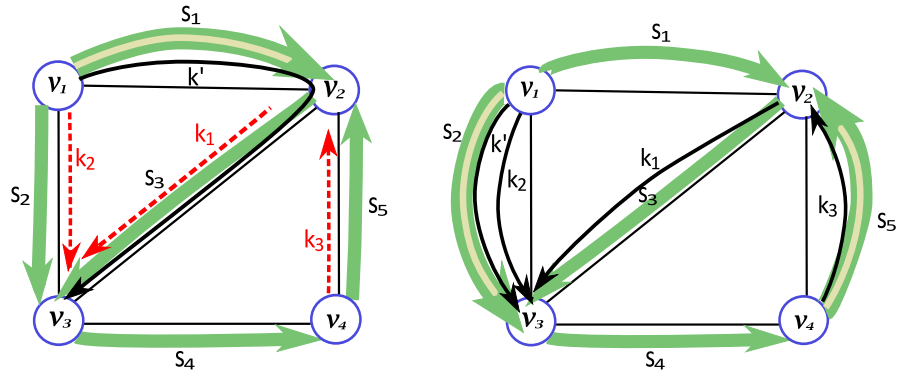
4.4.3 Scenario 3 : Possibility of upgrading/downgrading of port capacity

Scenario 3 starts with Scenario 2 and goes further by allowing some physical operations on the node ports :

- *Port upgrade* : replacement of an OC-48 port by an OC-192 port on the same wavelength ;
- *Port downgrade* : replacement of an OC-192 port by an OC-48 port on the same wavelength.

We assume that we have an inventory where we store the OC-48 transport blades, replaced by the network operator, and the OC-192 transport blades purchased in anticipation of future updates.

Let us consider the example illustrated in Figure 4.3 where the network is the same as in Figure 4.2. The network provisioning is depicted in Figure 4.3(a) where we have 4 full capacitated optical hops s_1, s_3, s_4, s_5, each with one bandwidth unit capacity, except s_1 which has a two bandwidth unit capacity, and one empty optical hop s_2. In terms of traffic, we have three new connection requests :
$k_1 : v_2 \longmapsto v_3$ and $k_3 : v_4 \longmapsto v_2$ with bandwidth requirement equal to the transport wavelength capacity and
$k_2 : v_1 \longmapsto v_3$ with bandwidth requirement equal to 0.5 times the transport wavelength capacity.
Only request k_2 can be granted beacause for all possible lightpaths for k_1 and k_3, there is always at least one segment at full capacity (provisioning of the connection requests on segments s_1, s_3, s_4, s_5 has been made in a previous time period). Let us assume that there is a request k' from v_1 to v_3 with a bandwidth requirement equal to the transport capacity



(a) Request k_1 and k_2 cannot be routed

(b) Rerouting of $k' \rightarrow k_1$ and k_3 can be routed

Figure 4.3 – Upgrading/Downgrading of some ports

of a wavelength. If we accept to upgrade the capacity of the ports at the endpoints of the links (v_1, v_3) and (v_4, v_2) to two bandwidth units, all requests can be granted as shown in Figure 4.3(b). Request k' can be groomed with k_2 and routed on segment s_2. Note that we can downgrade the ports at the endpoints of link (v_1, v_2) as only one bandwidth unit is needed on segment s_1 (k' has been rerouted).

Before setting the model, we need to introduce the following variables :

$z_v^{g,t} \in \mathrm{Z}^+$ where $z_v^{g,t}$ is equal to the number of ports in use at time period t, with granularity $g \in G$

Objective function.

The objective function of Scenario 3 has two components : the first component corresponds to the objective of Scenario 2 and the second component is the upgrading/downgrading port cost. Its expression is as follows :

$$f^3_{\mathrm{OBJ}}(y,z) = f^2_{\mathrm{OBJ}}(y,z) + C^{\mathrm{PORT}}\left(\nabla_{192}(z^g) + \nabla_{48}(z^g)\right) \tag{4.13}$$

where

$$\nabla_{192}(z^g) = \sum_{v \in V} z_v^{192,t} - \sum_{v \in V} z_v^{192,t-1} \tag{4.14}$$

$$\nabla_{48}(z^g) = \sum_{v \in V} z_v^{48,t} - \sum_{v \in V} z_v^{48,t-1}. \tag{4.15}$$

We can simplify the expression of the upgrading/downgrading port cost by assuming that there are enough OC-48 transport cards in the inventory, resulting from legacy equipment in the context where network operators are, on average, moving toward larger transport capacities (OC-192 and even OC-768). Consequently, the expression of f^3_{OBJ} becomes :

$$f^3_{\text{OBJ}}(y,z) = f^2_{\text{OBJ}}(y,z) + C^{\text{PORT}} \nabla_{192}(z^g). \tag{4.16}$$

Constraints of Scenario 3 are similar to the constraints of Scenario 2, except that in constraints (4.6), we replace u_s^{R} by :

$$(1-\alpha_s)u_s + \alpha_e u'_s.$$

where $\alpha_s = 1$ if the transport capacity of segment s is modified, and 0 otherwise. If u_s = OC-48, then u'_s = OC-192. If u_s = OC-192, then u'_s = OC-48.

4.4.4 Scenario 4 : No restriction

$$f^4_{\text{OBJ}}(y,z) = \sum_{k \in K_t} b_k z_k \tag{4.17}$$

Constraints are as follows :

Transport capacity. The overall bandwidth of all working paths going through segment s should not exceed the transport capacity of the segment.

$$\sum_{p \in \mathscr{P}} a_{sp} x_p \leq u_s \qquad\qquad s \in S. \tag{4.18}$$

Flow decomposition. The overall number of flows going through path p is composed of

all connection requests routed over p :

$$x_p = \sum_{k \in K_t \cup K^{\mathrm{RER}}} b_k x_p^k \qquad p \in \mathscr{P}, \tag{4.19}$$

All or nothing. If incoming request k, going from v_s to v_d, is granted on path p ($x_p^k = 1$), all the bandwidth of request k must be served on the same lightpath p :

$$\sum_{p \in \mathscr{P}_{sd}} x_p^k = z_k \qquad k \in K^{\mathrm{ADD}} \cup K_t, \tag{4.20}$$

Note that for bifurcated flows, $x_p^k \in [0,1]$ represents the portion of the traffic of the request k going through lightpath p. Then, constraints (4.20) ensure that if any portion of request k is granted, all the bandwidth of k must be granted on paths between v_s and v_d.
Route all ongoing connections. If a connection has been granted in the previous time periods, this connection must be granted in the current time period.

$$z_k = 1 \qquad k \in K_{t-1}. \tag{4.21}$$

4.5 Numerical results

We implemented the mathematical models proposed in the previous section and conducted some numerical experiments in order to evaluate and compare the four scenarios for dynamic traffic grooming. We first describe the network and traffic instances in Section 4.5.1. We next provide the results obtained for each scenario, and then compare their performances.

4.5.1 Network and Traffic Instances

We use the EONET network with 20 nodes and 78 directed links. We assume a fiber capacity of 8 wavelengths per link.

The initial provisioning of the network is made of 756 request connections, i.e., 918 Gbps, with the following distribution among the different possible granularities : 137

OC-1 requests, 130 OC-3 requests, 163 OC-12 requests, and 326 OC-48 requests. At each time period, some connections are torn down and new incoming ones are established. We consider a traffic scenario with a stable overall traffic, i.e., with a number of incoming requests equal to the number of torn down ones. We call turn over rate the percentage of add/drop changes in the traffic, assuming the number of added requests is equal to the number of dropped ones. We consider 10 time periods, where for each time period, the turn over rate is equal to 5% (5 % add, 5 % drop) of the already established connections. The 5% of additional requests is defined as follows : it is made of 5% additional OC-g requests out of the current overall number of provisioned OC-g connections, for $g \in G = \{\text{OC}-1, \text{OC}-3, \text{OC}-12, \text{OC}-48\}$.

For a network with $|L|$ directed links, and W wavelengths, we need at most $2 \times W \times |L|$ ports at the endpoints of the optical hops. Very often, the network is saturated before we install so many ports due to the traffic distribution and the limitation on the number of hops in lightpaths. In our study, we allowed at most $.5 \times 2 \times W \times |L|$ ports for the endpoints of the optical hops, and $.3 \times 2 \times W \times |L|$ bypass ports (for the intermediate nodes in the optical hops where optical switching occurs). The number of ports is distributed uniformly among the nodes, while making sure that the number of ports is the same at the two endpoints of each link. At the beginning, all ports are supposed to be OC-48 ports. In all experiments, we assume that we start with a network provisioning where only OC-48 ports are used, i.e., all wavelengths have a transport capacity of 2.5 Gbps.

In bifurcated data flows scheme, assuming the use of the VCAT/LCAS protocols, the bandwidth flow of a connection flow can be partitioned into sub-flows granularity which is a multiple of VC-11, with VC-11 = $1/28\times$ STS-1 $\equiv 1/28\,\text{OC}-1 = 1/28 \times 51.84 = 1,095$ Mbps flow. Although we use continuous variables in the bifurcated models, we impose the x_p variables to take a value larger than 1,095 Mbps, assuming one would round up the continuous values to the closest value which a multiple of 1,095 Mbps.

4.5.2 Scenario 1

We first compare the throughput with respect to the new incoming requests under the assumption of bifurcated flows vs. non bifurcated flows for Scenario 1. Results are

summarized in Figure 4.4, where time periods are on the horizontal axis, and throughput on the vertical one. The throughput is indeed the grade of service (GoS), expressed in terms of throughput for the new incoming requests, i.e., the sum of the bandwidth requirements of the newly granted requests over the sum of the bandwidth requirements of all the new incoming requests.

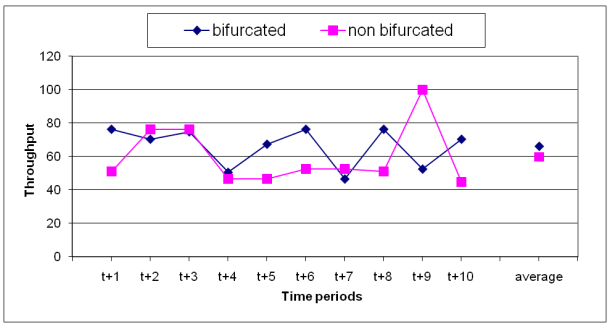


Figure 4.4 – Performance of Scenario 1 - Throughput GoS

While difference is significant, i.e., about 11% on average, we can observe that the throughout GoS is larger in the case of bifurcated flows. It is explained by the fact that, if no lightpath can be found to route the whole bandwidth of a request, the request can be granted under a bifurcated flow scheme, thanks to distributing its bandwidth requirement over different lightpaths.

4.5.3 Scenario 2

Figure 4.5 shows the same throughput GoS comparison than Figure 4.4, for Scenario 2, with the rerouting penalty coefficient C^{PENAL} equal to 0.1.

Throughput GoS differences between bifurcated and non bifurcated flows are smaller (4.6 % compared to 11 % on average). This is not surprising as allowing some ongoing connections to be rerouted, gives the chance to otherwise blocked connections to be accepted in the case of non bifurcated flows.

We next studied the impact of the rerouting penalty coefficient (C^{PENAL}) on the throughput GoS of the new granted requests. In Table 4.I, for various values of the C^{PENAL} coefficient, we provide the throughput GoS, as defined earlier, and the number

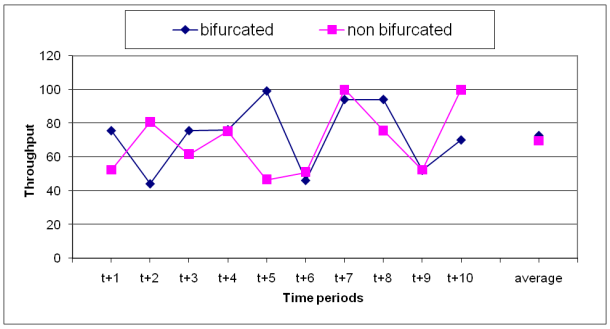


Figure 4.5 – Performance of Scenario 2 - Throughput GoS

of rerouted requests expressed in percentage of the current set of provisioned requests. The throughput GoS values are plotted on the graphs depicted in Figure 4.6 for the non bifurcated and the bifurcated flow cases, over a set of 10 time periods.

The best results with respect to the throughput GoS are obtained with the smallest value of rerouting penalty coefficient ($C^{\text{PENAL}} = 0.1$) in both bifurcated and non bifurcated flows. However, the number of rerouted connections increases if we decrease the penalty coefficient.

		C^{PENAL}			
	Flows	0.1	0.5	1	10
Throughput GoS (%) (average)	bifurcated	72.7	71.8	67.6	66.4
	non bifurcated	69.5	68.6	64.2	61.1
Rerouting (%) (average)	bifurcated	0.31	0.22	0.20	0.07
	non bifurcated	0.79	0.68	0.66	0.02

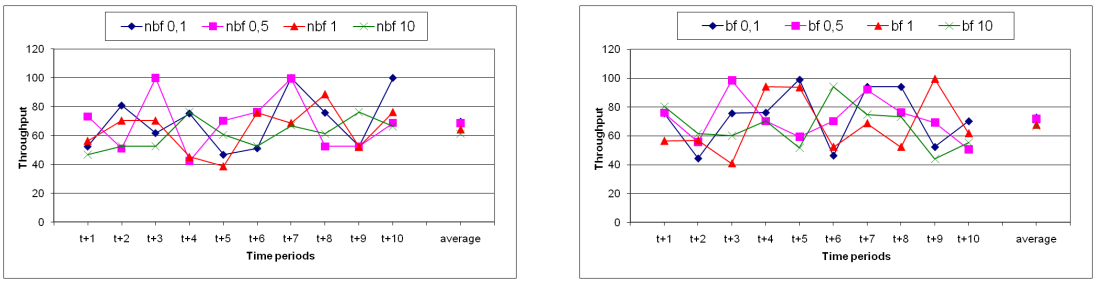
Tableau 4.I – Impact of the C^{PENAL} coefficient - Scenario 2

When the rerouting penalty coefficient C^{PENAL} is equal to 1, the throughput GoS values are larger than those obtained with Scenario 1. Indeed, the throughput GoSs in Scenario 2 are 67.6% and 64.2 % for bifurcated and non bifurcated flows respectively, compared to 66.1% and 59.7 % in Scenario 1. It means that, if we disturb some connections with an overall bandwidth amount B, we may be able to accept more than bandwidth amount of B with the additional granted connections. In such a case, the average number of disturbed connections is 0.20% and 0.66% of already established connections,

in bifurcated and no bifurcated flows, respectively.

When the rerouting penalty coefficient C^{PENAL} is equal to 10, the throughput GoS values are almost equal to those obtained with Scenario 1. Indeed, in both cases, bifurcated and non bifurcated flows, a small percentage of rerouting of already established connections is performed. This means that, it is almost impossible to accept an additional connection with an overall amount of bandwidth which is equal to 10 times the bandwidth of the rerouted connections.

The best compromise can be $C^{\text{PENAL}} = 0.5$. We get the best throughput GoS with an additional connection with bandwidth equal to B if we accept to disturb some connections with an overall bandwidth $2 \times B$. In such a case, the average number of rerouted connections is equal to 0.22% and 0.68% for bifurcated and non bifurcated flows, respectively.



(a) Non bifurcated flows

(b) Bifurcated flows

Figure 4.6 – Throughput GoS over a set of 10 time periods - Scenario 2

4.5.4 Scenario 3

We evaluated the performances of Scenario 3 for various values of the penalty parameters C^{PENAL} and C^{PORTS}. In Table 4.II, for each set of penalty parameter values, we provide the average (over a set of 10 time periods) throughput GoS (denoted by GoS^{T}), as well as the number of port upgrades (UP) and downgrades (DOWN).

We can observe that the throughput GoS (GoS^{T}) can be very high compared with the GoS^{T} values obtained with Scenario 1 or Scenario 2. Almost all connections can be accepted (up to 99.5%). This is due to the port upgrading.

4.5.4.1 C^{PORTS} parameter impact

From table 4.II, we can see that the effect of the C^{PORTS} penalty parameter is very important on the throughput GoS for both bifurcated and non bifurcated flows. Indeed, the throughput GoS varies from 65.2 % when C^{PORTS} = 10,000 to 99.5 % when C^{PORTS} = 1 for bifurcated flows, and from 63.8 % for C^{PORTS} = 10000 to 98.1 % for C^{PORTS} = 1 for non bifurcated flows.

	C^{PENAL}	C^{PORTS}	GoS^{T} (average) %	% reroutings	UP	DOWN
Bifurcated	1	1	99.5	0.2	5.7	4.5
	1	10	99.5	0.2	4.9	3.7
	1	100	98.4	0.3	3.4	2.3
	1	1,000	94.1	0.7	1.4	1.5
	1	10,000	65.2	1.0	0.0	1.0
Non bifurcated	1	1	98.1	0.0	6.7	4.3
	1	10	98.1	0.1	5.7	4.1
	1	100	98.1	0.4	2.9	2.1
	1	1,000	94.9	1.2	1.3	1.2
	1	10,000	63.8	1.8	0.0	1.0

Tableau 4.II – Performances of Scenario 3 wrt C^{PORTS}

Besides the effect on the throughput GoS, the upgrading coefficient C^{PORTS} has an effect on three other parameters,

- Number of upgraded ports : as expected, the higher C^{PORTS} is, the smallest the number of the upgraded ports.
- Number of downgraded ports : coefficient C^{PORTS} does not only affect the number of upgraded ports, but the number of downgraded ports as well.
- Percentage of disturbed connections : even if the disturbance coefficient C^{PENAL} remains unchanged, the percentage of rerouted connections increases with the increase of the coefficient C^{PORTS}. Indeed, when C^{PORTS} = 1, only 0.2 % of connections on average are rerouted in the case of bifurcated flows while no connection

is rerouted in the case of non bifurcated flows, and this percentage increases to 1.8 % (non bifurcated flows) and 1.0 % (bifurcated flows) when C^{PORTS} is equal to 10,000.

4.5.4.2 C^{PENAL} parameter impact

The C^{PENAL} penalty parameter impact is very similar to Scenario 2 when the upgrading coefficient C^{PORTS} is very large. Indeed, from Table 4.III, we can see that if C^{PORTS} = 10,000, the smallest is C^{PENAL}, the best is the throughput GoS and the highest is the number of the rerouted connections.

	C^{PENAL}	C^{PORTS}	GoS^{T} (average) %	% reroutings	UP	DOWN
Bifurcated	1	1	99.5	0.2	5.7	4.5
	10	1	99.5	0.1	3.7	3.9
	100	1	97.2	0.1	3.6	3.7
	1	10,000	65.2	1.0	0.0	1.0
	10	10,000	64.7	0.1	0.0	0.2
	100	10,000	63.2	0.1	0.0	0.3
Non bifurcated	1	1	98.1	0.0	6.7	4.3
	10	1	98.1	0.0	5.9	4.7
	100	1	97.4	0.0	5.2	6.2
	1	10,000	63.8	1.8	0.0	1.0
	10	10,000	58.3	0.2	0.0	0.3
	100	10,000	55.3	0.0	0.0	0.5

Tableau 4.III – Performances of Scenario 3 wrt C^{PENAL}

However, when the C^{PORTS} coefficient is small (e.g., 1), the effect of the C^{PENAL} coefficient is negligible on the throughput GoS. Indeed, for bifurcated flows, the throughput GoS is equal to 99.5 % for C^{PENAL} = 1 and 10 and equal to 97.2 % for C^{PENAL} = 100 and for non bifurcated flows the throughput GoSs are equal to 98.1 % for C^{PENAL} = 1 and 10 and equal to 97.4 % for C^{PENAL} = 100.

4.5.4.3 Combined C^{PENAL} and C^{PORTS} coefficient impacts

We can summarize the effect of the two coefficients in Table 4.IV. We will use the following symbols.

$\nearrow$: increases,

$\searrow$: decreases,

$\longrightarrow$: stable,

? : no or negligible effect.

C^{PENAL}	C^{PORTS}	GoS^{T} (average) %	# reroutings	UP	DOWN
?	$\nearrow$	$\searrow$	$\nearrow$	$\searrow$	$\searrow$
?	$\searrow$	$\nearrow$	$\searrow$	$\nearrow$	$\nearrow$
?	small value	?	?	?	?
$\nearrow$	high value	$\searrow$	$\searrow$	?	?
$\searrow$	high value	$\nearrow$	$\nearrow$	?	?

Tableau 4.IV – C^{PENAL} and C^{PORTS} impacts

4.5.5 Scenario 4

At each time period t, we must provision all ongoing connections in time period $t-1$ that are not dropped in time period t. The results are slightly better than in Scenario 2. Indeed, from Figure 4.7, we can see that the throughput GoS is equal to 73.9 % for bifurcated flows compared to 72.7 % for Scenario 2. The throughput GoS is equal to 72.9 % in the case of non bifurcated flows compared to 69.5 % in Scenario 2. However, the throughput GoS is small compared to the values obtained with Scenario 3. It means that even if we do not limit the number of rerouted connections, there is a limit on what

we can accept if no change in the nodal equipment of the network is done (e.g., port upgrading).

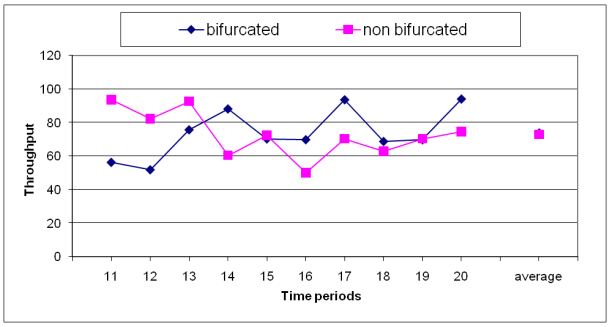


Figure 4.7 – Performance of Scenario 4 - Throughput GoS

4.6 Conclusion

In this paper, we investigated four different scenarios to provision connections in the context of dynamic grooming traffic. We proposed a mathematical model for each scenario with an objective function whose first priority is to maximize the throughput of the new granted requests. We considered two different flow models, namely bifurcated and non bifurcated flows.

The results confirm that, for all scenarios, the throughput GoS is higher in the case of bifurcated flows. Allowing the rerouting of some established connections increases the throughput. However, Scenario 4 confirms that there is a limit to what can be accepted even if we allow to disturb all established connections. The best results are obtained with Scenario 3, where we allow some port upgrading/downgrading. This can be viewed as the definition of some *hub* ports where we can groom a larger amount of traffic than the other ports. The definition of such ports is done by the mathematical model and depends on the distribution of the traffic in the network.

Future work should include taking into account the signal attenuation features as upgrading the capacity of a port has an impact of the amplification equipment to be added along the fibers.

CHAPITRE 5

DIRECTED *P*-CYCLE PROTECTION IN DYNAMIC WDM NETWORKS

Ammar Metnani and Brigitte Jaumard

Préambule : Dans le contexte d'un trafic dynamique certaines connexions en cours s'arrêtent et de nouvelles connexions arrivent à chaque période de temps. Chaque nouvelle connexion est routée sur le plus court chemin de la source à la destination.

Le but est d'utiliser les *p*-cycles protégeant les liens pour protéger tout le trafic du réseau contre les pannes simples de liens de façon efficace à chaque période de temps.

Nous proposons trois stratégies différentes pour la mise à jour des cycles de protection à chaque période de temps. Pour chaque stratégie, nous proposons un modèle de programmation linéaire en nombres entiers (PLNE). Nous utilisonss la technique de génération de colonnes pour résoudre chaque modèle proposé, ce qui nous permet d'obtenir des résultats avec un intervalle d'optimalité de moins de 1 % dans un temps de moins de 10 secondes.

Abstract : *p*-Cycle protection is an efficient pre-configured and pre-cross-connected protection scheme which can achieve ring-like recovery speed while retaining the capacity efficiency of mesh-based schemes. Thereafter, we study the stability and the efficient reconfiguration of *p*-cycles in the context of dynamic asymmetric traffic, a protection problem that is acknowledged to be difficult to solve, but with which we must deal in the context of intelligent optical networks (IONs).

While most studies have focused on static traffic, the few studies with a dynamic traffic assumption reports the necessity to use heuristics instead of Integer Linear Programming (ILP) tools because they are too time consuming. We claim that assuming appropriate ILP tools are used, such tools are scalable. We investigate three *p*-cycle updating strategies. We design and solve exact models for each strategy, using large scale optimization tools. Results show that, not only the proposed ILP tools are scalable, but in addition, that, by modifying only a very small fraction of the already established *p*-cycles, we get a protection level that is as good as if we would reset the whole set of *p*-cycles.

5.1 Introduction

While *p*-cycles have been studied a lot in the context of static and symmetrical traffic, traffic demand is usually changing dynamically and becoming more asymmetrical, consequently working paths need to be set dynamically and protection must also be adapted accordingly. Few papers have been devoted to the *p*-cycle design problem for survivable networks under dynamic asymmetric traffic. The objective of this paper is therefore to investigate further this problem.

p-Cycles are gaining in popularity among the protection schemes for WDM (Wavelength Division Multiplexing) networks due to their pre-configuration and pre-cross-connectivity features, resulting in ring-like recovery speed while retaining the capacity efficiency of mesh-based schemes [37]. In case of a link failure, which is the most frequent failure, only the two endpoints of the failing fiber link need to be reconfigured. Capacity efficiency results from the protection, by a *p*-cycle, not only of the on-cycle

links, but also of the straddling links (i.e., links which although not on-cycle links, have their two endpoints on the cycle).

As for the static *p*-cycle problem, two approaches have been explored for the dynamic *p*-cycle problem. There is the *off-line* approach, which consists in generating off-line either the set of all possible *p*-cycles [37, 126], or a restricted set of promising candidate *p*-cycles [34, 84]. While new incoming traffic is routed, *p*-cycles are selected among the cycles of the pre-computed set of cycles. The *on-line* approach, i.e., the one we adopted in this paper, consists in a dynamic *p*-cycle generation, where *p*-cycles are generated only when needed. While several authors have successfully used Integer Linear Programs (ILPs) in studies under static traffic, it has been strongly questioned in the case of dynamic traffic for scalability reasons, see, e.g., [126]. We believe there is a misunderstanding on the usefulness of ILP tools. Indeed, their strength is often under evaluated, as their use is frequently not pushed to their limit, and not sufficiently combined with the nowadays available large scale optimization tools. In addition, while ILP tools have often been perceived as to be used for exact solutions, they are also very powerful tools for global heuristic search. This is one of the endeavor of the present study to argue in favor of those arguments.

Previous studies include the work of Zhong *et al.* [126] where the authors adapted to dynamic traffic the heuristic method *ER-Based Unity-p-cycle* [124] proposed in the context of static traffic, see also the Chapter 10 of Grover [32]. For each new incoming request that is not protected by the existing *p*-cycles, they compute all the candidate *p*-cycles using the algorithm in [49] among which the candidate with the largest *Efficiency Ratio* is selected, and the process is repeated until the whole routing path of the new incoming request is protected. In [84], Ruan *et al.* compute at the outset a first set of *p*-cycles (i.e., it is an *off-line* generation) such that each link of the network is protected by at least one *p*-cycle, called *primary p-cycle*. The selection of the routing path for a new incoming request is made *on-line* at the time the request comes in, in order to fully exploit the fact that the links of that path can be already protected by the set of existing *p*-cycles. If one link of the routing path is not protected, an additional copy of the designated *primary p-cycle* for its protection is set up.

Most studies on dynamic traffic use the grade of service metric in order to evaluate the performance of their strategies. However, the transport capacity is often set to an arbitrary value with, e.g., a dedicated uniform share of 50% for protection on all the links, even if it is well know, that even under a uniform distribution of the traffic between all node pairs, not all links are used uniformly due to the particular topology of the network under study. For this reason, in the present study, we have decided not to put any limit on the transport capacity, and to use as in the static case, the network cost for the objective. However, we provide the bandwidth use and the network redundancy values in order to be able to evaluate the smallest blocking rate that would be encountered for a given budget, under the most efficient bandwidth distribution.

In this study, as in [31], we propose to examine dynamic provisioning with the framework of small-batch provisioning under asymmetric traffic, and not in the context of the on-line/dynamic provisioning of a single path or *p*-cycle. Indeed, even if a wide range of applications may be envisioned to require on-demand connection provisioning, it seems reasonable, in particular in core networks, that a delay in the range of few seconds up to few minutes, depending on the applications, can be reasonably tolerated between connection request and setup. Indeed, in the context of a core network, we are dealing with the establishment of lightpaths that can convey up to 10 Gbs or even 40 Gbs, and costs thousand of dollars to use. We can think of routers making the request for an additional lightpath on the basis of observed trends, slightly before the added capacity is fully needed, see [31, 62].

The paper is organized as follows. In the next section, we investigate three strategies for dynamic *p*-cycles. In Section 5.3, we develop the corresponding mathematical models in order to later evaluate the three strategies. All models can be solved optimally, thanks to large scale optimization tools, as explained in Section 5.4. In Section 5.5, we first examine the quality and the characteristics of the solutions, and then we evaluate the comparative performances of the three models. Conclusions are drawn in the last section.

5.2 Three Strategies

We propose three strategies of increasing complexity. The first one where no already established protection can be modified, a second one where a small percentage of *p*-cycles can be modified in order to increase the protection coverage of the new incoming requests, and the third one (an holistic strategy for performance evaluation only) where the whole protection is reset in order to minimize the protection cost while granting and protecting all requests. We use the same objective for all three strategies, i.e., to minimize the protection link cost.

5.2.1 No Disruption of the Established *p*-Cycles

Most previous studies adopted the assumptions of granting new requests and protecting them under the condition of no disruption of the ongoing ones, in order to fulfill the requirement of high availability or the fulfillment of the service level agreement for already established ones. Under such assumptions, only useless copies of *p*-cycles are dismantled in order to release resources for increasing the spare capacity. When a new request comes in, a working routing path is established along one of the shortest paths between the source and the destination nodes. New *p*-cycle copies, or new *p*-cycles are added, as needed, in order to ensure protection for all the links of this new route, with the aim of minimizing the network cost and therefore maximizing the protection sharing. If needed, the transport capacity of existing *p*-cycles is modified in order to guarantee the full protection of the new incoming connection requests. This strategy is very similar to the *Routing in Spare Capacity* developed by Schupke *et al.*in [88].

5.2.2 Limited *p*-Cycle Disruption

Depending on the applications, longer end-to-end delays can be acceptable. We can exploit this tolerance in order to briefly disrupt the protection of a limited number of tolerant connections, taking the risk of a failure on an unprotected connection request, in order to increase the number of protected granted connections. In this strategy, we propose to explore how many copies of *p*-cycles we need to disrupt in order to increase

significantly the number of additional requests we can grant with an adequate protection, and whether it corresponds to an acceptable disruption level.

5.2.3 Global Re-optimization

The objective is to reoptimize the whole set of p-cycles in order to get the most economical protection p-cycle scheme. It can be viewed as an holistic strategy in order to evaluate the quality and the stability of the other protection updating schemes. With such a strategy, we simply reuse one of the efficient models developed under the assumption of static traffic, see, e.g., [82] and citations therein.

5.3 Mathematical Models

In this section, we develop the mathematical models associated with the dynamic p-cycle strategies we discussed in the previous section. First, we define the notations, parameters and variables in Section 5.3.1, and then each one of the three mathematical models in the forthcoming sections.

5.3.1 Notations

We represent the optical network by a graph $G = (V, L)$ where V represents the set of optical nodes, indexed by v, and where L is associated with the set of optical directional links, indexed by ℓ. Each link has a cost, denoted by COST_ℓ, which represents, e.g., the number of working bandwidth units of the link or the link cost with respect to its terminal equipment.

Let T be the overall set of time periods over which we want to protect the network, indexed by t. We develop the mathematical model at an arbitrary period, say t, taking into account the new incoming and the terminating requests, as well as the traffic legacy, i.e., the set of ongoing protected requests. In order to alleviate the notations, unless needed, we will omit the t index when dealing over the current time period. The parameter w_ℓ (or w_ℓ^t in case of ambiguity) is the overall number of bandwidth units that need to be protected on link ℓ during the current time period.

Let $\mathscr{C}$ be the set of all potential cycles, indexed by c. Link $\ell \in L$ is an on-cycle link of p-cycle c if $a_\ell^c \in \{0,1\}$ is equal to 1 and 0 otherwise, and it is a straddling one with respect to p-cycle c if $s_\ell^c \in \{0,1\}$ is equal to 1, and 0 otherwise. Parameter c_ℓ^R represents the residual capacity of link ℓ, i.e., the overall transport capacity $\overline{c}_\ell$ of link ℓ minus w_ℓ. Parameter $\tilde{y}_c$ represents the number of copies of p-cycle c which are used in order to protect the ongoing requests that have been granted during the previous time periods.

In order to establish the mathematical models, we need two sets of integer variables. The first one, represented by vector $y = (y_c) \in \mathrm{Z}_+^{|\mathscr{C}|}$ is such that y_c is the number of required copies of p-cycle c during the current time period, in order to grant all new incoming requests, while the second one, $z = (z_c) \in \mathrm{Z}_+^{|\mathscr{C}|}$ is such that z_c is the number of dismantled copies of p-cycle c during period t in order to grant more efficiently the new incoming requests.

5.3.2 Strategy 1

In Strategy 1, no protection disruption of any previously granted request is allowed. Therefore, the objective function corresponds to the protection cost for the current time period.

$$f^{\mathrm{OBJ1}} = \min \sum_{\ell \in L} \left(\mathrm{COST}_\ell \sum_{c \in \mathscr{C}} a_\ell^c y_c \right)$$

$$\sum_{c \in \mathscr{C}} (a_\ell^c + s_\ell^c)\, y_c \geq w_\ell - \sum_{c \in \mathscr{C}^{\mathrm{PREVIOUS}}} (a_\ell^c + s_\ell^c)\, \tilde{y}_c \quad \ell \in L \tag{5.1}$$

$$\sum_{c \in \mathscr{C}} a_\ell^c y_c \leq c_\ell^R - \sum_{c \in \mathscr{C}^{\mathrm{PREVIOUS}}} a_\ell^c \tilde{y}_c \quad \ell \in L \tag{5.2}$$

$$y_c \in \mathrm{Z}^+ \quad c \in \mathscr{C} \tag{5.3}$$

where $\mathscr{C}^{\mathrm{PREVIOUS}}$ is the set of already established p-cycles.

Constraints (5.1) ensure that the overall new working traffic on each link is protected, while constraints (5.2) ensure that the overall protection capacity does not exceed the

residual capacity on each link.

5.3.3 Strategy 2

In Strategy 2, we allow a small fraction of protection disruption in order to protect more efficiently the new incoming traffic. The cost function must then take into account the cost of the disrupted copies of *p*-cycles as we assume that terminal equipment used for dismantled *p*-cycles are reused to establish the new *p*-cycles or to increase their transport capacity.

$$f^{\text{OBJ2}} = \min \sum_{\ell \in L} \text{COST}_\ell \left(\sum_{c \in \mathscr{C}} a_\ell^c y_c - \sum_{c \in \mathscr{C}^{\text{PREVIOUS}}} a_\ell^c z_c \right)$$

$$\sum_{c \in \mathscr{C}} (a_\ell^c + s_\ell^c) y_c - \sum_{c \in \mathscr{C}^{\text{PREVIOUS}}} (a_\ell^c + s_\ell^c) z_c \geq w_\ell - \sum_{c \in \mathscr{C}^{\text{PREVIOUS}}} (a_\ell^c + s_\ell^c) \tilde{y}_c \qquad \ell \in L \tag{5.4}$$

$$\sum_{c \in \mathscr{C}} a_\ell^c y_c - \sum_{c \in \mathscr{C}^{\text{PREVIOUS}}} a_\ell^c z^c \leq c_\ell^R - \sum_{c \in \mathscr{C}^{\text{PREVIOUS}}} a_\ell^c \tilde{y}_c \qquad \ell \in L \tag{5.5}$$

$$\sum_{c \in \mathscr{C}^{\text{PREVIOUS}}} z^c \leq \overline{z} \tag{5.6}$$

$$y_c, z_c \in \mathrm{Z}^+ \qquad c \in \mathscr{C}. \tag{5.7}$$

Constraints (5.4) ensure that the overall working traffic on each link is protected. Constraints (5.5) ensure that the overall protection capacity does not exceed the residual capacity on each link, while constraint (5.6) ensures that the number of dismantled copies does not exceed a given threshold.

5.3.4 Strategy 3

Strategy 3 is a theoretical strategy, in which we re-establish the whole protection p-cycle plan in order to accommodate the new incoming traffic, in the most possible optimized way. It therefore corresponds to a static p-cycle model as in [82].

$$f_{\text{OBJ}} = \min \sum_{\ell \in L} \text{COST}_\ell \left(\sum_{c \in \mathscr{C}} a_\ell^c \, y_c \right)$$

$$\sum_{c \in \mathscr{C}} (a_\ell^c + s_\ell^c) \, y_c \geq w_\ell \qquad \ell \in L \tag{5.8}$$

$$\sum_{c \in \mathscr{C}} a_\ell^c \, y_c \leq c_\ell^R \qquad \ell \in L \tag{5.9}$$

$$y_c \in \mathrm{Z}^+ \qquad c \in \mathscr{C}. \tag{5.10}$$

Again, the first set of constraints (5.8) corresponds to the protection demand constraints, and the second set (5.9) to the protection transport capacity constraints.

5.4 Solving the ILP Models

We discuss here how to solve efficiently the mathematical models developed in the previous section.

5.4.1 Large Scale Optimization Tools

As for the efficient computation of p-cycles in a static model, we have different choices for solving the three models developed in the previous section. Obviously, enumerating explicitly all possible cycles will not lead to a scalable solution scheme. Therefore, we can either develop a heuristic in order to generate explicitly only the most promising cycles as in [15, 34], assuming we can define a good metric to identify the most promising, or we can generate implicitly all potential cycles with the use of the column generation techniques combined with either a heuristic or a branch-and-price

for solving the ILP part as in, e.g., [59]. We propose to go in the direction of column generation techniques and to devise how to derive a near optimal ILP solution once we have computed the optimal LP solution.

Column generation techniques involve the decomposition of the original problem into a master problem corresponding to the models developed in Section 5.3 and a pricing problem corresponding to the on-line generation of a new cycle which is guaranteed to improve the current LP solution. The improvement guarantee is obtained thanks to the identification of a cycle with a negative reduced cost, where the reduced cost, an LP indicator (see, e.g., [12]), is defined as follows (mathematical model of Strategy 1) :

$$\overline{\text{COST}}_c = \text{COST}_c - \sum_{\ell \in L} u^1_\ell (a^c_\ell + s^c_\ell) + \sum_{\ell \in L} u^2_\ell a^c_\ell,$$

where COST_c is the unit cost of cycle c, given by $\text{COST}_c = \sum_{\ell \in L} a^c_\ell \text{COST}_\ell$, and where $u^1_\ell \geq 0$ and $u^2_\ell \leq 0$ are the dual variables of constraints (5.1) and (5.2) respectively.

At each iteration of the column generation algorithm, the pricing problem looks for a cycle with a negative reduced cost (but not necessarily the most negative one). If such a cycle is found, it is added to the master problem, which is solved again optimally. New dual variable values are generated and again thrown in the pricing problem, to generate another cycle. This process is repeated until no cycle with a negative reduced cost is found, then we can claim that the current LP solution is optimal. Let us denote by $f^\star_{\text{LP}}$ the optimal solution of the linear relaxation of say the mathematical formulation of Strategy 1. Very often, $f^\star_{\text{LP}}$ is not associated with an integer vector $y^\star$ but only provides a lower bound on the optimal integer value, say $f^\star_{\text{ILP}}$. Solving the ILP deduced from the constraint matrix derived from the set of columns generated in order to obtain $f^\star_{\text{LP}}$ leads to a feasible integer solution whose value, $\tilde{f}_{\text{LP}}$, is often near optimal, i.e., such that the optimality gap $\tilde{f}_{\text{ILP}} - f^\star_{\text{LP}}$ is often very small, i.e., less than 1 %.

5.4.2 Dynamic *p*-Cycle Configuration

At the beginning of each time period, we first take care of the ending requests and free, if there exists some, the unnecessary copies of *p*-cycles. New requests are routed on one of the shortest routes from their source to their destination. It is not necessarily the best strategy in order to make the most efficient use of, e.g., the bandwidth, but we proceed with such a sequential approach in order to simplify the study of the dynamic reconfiguration of the *p*-cycles.

5.5 Computational Experiments

We first describe the data instances, network and traffic, which we use for our experiments. We next evaluate the quality and the characteristics of the solutions in Section 5.5.2. Next, we conduct a network cost analysis in Section 5.5.3 and a protection performance comparison of the three updating strategies in Section 5.5.4.

5.5.1 Network and Traffic Instances

We use two network instances, the COST239 network, and the highly connected NY network. Key characteristics of the two networks are provided in Table 5.I. Link costs are indicated in the last column.

Network	$\|V\|$	$\|L\|$	Average Node Degree	Link Costs
COST239	11	52	4.7	see [6]
NY	16	98	6.1	Unit costs

Tableau 5.I – Network Instances

We assume that, at the outset, we already have 2,000 established request connections, for all instances. At each time period, some connections are torn down and new incoming ones are established. We consider three different traffic scenarios, each with a stable overall traffic, but with different turn over ratios where the turn over ratio is defined by the number of ADD/DROP connection requests. In the case of stable traffic, number of incoming requests is equal to the number of torn down ones. The first traffic scenario has a turn over ratio of 10 %, the second one 20 % and the third one 30 %.

We perform experiments on the three updating strategies, first on the network cost in Section 5.5.3 and then on the network protection performance in Section 5.5.4.

Experiments were conducted on a dual core AMD Opteron machine (2.5 Ghz, 16Gb RAM), using the CPLEX package.

5.5.2 Solution Qualities and Characteristics

Although we did not develop a branch-and-price method in order to guarantee the full optimality of the solutions, thanks to the good qualities of the lower and upper bounds, the optimality gaps were all smaller than 1%, sometimes even of the order of 0.1 % or less. Such a small optimality gap makes the development of a branch-and-price algorithm meaningless. In addition, the computing times required to obtain such high quality solutions are smaller than 10 seconds, therefore the proposed solution scheme is highly scalable, taking into account that several improvements could be easily added to reduce further the computing times (e.g., a heuristic for a "warm" start instead of starting from scratch the solution of the linear programs).

We describe the characteristics of the solutions in Table 5.II for all three values of the turn over ratio (10 %, 20 % and 30 %), i.e., we provide the number of p-cycles, the overall number of p-cycle copies and the average length of the p-cycles. For the average length of the p-cycles, we distinguish the length of the disrupted copies of the p-cycles and of the newly added p-cycles or copies of p-cycles.

In Table 5.II, for all three turn over ratios, we observe that the overall number of copies with Strategy 2 is smaller than with Strategy 1, and even smaller in Scenario 3, meaning that the overall protection bandwidth capacities is reduced with Strategy 2, and further reduced with Strategy 3. This is further investigated with the results presented in the histograms of Fig. 5.1. For COST239, we provide, for 9 links, the values of the required bandwidth for protection (vertical axis). For every pair of connected nodes, the two opposite links have the same cost and their indices are ℓ and $\ell + |E|$ (and are consecutive on the horizontal axis). For each time period, we keep note of those bandwidth capacity requirements, and depending of whether those requirements were present for every time period or only for a few of them, we represented them with a different color,

(a) 10% ADD/DROP

Network	Strategy	# p-cycles	# p-cycle copies	average length of p-cycle copies all / disrupted / added		
COST239	1	21.3	139.5	9.1	0.0	3.8
	2	24.7	121.5	9.7	9.1	8.6
	3	17.6	116.5	9.8	9.8	9.8
NY	1	35.8	137.5	9.8	0.0	6.0
	2	28.2	123.4	10.1	11.7	10.7
	3	18.0	117.5	10.5	10.5	10.5

(b) 20% ADD/DROP

Network	Strategy	# p-cycles	# p-cycle copies	average length of p-cycle copies all / disrupted / added		
COST239	1	24.2	161.8	8.5	0.0	3.9
	2	22.7	132.6	9.3	8.5	8.6
	3	14.2	118.1	10.0	10.0	10.0
NY	1	40.6	163.6	8.9	0.0	5.1
	2	32.0	138.4	9.3	12.1	10.5
	3	21.5	130.9	9.6	9.6	9.6

(c) 30% ADD/DROP

Network	Strategy	# p-cycles	# p-cycle copies	average length of p-cycle copies all / disrupted / added		
COST239	1	26.5	139.3	9.4	0.0	4.5
	2	24.9	125.9	9.5	9.3	8.7
	3	16.0	111.2	10.2	10.2	10.2
NY	1	38.3	163.6	9.1	0.0	5.6
	2	37.3	142.2	9.2	11.0	10.0
	3	21.7	134.0	9.4	9.4	9.4

Tableau 5.II – Solution Characteristics

as indicated on the top of the histograms of Fig. 5.1.

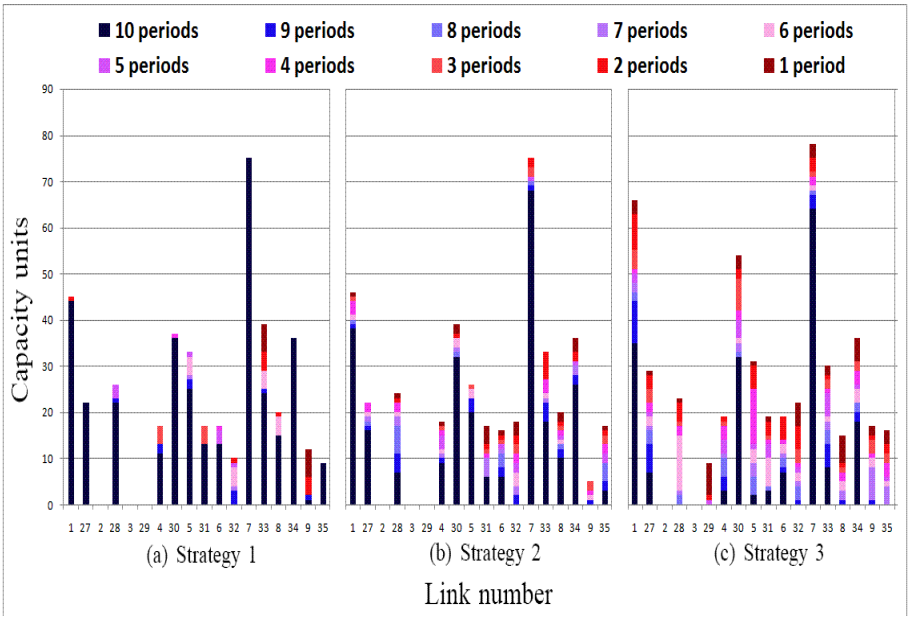
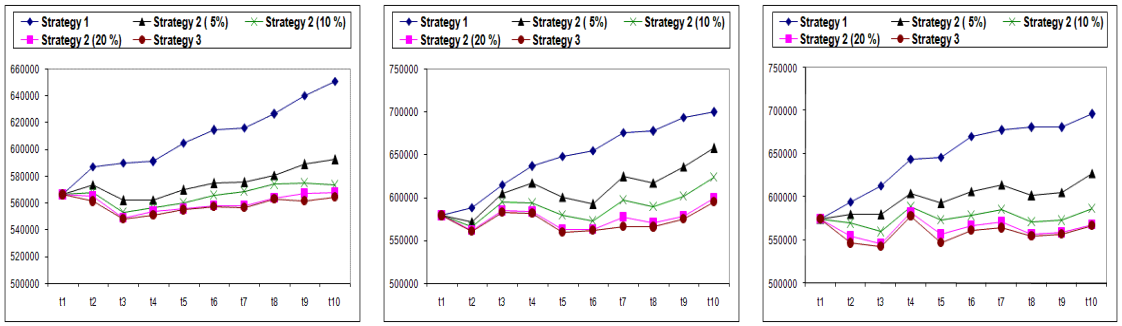


Figure 5.1 – Protection transport capacity requirements

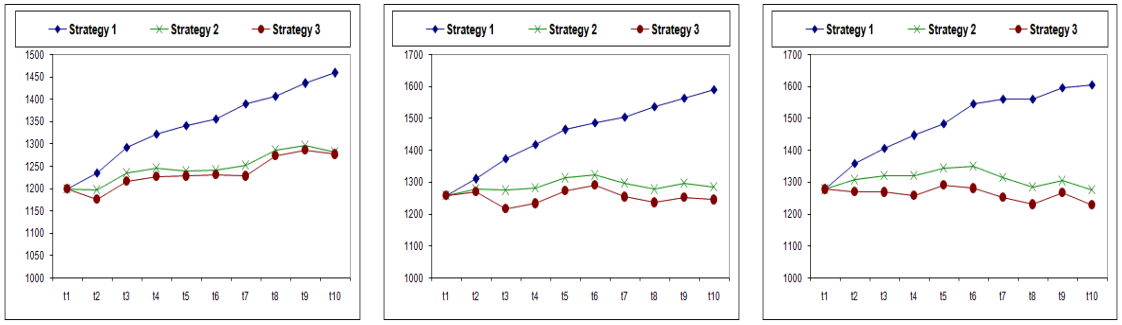
We observe that, due to the cost structure (larger values), some links are not used, or poorly used, see, e.g., link 3 whose cost value is 1090, while some other links are widely used, see, e.g., link 7 whose cost value is 320. On the contrary of most studies in the dynamic context, we decided to minimize the cost rather than the blocking rate, and not to impose capacity constraints, with the argument that transport capacities cannot be set arbitrarily. Results depicted in Figure 5.1 clearly show that transport capacities values should be set, taking into account the cost structure as well as the topology and traffic of the network. This could be done by setting an a priori budget for protection capacity and adding a constraint of the type : $\sum_{\ell \in L} c_\ell^R \leq c^R$ or $\sum_{\ell \in L} \overline{c}_\ell \leq \overline{c}$, in an optimization model where routing and working capacities are jointly taken care.

Strategy 1 entails keeping all p-cycle copies as long as they are protecting some connection requests, and very often those copies are not very capacity or cost efficient. In addition, with Strategy 1, we need to add more p-cycle copies or new p-cycles in order to be able to offer a protection to the new incoming traffic without being able to compact

the remaining traffic on the existing *p*-cycles as it would entail disturbances. This in turn leads to an increased number of *p*-cycles or at least of *p*-cycle copies. On the other hand, the remaining *p*-cycles only offer some protection to the new incoming traffic, leading to the addition of new *p*-cycles that are less lengthly that those already established, and explain why the average length of the *p*-cycles (or *p*-cycle copies) is decreasing. While the average length of the *p*-cycles is smaller for Strategy 1 and larger for Strategy 3, this is true for the two networks.



(a) COST239 network



(b) NY network

Figure 5.2 – *p*-Cycle protection costs

5.5.3 Network Cost Analysis

The focus of this study is the investigation of the overall costs of all required *p*-cycle copies in order to ensure 100 % protection during each time period. Results are summarized in Fig. 5.2, where time periods are on the horizontal axis, and network costs on the vertical one. For COST239 network, we did experiments with three turn over

rates, 5%, 10 % and 20 %, and with 10 % for NY network. Indeed, we observe that 10 % is a good compromise between the quality of the solutions with respect to the cost and the percentage of disrupted *p*-cycles. We observe that, very quickly, even with 5 % of protection reconfiguration, Strategy 2 is rather close to Strategy 3, i.e., the optimal one with respect to the network cost.

We observe that, in Strategy 3, due to the overall re-optimization, without any limit on the number of disrupted *p*-cycle copies, the overall *p*-cycle cost can be reduced by up to 30 % in comparison to Strategy 1, i.e., the opposite strategy where not a single disruption is allowed. However, Strategy 3 is not realistic due its unnecessary high level requirement of control operations. If we turn our attention to Strategy 2, we notice that, with a low disruption level, i.e., 10 % in our study, results are very close to those obtained with the holistic Strategy 3. Curves of Fig. 5.2 illustrate the evolution of the protection cost throughout the 10 time periods, for COST239 network in Fig. 5.2(a) and for NY network in Fig. 5.2(b).

5.5.4 Performance Analysis

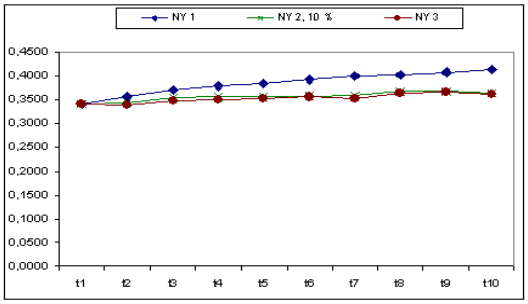


Figure 5.3 – *p*-Cycle protection ratio in NY Network

We investigate the performance of the protection throughout the protection ratio, i.e., the ratio of the protection required bandwidth over the working required bandwidth in Fig. 5.3 for NY network with a 10 % disruption level. Again, we observe that we obtain very similar results for Strategies 2 and 3, i.e., Strategy 2 is near optimal with respect to the network ratio.

5.6 Conclusion

In this paper, we have investigated the reconfiguration and the stability of *p*-cycles in a dynamic asymmetric traffic context. First, we have designed and developed efficient and scalable ILP models and showed that ILP tools remains of great interest even under dynamic traffic. We have also demonstrated that, by reconfiguring a small fraction of the *p*-cycles, we can significantly increase the number of granted requests. In practice, those reconfigurations could be restricted to tolerant connection requests in order to ensure that quality of service and therefore service level agreements (SLAs)remain fulfilled. Transport capacities, whether differentiated for working and protection capacities or not, need to be carefully assessed in order to take into account the network and traffic characteristics, but also the cost parameters.

CHAPITRE 6

DYNAMIC PROVISIONING AND STABILITY OF *P*-CYCLES IN WDM NETWORKS

Ammar Metnani and Brigitte Jaumard

Préambule : Dans le même contexte que le chapitre précédent, le but est d'étudier la stabilité des *p*-cycles protégeant les liens dans le contexte d'un trafic dynamique. Tout le trafic du réseau doit être protégé contre les pannes simples de liens.

En plus du modèle proposé dans le chapitre précédent, nous proposons un nouveau modèle mathématique basé sur la minimisation des ports constituant les *p*-cycles ajoutés à chaque période de temps. Dans le nouveau modèle on prend en considération l'affectation de longueurs d'onde pour chaque *p*-cycle considéré et la contrainte de continuité de longueurs d'onde. Nous montrons que dépendamment de la fonction objective à considérer, les *p*-cycles peuvent être très stables.

Abstract-*p*-Cycles correspond to an efficient pre-configured and pre-cross connected protection scheme which can achieve ring-like recovery speed while retaining the capacity efficiency of mesh-based schemes. While most studies focused on static traffic, we study here the stability and the efficient reconfiguration of *p*-cycles in the context of dynamic traffic.

We design two new highly scalable mathematical models and algorithms for dynamic *p*-cycles. We consider two objectives, the classical one with the minimization of the spare bandwidth requirements, and the objective of minimizing the number of optical bypasses to be newly established or reset while re-using as much as possible the previously established optical bypasses. We use integer linear programming (ILP) formulations relying on decomposition techniques. Results confirm that, not only the proposed models and algorithms are highly scalable, but in addition, show that *p*-cycles are highly stable protection schemes.

6.1 Introduction

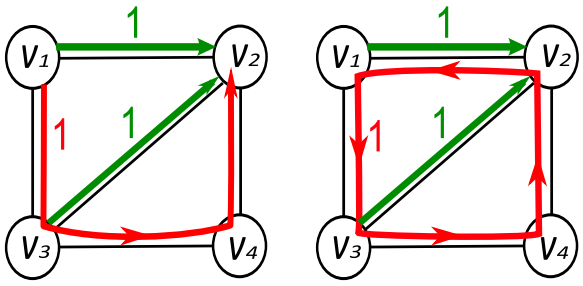
While *p*-cycles have been studied a lot in the context of static and symmetrical traffic, traffic demand is changing dynamically and becoming more asymmetrical, especially with the steady growth of the video traffic over the Internet. Consequently working paths need to be set dynamically and protection provisioning must be adapted accordingly. Few papers have been devoted to the *p*-cycle design problem for survivable networks under dynamic asymmetric traffic. The objective of this paper is therefore to investigate further this issue.

p-Cycles are gaining in popularity among the protection schemes for WDM (Wavelength Division Multiplexing) networks due to their pre-configured and pre-cross connectivity features, resulting in ring-like recovery speed while retaining the capacity efficiency of mesh-based schemes. In case of a link failure, which is the most frequent failure, only the two endpoints of the failing fiber link need to be reconfigured. Capacity efficiency results from the protection, by a *p*-cycle, of not only the on-cycle links, but also of the straddling links (i.e., links which although not on-cycle links, have their two

endpoints on the cycle).

In this study, instead of focusing on measuring the efficiency of p-cycles with their bandwidth requirements, we will also look at the number of optical bypass reconfigurations in view of faster recovery delays, see, e.g., [76], and of reduced network cost, see, e.g., [40]. Let us highlight the meaningfulness of such a performance criterion, with a comparison on an example between what happens in a single link shared (SLP) protection scheme and in a p-cycle scheme, under a single failure scenario.

Consider the example represented in Figure 6.1, with two unit connections $v_1 \to v_2$ and $v_3 \to v_2$. In case of a link failure under SLP scheme, the traffic is rerouted on the protection path around the failed link, meaning that three protection units are needed to protect the two connections. Indeed, link $v_1 \to v_2$ is protected by 1 unit on the protection path $v_1 \to v_3 \to v_4 \to v_2$, and link $v_3 \to v_2$ is also protected by 1 unit of spare capacity on path $v_3 \to v_4 \to v_2$, with a shared one unit of spare capacity on links $v_3 \to v_4$ and $v_4 \to v_2$. Under a directed p-cycle scheme (Figure 6.1(b)), the necessary spare capacity on each link of the p-cycle is one unit. Thus, four protection units are needed.



(a) Shared Link Protection (b) p-Cycle Protection

Figure 6.1 – Number of bypass reconfigurations in p-cycle vs. SLP protection schemes

However, in case of a failure on link $v_3 \to v_2$ under the p-cycle protection scheme, only the optical bypasses at nodes v_3 and v_2 need to be reconfigured. Optical bypass at node v_4 is already pre-configured. Under the SLP scheme, the optical bypasses of three nodes v_2, v_3 and v_4 need to be reconfigured (or at least checked for v_4 as spare capacity is shared on link $v_3 \to v_4$). In general networks, depending on the length of the protection paths in SLP, more optical bypass reconfigurations are needed, while, in any case, only

two such reconfigurations are needed (per single link failure) in a *p*-cycle scheme.

As for the static *p*-cycle problem, two design approaches have been explored for the dynamic *p*-cycle problem. There is the *off-line* approach, which consists in an off-line generation of either the set of all possible *p*-cycles [126], or of a restricted set of promising candidate *p*-cycles [84]. While new incoming traffic is routed, *p*-cycles are selected among the cycles of the pre-computed set of cycles. The *on-line* approach, i.e., the one we adopted in this paper, consists in a dynamic *p*-cycle generation, where new *p*-cycles are generated only when needed, in order to improve the current solution (i.e., *p*-cycle scheme). While several authors have successfully used Integer Linear Programs (ILPs) in studies under static traffic, it has been strongly questioned in the case of dynamic traffic for scalability reasons, see, e.g., [126]. We believe there is a misunderstanding on the usefulness of ILP tools. Indeed, their strength is often under estimated, as their use is frequently not pushed to their limit, and not sufficiently combined with the nowadays available large scale optimization tools [66] [5]. In addition, while ILP tools have often been perceived as to be used only for exact solutions, they are also very powerful tools for global heuristic search. This is one of the endeavor of the present study to argue in favor of those arguments.

Previous studies include the work of Zhong *et al.* [126] where the authors adapted to dynamic traffic the heuristic method *ER-Based Unity-p-cycle* [124] proposed in the context of static traffic. For each new incoming request that is not protected by the existing *p*-cycles, they compute all the candidate *p*-cycles using the algorithm in [49] among which the candidate cycle with the largest *Efficiency Ratio* is selected. The process is repeated until the whole routing path of the new incoming request is protected. In [84], Ruan *et al.* compute, at the outset, a first set of *p*-cycles (i.e., it is an *off-line* generation) such that each link of the network is protected by at least one *p*-cycle, called *primary p-cycle*. The selection of the routing path for a new incoming request is made *on-line* at the time the request comes in, in order to fully exploit the fact that the links of that path can be already protected by the set of existing *p*-cycles. If one link of the routing path is not protected, an additional copy of the designated *primary p-cycle* for its protection is set up. Note that there exists some works in the context of multicast traffic, see, e.g.,

Zhang and Zhong [118].

In this study, as in [31], we propose to examine dynamic provisioning within the framework of small-batch provisioning under asymmetric traffic, and not in the context of the on-line/dynamic provisioning of a single path or *p*-cycle. Indeed, even if a wide range of applications may be envisioned to require on-demand connection provisioning, it seems reasonable, in particular in core networks, that a delay in the range of a few seconds up to a few minutes, depending on the applications, can be reasonably tolerated between connection request and setup. Indeed, in the context of a core network, we are dealing with the establishment of light-paths that can convey up to 10 Gbs or even 40 Gbs, and costs thousand of dollars to use. Routers make requests for additional light-paths on the basis of observed trends, slightly before the added capacity is fully needed, see, e.g., [31, 62].

The paper is organized as follows. In Section 6.2, we provide some additional motivation and insight on dynamic provisioning of *p*-cycles. Scalable mathematical models are proposed and detailed in Section 6.3 for the two objectives discussed in Section 6.2. Efficient solutions of those mathematical models are delineated in Section 6.4. Numerical results are presented in Section 6.5. Conclusions are drawn in the last section.

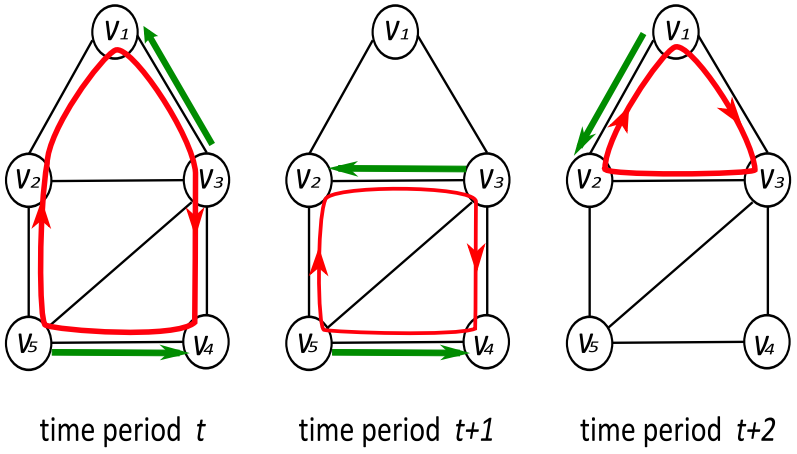
6.2 Dynamic *p*-Cycle Protection

We provide here more motivation and insights on dynamic *p*-cycles. Firstly, we come back on the optimization criteria to be used, then on the interest of a temporary and very short disruption of a limited number of *p*-cycles, and lastly on the different categories of dynamic traffic.

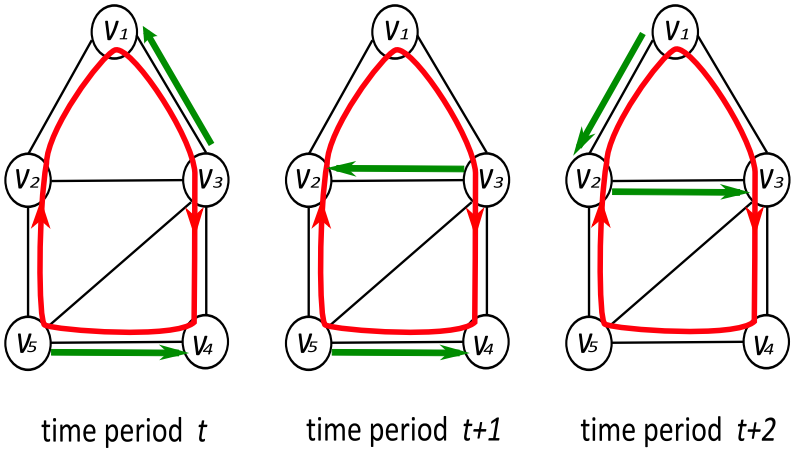
6.2.1 Bandwidth vs. bypass reconfigurations

We propose to investigate two objectives in the present study. The first one is the classical objective that has been used in the literature so far, i.e., the minimization of the spare capacity (or protection capacity). The second one, which might be more meaningful for network designers/operators, corresponds to the minimization of the number

of optical bypasses to be established (at the outset), or the number of optical bypass reconfigurations or re-utilization in the context of a dynamic traffic.



(a) Minimizing the protection bandwidth capacity



(b) Minimizing the number of optical bypass reconfigurations

Figure 6.2 – Bandwidth capacity vs. number of optical bypass reconfigurations

Let us have a look at the example in Figure 6.2 where we represented a small network and its traffic over three successive time periods. In Figure 6.2(a), the p-cycle is not very stable, as it is modified at each time period in order to minimize the protection bandwidth requirements. On the contrary, in Figure 6.2(b), the p-cycle is quite stable, i.e., remains the same over the first two time periods and is slightly modified in time period $t+2$, as we aim at minimizing the number of optical bypass (re)configurations. Indeed, we need three optical bypass (re)configurations going from time period $t+1$ to $t+2$ in

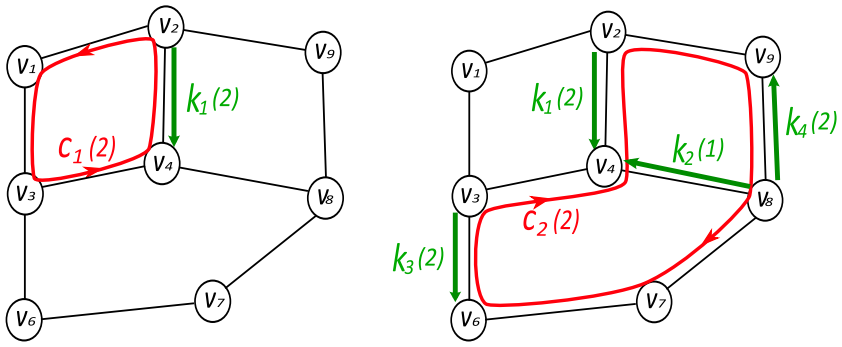
Figure 6.2(b), while we need 4 (re)configurations in Figure 6.2(a). Stability comes at the expense of not optimizing the protection bandwidth usage, but it is not an issue as, in any case, the whole spare capacity is needed during the most busy (peak) time period in order to guarantee a 100% protection against single link failures.

6.2.2 Limited protection disruption

Another point that we will exploit in the mathematical models proposed in Section 6.3, is a p-cycle updating strategy where a small percentage of p-cycles can be modified in order to ensure the protection coverage of the new incoming requests without additional resources. It consists in disrupting the protection of some requests (while keeping the primary routing as is) for a very short time. As will be observed in the numerical results, such a strategy ends up in considering moderate lengthly p-cycles, and therefore, not anymore very long p-cycles as in the static traffic p-cycle design.

Even if we did not explicitly consider different classes of traffic, such a tolerance is of interest in order to briefly disrupt the protection of a limited number of tolerant connections, taking the very low risk of a failure on an unprotected connection request at the time of a p-cycle rerouting, in order to increase the number of protected granted connections. We propose to explore how many copies of p-cycles we need to disrupt in order to increase significantly the number of additional requests we can grant with an adequate (and overall stable) protection, and whether it corresponds to an acceptable disruption level.

Figure 6.3 shows an example of p-cycle disruption. At time period t, three new connections k_2, k_3 and k_4 are added and a new p-cycle c_2 is established to protect them. The p-cycle c_1 protects the connection k_1 at time period $t-1$. Even if the connection k_1 is not dropped at time t, to efficiently protect all connections, p-cycle c_1 is dismantled at time period t (while maintaining its pre-cross connection at node v_4), putting the connection k_1 under the protection of c_2.



(a) Period $t-1$ (b) Period t

Figure 6.3 – p-Cycle disruption

6.2.3 Dynamic Batch Traffic

Throughout the paper, we will assume that traffic instances are described by a set of connection requests, whose working provisioning has been conducted using shortest path routing. Consequently, once the working provisioning is completed, we are left with a set of links such that, for each link, we are given a number of working bandwidth units to protect against single link failure.

Dynamic provisioning can have a different meaning depending on the connection management and control network context. Clearly, in any network, connections do not remain static and the lower the network layer, the less frequent are the changes. An accurate traffic modeling is needed in order to ensure an efficient network provisioning and its ability to survive unpredicted traffic changes. However, depending whether we deal with traffic engineering, or network engineering or network planning (see, e.g., Mukherjee [71] for definitions), dynamic traffic has a different interpretation.

In the context of a core network, even if a wide range of applications may be envisioned and require on-demand connection provisioning, it seems reasonable that a delay in the range of few seconds up to few minutes, depending on the applications, can be reasonably tolerated between connection request and setup times. This is the reason why, in this study, we propose to examine dynamic provisioning with the framework of small-batch provisioning.

Let T be the overall set of time periods, indexed by t, where each time period is

associated with a batch of some add/drop connection requests (to be provisioned/dropped at the end/beginning of the time period). Let Λ be the set of available wavelengths (the same number for each directional fiber link), with generic index λ and $W = |\Lambda|$.

6.3 Mathematical Models

In this section, we develop the mathematical models associated with the dynamic p-cycle strategies we discussed in the previous section. First, we define the notations, parameters and variables in Section 6.3.1, and then each one of the two mathematical models in the forthcoming sections.

6.3.1 Notations

We represent the optical network by a directed graph $G = (V, L)$ where V represents the set of optical nodes, indexed by v, and where L is associated with the set of optical fiber directional links, indexed by ℓ. Each link has a cost, denoted by COST_ℓ, which represents, e.g., the number of working bandwidth units of the link or the link cost with respect to its intermediate/endpoint equipment. We denote by $\omega^+(v)$ (resp. $\omega^-(v)$) the set of outgoing (resp. incoming) links of node v, with $\omega(v) = \omega^+(v) \cup \omega^-(v)$.

Let T be the overall set of time periods over which we want to protect the network, indexed by t. We develop the mathematical model at an arbitrary period, say t, taking into account the new incoming and the terminating requests, as well as the traffic legacy, i.e., the set of ongoing protected requests. In order to alleviate the notations, unless needed, we will omit the t index when dealing with the current time period. The parameter w_ℓ (or w_ℓ^t in case of ambiguity) is the number of bandwidth units that need to be protected on link ℓ during the current time period.

Let $\mathscr{C}$ be the set of all potential cycles, indexed by c. Link $\ell \in L$ is an on-cycle link of p-cycle c if $a_\ell^c \in \{0,1\}$ is equal to 1 and 0 otherwise, and it is a straddling one with respect to p-cycle c if $s_\ell^c \in \{0,1\}$ is equal to 1, and 0 otherwise. $\mathscr{C}^{\text{PREVIOUS}}$ denotes the set of previously established and maintained p-cycles during the previous time period. Note that $\mathscr{C}^{\text{PREVIOUS}}$ is a very small subset of $\mathscr{C}$ as we are not dealing with potential but

with established p-cycles. Parameter c_ℓ^{R} represents the residual capacity of link ℓ, i.e., the overall transport capacity $\overline{c}_\ell$ of link ℓ minus w_ℓ ($c_\ell^{\mathrm{R}} = \overline{c}_\ell - w_\ell$). Parameter $y_{p(\gamma)}^{t-1}$ represents the number of copies of p-cycle c which are used in order to protect the ongoing requests that have been granted during the previous time periods. As we will see in Section 6.4.1, there is a way to only *implicitly* enumerate the set of potential cycles, so that the proposed models can be easily solved in very short computing times.

To finalize the setting of the mathematical models, we now introduce the sets of variables. The first set of variables, represented by vector $y = (y_{p(\gamma)}^t) \in \mathrm{Z}_+^{|\mathscr{P}|}$ is such that $y_{p(\gamma)}^t$ is the number of new required copies of p-cycle c during the current time period, in order to grant all new incoming requests, while the second one, $z = (z_c) \in \mathrm{Z}_+^{|\mathscr{P}|}$ is such that z_c is the number of dismantled copies of p-cycle c during period t in order to grant more efficiently the new incoming requests. Note that a p-cycle can be dismantled either because it is no more useful in order to protect some working bandwidth, or because it is more advantageous to replace it by other more bandwidth efficient p-cycle(s). We allow the disruption of a limited number ($\overline{z}$) of copies of p-cycles and will investigate the best compromise between p-cycle rerouting and bandwidth cost or number of bypass reconfigurations in Section 6.5.

6.3.2 Model 1 : Dynamic p-cycles with a minimum bandwidth cost objective

The first model can be written as follows. The objective function, to be minimized, evaluates the link cost :

$$\min f_1^{\mathrm{OBJ}} = \sum_{\ell \in L} \mathrm{COST}_\ell \left(\sum_{c \in \mathscr{C}} a_\ell^c \, y_{p(\gamma)}^t - \sum_{c \in \mathscr{C}^{\mathrm{PREVIOUS}}} a_\ell^c \, z_c \right).$$

The set of constraints can be written as follows :

$$\sum_{c\in\mathscr{C}} (a_\ell^c + s_\ell^c)\, y_{p(\gamma)}^t - \sum_{c\in\mathscr{C}^{\text{PREVIOUS}}} (a_\ell^c + s_\ell^c)\, z_c \geq$$
$$w_\ell - \sum_{c\in\mathscr{C}^{\text{PREVIOUS}}} (a_\ell^c + s_\ell^c)\, y_{p(\gamma)}^{t-1} \qquad \ell \in L \tag{6.1}$$

$$\sum_{c\in\mathscr{C}} a_\ell^c\, y_{p(\gamma)}^t - \sum_{c\in\mathscr{C}^{\text{PREVIOUS}}} a_\ell^c\, z_c \leq$$
$$c_\ell^{\text{R}} - \sum_{c\in\mathscr{C}^{\text{PREVIOUS}}} a_\ell^c\, y_{p(\gamma)}^{t-1} \qquad \ell \in L \tag{6.2}$$

$$\sum_{c\in\mathscr{C}^{\text{PREVIOUS}}} z_c \leq \overline{z} \tag{6.3}$$

$$y_{p(\gamma)}^t, z_c \in \mathrm{Z}^+ \qquad c \in \mathscr{C} \tag{6.4}$$

Constraints (6.1) ensure that the overall working traffic on each link is protected, while taking into account the protection than can ensured by the previously established and maintained p-cycles. Constraints (6.2) ensure that the overall protection capacity does not exceed the available residual capacity on each link, while constraint (6.3) ensures that the number of disrupted copies does not exceed a given threshold.

6.3.3 Model 2 : Dynamic p-cycles with a minimum number of optical bypass reconfigurations

The second model aims at evaluating the number of optical bypass reconfigurations when updating/modifying p-cycles following some traffic variations, subject to the wavelength continuity constraints.

We therefore introduce the notion of a light-cycle, denoted by γ, similarly to a lightpath in [71], by the combination $\gamma = (c, \lambda)$ of a cycle c and a wavelength λ assigned to it. Consequently, let Γ_λ be the set of all potential light-cycles with wavelength λ and $\Gamma = \bigcup_{\lambda\in\Lambda} \Gamma_\lambda$ be the set of all potential light-cycles, indexed by γ.

Following those definitions, the wavelength continuity constraints are expressed throughout the definition of the light-cycle configuration, i.e., a potential light-p-cycle : each light-cycle configuration is in one to one correspondence with a wavelength.

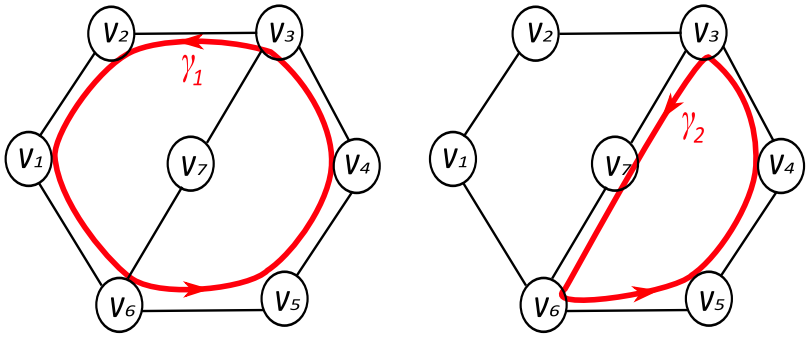
The objective function of Model 2, to be minimized, evaluates the number of OXC port reconfigurations :

$$\min f_2^{\mathrm{OBJ}} = \sum_{\lambda\in\Lambda}\sum_{\gamma\in\Gamma_\lambda}\sum_{\ell=(v,v')\in L}\sum_{\ell'\in omega^+(v')} \left(\underbrace{a_\ell^\gamma a_{\ell'}^\gamma}_{\text{Part 1}} - \underbrace{\sum_{\gamma'\in\Gamma_\lambda^{t-1}} (a_\ell^\gamma a_{\ell'}^\gamma)(a_\ell^{\gamma'} a_{\ell'}^{\gamma'})}_{\text{Part 2}} \right) y_\gamma \tag{6.5}$$

On each copy of a light-p-cycle, a node pre-cross connection (i.e., an optical bypass) involves two ports : One input port on the incoming on-cycle link and one output port on the outgoing on-cycle link, with both links assigned to the same wavelength, say λ. Such a pre-cross connection, within the context of a light-p-cycle configuration γ, can be associated to the product $a_\ell^\gamma a_{\ell'}^\gamma$, with parameters a_ℓ^γ and $a_{\ell'}^\gamma$ similarly defined as a_ℓ^c and $a_{\ell'}^c$ in Subsection 6.3.1. The product $a_\ell^\gamma a_{\ell'}^\gamma$ is equal to 1 if there is an operational pre-cross connection with incoming link ℓ and outgoing link ℓ' in light-cycle configuration γ, 0 otherwise.

At time period $t-1$, a pre-cross connection $a_\ell^\gamma a_{\ell'}^\gamma$ on p-cycle $\gamma = (c,\lambda) \in \Gamma_\lambda$ can be reused during time period t in a light-p-cycle $\gamma' = (c',\lambda') \in \Gamma^{\lambda'}$, if and only if light-$p$-cycle γ is dismantled and $\lambda = \lambda'$. An example of such a case is illustrated in Figure 6.4 where we show the evolution of a light-p-cycle from period $t-1$ to period t. While light-p-cycle γ_1, in operation during period $t-1$, is dismantled one period later, the pre-cross connections at nodes v_4 and v_5 can be preserved and reused in light-p-cycle γ_2 at time period t. The pre-cross connection cost of light-p-cycle γ_2 is therefore reduced to 3 (instead of 5 if we do care of re-using the previously established pre-cross connections).

Coming back to the analytical expression of the objective function f_2^{OBJ}, the term called "Part1" counts the number of pre-cross connections of the light-cycle configurations, and subtract from it (term "Part2") those which have been already set in eliminated light-p-cycles and can therefore be reused.



(a) Time period $t-1$ (b) Time period t

Figure 6.4 – Reuse of pre-cross connections

The set of constraints follows and is very similar to the one of Model 1, with the additional concern about the wavelength continuity constraints in order to be able to compute accurately the number of pre-cross connections, see constraints (6.8). Consequently, instead of dealing with cycles, we deal with light-cycles.

$$\sum_{\gamma\in\Gamma}(a_\ell^\gamma+s_\ell^\gamma)\,y_\gamma-\sum_{\gamma\in\Gamma^{t-1}}(a_\ell^\gamma+s_\ell^\gamma)\,z_\gamma\geq w_\ell-\sum_{\gamma\in\Gamma^{t-1}}(a_\ell^\gamma+s_\ell^\gamma)\,\tilde{y}_\gamma \quad \ell\in L \tag{6.6}$$

$$\sum_{\gamma\in\Gamma}a_\ell^\gamma y_\gamma-\sum_{\gamma\in\Gamma^{t-1}}a_\ell^\gamma z_\gamma\leq c_\ell^{\mathrm{R}}-\sum_{\gamma\in\Gamma^{t-1}}a_\ell^\gamma\tilde{y}_\gamma \quad \ell\in L \tag{6.7}$$

$$\sum_{\gamma\in\Gamma^\lambda}a_\ell^\gamma y_\gamma-\sum_{\gamma\in\Gamma_\lambda^{t-1}}a_\ell^\gamma z_\gamma\leq 1-\sum_{\gamma\in\Gamma_\lambda^{t-1}}a_\ell^\gamma\tilde{y}_\gamma \quad \ell\in L,\lambda\in\Lambda \tag{6.8}$$

$$\sum_{\gamma\in\Gamma^{t-1}}z_\gamma\leq\overline{z} \tag{6.9}$$

$$y_\gamma,z_\gamma\in\{0,1\} \qquad \gamma\in\Gamma \tag{6.10}$$

Constraints (6.6) ensure that the overall working traffic on each link is protected, again taking into account the legacy light-p-cycles and their available protection re-

sources. Constraints (6.7) ensure that the overall protection capacity does not exceed the available residual capacity on each link, while constraint (6.9) ensures that the number of dismantled copies does not exceed a given threshold.

6.4 Solving the ILP Models

We discuss here how to solve efficiently the mathematical models developed in the previous section.

6.4.1 Large Scale Optimization Tools

As for the efficient computation of *p*-cycles in a static model, we have different choices for solving the models developed in the previous section. Obviously, enumerating explicitly all possible cycles or light-cycles will not lead to a scalable solution scheme. Therefore, we can either develop a heuristic in order to generate explicitly only the most promising cycles or light-cycles as in [15], assuming we can define a good metric to identify the most promising, or we can generate implicitly all potential cycles or light-cycles with the use of the column generation techniques combined with either a heuristic or a branch-and-price for solving the ILP part as in, e.g., [56]. We remind the reader that column generation techniques are for solving a linear program (LP) with a large/huge number of variables, and where the coefficients of the constraint matrix can be implicitly defined, see, e.g., Chvatal [12] or Lasdon [63] for more details. We propose to go on with the use of column generation techniques and to devise how to derive an optimal or a near optimal ILP solution once we have computed the optimal Linear Programming (LP) solution.

The first step is to devise a decomposition scheme, in order to avoid the *explicit* enumeration of all potential cycles, a clear obstacle in order to envision a scalable solution scheme. For doing so, the original problem is decomposed into a so-called master problem corresponding here to the model developed in Section 6.3 and a pricing problem corresponding here to the on-line generation of a new cycle/light-cycle which is guaranteed to improve the current LP solution (again, the reader who is not familiar with linear

programming or column generation is referred to, e.g., [12] or [63]).

6.4.2 Pricing Problems

In this section, we will develop the pricing problem (generation of a new potential p-cycle) for Model 1, and let the reader deduce the pricing problem for Model 2 (generation of a new light-p-cycle). Both pricing problems are identical except for the expression of the reduced cost, as the objective of Model 2 differs from the one of Model 1. Whether we deal with the generation of a cycle or a light-cycle does not affect the set of constraints due to the wavelength continuity constraint : it only means that each light-cycle is the combination of a cycle and a wavelength.

We now describe the objective and set of constraints for the pricing problem of Model 1. The improvement guarantee of a new cycle is obtained thanks to the identification of a cycle with a negative reduced cost, where the reduced cost, an LP indicator (see, e.g., [12]), is defined as follows, for Model 1 :

$$\overline{\text{COST}}_c = \text{COST}_c - \sum_{\ell \in L} u_\ell^1 (a_\ell^c + s_\ell^c) + \sum_{\ell \in L} u_\ell^2 a_\ell^c,$$

where COST_c is the unit cost of cycle c, given by $\text{COST}_c = \sum_{\ell \in L} a_\ell^c \text{COST}_\ell$, and where $u_\ell^1 \geq 0$ and $u_\ell^2 \leq 0$ are the values of the dual variables of constraints (6.1) and (6.2) respectively.

The constraints can be written as follows :

$$\sum_{\ell \in omega^{+}(v)} a_\ell = y_v \qquad v \in V \tag{6.11}$$

$$\sum_{\ell \in omega^{-}(v)} a_\ell = y_v \qquad v \in V \tag{6.12}$$

$$\sum_{\ell \in omega^{+}(V')} a_\ell \geq y_v + y_{v'} - 1 \qquad V^{'} \subset V,$$

$$3 \leq |V'| \leq |V| - 3, v \in V', v' \in V \setminus V' \tag{6.13}$$

$$y_v \geq a_\ell + s_\ell \qquad \ell \in \omega(v), v \in V \tag{6.14}$$

$$s_\ell + a_\ell + 1 \geq y_v + y_{v'} - a_{-\ell} \qquad \ell = (v, v') \in L; v, v' \in V \tag{6.15}$$

$$a_\ell + a_{-\ell} \leq 1 \qquad \ell \in L \tag{6.16}$$

$$a_\ell + s_{-\ell} \leq 1 \qquad \ell \in L \tag{6.17}$$

$$y_v \in \{0,1\} \qquad v \in V \tag{6.18}$$

$$a_\ell, s_\ell \in \{0,1\} \qquad \ell \in L \tag{6.19}$$

Constraints (6.11), (6.12) and (6.14-6.17) define conditions to build a cycle and determine the protected links. Constraints (6.11) and (6.12) ensure that the degree of each node is equal to 0 or 2 (each on cycle node has one incoming on cycle link and one outgoing on cycle link). Constraints (6.14) ensure that an adjacent link to an on cycle node cannot be both an on cycle link and a straddling link at the same time, while constraints (6.15) ensure that a link ℓ whose two endpoints are on cycle nodes is considered as an on cycle link or as a straddling link or if none of the two previous cases occur, its reverse link $-\ell$ is an on cycle link. Constraints (6.16) and (6.17) ensure that if a link ℓ is considered as an on cycle link, its reverse link $-\ell$ cannot be considered as an on cycle link or as a straddling link.

Constraints (6.13) are the classical subtour elimination constraints, well known in the context of the Traveling Salesman Problem, see, e.g., [72]. Note that to overcome the difficulty of the exponential number of those constraints, we consider them as lazy constraints, meaning that the pricing problem is first solved without any of the subtour

constraints, and then resolved with the addition of those (or a small number of those) which are violated. Several rounds may be needed before all subtour constraints are satisfied. In practice, only two rounds were necessary in order to get all subtour inequalities satisfied, with an explicit addition of only 1% of them on the average.

6.4.3 Deriving an optimal or a near optimal ILP solution

We first summarize the column generation solution algorithm in which we alternate between the solution of the so-called Restricted Master Problem (RMP) and the pricing problem. We next explain how we derive an optimal or a near optimal integer solution.

Assume we consider time period t. Firstly, some initial columns are generated to define some initial configurations in order to set a first so-called Restricted Master Problem (RMP) with a constraint matrix made of a subset of the columns of the Master Problem. Then, the current RMP is optimally solved, and the optimal values of the dual variables become available. These values are input parameters for the pricing problems. So next, the pricing problem is solved. As soon as the pricing problem reaches a solution with a negative reduced cost, then a so-called augmenting configuration has been found, i.e., a configuration which, if added to the restricted master problem, will improve the current value of its objective function. Note that although solving *exactly* the pricing problem would lead to the best one step ahead improvement of the objective function of the master problem, it is well known that it is more efficient over the long run, to stop its solution as soon as a solution with a negative reduced cost has been reached (compromise between the required time to get an optimal solution and the number of times the pricing problem needs to be solved : it is more efficient, in practice, to solve the pricing problem more often and to use the first solution associated with a negative reduced cost instead of the optimal solution).

The RMP is optimally solved to get new dual variables after the solution of the pricing problem, and then, the solution process resumes to solving again the pricing problem. The process is repeated until the minimum reduced cost of the pricing problem is positive, in which case we can conclude that the optimal solution of the LP relaxation of the Master Problem has been reached. Note that in this section, terms configurations and

columns will be used interchangeably as adding a configuration to the master problem corresponds to adding a column to its constraint matrix.

Once the LP relaxation has been optimally solved using the column generation technique, we need to obtain an optimal or a near optimal integer solution.

In order to solve exactly the ILP model assuming the use of Column Generation (CG) techniques for solving the LP relaxation, one needs to use branch-and-cut methods, see Barnes *et al.* [5]. However, in practice, those methods may not be scalable and alternate scalable solutions exist, while still offering information on the ILP solution accuracy. Here, we use a branch-and-bound method (the one embedded in CPLEX [50]) on the constraint matrix made of the columns generated in order to reach the optimal solution of the LP relaxation.

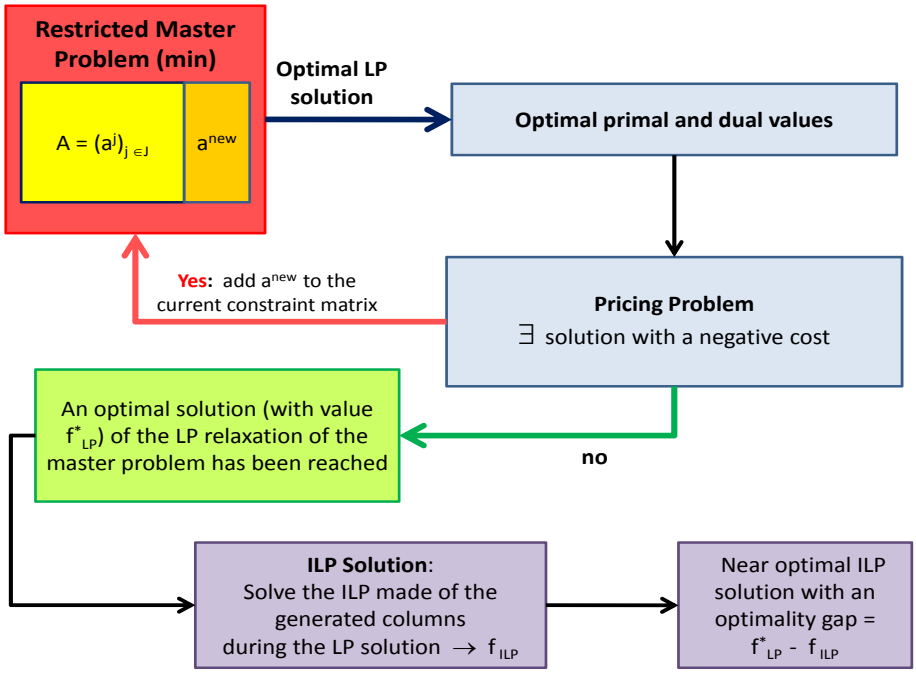


Figure 6.5 – ILP and Column Generation Algorithm

The overall solution scheme, i.e., column generation and derivation of an ILP solution, is summarized in the diagram of Fig. 6.5, using a generic notation for the constraints.

Let us denote by $f^{\star}_{\mathrm{LP}}$ the optimal solution of the linear relaxation of the mathematical formulation of one of the model of Section 6.3. Very often, $f^{\star}_{\mathrm{LP}}$ is not associated with

an integer vector but only provides a lower bound on the optimal integer value, say $f^{\star}_{\mathrm{ILP}}$. Solving the ILP deduced from the constraint matrix derived from the set of columns generated (with the use of a rounding off procedure) in order to obtain $f^{\star}_{\mathrm{LP}}$ leads to a feasible integer solution whose value, $\tilde{f}_{\mathrm{ILP}}$, is often near optimal, i.e., such that the optimality gap $\tilde{f}_{\mathrm{ILP}} - f^{\star}_{\mathrm{LP}}$ is often very small. This is what we will observe in the numerical results described in Section 6.5.

6.4.4 Building an initial solution for Model 2

At the outset time period, the objective is to minimize the number of pre-cross connections which are required in order to make all light-p-cycles operational. The expression of both the objective and of the constraints are simplified, so as to only deal with p-cycles and not with light-p-cycles. It can be written as follows :

$$\min f_2^{\mathrm{OBJ}(t=0)} = \sum_{\ell \in L} \sum_{c \in \mathscr{C}} a_\ell^c \, y^t_{p(\gamma)}$$

$$\sum_{c \in \mathscr{C}} (a_\ell^c + s_\ell^c) \, y^t_{p(\gamma)} \geq w_\ell \qquad \ell \in L \tag{6.20}$$

$$\sum_{c \in \mathscr{C}} a_\ell^c \, y^t_{p(\gamma)} \leq W \qquad \ell \in L \tag{6.21}$$

$$y^t_{p(\gamma)} \in \mathrm{Z}^+ \qquad c \in \mathscr{C} \tag{6.22}$$

The ILP solution of the above model provides the number of copies $y^t_{p(\gamma)}$ of each selected p-cycle configuration. In order to be able to solve Model 2 in the subsequent time periods, we need to associate a wavelength to each copy of a p-cycle, while minimizing the number of required wavelengths. This corresponds to a simple wavelength assignment problem that we decided to solve using a reformulation of it as a graph coloring problem on an undirected conflict graph $G_c = (V_c, L_c)$, where each node $v \in V_c$ of the conflict graph is associated with one p-cycle copy. An edge exists between two nodes of the conflict graph if the two p-cycle copies associated with the nodes share a link.

We used the classical and well known DSATUR heuristic (Brélaz [9]) to solve the graph coloring problem.

6.5 Computational Results

We first describe the data instances and next the experiments we conducted with the two models. For each objective (or model), we evaluate the solutions, i.e., protection schemes, of the two models using the two metrics induced by the two objectives, i.e., the protection bandwidth requirements (even if not directly optimized in Model 2) and the number of new pre-cross connections that need to be established after each traffic variation (even if not directly optimized in Model 1).

6.5.1 Network and Traffic Instances

We use two network instances, the EONET network, and the highly connected NY network. Key characteristics of the two networks are provided in Table 6.I. We use unit link costs.

Network	$\|V\|$	$\|L\|$	Average Node Degree
EONET	20	78	3.9
NY	16	98	6.1

Tableau 6.I – Network Instances

We assume that, at the outset, we already have 2,000 established request connections, for all traffic instances. We randomly generated them, with a uniform distribution on the set of origin and destination node pairs. The initial set of p-cycles is computed using, e.g., the column generation algorithm of [90] for simple p-cycles. Then, wavelength assignment is made using a reformulation of the wavelength assignment problem into a graph coloring problem, solved using the classical DSATUR heuristic [9]. At each time period, some connections are torn down and new incoming ones are established. We consider a traffic scenario with a stable overall traffic, i.e., with a number of incoming requests equal to the number of torn down ones. Let $d^t_{sd\uparrow}$ be the number of added traffic requests from v_s to v_d and $d^t_{sd\downarrow}$ the number of dropped traffic requests from v_s to v_d during

time period t. Let ADD_t (resp. DROP_t) be the percentage of all added (resp. dropped) traffic requests at time period t, defined by the ratios

$$\text{ADD}_t = \frac{\sum_{\{v_s,v_d\}\in\mathscr{SD}} d^t_{sd\uparrow}}{\sum_{\{v_s,v_d\}\in\mathscr{SD}} d_{sd}} \times 100$$

$$\text{DROP}_t = \frac{\sum_{\{v_s,v_d\}\in\mathscr{SD}} d^t_{sd\downarrow}}{\sum_{\{v_s,v_d\}\in\mathscr{SD}} d_{sd}} \times 100.$$

where $\mathscr{SD}$ denotes the set of source and destination node pairs with some traffic.

In the case of stable traffic, $\text{ADD}^t = \text{DROP}^t$.

In order that the parameters properly reflect the added/dropped traffic that impact the settings of the optical bypasses, we generated traffic instances such that

$$d^t_{sd\uparrow} \times d^t_{sd\downarrow} = 0$$

so that, for a given traffic instance, we do not have, e.g., both 10 added and 10 dropped traffic units between $\{v_i, v_j\}$ at the same time period, i.e., an artificial traffic turn over, at least from the perspective of the equipment resetting. It can be viewed as performing a pre-processing so as to eliminate the added connection requests with identical provisioning needs than the dropped ones.

Programs were run on a dual core AMD Opteron machine (2.5 Ghz, 16Gb RAM), using the CPLEX package [50].

For each set of experiments, we provide two graphics : in the first one, we compare the performance (with respect to a given criterion) without/with limits on the lengths of the p-cycles, and in the second one, only the curves without assuming any length limit on the p-cycles are provided in order to facilitate the comparison of the compared curves. In each graphic, we provide results for the two models. For Model 1, in the left graphic, we provide the results for two turn over rates (5% and 10%), , i.e., the percentage of already established p-cycle copies that can be altered at each time period (relative value of $\overline{z}$), without and with a limit of 10 links on the length of the p-cycles. For Model 2, we

provide the results for a single turn over rate (10 %) which, in practice, is never reached, and again without and with a limit of 10 links on the length of the *p*-cycles.

We observed that, while there are performance differences when imposing limits on the lengths of the *p*-cycles, they are meaningless for the second objective (minimizing the number of optical bypass reconfigurations). Differences are more apparent, without being drastic, for the first objective. It is well known that the length of the *p*-cycles may have an impact on the recovery delay, as in spite of the pre-cross connected features, the signal may need to be regenerated or at least amplified depending on the lengths of the optical links. See Section 6.5.6 for more detailed comments.

To simplify the presentation of the results, we will consistently use the terminology *p*-cycles, while to be accurate, we should use *p*-cycles in the context of Model 1, and light-*p*-cycles in the context of Model 2.

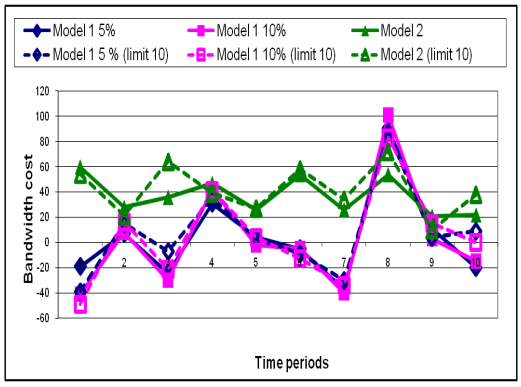
6.5.2 Solution Qualities and Characteristics

Although we did not develop a branch-and-price method in order to guarantee the full optimality of the solutions, thanks to the good qualities of the lower and upper bounds, the optimality gaps were all smaller than 1%, sometimes even of the order of 0.1 % or less. Such a small optimality gap makes the development of a branch-and-price algorithm meaningless. In addition, the computing times required to obtain such high quality solutions are smaller than 10 seconds, therefore the proposed solution schemes for solving the two optimization models are highly scalable, taking into account that several improvements could be easily added to reduce further the computing times (e.g., a heuristic for a "warm" start instead of starting from scratch the solution of the linear programs).

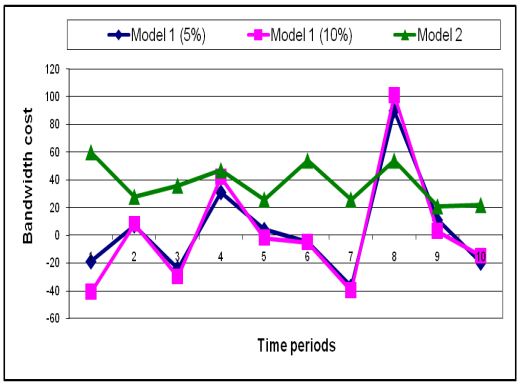
6.5.3 Bandwidth Cost under Dynamic Traffic

The bandwidth variation cost, at time period t, is the difference between the overall protection bandwidth cost at time period t and the overall protection bandwidth cost at time period $t-1$. Results are summarized in Figure 6.6, where time periods are on the horizontal axis, and bandwidth variation cost on the vertical one. We did experiments

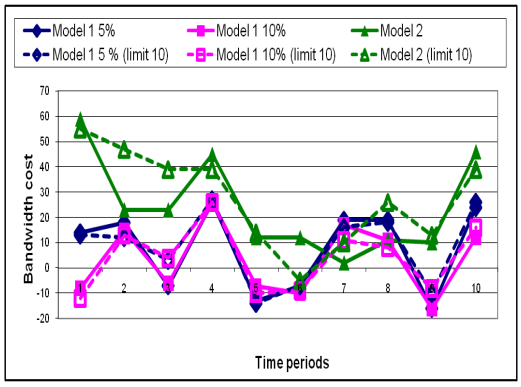
with two turn over rates, 5 % and 10 % for the first model, and 10 % for the second one. For the second model, even if the threshold value was set to 10 %, the number of altered *p*-cycle copies never exceeded an average value of 1.5 % in all traffic scenarios for the two network instances. We observe that the first model gives the best results in



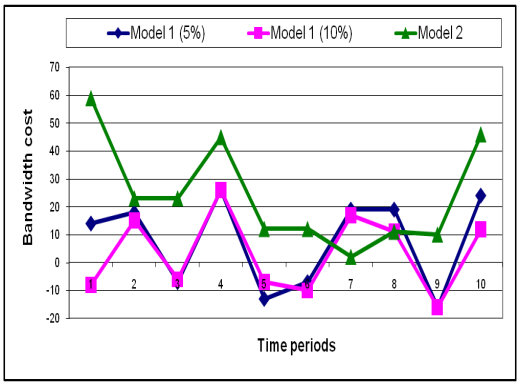
(a) EONET network : with/without length limits on the *p*-cycles



(b) EONET network : without any length limits on the *p*-cycles



(c) NY network : with/without length limits on the *p*-cycles



(d) NY network : without any length limit on the *p*-cycles

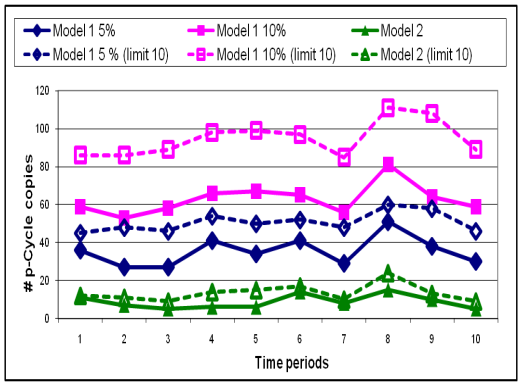
Figure 6.6 – Variations of the *p*-Cycle bandwidth costs in Models 1 & 2

terms of bandwidth cost. The best quality with respect to the cost is obtained by the first model with turn over rate of 10 %. Note that, however, for Model 1, results do not vary very much whether $\overline{z}$ is set to 5% or 10%, see [68] for additional results. Recall that the objective function of the first model is to minimize the overall bandwidth cost. However, even if the objective function of the second model is to minimize the number of optical

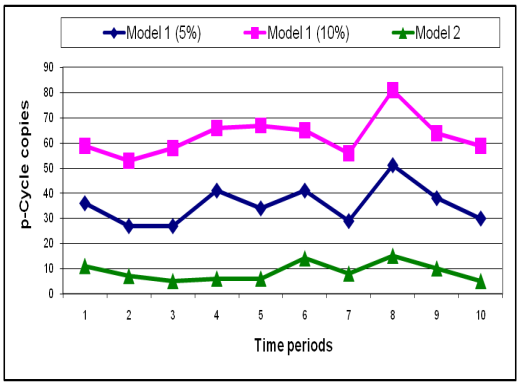
bypass (re)configurations at each time period, we can see that the bandwidth cost is not very high compared to the the bandwidth cost of the first model.

6.5.4 *p*-Cycle stability under dynamic traffic

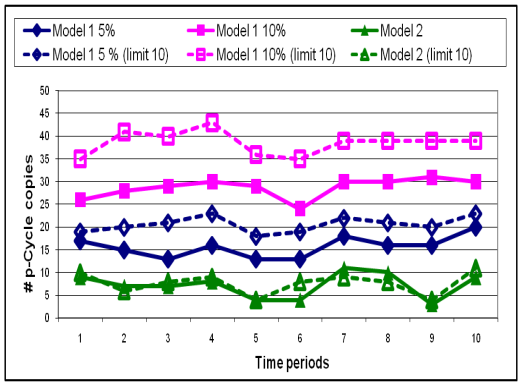
We investigate here the stability of the *p*-cycles under dynamic traffic using the number of reconfigured *p*-cycles as a stability indicator. Figure 6.7 shows the number of



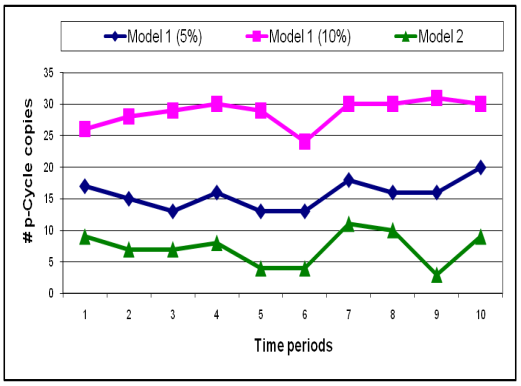
(a) EONET network : with/without length limits on the *p*-cycles



(b) EONET network : without any length limit on the *p*-cycles



(c) NY network : with/without length limits on the *p*-cycles



(d) NY network : without any length limit on the *p*-cycles

Figure 6.7 – Stability of *p*-cycles

reconfigured/added *p*-cycles at each time period. It is evaluated, at time period t with the sum of the number of altered *p*-cycles and of the number of added *p*-cycles. In practice, it corresponds to the number of *p*-cycles a network operator would have to set up

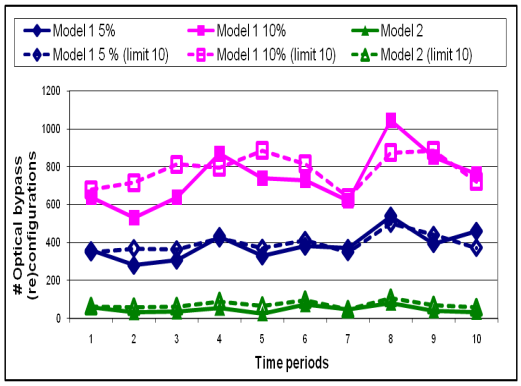
or modify at each time period. We can see that Model 2 clearly outperforms Model 1 in terms of the number of reconfigured/added *p*-cycles. Indeed, the average number of reconfigured/added *p*-cycles for Model 2 varies between 7.2 and 7.7 for NY network (see Figure 6.7(c)) and between 8.7 and 13.4 for EONET network (see Figure 6.7(a)), while the average number of reconfigured *p*-cycles for Model 1 with a 10 % alteration threshold ($\overline{z}$), varies between 28.7 and 38.6 for NY network and between 62.8 and 94.8 for EONET network. Thus, with Model 2, the number of reconfigured *p*-cycles can be reduced by 80 % in NY network and by 86.1 % in EONET network, in comparison with Model 1 assuming a 10 % alteration rate (which, in addition, is far from being reached).

One may ask how the current model compares with a protected working capacity envelope (PWCE) model. For a given network topology with specified transport capacities, the PWCE model [92] first assigns some spare capacity to each edge to create an envelope of protected working capacities. It is therefore attractive because it does not need protection resources to be dynamically configured. Zhang *et al.* [123] compared the PWCE protection paradigm with the classical one where primary resources are set prior to protection ones, with the objective of maximizing the grade of service. In the context of 100% provisioning, the question would be : what are the minimum required transport capacities in order to guarantee full protection under any dynamic pattern ? It is not an easy question to answer. If dynamic traffic patterns cannot be accurately forecast, it appears difficult to guarantee a 100% protected provisioning with PWCE without over-provisioning the network. On the other hand, in the case of known dynamic traffic patterns, a good PWCE may be quite close to the set of stable *p*-cycles, leading to a kind of convergence of the two protection paradigms.

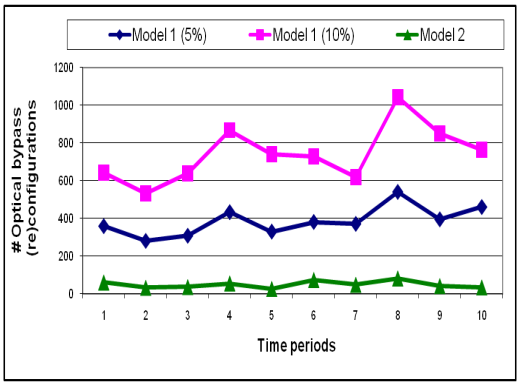
p-cycle-based PWCE design considered here achieves lower blocking probabilities than the dynamic p-cycle design proposed earlier by us and can achieve virtually the same blocking probability as the shared path protection scheme in spite of its substantially lower computational complexity.

6.5.5 Pre-cross connection updates

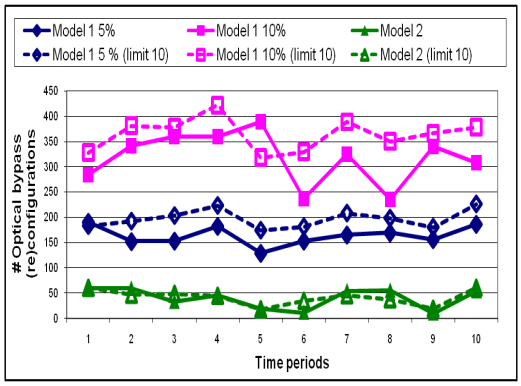
The second indicator of stability that we now investigate is the number of pre-cross connection updates at each time period. Figure 6.8 shows that the average number of



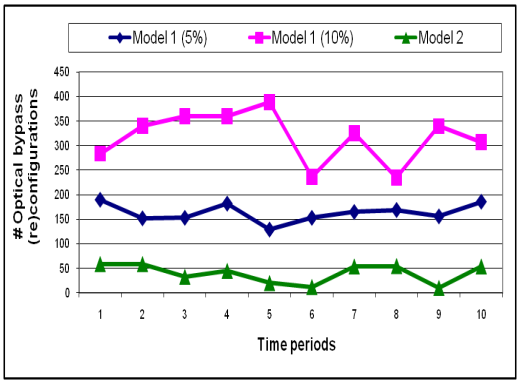
(a) EONET network : with/without length limits on the *p*-cycles



(b) EONET network : without any length limit on the *p*-cycles



(c) NY network : with/without length limits on the *p*-cycles



(d) NY network : without any length limit on the *p*-cycles

Figure 6.8 – Re-using pre-cross connections

pre-cross connection updates for Model 2 varies between 40.1 and 41.9 for NY network (see Figure 6.8(c)) and between 48.8 and 72.9 for EONET network (see Figure 6.8(a)), while the average number of pre-cross connection updates for Model 1 with a 10 % turn over rate, varies between 317.8 and 363.9 for NY network and between 742.3 and 782.0 for EONET network. Thus, with Model 2, the number of pre-cross connection updates can be reduced by 88.5 % in NY network and by 93.4 % in EONET network, compared

to Model 1 with a 10 % traffic turn over rate. It is a huge difference.

If we turn our attention to the results in section 6.5.4, we can see that the reduction made by Model 2 in terms of number of reconfigured/added *p*-cycles compared to Model 1 varies between 74.9 % and 80 % in NY network and between 85.9 % and 86.1 % in EONET network, while the reduction in terms of the number of pre-cross connection updates varies between 87.4 % and 88.5 % in NY network and between 90.7 % and 93.4 % in EONET network. The reduction (Model 2 vs. Model 1) in terms of the number of pre-cross connection updates is therefore very significant compared to its reduction in terms of the number of *p*-cycle copies, and can be explained by : *(i)* the average length of the reconfigured/added *p*-cycles is greater in Model 1 and *(ii)* pre-cross connections of the reconfigured/added *p*-cycles in Model 2 are saved ones from the previously established *p*-cycles that are re-used in some of the newly established *p*-cycles.

6.5.6 Effect of limiting *p*-cycle lengths

The first observation we made is that, if we do not limit the length of *p*-cycles, the longest one is an Hamiltonian cycle in NY network and a 19 node cycle in the 20 node EONET network (note that it is impossible to get a Hamiltonian cycle in the EONET network due to the characteristics of the topology). The effect of limiting the lengths of the *p*-cycles is the same on the two models : performances (i.e., bandwidth costs and number of optical bypass reconfigurations) are better when we do not limit the length of *p*-cycles. Let us discuss the results of Model 1 with a 10 % turn over rate (results are very similar when the turn over rate is equal to 5 %). The bandwidth costs are slightly larger when we limit the length of *p*-cycles. They are 3.9 (Model 1) and 27.7 (Model 2) in NY network compared to 3.4 and 24.3 when there is no limit on the length. In EONET network, they are 4.2 (Model 1) and 41.7 (Model 2) compared to 2.1 and 37.4 without any limit.

The effect on the number of *p*-cycle copies is, however, significant in Model 1. The average number of manipulated *p*-cycle copies is equal to 38.6 (Model 1) and 7.7 (Model 2) in NY network compared to 28.7 and 7.2 when there is no limit on the *p*-cycle lengths. Those values are 94.8 (Model 1) and 13.4 (model 2) compared to 62.8 and 8.7 when we

do not put a limit to *p*-cycle lengths in the EONET network.

The average number of bypass (re)configurations is equal to 363.9 (Model 1) and 41.9 (Model 2) in NY network when there is a limit on the length of *p*-cycles compared to 317.8 and 40.1 without limit. In EONET network the values are 782 (Model 1) and 72.9 (Model 2) compared to 742.3 and 48.8. The effect on the number of bypass (re)configurations is not as significant as on the number of *p*-cycle copies. This is due to the length of *p*-cycles in both cases (with and without any limit). Indeed, when there is a limit on the *p*-cycle lengths, the average length of all *p*-cycles is equal to 9.59 (Model 1) and 7.06 (Model 2) in NY network and 8.64 (Model 1) and 7.29 (Model 2) in EONET network. However when there is no limit on the length of *p*-cycles, the average length of all *p*-cycles is equal to 13.39 (Model 1) and 8.34 (Model 2) in NY network and 12.99 (Model 1) and 9.25 (Model 2) in EONET network .

6.6 Conclusion

In this paper, we revisited the controversial issue of the stability of *p*-cycles under dynamic traffic. While we first investigated this issue in [68] and found out that the stability was moderate under the classical objective of the minimum spare capacity protection design, numerical results showed that we indeed have a significant stability of the *p*-cycles when we grant the new incoming requests with the objective of minimizing the number of OXC pre-cross connection reconfigurations. The effect of limiting the *p*-cycle lengths has also been investigated, and we observed that, under the objective of minimizing the optical bypass reconfigurations, very few lengthly *p*-cycles are generated. Last, numerical results evidenced the high scalability of the optimization models which were designed to conduct the paper study and therefore to perform *p*-cycle updates under dynamic traffic.

Future work will include the study of the stability of the shared path protection scheme under the same objective, and the investigation of how many optical bypass reconfigurations are needed, due to the link protection share, under dynamic traffic.

CHAPITRE 7

STABILITY OF FIPP *P*-CYCLES UNDER DYNAMIC TRAFFIC IN WDM NETWORKS

Ammar Metnani and Brigitte Jaumard

Préambule : Contrairement aux *p*-cycles de base qui ont pour but de protéger chacun des liens du réseau, les *p*-cycles protégeant les connexions ou FIPP *p*-cycles protègent les connexions ou les flots de données de bout-en-bout. Une connexion est protégée par un seul FIPP *p*-cycle.

Dans un contexte de trafic dynamique, le but est de protéger toutes les connexions du réseau contre les pannes simples de liens. Dans la mesure où on est en présence d'un changement continuel du trafic que chaque FIPP *p*-cycle protège, la stabilité des *p*-cycles devient très importante.

Nous proposons un modèle mathématique pour augmenter la stabilité des FIPP *p*-cycles. En présence de la contrainte de continuité de longueurs d'onde, nous prenons en considération l'affectation de longueur d'onde pour chaque FIPP *p*-cycle.

Nous utilisons la technique de génération de colonnes pour résoudre le problème. Dans la mesure où le problème auxiliaire doit définir un cycle ainsi que les connexions qu'il protège, nous proposons une décomposition hiérarchique du problème auxiliaire en deux sous-problèmes, ce qui réduit le temps d'exécution. L'un définit les cycles de protection et l'autre trouve la meilleure combinaison des connexions qu'il protège.

Abstract-Application opportunities associated with video, voice and data triple play result in a dramatic demand increase in metro transport networks, with traffic patterns becoming increasingly dynamic and difficult to predict. This is driving the need of core networks with a high degree of flexibility and multi granularities to carry traffic.

We propose to investigate the question of what this means in terms of dynamic protection provisioning. In other words, we want to study how much a dynamic traffic affects the protection structures in a survivable WDM network and how stable are the protection structures under dynamic traffic.

While most studies on the stability of protection structures has been conducted on *p*-cycles and link shared protection, we study here the stability of FIPP *p*-cycles under dynamic traffic. For doing so, we design and develop a scalable mathematical model that we solve using large scale optimization tools. Numerical results show that FIPP *p*-cycles are remarkably stable under the evaluation of the number of optical bypass reconfigurations under dynamic traffic.

7.1 Introduction

Following the development of new applications and new connectivity paradigms (e.g., multicast or point-to-multipoint connectivity, or even point-to-cloud connectivity), service providers look at network design options to determine how best to achieve a scalable, reliable and flexible core network. While such designs call for flexible reconfigurable networks, offering network flexibility and reconfigurable lightpaths, one question is how much flexibility do we need for protection provisioning. This is the question that we propose to investigate in this paper.

Quite a few studies have been conducted on dynamic protection provisioning, but very few of them look at the protection stability under dynamic traffic.

Among the recent studies on dynamic protection provisioning, we can cite the following references. Shen and Grover [92] survey various dynamic provisioning protection methods and compare them from the aspects of operational complexity and blocking performance. Yang *et al.* [113] look at dynamic and flexible bandwidth on demand in

Metro agile all-optical ring networks. But none of these references studied the stability of the protection provisioning.

The only studies we are aware of with respect to protection stability have been conducted in the context of *p*-cycle based protection mechanisms. *p*-Cycles are kind of a unique protection scheme due to their pre-configured and pre-cross connected characteristics. They are known to offer a protection with a ring-like recovery speed while retaining the capacity efficiency of mesh-based schemes [37]. In case of a link failure, which is the most frequent failure, only the two endpoints of the failing fiber link need to be reconfigured. Capacity efficiency emanates from the protection, by a *p*-cycle, of not only the on-cycle links, but also of the straddling links (i.e., links which although not on-cycle links, have their two endpoints on the cycle). We review below the few studies which have been published within the context of dynamic and asymmetrical traffic in *p*-cycles or their variants, in addition to some concerns about *p*-cycle stability.

He and Somani [42] have quite thoroughly compare the performance of *p*-cycles vs. shared link/path protection with respect to capacity efficiency and recovery time. Their conclusion was that *p*-cycles offer the best compromise : they do use more capacity than the other protection schemes, but pre-cross connection and pre-configuration of the protection patterns (cycles) allow them to be advantageous overall, especially in highly connected networks.

In the context of static traffic, the classical way is firstly to route traffic and then to establish the *p*-cycle protection. In the context of dynamic traffic, where we have to deal with new dynamic incoming/terminating batches of requests, the algorithms that we can find in the literature proceed in the same two step way. This leads to a *p*-cycle protection whose stability and capacity efficiency have been questioned, depending on the traffic changes. Previous studies on dynamic *p*-cycles include the work of Zhong *et al.* [126] where the authors adapted to dynamic traffic the heuristic method *ER-Based Unity-p-cycle* [124] proposed in the context of static traffic, see also Chapter 10 of Grover [32].

In [84], Ruan *et al.* compute, at the outset, a first set of *p*-cycles (i.e., it is an *off-line* generation) such that each link of the network is protected by at least one *p*-cycle, called *primary p-cycle*. The selection of the routing path for a new incoming request is made

on-line at the time the request comes in, in order to fully exploit the fact that the links of that path can be already protected by the set of existing *p*-cycles. If one link of the routing path is not protected, an additional copy of the designated *primary p-cycle* for its protection is set up.

In Zhang *et al.* [121], the authors present a necessary and sufficient condition that guarantees *p*-cycles to have enough protection bandwidth when there is a new incoming request. Based on that condition, the authors propose a heuristic scheme which takes care of the request admission and provisioning. They next compare their heuristic with a path protection scheme, and find out that their heuristic (a link protection scheme) has comparable performance to path protection with respect to the connection satisfaction ratio.

In [68] and [54], Jaumard and Metnani investigated the stability of *p*-cycles under dynamic traffic with different objectives, the required spare bandwidth and the number of optical bypass reconfigurations. Under the latter objective, they observe that the *p*-cycles are stable under dynamic traffic whether the overall amount of traffic remains stable or is incremental. In this study, we extend these works to FIPP *p*-cycles under dynamic traffic. Although we are now dealing with a path protection scheme rather than a link protection one, we observe a similar stability behavior under the objective of minimizing the number of bypass reconfigurations. Mathematical models are, however, more complex, as well as their solution.

The paper is organized as follows. In the next section, we describe the problem statement. In Section 7.3, we describe the optimization model, and its solution in Section 7.4. Numerical results are presented in Section 7.5 and show a remarkable stability of the FIPP *p*-cycles. Conclusions are drawn in the last section.

7.2 Problem Statement : FIPP *p*-cycle Protection Design under Dynamic Traffic

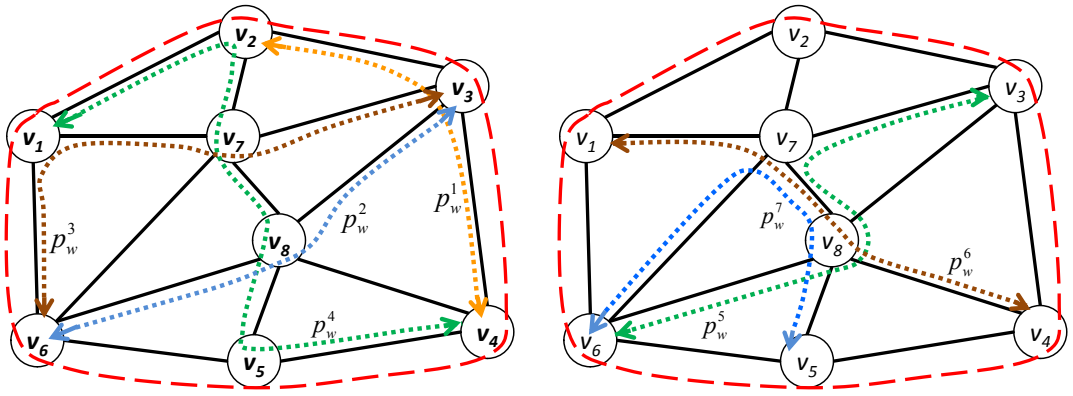
We propose to investigate the problem of the updating of FIPP *p*-cycles under dynamic traffic, and then to study the stability of the FIPP *p*-cycle structures in the context of protection against single link failures. While most studies on *p*-cycles and their variants

look at the minimization either of the spare capacity or of a bandwidth cost, we suggest to examine the minimization of the number of OXC ports, i.e., the number of port (OXC) reconfigurations in the context of a dynamic traffic.

7.2.1 FIPP *p*-cycles

Recall that failure independent path-protecting (FIPP) *p*-cycles [59] are an extension of the basic concept of *p*-cycles to provide end-to-end failure independent path switching against either span or node failures. In a transparent or translucent optical network, the property of pre-cross-connection of protection paths can be even more important than just increasing restoration-speed : when optical protection paths are pre-cross-connected, they can be guaranteed in advance to be operational when required. FIPP *p*-cycles therefore offer a fully pre-cross-connected protection scheme, alternative to SBPP (Shared Bandwidth Path Protection) in which protection paths must be assembled on the fly using spare wavelength channels.

To explain the concept of FIPP *p*-cycles, we first recall the concepts of working and protection paths. A working path between two nodes v_s (source node) and v_d (destination node) is a primary operational path in order to route the traffic between v_s and v_d. A protection path between v_s and v_d is a backup path that is used to reroute the traffic in case of a failure on the working path. A FIPP *p*-cycle is a cyclical protection structure that offers pre-cross connected path protection to any working path such that its two endpoints lie on the cycle, under the assumption that the working path is either fully supported by on-line links (on-line working path), or by no on-line links (straddling working path), or only partially by on-line links (partially on-line working path / partially straddling working path). In practice, most studies have limited themselves to the case of working paths such as those illustrated in Figure 7.1(a) with paths p_w^3 and p_w^4, i.e., made of at most three segments, each segment with consecutive either on-line or non on-line links. In both examples of Figure 7.1, the FIPP *p*-cycle is represented by a red dashed line while the working paths are represented by dotted lines. In Figure 7.1(a), the working path $p_w^2 = v_3 - v_8 - v_6$ is a straddling working path. If, e.g., the link $\{v_6, v_8\}$ fails, protection paths $v_6 - v_1 - v_2 - v_3$ and $v_6 - v_5 - v_4 - v_3$ can be used to restore the



(a) Different types of protected working paths (b) Disjointness requirements

Figure 7.1 – FIPP p-cycle example

traffic between v_3 and v_6. In the same figure, the working path $p_w^1 = v_2 - v_3 - v_4$ is a fully on-cycle one and the working path $p_w^3 = v_6 - v_1 - v_7 - v_3$ is a partially on-cycle (or a partially straddling) one. If, for instance, the on-cycle link $\{v_3, v_4\}$ or $\{v_2, v_3\}$ of working path p_w^1 fails, the protection path $v_2 - v_1 - v_6 - v_5 - v_4$ is used to restore the traffic between v_2 and v_4. A special case of a partially on-cycle working path corresponds to the so-called z-relationship. This is the case of the protection path used to protect the working path $p_w^4 = v_1 - v_2 - v_7 - v_8 - v_5 - v_4$: the protection path depends on which working link fails. For example, the protection path $v_1 - v_6 - v_5 - v_4$ is used to recover from a failure on link $\{v_1, v_2\}$, and the protection path $v_1 - v_2 - v_3 - v_4$ is used to recover from a failure in link $\{v_5, v_4\}$.

In the general case, in order to avoid contention for protection resources in case of a failure, a set of working paths can share the same FIPP p-cycle if :

(i) The working paths are mutually link disjoint, or

(ii) There exists at least a set of protection paths (with one protection path for each working path) which are pairwise link disjoint.

If the disjointness property imposes node-disjoint paths (except for the endpoints), then 100 % node protection is ensured, otherwise only link protection is guaranteed.

In Figure 7.1, working paths $p_w^1 = v_2 - v_3 - v_4$, $p_w^2 = v_3 - v_8 - v_6$ and $p_w^3 = v_6 - v_1 -$

$v_7 - v_3$ can share the same FIPP p-cycle as they are mutually link/node disjoint. Working paths $p_w^6 = v_4 - v_8 - v_7 - v_1$ and $p_w^7 = v_6 - v_7 - v_8 - v_5$ can share the same FIPP p-cycle even if their working paths are not link disjoint as their protection paths are disjoint (use protection path $v_1 - v_2 - v_3 - v_4$ to protect working path p_w^6 and protection path $v_6 - v_5$ to protect working path p_w^7). However, working paths p_w^5 and p_w^6 cannot share the same FIPP p-cycle as they do not meet any of the disjointness conditions.

To the best of our knowledge, there exists no integer linear programming formulation which includes all particular cases. Under the assumption of link protection only, either the z case [59, 83] or the (*ii*) case [56] is taken care but never both cases at the same time. Surprisingly, node protection has never been carefully discussed or dealt with. For the present study, we extended the less complex formulation to the dynamic case, i.e., the one of [83] that encompasses the z relationship, while relaxing (*ii*). We did not require node-disjointness in (*i*).

7.2.2 Dynamic Batch Traffic

Throughout the paper, we will assume that traffic instances are described by a set Φ of traffic flows, indexed by φ, where each traffic flow has a unit wavelength capacity, i.e., corresponds to the aggregation of several requests. For a given pair of source and destination $\{v_s, v_d\}$, we are therefore given the number of symmetrical flows to provision, denoted by d_{sd} (= d_{ds}). We will assume that the routing of the working paths is given, and that it has been performed using a shortest path routing (the same shortest path for the provisioning of all the requests between a given pair of nodes).

Dynamic provisioning can have a different meaning depending on the connection management and control network context. Clearly, in any network, connections do not remain static and the lower the network layer, the less frequent are the changes. An accurate traffic modeling is needed in order to ensure an efficient network provisioning and its ability to survive unpredicted traffic changes. However, depending whether we deal with traffic engineering, or network engineering or network planning (see, e.g., Mukherjee [71] for definitions), dynamic traffic has a different interpretation.

In the context of a core network, even if a wide range of applications may be envisio-

ned and require on-demand connection provisioning, it seems reasonable that a delay in the range of few seconds up to few minutes, depending on the applications, can be reasonably tolerated between connection request and setup times. This is the reason why, in this study, we propose to examine dynamic provisioning within the framework of small-batch provisioning.

Let T be the overall set of time periods, indexed by t, where each time period is associated with a batch of some add/drop traffic flows (to be provisioned/dropped at the end/beginning of the time period). Let Λ be the set of available wavelengths (the same number for each link), with generic index λ and $W = |\Lambda|$. At a given time period t, let Λ^{USED} be the set of wavelengths in use by the protection of the ongoing traffic flows[1].

7.3 Optimization Model

The optimization model that we propose relies on the notion of configurations in order to allow the use of large scale optimization techniques for its solution. We therefore first provide the definition of a configuration (Section 7.3.1), then of the additional parameters and of the variables (Section 7.3.2), before setting the optimization model (Section 7.3.4). Solution of the model will be discussed in Section 7.4.

7.3.1 Notion of configuration

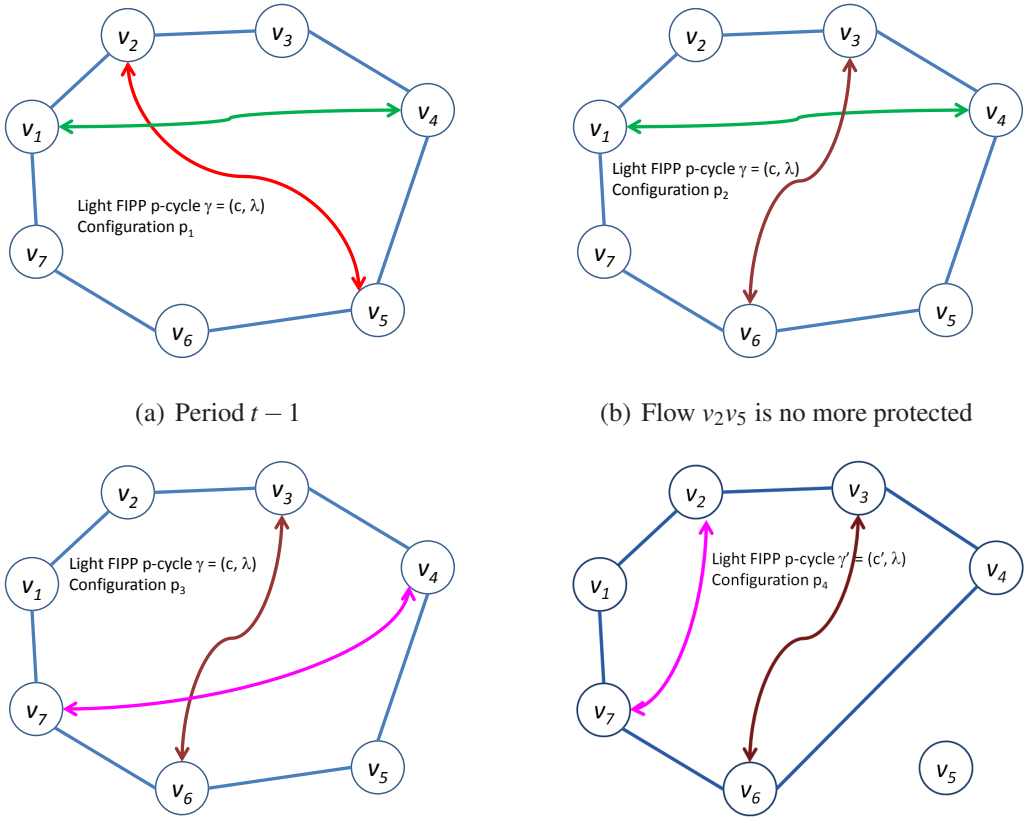
In the context of dynamic traffic, we have to take into account the still needed legacy protection reservations, and therefore we have to deal with the wavelength assignment of the protection reservations of the on-going working routes (and lightpaths). Under the wavelength continuity assumption, the FIPP p-cycles which are used, are such that an identical wavelength is used on all the links of every unit FIPP p-cycle. Consequently, wavelength assignment has to be taken into account in the definition of new FIPP p-cycles or the re-use/modification of the already established ones.

We therefore define the notion of a light-cycle, denoted by γ, similarly to a light-path

[1]Note that assuming different transport capacities for the set of links would correspond to minor modifications in the forthcoming mathematical model.

in [71], by the combination $\gamma = (c, \lambda)$ of a cycle c and a wavelength λ assigned to it.

A configuration $p = (p_{sd}^{\gamma})$ is defined by a light-cycle γ and the set of flows that γ protects as follows : $p_{sd}^{\gamma} \in \{0, 1, 2\}$ defines the number of protection units provided by light-cycle γ to a given unit flow between v_s and v_d. Note that, for a given light-cycle, several potential configurations can be generated, but only one configuration can be selected in the final/optimal solution as each configuration is associated with a wavelength.



(a) Period $t-1$

(b) Flow v_2v_5 is no more protected

(c) New flows can be protected

(d) Reusing some optical bypasses

Figure 7.2 – Possible configuration changes for a given light-cycle $\gamma = (c, \lambda)$

In the present study, we want to investigate the stability of the FIPP p-cycles under dynamic traffic, in terms of the number of optical bypasses that need to be reset from one time period to the next. Two cases can occur : *(i)* the FIPP light p-cycle stays as is, but the flows that are protected by it may vary, *(ii)* the FIPP light p-cycle is not maintained as is, but some of its optical bypasses are saved in order to build a new or new FIPP p-

cycles. The first case is illustrated in Figure 7.2(a) which shows a configuration p where its light-cycle γ protects two flows $\varphi_{14} = v_1v_4$ and $\varphi_{25} = v_2v_5$ at time period $t-1$. At time period t, the flows protected by the light-cycle γ can be :

- The same as in time period $t-1$. The configuration p remains unchanged in time period t.
- Different from the ones in period $t-1$, either partially different as in Figure 7.2(b), or completely different as in Figure 7.2(c). In both cases, the configuration associated with γ changes, but the FIPP light p-cycle remains as is, and therefore there is no need of resetting the optical bypasses.

The second case is illustrated in Figure 7.2(d) where the light p-cycle is modified (it does not include node v_5 anymore), but the reset of its nodal optical bypasses is limited to those at nodes v_4 and v_6, the other ones are inherited from the previous (partially dismantled) light p-cycle from time period $t-1$.

The above examples shows that, even if configurations change, their associated light-cycles remain either stable, or can be largely reused with respect to their set of optical bypasses.

7.3.2 Notations

We represent the optical network by a graph $G = (V, L)$ where V represents the set of optical nodes, indexed by v, and where L is associated with the set of optical (undirected) links, indexed by ℓ. Each link is characterized by its transport capacity (i.e., the number of wavelengths or the number of bandwidth units it carries) $\overline{c}_\ell$. To simplify the exposure of the mathematical model, we will assume, from now on, that $\overline{c}_\ell = W$, for all $\ell \in L$.

We develop the mathematical model at an arbitrary time period, say t, taking into account the new incoming and the terminating requests. WP_{sd} represents the set of working routes between v_s and v_d.

Let Γ_λ be the set of all potential light-cycles with wavelength λ and $\Gamma = \bigcup_{\lambda \in \Lambda} \Gamma_\lambda$ be the set of all potential light-cycles, indexed by γ. Let $\mathscr{P}$ be the set of all potential

configurations and $\mathscr{P}_\gamma$ be the set of configurations generated with the use of light-cycle γ. We denote by C^{t-1} the set of FIPP p-cycles which have been established or maintained during the previous time period. Note that C^{t-1} is a very small subset of Γ as we are not dealing with potential, but with established or maintained light-cycles (this is also the reason of the change of notation, i.e., C vs. Γ) We use the following set of binary parameters : $a_{\ell,\gamma} \in \{0,1\}$ such that $a_{\ell,\gamma} = 1$ if $\gamma = (c,\lambda)$ and link $\ell \in L$ is an on-cycle link of p-cycle c, and $a_{\ell,\gamma} = 0$ otherwise.

We next introduce the sets of variables.

7.3.3 Variables

The first set of binary variables, represented by vector $y^t = (y^t_{p(\gamma)}) \in \{0,1\}$ is such that $y^t_{p(\gamma)}$ is equal to 1 if configuration p with light-cycle $\gamma \in \Gamma^t$ is selected during time period t, and equal to 0 otherwise. The second set of binary variables, represented by vector $x^t = (x^t_{p(\gamma)}) \in \{0,1\}$ where $x^t_{p(\gamma)}$ is equal to 1 if configuration p, which makes use of light-p-cycle $\gamma \in C^{t-1}$, is selected during time period t, and equal to 0 otherwise. In other words, y^t is related to the newly built configurations, while x^t is related to the previously generated configurations, and used either as such or "recycled", i.e., with an updated set of protected flows.

Depending on the applications, longer end-to-end delays can be acceptable. We exploit this tolerance in order to briefly disturb the protection of a limited number of tolerant flows, in order to increase the number of protected granted flows and to minimize the protection cost. For this reason, we allow the re-arrangement of a limited number ($\overline{z}$) of copies of light-p-cycles.

The third set of variables, $z = (z^{t-1}_\gamma) \in \{0,1\}$ is such that z^{t-1}_γ is equal to 1 if light p-cycle γ is dismantled during period t in order to grant more efficiently the new incoming requests, and to allow, accordingly, some light-cycle re-arrangement, and equal to 0 otherwise. Note that a FIPP p-cycle can be dismantled either because it is no more useful in order to protect some working flows, or because it is more advantageous to replace it by another (other) more bandwidth efficient light-cycle(s).

7.3.4 Mathematical Model

The mathematical model aims at evaluating the port cost, and the number of optical bypasses (e.g., OXC ports) that need to be reset when updating/modifying FIPP p-cycles following some traffic variations, subject to the wavelength continuity constraints. Those last constraints are expressed through the definition of a light-cycle : each light-cycle is in one to one correspondence with a wavelength.

7.3.4.1 Objective

The objective function, to be minimized, evaluates the number of optical bypass (e.g., OXC port) reconfigurations :

$$f^{\text{OBJ}} = \sum_{\lambda \in \Lambda} \sum_{\gamma \in \Gamma_\lambda} \text{COST}^t_{\gamma(\lambda)} \left(\sum_{p \in \mathscr{P}_\gamma} y^t_{p(\gamma)} \right)$$

where

$$\text{COST}^t_{\gamma(\lambda)} = \sum_{\ell=(v,v') \in L} \sum_{\ell'=(v',v''),v \neq v''} \left(\underbrace{a_{\ell,\gamma} a_{\ell',\gamma}}_{\text{Part 1}} - \underbrace{\sum_{\gamma' \in C^{t-1}_\lambda} (a_{\ell,\gamma} a_{\ell',\gamma})(a_{\ell,\gamma'} a_{\ell',\gamma'}) z^{t-1}_{\gamma'}}_{\text{Part 2}} \right)$$

On each *wavelength* copy of a FIPP light p-cycle, a node pre-cross-connection involves two ports : one input port on the incoming on-cycle link (ℓ) and one output port on the outgoing on-cycle link (ℓ'), with both links assigned to the same wavelength, say λ. Such a pre-cross-connection, within the context of a wavelength configuration (or FIPP light p-cycle $\in \Gamma_\lambda$) γ, is associated with the product $a_{\ell,\gamma} a_{\ell',\gamma}$, with parameters $a_{\ell,\gamma}$ and $a_{\ell',\gamma}$ as defined in Subsection 7.3.2. The product is equal to 1 if there is an operational pre-cross-connection with incoming link ℓ on wavelength λ and outgoing link ℓ' on wavelength λ in configuration $p^{\gamma,\lambda}$, 0 otherwise.

A legacy nodal pre-cross-connection $a_{\ell,\gamma} a_{\ell',\gamma}$ on light p-cycle $\gamma \in \Gamma_\lambda$ that is still operational during time period $t-1$ (meaning that $\gamma \in C^{t-1}_\lambda$) can be reused during time

period t in a light p-cycle $\gamma' \in \Gamma_{\lambda'}$, if and only if light p-cycle γ is partially dismantled, $\lambda = \lambda'$, and links ℓ and ℓ' are involved in the same new light-cycle.

Coming back to the analytical expression of the objective function, the term called "Part 1" is equal to 1 if there is a pre-cross-connection at node v (with incoming link ℓ and outgoing link ℓ', on wavelength λ) and is equal to 0 otherwise, while the term called "Part 2" is equal to 1 if there was already such a pre-cross-connection (which can be reused if useful) at time period $t-1$, and 0 otherwise. Subtracting term 2 to term 1 allow the computation of new nodal pre-cross connections. Not only we need to compute the subtraction for every pair of incoming and outgoing links at every node, but also for every wavelength (under the wavelength continuity assumption).

7.3.4.2 Constraints

The set of constraints can be written as follows.

$$\sum_{\gamma\in\Gamma}\sum_{p\in\mathscr{P}_\gamma} p^{\gamma}_{sd}\, y^{t}_{p(\gamma)} + \sum_{\gamma\in C^{t-1}}\sum_{p\in\mathscr{P}_\gamma} p^{\gamma}_{sd}\, x^{t}_{p(\gamma)} \geq d_{sd} \qquad \{v_s, v_d\} \in \mathscr{S}\mathscr{D} \tag{7.1}$$

$$\sum_{\gamma\in\Gamma_\lambda} a_{\ell,\gamma}\left(\sum_{p\in\mathscr{P}_\gamma} y^{t}_{p(\gamma)}\right) + \sum_{\gamma\in C^{t-1}_{\lambda}} a_{\ell,\gamma}\left(\sum_{p\in\mathscr{P}_\gamma} x^{t}_{p(\gamma)}\right) \leq 1 \qquad \ell \in L, \lambda \in \Lambda \tag{7.2}$$

$$z^{t-1}_{\gamma} + \sum_{p\in\mathscr{P}_\gamma} x^{t}_{p(\gamma)} = 1 \qquad \gamma \in C^{t-1} \tag{7.3}$$

$$\sum_{\gamma\in C^{t-1}} z^{t-1}_{\gamma} \leq \overline{z} \tag{7.4}$$

$$y^{t}_{p(\gamma)}, x^{t}_{p(\gamma)}, z^{t-1}_{\gamma} \in \{0,1\} \qquad \gamma \in \Gamma. \tag{7.5}$$

Constraints (7.1) ensure that the overall working traffic on each link is protected. Note that, in order to alleviate the notations, the index t is not explicitly mentioned in parameters p^{γ}_{sd} and d_{sd}. Constraints (7.2) ensure that a maximum of one light-cycle is

established on each wavelength. Constraints (7.3) ensure that each light-cycle inherited from some previous period is considered in a configuration of the final solution or dismantled. If the light-cycle $\gamma = (c, \lambda)$ associated with configuration p at time period t, was associated with configuration p' at time period $t-1$, note that variable z_{γ}^{t-1} is not equal to 1. Indeed, even if γ is associated with a new configuration, light-cycle γ is not dismantled, see constraints (7.3). In addition, if several configurations are generated for the light-cycle γ, at most one configuration will be considered in the final solution at time period t. Constraint (7.4) ensures that the number of dismantled copies does not exceed a given threshold.

Observe that we do not impose explicitly to reuse the previously established configurations with the same ongoing requests : it is indirectly taken care of with the objective of minimizing the number of bypass reconfigurations. As we will see in Section 7.5, numerical experiments confirmed that point.

7.4 Solving the Optimization Model

We discuss here how to solve efficiently the ILP model developed in the previous section.

7.4.1 Large Scale Optimization Tools

As for the efficient computation of FIPP p-cycles in a static model, we have different choices for solving the mathematical model developed in Section 7.3.4. Obviously, enumerating explicitly all possible light-cycles as well as all possible configurations will not lead to a scalable solution scheme. Therefore, we can either develop a heuristic in order to generate explicitly only the most promising light-cycles, assuming we can define a good metric to identify the most promising ones, or we can generate *implicitly* all potential light-cycles with the use of the column generation technique (see, e.g., [12, 63]) combined with either a heuristic or a branch-and-price for solving the ILP part. We propose to go in this last direction with the use of column generation techniques and to devise how to derive a near optimal ILP solution once we have computed the optimal LP

solution.

Column generation techniques involve the decomposition of the original problem into a master problem corresponding here to the model developed in Section 7.3.4 and a pricing problem corresponding here to the on-line generation of a new light-cycle configuration which is guaranteed to improve the current LP solution (the reader who is not familiar with linear programming or column generation is referred to, e.g., [12] or [63]). The improvement guarantee is obtained thanks to the identification of an "augmenting" light-cycle configuration, i.e., a configuration with a negative reduced cost in the pricing problem, again see, e.g., [12] in case of non familiarity with linear programming tools. The negative reduced cost $\overline{\text{COST}}^t_{\gamma(\lambda)}$ of the light-cycle configuration variable $y^t_{p(\gamma)}$ is defined as follows :

$$\overline{\text{COST}}^t_{\gamma(\lambda)} = \text{COST}^t_{\gamma(\lambda)} - \sum_{\{v_s,v_d\}\in\mathscr{SD}} u^1_{sd}\, p^{\gamma}_{sd} + \sum_{\ell\in L} u^2_{\ell,\lambda}\, a_{\ell,\gamma}$$

where

- $\text{COST}^t_{\gamma(\lambda)}$ is the unit cost (or weight) of light-cycle γ as described in (7.3.4.1) and (7.3.4.1), and
- $u^1_{sd} \geq 0$ and $u^2_{\ell,\lambda} \leq 0$ are the dual variables of constraints (7.1) and (7.2) respectively.

As in [83], we use a decomposition of the pricing problem into two pricing models in order to avoid unnecessary (and somewhat costly) generation of new light-cycles. The first pricing problem, denoted by LCCP, generates an augmenting configuration with a new light-cycle, together with the information of which fraction of the demand is protected by the light-cycle. The second one, denoted by LCPP, generates an augmenting configuration which re-uses one of the previously generated light-cycle (involved in another configuration and provided as an input parameter). Both pricing problems are detailed in the two forthcoming paragraphs.

7.4.1.1 Light Cycle Configuration Pricing (LCCP) Problem

The Light-Cycle Configuration Pricing (LCCP) problem aims at finding a light-cycle configuration that can improve further the current value of the linear programming relaxation. Input parameters are the values of dual variables (in the expression of the reduced cost of variable $y^t_{p(\gamma)}$, i.e., the objective of the LCCP model) and the output is therefore a new light-cycle configuration, i.e., a new light-cycle together with the fraction of the traffic that it protects.

To that end, the LCCP looks at a new light-cycle configuration with minimum reduced cost. If the reduced cost is negative, then there exists an improving configuration, otherwise no further improving light-cycle configuration exists. For a given wavelength λ and set of dual values, the LCCP problem outputs a light-cycle configuration $\gamma = (c, \lambda)$, i.e., builds a cycle and identifies the flows that can be protected by γ.

The LCCP problem has four sets of binary variables :

- $a_\ell \in \{0,1\}, \ell \in L$. a_ℓ is equal to 1 if cycle c crosses the link ℓ, and 0 otherwise
- $y_v \in \{0,1\}, v \in V$. y_v is equal to 1 if node v is traversed by the cycle, and 0 otherwise,
- $t_{sd} \in \{0,1\}, \{v_s, v_d\} \in \mathscr{SD}$. $o_{sd} = 1$ if and only if flow φ_{sd} is protected (whether on-cycle or straddling), and equal to 0 otherwise
- $s_{sd} \in \{0,1\}, \{v_s, v_d\} \in \mathscr{SD}$. $s_{sd} = 1$ if and only if flow φ_{sd} is protected and straddles the cycle, and equal to 0 otherwise.

For $S \subseteq V$, let $\delta(S)$ be the set of links incident to a node in S. The objective function of the LCCP problem for the wavelength λ is defined as follows :

$$\min \ \text{COST}^t_{\gamma(\lambda)} - \sum_{\{v_s,v_d\} \in \mathscr{SD}} u^1_{sd} \left(o_{sd} + s_{sd}\right) + \sum_{\ell \in L} u^2_{\ell,\lambda} a_\ell$$

where $u^1_{sd} \geq 0$ and $u^2_{\ell,\lambda} \leq 0$ are the dual variables of constraints (7.1) and (7.2) respecti-

vely. The set of constraints can be written as follows :

$$\sum_{\ell \in \delta(v)} a_\ell = 2y_v \qquad v \in V \tag{7.6}$$

$$\sum_{\ell \in \delta(S)} a_\ell \geq 2(y_i + y_j - 1) \qquad v_i \in S, v_j \in V \setminus S, S \subset V,$$

$$3 \leq |S| \leq |V| - 3 \tag{7.7}$$

$$o_{sd} \leq y_v \qquad \varphi_{sd} \in \Phi, v \in \{v_s, v_d\} \tag{7.8}$$

$$2s_{sd} + a_\ell - o_{sd} \leq 1 \qquad \varphi_{sd} \in \Phi, \ell \in \mathrm{WP}_{sd} \tag{7.9}$$

$$\sum_{\varphi_{sd} \in \Phi : \mathrm{WP}_{sd} \ni \ell} o_{sd} \leq 1 \qquad \ell \in L \tag{7.10}$$

$$o_{sd}, s_{sd} \in \{0,1\}, \varphi_{sd} \in \Phi \qquad \{v_s, v_d\} \in \mathscr{SD} \tag{7.11}$$

$$a_\ell \in \{0,1\} \qquad \ell \in L \tag{7.12}$$

$$y_v \in \{0,1\} \qquad v \in V. \tag{7.13}$$

Constraints (7.6) force the degree of each node to be 0 or 2, in order to get one or several cycles (any node belonging to a cycle has an even degree, nodes that do not belong to a cycle have a null degree). In order to be able to properly identify straddling links, we cannot generate more than one cycle at a time : this is the purpose of the next set of constraints. Constraints (7.7) ensure that each cut[2] separating two visited nodes (i.e., nodes belonging to the light cycle) is crossed at least twice, in order to ensure the elimination of multiple cycles. Those constraints are often called subtour elimination constraints in the literature on the traveling salesman problem, see,e.g., [72]. Constraints (7.8) ensure that each protected flow has both of its end nodes crossed by the light-cycle. Constraints (7.9) determine whether a protected connection is a straddling flow to the light-cycle or not. Constraints (7.10) ensure that only mutually disjoint flows are protected.

[2]In graph theory, a cut is a partition of the vertices of a graph into two disjoint subsets.

7.4.1.2 Light Cycle Packing Problem (LCPP)

The Light Cycle Packing Problem (LCPP) generates, for given values of the dual variables (in the objective of the LCPP model, made of the reduced cost of variable $y^t_{p(\gamma)}$) and for a given light-cycle $\gamma = (c, \lambda)$, a subset of mutually disjoint flows, which can be protected by γ. Such flows must be associated with a negative reduced cost in order to guarantee that the generated configuration is an augmenting one, i.e., will improve the current value of the objective of the master problem (i.e., the model described in Section 7.3.4).

The LCPP model uses only one set of binary variables :

- $o_{sd}, \{v_s, v_d\} \in \mathscr{SD}$, such that o_{sd} = 1 if and only if the flow φ_{sd} is protected by the light-cycle γ, and $o_{sd} = 0$ otherwise.

Under the assumption that we are given a light cycle, the term $\text{COST}^t_{\gamma(\lambda)} + \sum_{\ell \in L} u^2_{\ell,\lambda} a_{\ell,\gamma}$ is a constant in the reduced cost expression of (7.4.1.1). Consequently, minimizing the reduced cost amounts to maximizing the so-called revenue of the (input provided) traffic flows protected by the light-cycle, defined as follows. Let Φ^γ be the set of flows φ_{sd} such that v_s and v_d are crossed by light-cycle γ. The revenue r_{sd} of flow $\varphi_{sd} \in \Phi^\gamma$ is defined as follows : $r_{sd} = u^1_{sd}$ if flow φ_{sd} is fully or partially on light-cycle γ, otherwise $r_{sd} = 2u^1_{sd}$. The objective function of the LCPP can then be written :

$$\max \quad \sum_{\{v_s, v_d\} \in \mathscr{SD}} r_{sd}\, o_{sd}$$

subject to :

$$\sum_{\varphi_{sd} \in \Phi^\gamma : \text{WP}_{sd} \ni \ell} o_{sd} \leq 1 \qquad \ell \in L \tag{7.10}$$

$$o_{sd} \in \{0, 1\} \qquad \varphi_{sd} \in \Phi^\gamma \tag{7.14}$$

with the same set of constraints, i.e., (7.10), as in the LCCP model.

Figure 7.3 represents an example of a network at time period t. The FIPP p-cycle is represented by red dashed line. The protected flows by this FIPP p-cycle in time period

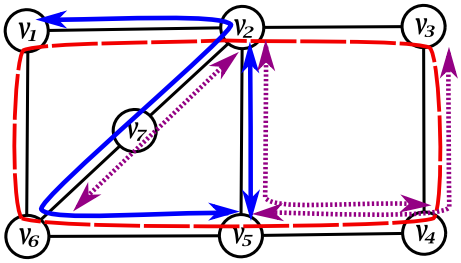


Figure 7.3 – LCPP example

$t-1$, $\varphi_{15} = v_1v_5$ and $\varphi_{25} = v_2v_5$ are represented by solid blue lines. The new flows $\varphi_{26} = v_2v_6$, $\varphi_{24} = v_2v_4$ and $\varphi_{35} = v_3v_5$ are represented by dark violet dotted lines. If we consider that the dual variables u^1_{sd} corresponding to the flows represented in the network are all equal to 1, we can see from the network that the revenues :

- $r_{25} = r_{26} = 2$ (straddling flows),
- $r_{15} = r_{24} = r_{35} = 1$

The set of constraints (7.10) obtained from the links are :

$$
\begin{aligned}
&v_1v_2 : o_{15} \leq 1 \\
&v_2v_5 : o_{24} + o_{25} \leq 1 \\
&v_2v_7 : o_{15} + o_{26} \leq 1 \\
&v_3v_4 : o_{35} \leq 1 \\
&v_4v_5 : o_{24} + o_{35} \leq 1 \\
&v_5v_6 : o_{15} \leq 1 \qquad \text{(redundant)} \\
&v_6v_7 : o_{15} + o_{26} \leq 1 \qquad \text{(redundant)}
\end{aligned}
$$

As a result of the LCPP, given the cycle represented in Figure 7.3, the protected flows are φ_{25}, φ_{26} and φ_{35}. Note that we do not force explicitly the protection of the same flows than in the previous periods, but only implicitly through the overall minimization of the number of optical bypass resets. Here, for instance, the flow φ_{15} is not protected by the cycle although it was protected at time period $t-1$. As will be seen in the nume-

rical results in Section 4.5, in practice, nearly all previously protection relationships are preserved from one time period to the next.

7.4.1.3 Column Generation Algorithm

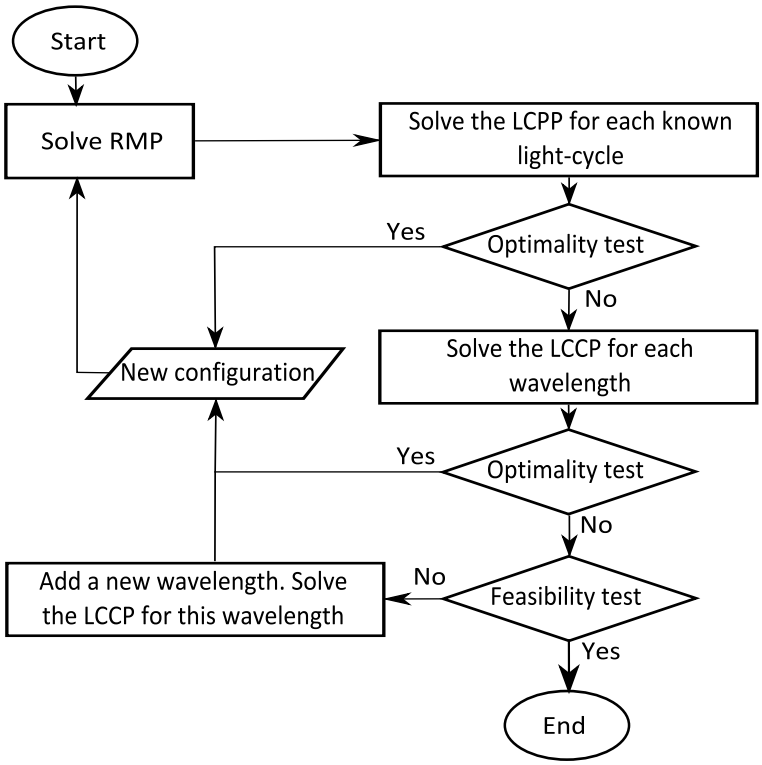


Figure 7.4 – Column generation algorithm

We provide here the detail of the column generation solution algorithm in which we alternate between the two pricing problems. An outline is presented in the flow chart of Figure 7.4, and a more detailed description is given in Algorithm 7.1. Assume we consider time period t.

First, some initial columns are generated to define some initial configurations in order to set a first so-called Restricted Master Problem (RMP) with a constraint matrix made of a subset of the columns of the Master Problem. Then, the current RMP is optimally solved, and the optimal values of the dual variables become available. These values are input parameters for the pricing problems. So next, pricing problems (LCPP

and LCCP) are solved according to the strategy described in Algorithm 7.1. Whenever a pricing problem has a solution with a negative reduced cost, then an so-called augmenting configuration has been found, i.e., a configuration which, if added to the master problem, will improve the current value of the objective function of the restricted master problem. Note that although solving *exactly* those pricing problems would lead to the best one step ahead improvement of the objective function of the master problem, it is well known that it is more efficient over the long run, to stop their solution as soon as a solution with a negative reduced cost has been reached (it has to do with the compromise between the required time to get an optimal solution and the number of times pricing problems are solved : it is more efficient, in practice, to solve pricing problems more often while only using their first solutions associated with a negative reduced cost instead of their optimal solutions).

The RMP is optimally solved to get new dual variables after the solution of any pricing problem, and then, the solution process resumes to solving pricing problems in sequence. The process is repeated until the reduced cost of all pricing problems (LCPP and LCCP) is not negative anymore, in which case we can conclude that the optimal solution of the LP relaxation of the Master Problem has been reached.

In short, the column generation algorithm consists in firstly generating configurations while reusing the established, still operational, light-cycles of the previous time periods, and then, if needed (e.g., in order to improve further the current solution) to generate augmenting configurations with new light-cycles. Note that in this section, terms configurations and columns will be used interchangeably as adding a configuration to the master problem corresponds to adding a column to its constraint matrix.

Note that we may add one column at a time, i.e., as soon as an improving configuration has been found, or several at a time. The advantage of adding one at a time is to benefit from it when generating the next ones, i.e., to generate strictly improving configurations. On the other hand, adding several columns at the same time saves some iterations in order to re-optimize the master problem (in comparison with the sum of iterations it takes in a scheme where columns are successively added one at a time), but at the expense of generating an overall larger number of columns. For those reasons, as

Algorithm 7.1 Column Generation Algorithm

1: CYC$(C^{t-1}) \leftarrow$ set of light-cycles in C^{t-1}
2: CYC$^t \leftarrow$ set of light-cycles generated at time period t
3: CYC$^t = \emptyset$ at the outset of time period t
4: AUGMENT $\leftarrow$ .true.
5:
6: Build an initial solution using a set of artificial columns and the configurations selected in time period $t-1$.
7:
8: **while** AUGMENT = .true., i.e., there exists an improving configuration **do**
9: **for all** $\gamma \in$ CYC$(C^{t-1}) \cup$ CYC^t **do**
10: Solve the LCPP model
11: **if** an augmenting column has been found **then**
12: Add the augmenting column to the master problem
13: Re-optimize the master problem up to optimality
14: go back to Step 9
15: **end if**
16: **end for**
17: AUGMENT $\leftarrow$ .false.
18: **end while**
19:
20: **for all** each wavelength **do**
21: Solve the LCCP model
22: **if** an augmenting column has been found **then**
23: Add the augmenting column to the master problem
24: Re-optimize the master problem up to optimality
25: AUGMENT $\leftarrow$ .true.
26: **end if**
27: **end for**
28: **if** AUGMENT $=$ .true. **then**
29: Go back to Step 8
30: **end if**
31:
32: **if** no augmenting column exists anymore and the current solution is feasible **then**
33: An optimal LP solution has been found : Stop.
34: **else**
35: {The network is missing transport capacity in order to ensure a protection to all connection requests }
36: Add a new wavelength λ to Λ^{USED}
37: Solve the LCCP model with λ
38: If any, add the newly generated configuration to the master problem
39: Restore the optimality conditions for the master problem
40: Go back to Step 8
41: **end if**

will be described in the sequel, we choose to re-optimize the master problem as soon as an augmenting column (configuration) is found.

The strategy that was adopted in Algorithm 7.1 has been selected after various experiments in which we compared different strategies, each of them associated with a different sequence for solving the two pricing problems. It corresponds to the strategy, among all tested ones, which gave the best results in terms of computing times.

In Step 8, the goal is to find, for each already existing light-cycle, a set of ongoing flows that can be protected by this light-cycle and where the resulting configuration improves the current solution. We reach Step 20 when no more augmenting configuration resulting from the already generated (or inherited from previous periods) light-cycles can be found. We then try to find new augmenting configurations, using new light-cycles. When it is no more possible to find new augmenting configurations, either with the previously generated light-cycles, or with new light-cycles, we distinguish two cases. Either we have found a feasible solution (i.e., the current LP solution does not contain any of the artificial columns introduced at the outset), and then the algorithm stops, or we need to add a new wavelength in order to be able to generate new augmenting configurations, in order to obtain a feasible solution that can ultimately allow the protection provisioning of all connection requests.

Once the LP relaxation has been optimally solved using the column generation (CG) technique, we need to obtain an optimal or a near optimal integer solution.

In order to solve exactly the ILP model assuming the use of CG for solving the LP relaxation, one needs to use branch-and-cut methods, see Barnes *et al.* [5]. However, in practice, those methods may not be scalable and alternate scalable solutions exist, while still offering information on the ILP solution accuracy. Here, we use a branch-and-bound method (the one embedded in Cplex [50]) on the constraint matrix made of the columns generated in order to reach the optimal solution of the LP relaxation. In addition, as the number of generated columns was quite large, we had to embed the branch-and-bound in a rounding off scheme in which we iteratively select a fractional variable, round it off to its nearest integer feasible value, and re-optimize the LP relaxation until all variables have integer values.

The overall solution scheme, i.e., column generation and derivation of an ILP solution, is summarized in the diagram of Figure7.5, using generic notation for the constraint constraint.

Let us denote by $f^{\star}_{\mathrm{LP}}$ the optimal solution of the linear relaxation of the mathematical formulation of the model of Section 7.3.4. Very often, $f^{\star}_{\mathrm{LP}}$ is not associated with an integer vector but only provides a lower bound on the optimal integer value, say $f^{\star}_{\mathrm{ILP}}$. Solving the ILP deduced from the constraint matrix derived from the set of columns generated (with the use of a rounding off procedure) in order to obtain $z^{\star}_{\mathrm{LP}}$ leads to a feasible integer solution whose value, $\tilde{f}_{\mathrm{ILP}}$, is often near optimal, i.e., such that the optimality gap $\tilde{f}_{\mathrm{ILP}} - f^{\star}_{\mathrm{LP}}$ is often very small. This is what we will observe in the numerical results described in Section 4.5.

7.4.2 Building an initial solution

At the outset time period, the objective is to minimize the number of pre-cross-connections which are required in order to make all the p-cycles operational. The expression of both the objective and of the constraints are simplified and can be written as follows :

$$\min f^{\mathrm{OBJ}(t=0)} = \sum_{\ell \in L} \sum_{\gamma \in \Gamma} a_{\ell,\gamma}\, y^{t}_{p(\gamma)}$$

$$\sum_{\gamma \in \Gamma} \sum_{p \in \mathscr{P}_{\gamma}} p^{\gamma}_{sd}\, y^{t}_{p(\gamma)} \geq d_{sd} \qquad \{v_s, v_d\} \in \mathscr{SD} \tag{7.15}$$

$$y^{t}_{p(\gamma)} \in \mathrm{Z}^{+} \qquad \gamma \in \Gamma \tag{7.16}$$

The ILP solution of the above model provides the number of copies $y^{t}_{p(\gamma)}$ of each selected p-cycle configuration. In order to be able to solve the model in the subsequent time periods, we need to associate a wavelength with each copy of a p-cycle configuration, while minimizing the number of required wavelengths. This corresponds to a simple wavelength assignment problem that we solved using a reformulation of it as a graph coloring problem on an undirected conflict graph $G_{\gamma} = (V_{\gamma}, L_{\gamma})$, where each node

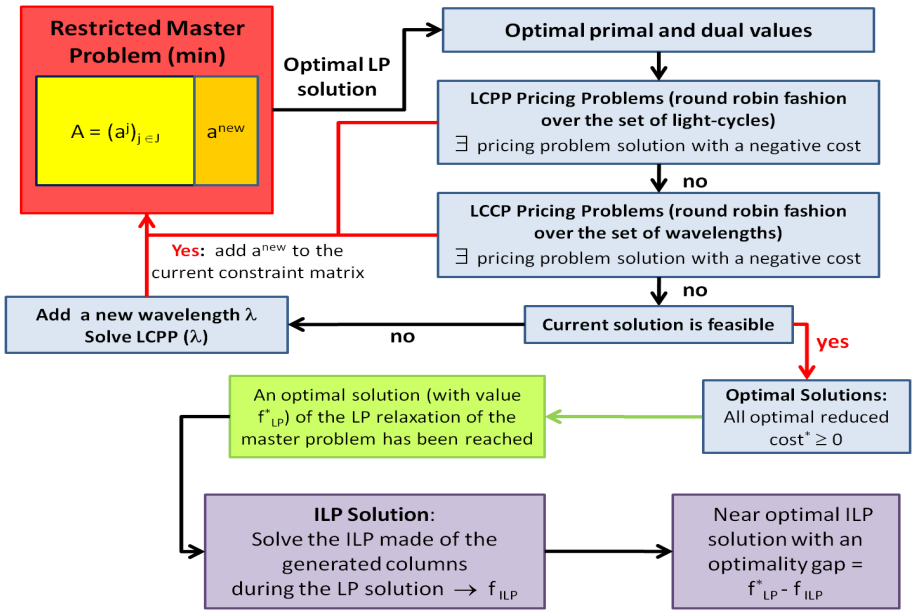


Figure 7.5 – ILP and Column Generation Algorithm

$v \in V_\gamma$ of the conflict graph is associated with one p-cycle configuration copy. An edge exists between two nodes of the conflict graph if the two p-cycle copies associated with the nodes share a link. We used the classical and well known DSATUR heuristic [9] to solve the graph coloring problem.

7.5 Computational Results

We first describe the data instances and next the various experiments we conducted with the model of Section 7.3.4. We evaluate the solutions, i.e., protection schemes, of the model looking at the percentage and characteristics of the modified configurations from one time period to the next.

7.5.1 Network and Traffic Instances

We use the COST239 network (11 nodes, 26 undirected links) and the NY network (16 nodes, 49 undirected links), over 10 time periods.

We assume that, at the outset, we already have an average of 4 established request connections between each $\{v_s, v_d\} \in \mathscr{S}\mathscr{D}$, for all traffic instances. At each time period,

some connections are torn down and new incoming ones are established. We consider two different traffic scenarios, one with a stable overall traffic, and the second one with incremental traffic. Let $d^t_{ij\uparrow}$ be the number of added traffic units of flow φ_{ij} and $d^t_{ij\downarrow}$ the number of dropped traffic units of flow φ_{ij} during time period t. Let ADD_t (resp. DROP_t) be the percentage of all added (resp. dropped) flow units at time period t, defined by the ratios

$$\text{ADD}_t = \frac{\sum\limits_{\{v_s,v_d\}\in\mathscr{SD}} d^t_{ij\uparrow}}{\sum\limits_{\{v_s,v_d\}\in\mathscr{SD}} d_{ij}} \times 100$$

$$\text{DROP}_t = \frac{\sum\limits_{\{v_s,v_d\}\in\mathscr{SD}} d^t_{ij\downarrow}}{\sum\limits_{\{v_s,v_d\}\in\mathscr{SD}} d_{ij}} \times 100.$$

In the case of stable traffic, the number of incoming requests ADD^t is equal to the number of torn down ones DROP^t, while for incremental traffic, the number of incoming requests ADD^t is greater than the number of torn down ones DROP^t.

In order that the parameters properly reflect the added/dropped traffic that impact the settings of the optical bypasses, we generated traffic instances such that

$$d^t_{ij\uparrow} \times d^t_{ij\downarrow} = 0$$

so that, for a given traffic instance, we do not have, e.g., both 10 added and 10 dropped traffic units between $\{v_i, v_j\}$ at the same time period, i.e., an artificial traffic turn over, at least from the perspective of the equipment resetting. It can be viewed as performing a pre-processing so as to eliminate the added connection requests with identical provisioning needs than the dropped ones. In our study, we consider one stable traffic instance with ADD = 5 % and DROP = 5 % for all timeperiods, and one incremental traffic instance with DROP = 5 % and ADD = 10 % for all time periods.

Programs were run on a dual core AMD Opteron machine (2.5 Ghz, 16Gb RAM), using the CPLEX package (release 12).

7.5.2 FIPP *p*-Cycle stability under dynamic traffic

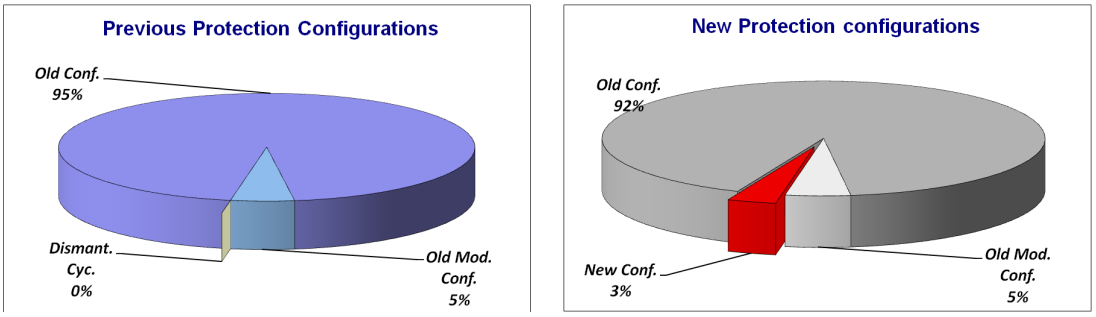
At time period t, three types of configurations can be found in the final solution :

NC type. It is made of configurations generated using new light-cycles,

NOC type. It consists of new configurations generated using the previously generated and established light-cycles $\gamma \in C^{t-1}$,

OC type, i.e., configurations generated during the previous time periods, and which are unchanged (same light-cycles, same protected flows).

We also have the set of dismantled configurations, so-called of DC type. Note that the NOC configurations are generated by the LCPP pricing problem using light-cycles inherited from the previous time period. However, the NC configurations can be either generated by the LCCP pricing problem or by the LCPP pricing problem using a light-cycle generated earlier by the LCCP during the same time period t. For both NOC and OC configurations, the light-cycles associated with the configurations have been established during previous time periods, and hence, are stable in time period t. However, for the first type NC, some resetting operations have been performed at some or at all nodes of the light-cycles associated with those configurations. There is a possibility to reuse some already established optical bypasses from disrupted light-cycles, as discussed in Section 7.3.4.

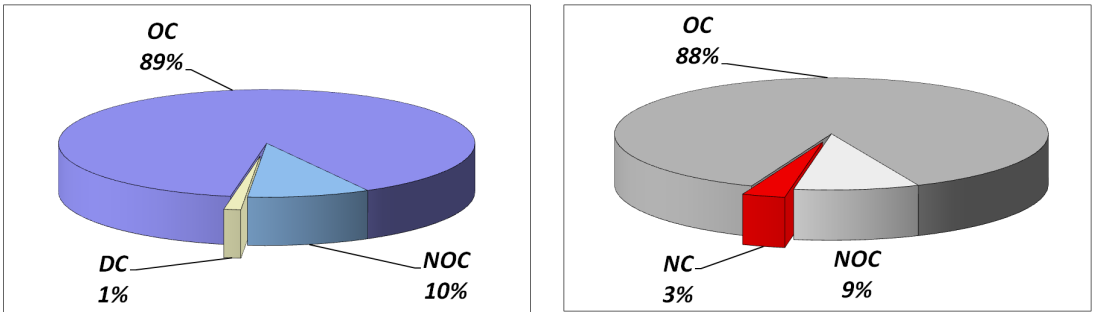


(a) Time period $t-1$ (b) Time period t

Figure 7.6 – Protection Configurations - Stable Dynamic Traffic - COST239 Network

Figures 7.6 and 7.7 represent the percentages of the three protection configuration types for the COST239 and the NY network instances, with a stable dynamic traffic, i.e., same number of added and dropped connection requests, respectively. Percentages are average percentages over the various time periods, compared with their previous time periods. Figures 7.6(a) and 7.6(b) show that, in a stable dynamic traffic environment, only a small percentage of configurations is changed. Indeed, in Figure 7.6(a), we see that all light-cycles of time period $t-1$ are all reused in time period t : No light-cycles of time period $t-1$ are dismantled. Indeed, 95.1 % of the configurations are of OC type (no modifications of the light-cycles and of their protected flows), i.e., are stable configurations inherited from previous time periods ; and in only 4.9 % of all light-cycles, some protected flows have changed. Hence, 100 % of previously established light-cycles are re-used and they are adequate for the protection of the traffic in time period t.

Figure 7.6(b) represents the average distribution of the various types of configurations in each time period. On the average, 91.7 % of established configurations are the same as in the previous time period (OC configurations), and 4.7 % of configurations are associated with light-cycles established in previous time periods (NOC configurations). Only 3.6 % of the configurations are newly generated ones in the current time period t (NC configurations).

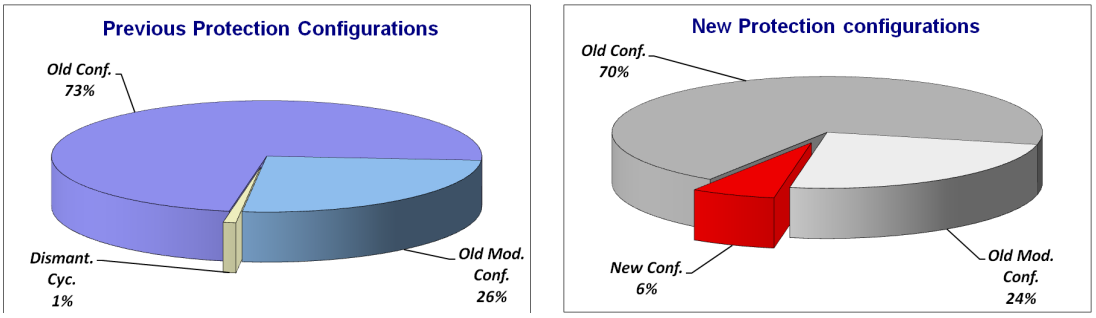


(a) Time period $t-1$ (b) Time period t

Figure 7.7 – Protection Configurations 5 % 5 % NY

Very similar results are observed with the NY network instance (Figures 7.7(a) and 7.7(b)) with an average of 98.7 % of already established light-cycles which remain un-

changed and which are re-used to protect the ongoing traffic. At each time period, an average of 96.8 % of the configurations, in the final solution, are associated with light-cycles generated in the previous time period. We can see that, under a stable dynamic traffic assumption, an average of 3.6 % in COST239 and 3.2 % in NY network of the configurations are newly established ones in order to improve the stability of the already established protection scheme.

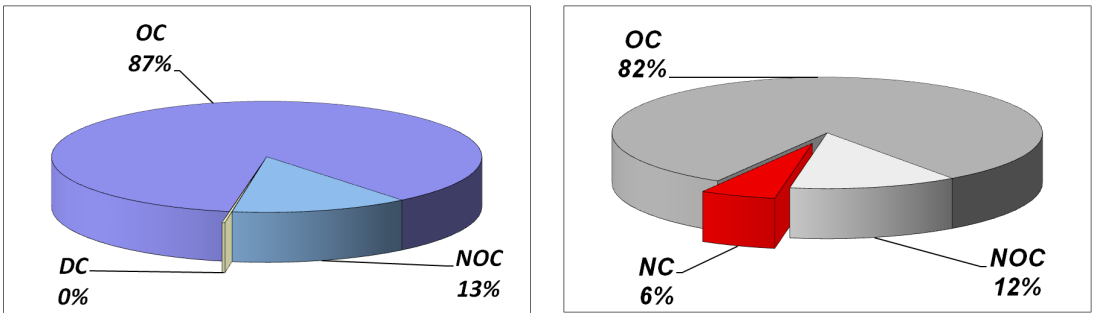


(a) Time period $t-1$ (b) Time period t

Figure 7.8 – Protection Configurations 5 % 10 % COST239

In the context of an incremental dynamic traffic, we can see from Figures 7.8 and 7.9 that the main difference, compared to the stable dynamic traffic environment, lies in the larger number of NOC configurations. Indeed, in the COST239 network, we can see that the number of NOC configurations represents 24.2 % of all configurations in the final solution at each time period. This is due to the fact that the number of added requests is twice the number of dropped ones and some rearrangements need to be performed (those correspond to the findings of the LCPP pricing problem) in order to maximize the stability of the already established light-cycles. We can see also that the number of NC configurations (i.e., brand new configurations) is only slightly larger than in the stable traffic, 6 % compared to 3 %. This is no surprise, since the protection scheme must follow the increase of working traffic.

Still, we must observe that the percentage of the new (NC) configurations (6 %) is almost equal to the percentage of traffic growth (10 % - 5 % = 5 %). It proves that the FIPP p-cycle protection scheme is very effective and stable, and that the model developed in Section 7.3.4 is accurate in translating the behavior of the FIPP p-cycles in a dynamic



(a) Time period $t-1$ (b) Time period t

Figure 7.9 – Protection Configurations 5 % 10 % NY

traffic environment.

7.6 Conclusion and Future Work

In this paper, we investigated the stability issue of the FIPP p-cycle protection structure. We found that using the number of required switching reconfigurations, FIPP p-cycles are highly stable and the percentage of required switching reconfigurations follows the percentage of traffic increase in the context of incremental traffic, even in the presence of torn down requests.

Future work will include the development of a more complex model where we will explicitly model at least two classes of traffic : real-time traffic which does not support any setting disruption without hampering its QoS, and non real-time traffic which can support minor disruption, especially with regard to its protection.

CHAPITRE 8

CONCLUSION

Les deux sujets traités dans cette thèse représentent deux domaines très importants dans les réseaux optiques WDM. Le groupage de trafic est devenu important à cause de la grande quantité de données qu'une longueur d'onde peut transporter et de la disparité très importante entre la capacité de transport d'une longueur d'onde et de la bande passante des requêtes individuelles. La protection de trafic est devenue encore plus importante du fait qu'une même fibre peut transporter plusieurs dizaines de longueurs d'onde et donc un nombre très élevé de requêtes.

Dans un réseau en opération, le trafic change continuellement ce qui oblige l'opérateur du réseau à s'adapter à ce changement. La stratégie que l'opérateur doit adopter est très importante pour pouvoir utiliser les ressources du réseau de façon efficace.

8.1 Synthèse des résultats obtenus

Pour le groupage de trafic, nous avons proposé quatre scénarios différents pour maximiser la bande passante des connexions acceptées à chaque période de temps.

Pour chacun des scénarios, nous avons considéré deux modèles de trafic différents. Le premier consiste à router tout le trafic d'une même connexion sur un seul chemin de la source à la destination. Dans le deuxième modèle, en se basant sur le principe de démultiplexage, une même connexion peut être divisée en plusieurs flots qui peuvent utiliser des chemins différents de la source à la destination, c'est ce qu'on a appelé les flots bifurqués. Le fait de considérer les flots bifurqués, améliore le taux d'acceptation des connexions. Le temps de calcul est, quant à lui, beaucoup plus petit comparativement au premier modèle de trafic (non bifurqué).

Le fait d'accepter le dérangement d'un certain nombre de connexions déjà en place, permet d'accepter plus de connexions à chaque période de temps. Le reroutage de connexions est une stratégie déjà connue dans la littérature, cependant il faut avoir une limite sur le

nombre de connexions qu'on peut déranger à chaque période de temps.

Le fait de ne pas imposer de limite sur le nombre de connexions qu'on peut déranger, que nous avons exploré en reconstruisant un nouveau schéma de groupage à chaque période de temps, nous permet d'augmenter le taux de trafic accepté. Cependant, nous considérons que cette augmentation n'est pas si grande pour justifier l'adoption de cette stratégie du fait qu'elle implique beaucoup de dérangements pour le trafic et pour l'opérateur du réseau.

La stratégie qui consiste à faire des augmentations/diminutions de la capacité des ports avec la possibilité de reroutage de connexions représente une option plus intéressante dans la mesure où le dérangement est minimal alors que le taux d'acceptation des connexions est très grand comparativement aux autres stratégies.

La protection par *p*-cycles apparaît comme un paradigme très intéressant, parmi les paradigmes de protection, dans la mesure où elle représente un bon compromis entre le temps de restauration et l'utilisation efficace de la capacité dans les réseaux optiques.

Dans le contexte de trafic dynamique nous avons proposé différentes stratégies pour l'utilisation efficace des *p*-cycles de protection. Bien qu'elle n'ait jamais été étudiée de manière approfondie dans la littérature, plusieurs émettaient des doutes sur la stabilité des *p*-cycles dans un contexte de trafic dynamique. Notre étude a montré que, dépendamment de l'objectif à atteindre lors du choix des *p*-cycles, ils peuvent être très stables même dans un trafic très variable où plusieurs connexions s'arrêtent et plusieurs autres arrivent à chaque période de temps. La minimisation du nombre de ports qui composent les *p*-cycles qu'on doit ajouter à chaque période de temps représente le meilleur objectif à considérer pour augmenter la stabilité des *p*-cycles dans le contexte d'un trafic dynamique.

L'utilisation de la technique de génération de colonnes, qui consiste à décomposer le problème traité en deux problèmes soit le problème maître et le problème auxiliaire, nous a permis d'obtenir des résultats quasi-optimaux avec des temps d'exécutions très courts pour les plus grands réseaux qu'on trouve dans la littérature.

L'étude de la protection par FIPP *p*-cycles nous a permis de constater que, comme dans les *p*-cycles de base, les FIPP *p*-cycles sont très stables et, dans un trafic évolutif,

le pourcentage des *p*-cycles ajoutés à chaque période de temps est quasi-identique au pourcentage de l'augmentation de trafic dans le réseau. La décomposition hiérarchique du problème auxiliaire permet de réduire de façon importante le temps d'exécution.

8.2 Conclusions

La caractéristique du trafic dynamique dans les réseaux optiques augmente la complexité des problèmes traités. En effet, cette caractéristique nous impose, lors de la résolution des deux problèmes traités dans la présente thèse, de trouver des solutions en un court laps de temps si on veut mettre en pratique les méthodes de résolution proposées. Cela nous oblige à considérer des compromis entre la qualité de la solution obtenue et le temps d'exécution des algorithmes proposés.

Le problème de groupage de trafic reste un problème très complexe à résoudre, surtout dans le contexte d'un trafic dynamique. Les résultats obtenus nous ont montré que des gains, en termes de taux d'acceptation des connexions, peuvent être faits par les différentes stratégies proposées. Cependant, de meilleurs résultats peuvent être obtenus grâce à la combinaison des bons algorithmes de résolution avec des petits changements matériels aux niveaux des composantes des réseaux (ex. augmentations/diminutions des capacités des ports).

La protection par *p*-cycles représente un paradigme très prometteur dans les réseaux optiques WDM. Nous avons montré qu'on peut avoir de très bons résultats au problème de protection par *p*-cycles avec un temps de calcul très raisonnable en utilisant des bonnes méthodes de résolution. Plusieurs chercheurs prétendent que les *p*-cycles ne sont pas stables, ce qui les rend moins adaptés au contexte de trafic dynamique. Nous avons montré que, dépendamment de l'objectif considéré, les *p*-cycles peuvent être très stables.

8.3 Travaux futurs

Parmi les travaux futurs que nous prévoyons, il y a la considération de plusieurs classes de service pour le trafic du réseau. Ceci nous donnera la possibilité de ne déranger

que les connexions qui tolèrent un peu de retard, pour le problème de groupage de trafic, et les *p*-cycles pour ce type de connexions pour le problème de protection de trafic. En effet, certains types de trafic peuvent tolérer un dérangement ou un retard de l'ordre de quelques millisecondes voire quelques secondes.

Les résultats que nous avons obtenus sur la stabilité des *p*-cycles montrent qu'il y a quand même un petit nombre de dérangements des *p*-cycles ou des liens des *p*-cycles. Nous croyons qu'on peut atteindre une stabilité totale où il n'y a pratiquement pas de dérangements dans le cas d'un trafic stable, en combinant les méthodes que nous avons utilisées pour la protection avec le routage sélectif des nouvelles connexions à chaque période de temps.

Bien que l'étude combinée des deux problèmes, groupage de trafic et protection par *p*-cycles, ait été proposée dans la littérature, et en se basant sur les difficultés que nous avons rencontré lors de l'étude de chacun des problèmes séparément, nous croyons que dans l'état actuel des choses, l'étude indépendante de chaque problème reste la méthode la plus efficace, surtout dans le contexte d'un trafic dynamique.

BIBLIOGRAPHIE

[1] J. Ahmed, F. Solano, P. Monti et L. Wosinska. Traffic re-optimization strategies for dynamically provisioned WDM networks. Dans *Conference on Optical Network Design and Modeling - ONDM*, pages 1–6, 2011.

[2] R. Asthana, Y.N. Singh et W.D. Grover. *p*-cycles : An overview. *IEEE Communications Surveys & Tutorials*, 12(1):97–111, 2010.

[3] A. Atamtürk et D. Rajan. Partition inequalities for capacitated survivable network design based on directed *p*-cycles. *Discrete Optimization*, 5:415–433, 2008.

[4] D. Baloukov, W.D. Grover et A. Kodian. Toward jointly optimized design of failure-independent path-protecting *p*-cycle networks. *Journal of Optical Networking*, 7(1):62–79, 2008.

[5] C. Barnhart, E.L. Johnson, G.L. Nemhauser, M.W.P. Savelsbergh et P.H. Vance. Branch-and-price : Column generation for solving huge integer programs. *Operations Research*, 46(3):316–329, 1998.

[6] P. Batchelor *et al.* Ultra high-capacity optical transmission networks : Final report of action COST 239. Rapport technique, Faculty of Electrical Engineering and Computing, University of Zagreb, 1999.

[7] G. Bernstein, D. Caviglia, R. Rabbat et H. Van Helvoort. VCAT/LCAS in a clamshell. *IEEE Communications Magazine*, 44(5):34 – 36, 2006.

[8] R. Berry et E. Modiano. Reducing electronic multiplexing costs in SONET/WDM rings with dynamically changing traffic. *IEEE Journal on selected areas in communications*, 18(10):1961 – 1971, 2000.

[9] D. Brélaz. New methods to color the vertices of a graph. *Communications of the ACM*, 22(4):251–256, 1979.

[10] B. Chen, G.N. Rouskas et R. Dutta. A Framework for Hierarchical Traffic Grooming in WDM Networks of General Topology. *2nd International Conference on Broadband Networks*, 1:155–164, 2005.

[11] B. Chen, G.N. Rouskas et R. Dutta. On hierarchical traffic grooming in WDM networks. *IEEE/ACM Transactions on Networking*, 16(5):1226–1238, 2008.

[12] V. Chvatal. *Linear Programming*. Freeman, 1983.

[13] D. Crawford. Fiber optic cable dig-ups - causes and cures. *Network Reliability and Interoperability Council website http ://www.nric.org /pubs/nric1/sections/abody.pdf*, Dernière visite : mars 2011.

[14] O. Crochat et J. Y. Le Boudec. Design Protection for WDM Optical Networks. *IEEE Journal on Selected Areas in Communications JSAC*, 16(7):1158–1165, 1998.

[15] J. Doucette, D. He, W.D. Grover et O. Yang. Algorithmic approaches for efficient enumeration of candidate *p*-cycles and capacitated *p*-cycle network design. Dans *Proceedings of IEEE/VDE Workshop on Design of Reliable Communication Networks - DRCN*, pages 212–220, 2003.

[16] A.C. Drummond et N.L.S. da Fonseca. Fair and efficient dynamic traffic grooming algorithm for WDM mesh networks. Dans *IEEE Global Telecommunications Conference - GLOBECOM*, pages 1–6, 2009.

[17] A.C. Drummond et N.L.S. da Fonseca. Fairness in zone-based algorithms for dynamic traffic grooming in WDM mesh networks. *IEEE/OSA Journal of Optical Communications and Networking*, 2(6):305 – 318, 2010.

[18] R. Dutta, A.E. Kamal et G.N. Rouskas. *Traffic Grooming for Optical Networks Foundations, Techniques and Frontiers*. Springer, 2008.

[19] R. Dutta et G.N. Rouskas. Traffic Grooming in WDM Networks : Past and Future. *IEEE Network*, 16(6):46–56, 2002.

[20] A. Eshoul et H.T. Mouftah. Performance Evaluation of Dynamic P-Cycle Protection Methods in WDM Optical Networks. *11th International Conference on Transparent Optical Networks ICTON*, pages 1–4, 2009.

[21] A. Eshoul et H.T. Mouftah. Survivability approaches using *p*-cycles in WDM mesh networks under static traffic. *IEEE/ACM Transactions on Networking (TON)*, 17(2):671–683, 2009.

[22] J. Fang et K. Somani. IP Traffic Grooming Over WDM Optical Networks. *IEEE Conference on Optical Network Design and Modeling*, pages 393–402, 2005.

[23] C. Ge, N. Bai, X. Sun et M. Zhang. Iterative joint design approach for failure-independent path-protecting *p*-cycle networks. *Journal of Optical Networking*, 6 (12):1329–1339, 2007.

[24] A. Gençata et B. Mukherjee. Virtual-Topology Adaptation for WDM Mesh Networks Under Dynamic Traffic. *IEEE/ACM Transactions on Networking*, 11(2): 236– 247, 2003.

[25] O. Gerstel et R. Ramaswami. Optical layer survavibility : A service perspective. *IEEE Communications Magazine*, 38(3):104 – 113, 2000.

[26] O. Gerstel, R. Ramaswami et G.H. Sasaki. Fault Tolerant Multiwavelength Optical Rings with Limited Wavelength Conversion. *IEEE Journal on Selected Areas in Communications JSAC*, 16(7):1166–1178, 1998.

[27] O. Gerstel, R. Ramaswami et G.H. Sasaki. Cost-effective traffic grooming in WDM rings. *IEEE/ACM Transactions on Networking*, 8(5):618 – 630, 2000.

[28] O. Gerstel et H. Raza. Predeployment of resources in agile photonic networks. *IEEE Journal of Lightwave Technology*, 22(10):2236 – 2244, 2004.

[29] G.Li, D. Wang, C. Kalmanek et R. Doverspike. Efficient distributed restoration path selection for shared mesh restoration. *IEEE/ACM Transactions on Networking*, 11(5):761–771, 2003.

[30] W. Goralski. *Sonet/SDH*. Osborne/McGraw-Hill, 3 édition, 2002.

[31] W. Grover. Globally optimal distributed synchronous batch reconfiguration for efficient hazard-free dynamic provisioning : How an entire network can " think globally and act locally" ? Dans *Proceedings of IEEE/VDE Workshop on Design of Reliable Communication Networks - DRCN*, pages 1–8, 2007.

[32] W.D. Grover. *Mesh-Based Survivable Networks*. Prentice-Hall, 2004.

[33] W.D. Grover. The protected working capacity envelope concept : An alternative paradigm for automated service provisioning. *IEEE Communications Magazine*, 42(1):62–69, 2004.

[34] W.D. Grover et J. Doucette. Advances in optical network design with *p*-cycles : Joint optimization and pre-selection of candidate *p*-cycles. Dans *Proc. IEEE LEOS Summer Topical Meetings*, pages 49–50, 2002.

[35] W.D. Grover, J. Doucette, A. Kodian, D. Leung, A. Sack, M. Clouqueur et G. Shen. Design of survivable networks based on p-cycles. In Resende, Mauricio G.C. and Pardalos, Panos M. (Eds.). *Handbook of Optimization in Telecommunications*, pages 391–434, 2006.

[36] W.D. Grover et G. Shen. Extending the *p*-cycle concept to path-segment protection. Dans *IEEE International Conference on Communications - ICC*, volume 2, pages 1314 – 1319, 2003.

[37] W.D. Grover et D. Stamatelakis. Cycle-oriented distributed preconfiguration : Ring-like speed with mesh-like capacity for self-planning network restoration. Dans *IEEE International Conference on Communications - ICC*, pages 537–543, 1998.

[38] W.D. Grover et D. Stamatelakis. Bridging the ring-mesh dichotomy with *p*-cycles. Dans *Proceedings of IEEE/VDE Workshop on Design of Reliable Communication Networks - DRCN*, pages 92–104, Munich, Germany, 2000.

[39] P.K. Gummadi, M.J. Pradeep et C. Siva Ram Murthy. A Segmented Backup Scheme for Dependable Real Time Communication in Multihop Networks. *IEEE/ACM Trans. Networking*, 11:81–94, 2003.

[40] M. Gunkel, R. Leppla, M. Wade, A. Lord, D. Schupke, G. Lehmann, C. Furst, S. Bodamer, B. Bollenz, H. Haunstein, H. Nakajima et J. Martensson. A cost model for the WDM layer. Dans *International Conference on Photonics in Switching (PS)*, pages 1–6, Herakleion (Crete), Greece, 2006.

[41] W. He, J. Fang et A.K. Somani. A *p*-cycle based survivable design for dynamic traffic in WDM networks. *IEEE Global Telecommunications Conference GLOBECOM*, 4:1869–1873, 2005.

[42] W. He et A. Somani. Comparison of protection mechanisms : Capacity efficiency and recovery time. Dans *IEEE International Conference on Communications - ICC*, pages 2218–2223, 2007.

[43] P-H. Ho et H.T. Mouftah. A framework for service-guaranteed shared protection in WDM mesh networks. *IEEE Communications Magazine*, 40(2):97–103, 2002.

[44] P-H. Ho, J. Tapolcai et T. Cinkler. Segment shared protection in mesh communications networks with bandwidth guaranteed tunnels. *IEEE/ACM Transactions on Networking*, 12(6):1105–1118, 2004.

[45] Q.D. Ho et M.S. Lee. Connection level active rerouting in WDM mesh networks with traffic grooming capability. Dans *ICC*, volume 5, pages 2415–2420, 2006.

[46] Q.D. Ho et M.S. Lee. A zone-based approach for scalable dynamic traffic grooming in large WDM mesh networks. *Journal of Lightwave Technology*, 25(1): 261–270, 2007.

[47] J.Q. Hu et B. Leida. Traffic grooming, routing, and wavelength assignment in optical WDM mesh networks. Dans *Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, volume 1, pages 495–501, 2004.

[48] S. Huang, M. Xia, C. Martel et B. Mukherjee. Survivable Multipath Traffic Grooming in Telecom Mesh Networks With Inverse Multiplexing. *IEEE/OSA Journal of Optical Communications and Networking*, 2(8):545–557, 2010.

[49] H. Hwang, S. Ahn, Y. Yoo et C.S. Ki. Multiple shared backup cycles for survivable optical mesh networks. Dans *International Conference on Computer Communications and Networks - ICCCN*, pages 284–289, 2001.

[50] *ILOG CPLEX 11.0 Documentation*. ILOG Inc., Gentilly, France, 2008.

[51] B. Jaumard, A. Houle et Y. Solari. Dimensioning WDM Optical Network with Minimum MSPP Configuration. Dans *IASTED International Conference OCSN (Optical Communications Systems and Networks)*, pages 826–833, 2004.

[52] B. Jaumard, H. Li et S. Sebbah. Design of multi-granularity directed segment *p*-cycles. Dans *2010 IEEE Sarnoff Symposium*, pages 1–5, 2010.

[53] B. Jaumard, H. Li et S. Sebbah. Design of path-segment-protecting *p*-cycles in survivable wdm mesh networks. Dans *14th International Telecommunications Network Strategy and Planning Symposium (NETWORKS)*, pages 1–6, 2010.

[54] B. Jaumard et A. Metnani. Stability of *p*-cycles under dynamic traffic. Dans *IEEE Global Telecommunications Conference - GLOBECOM*, pages 1–5, 2010.

[55] B. Jaumard, C. Meyer, B. Thiongane et Y. Xiao. ILP Formulations and Optimal Solutions for the RWA Problem. *IEEE Global Telecommunications Conference GLOBECOM*, 3:1918 – 1924, 2004.

[56] B. Jaumard, C. Rocha, D. Baloukov et W.D. Grover. A column generation approach for design of networks using path-protecting *p*-cycles. Dans *Proceedings of IEEE/VDE Workshop on Design of Reliable Communication Networks - DRCN*, 2007.

[57] M.S. Kiaei, C. Assi et B. Jaumard. A Survey on the p-Cycle Protection Method. *IEEE Communications Surveys & Tutorials*, 11(3):53–70, 2009.

[58] S. Kini, M. Kodialam, T.V. Laksham et C. Villamizar. Shared Backup Label Switched Path Restoration. *Internet Draft, draft-kinirestoration-shared-backup-00.txt*, 2000.

[59] A. Kodian et W.D. Grover. Failure-independent path-protecting *p*-cycles efficient and simple fully preconnected optical-path protection. *Journal of Lightwave Technology*, 23(10):3241–3259, 2005.

[60] A. Kodian, W.D. Grover et J. Doucette. A disjoint route sets approach to design of failure-independent path-protection *p*-cycle networks. Dans *Proceedings of IEEE/VDE Workshop on Design of Reliable Communication Networks - DRCN*, 2005.

[61] V.R. Konda et T.Y. Chow. Algorithm for Traffic Grooming in Optical Networks to Minimize the Number of Transceivers. Dans *IEEE Workshop on High Performance Switching and Routing,*, pages 218–221, 2001.

[62] J. Kuri, M. Gagnaire, N. Puech, E. Dotaro et R. Douville. Routing and wavelength assignment of scheduled lighpath demands. *IEEE Journal on Selected Areas in Communications*, 21(8):1231– 1240, 2003.

[63] L.S. Lasdon. *Optimization Theory for Large Systems*. MacMillan, New York, 1970.

[64] C. Liu et L. Ruan. Dynamic provisioning of survivable services using path-segment protecting *p*-cycles in WDM networks. Dans *International Conference on Computer Communications and Networks - ICCCN*, pages 275–280, 2006.

[65] K. Lo, D. Habibi, Q.V. Phung, A. Rassau et H.N. Nguyen. Efficient p-Cycles Design By Heuristic p-Cycle Selection and Refinement for Survivable WDM Mesh Networks. *IEEE Global Telecommunications Conference GLOBECOM*, pages 1–5, 2006.

[66] M.E. Lübbecke et J. Desrosiers. Selected topics in column generation. *Operations Research*, 53:1007–1023, 2005.

[67] G. Maier, A. Pattavina, S. De Patre et M. Martinelli. Optical Network Survivability : Protection Techniques in the WDM Layer. *Photonic Network Communications*, 4(3-4):251–269, 2002.

[68] A. Metnani et B. Jaumard. Directed *p*-cycle protection in dynamic WDM networks. Dans *IEEE International Workshop on Reliable Networks Design and Modeling - RNDM*, pages 1–6, St Petersburg, Russia, 2009.

[69] A. Metnani et B. Jaumard. Dynamic protection provisioning with fipp *p*-cycles in wdm networks. Dans *IEEE 15th International Conference on Optical Network Design and Modeling - ONDM*, pages 1–6, 2011.

[70] G. Mohan et C.S.R. Murthy. Lightpath restoration in WDM optical networks. *IEEE Network Magazine*, 6(14):24–32, 2000.

[71] B. Mukherjee. *Optical WDM Networks*. Springer, 2006.

[72] G.L. Nemhauser et L.A. Wolsey. *Integer and Combinatorial Optimization*. Wiley, 1999, reprint of the 1988 edition.

[73] D.P. Onguetou et W.D. Grover. Altering Grooming Decisions to Enhance p-Cycle Design Efficiency. *IEEE Global Telecommunications Conference, GLOBECOM*, pages 1–5, 2009.

[74] D.P. Onguetou et W.D. Grover. Solution of a 200-Node p-Cycle Network Design Problem with GA-Based Pre-Selection of Candidate Structures. *IEEE International Conference on Communications ICC*, pages 1–5, 2009.

[75] D.P. Onguetou et W.D. Grover. A Two-Hop Segment Protecting Paradigm that Unifies Node and Span Failure Recovery under p-Cycles. *IEEE Communications Letters*, 14(11):1080–1082, 2010.

[76] D.P. Onguetou et W.D. Grover. *p*-Cycle protection at the glass fiber level. *Computer Communications*, page to appear, 2011.

[77] C. Ou, K. Zhu, H. Zang, L.H. Sahasrabuddhe et B. Mukherjee. Traffic grooming for survivable wdm networks - shared protection. *IEEE Journal on Selected Areas in Communications*, 21(9):1367–1383, 2003.

[78] C. Ou, K. Zhu, J. Zhang, H. Zhu, B. Mukherjee, H. Zang et L. Sahasrabuddhe. Traffic grooming for survivable WDM networks : dedicated protection. *Journal of Optical Networking*, 3(1):50–74, 2004.

[79] R. Ramaswami et K.N Sivarajan. *Optical Networks : A Practical Perspective (Second Edition)*. Morgan Kaufmann Publishers Inc, 2002.

[80] R. Ramaswami, K.N. Sivarajan et G.H. Sasaki. *Optical Networks - A Practical Perspective*. Morgan Kaufmann, 3nd edition édition, 2009.

[81] L.C. Resendo, L.C. Calmon et M.R.N. Ribeiro. Simple ILP Approaches to Grooming, Routing and Wavelength Assignment in WDM Mesh Networks. *SBMO/IEEE MTT-S International Conference on Microwave and Optoelectronics*, pages 616–619, 2005.

[82] C. Rocha et B. Jaumard. Revisiting *p*-cycles / FIPP *p*-cycles vs. shared link / path protection. Dans *International Conference on Computer Communications and Networks - ICCCN*, pages 1–6, 2008.

[83] C. Rocha et B. Jaumard. Efficient computation of FIPP *p*-cycles. *to appear in Telecommunication Systems*, 2011.

[84] L. Ruan et Tang F. Dynamic establishment of restorable connections using *p*-cycle protection in WDM networks. Dans *International Conference on Broadband Networks*, volume 1, pages 137 – 144, 2005.

[85] M. Saad et Z. Luo. Reconfiguration With No Service Disruption in Multifiber WDM Networks. *Journal of Lightwave Technology*, 23(10):3092–3104, 2005.

[86] C.V. Saradhi et C.S.R. Murthy. Dynamic establishment of differentiated survivable lightpaths in WDM mesh networks. *Computer Communications*, 27(3): 273–294, 2004.

[87] D.A. Schupke. An ILP for optimal *p*-cycle selection without cycle enumeration. Dans *Proceedings of the 8th Working Conference on Optical Network Design and Modelling (ONDM)*, Ghent, Belgium, 2004.

[88] D.A. Schupke, W.D. Grover et M. Clouqueur. Strategies for enhanced dual failure restorability with static or reconfigurable *p*-cycle networks. Dans *IEEE International Conference on Communications - ICC*, volume 3, pages 1628– 1633, 2004.

[89] S. Sebbah et B. Jaumard. Efficient and scalable design of protected working capacity envelope. Dans *13th International Telecommunications Network Strategy and Planning Symposium - Networks*, pages 1 – 21, 2008.

[90] S. Sebbah et B. Jaumard. Survivable WDM networks design with non-simple *p*-cycle-based PWCE. Dans *IEEE Global Telecommunications Conference - GLOBECOM*, pages 1–6, 2008.

[91] S. Sebbah et B. Jaumard. PWCE design in survivable WDM Networks using unrestricted shape *p*-structure patterns. Dans *IEEE* 22^{nd} *Canadian Conference on Electrical and Computer Engineering 2009 (CCECE 2009), St Johns*, pages 279 – 282, 2009.

[92] G. Shen et W. D. Grover. Design and performance of protected working capacity envelopes based on *p*-cycles for dynamic provisioning of survivable services. *Journal of Optical Networking*, 4:361–390, 2005.

[93] G. Shen et W.D. Grover. Extending the *p*-cycle concept to path segment protection for span and node failure recovery. *IEEE Journal on Selected Areas in Communications*, 21(8):1306–1319, 2003.

[94] G. Shen et W.D. Grover. Survey and performance comparison of dynamic provisioning methods for optical shared backup path protection. *2nd International Conference on Broadband Networks*, 2:1310 – 1319, 2005.

[95] M. Sivakumar, K.M. Sivalingam et S. Subramaniam. On Factors Affecting the Performance of Dynamically Groomed Optical WDM Mesh Networks. *Workshop on High Performance Switching and Routing HPSR*, pages 411 – 415, 2005.

[96] M. Sivakumar, K.M. Sivalingam et S. Subramaniam. On factors affecting the performance of dynamically groomed optical WDM mesh networks. *Photonic Network Communications*, 12(1):15–28, 2006.

[97] T. Stidsen et T. Thomadsen. Joint optimization of working and *p*-cycle protection capacity. Rapport technique, Technical Univ. of Denmark, 2004.

[98] J. Tapolcai et P-H. Ho. Linear Formulation for Segment Shared Protection. *In Proceeding of the SPIE OptiComm*, pages 49–58, 2003.

[99] S. Thiagarajan et A. Somani. Traffic grooming for survivable mesh networks. *Special Issue on Protection/Restoration Meets the reliability challenge, Optical Network Magazine*, pages 88–98, 2002.

[100] B. Todd et J. Doucette. Multi-Flow Optimization Model for Design of a Shared Backup Path Protected Network. *IEEE International Conference on Communications ICC*, pages 131 – 138, 2008.

[101] M. Tornatore, A. Baruffaldi, H. Zhu, B. Mukherjee et A. Pattavina. Dynamic traffic grooming of subwavelength connections with known duration. Dans *Optical Fiber Communication Conference - OFC*, 2007.

[102] B. Vignac, B. Jaumard et F. Vanderbeck. Hierarchical optimization procedure for traffic grooming in wdm optical networks. Dans *Conference on Optical Network Design and Modeling - ONDM*, pages 1–6, 2009.

[103] G. Walter. *SONET/SDH (Third Edition)*. Osborne/McGraw-Hill, 2002.

[104] J. Wang, W. Cho, V.R. Vemuri et B. Mukherjee. Improved Approaches for Cost-effective Traffic Grooming in WDM Ring Networks : ILP Formulations and single-hop and multihop connections. *IEEE Journal of Lightwave Technology*, 19 (11):1645 – 1653, 2001.

[105] A. Wason et R.S. Kaler. Lightpath rerouting algorithm to enhance blocking performance in all-optical WDM network without wavelength conversion. *Optical Fiber Technology*, 16(3):146–150, 2010.

[106] E.W.M. Wong, A.K.M. Chan et T.-S.P. Yum. A taxonomy of rerouting in circuit-switched networks. *IEEE Communications Magazine*, 37(11):116 – 122, 1999.

[107] B. Wu, K.L. Yeung et P-H.Ho. ILP formulations for *p*-cycle design without candidate cycle enumeration. *IEEE/ACM Transactions on Networking*, 18(1):284–295, 2010.

[108] C. Xin. Blocking analysis of dynamic traffic grooming in mesh WDM optical networks. *IEEE/ACM Transactions on Networking*, 15(3):721 – 733, 2007.

[109] C. Xin et B. Wang. Logical Topology Design for Dynamic Traffic Grooming in Mesh WDM Optical Networks. *IEEE ICC*, 3:1792–1796, 2005.

[110] C. Xin, B. Wang, X. Cao et J. Li. Logical Topology Design for Dynamic Traffic Grooming in WDM Optical Networks. *Journal of Lightwave Technology*, 24(6): 2267–2275, 2006.

[111] Y. Xin, M. Shayman, R.J. La et S.I. Marcus. A Pure Framework for Cost-effective Virtual Ring Based Traffic Grooming in WDM Optical Networks. 2^{nd} *International Conference on Broadband Networks*, pages 302 – 307, 2005.

[112] K. Yang et K. Silvalingam. Routing in SONET/VCAT based optical WDM networks. Dans *International Conference on Broadband Networks*, September 2009.

[113] W. Yang, A. Paredes, H. Schriemer et T.J. Trevor. Protection of dynamic and flexible bandwidth on demand in metro agile all-optical ring networks. *Journal of Optical Communications and Networking*, 1(2):160–169, 2009.

[114] W. Yao, M. Li et B. Ramamurthy. Rerouting Scheme for Dynamic Traffic Grooming in Optical WDM Mesh Networks. *IEEE Communications Society, GLOBCOM*, pages 1773–1797, 2004.

[115] W. Yao, M. Li et B. Ramamurthy. Performance Analysis of Sparse Traffic Grooming in WDM Mesh Networks. *IEEE International Conference on Communications ICC*, 3:1766–1770, 2005.

[116] W. Yao et B. Ramamurthy. Dynamic traffic grooming using fixed-alternate routing in WDM mesh optical networks. Dans *First Workshop on Traffic Grooming in WDM Networks, Co-located with BroadNets 2004*, San Jose, California, USA, 2004.

[117] W. Yao et B. Ramamurthy. Rerouting schemes for dynamic traffic grooming in optical wdm networks. *Computer Networks*, 52:1891–1904, 2008.

[118] F. Zhang et Z. Zhang. Performance evaluation of *p*-cycle based protection methods for provisioning of dynamic multicast sessions in mesh WDM networks. *Photonic Network Communications*, 16(2):127–138, 2008.

[119] F. Zhang et W.-D. Zhong. A novel path-protecting *p*-cycle heuristic algorithm. Dans *Proceeding of the International Conference on Transparent Optical Networks (ICTON)*, volume 3, pages 203–206, Nottingham, UK, 2006.

[120] H. Zhang et O. Yang. Finding protection cycles in dwdm networks. Dans *Proceedings of IEEE International Conference on Communications (ICC 2002)*, volume 5, pages 2756–2760, 2002.

[121] W. Zhang, X. Du, K. Nygard et T. Wang. Self-protecting networking using dynamic *p*-cycle construction within link capacity constraint. Dans *IEEE International Conference on Communications - ICC*, pages 1–6, 2009.

[122] Y. Zhang, P. Chowdhury, M. Tornatore et B. Mukherjee. Energy efficiency in telecom optical networks. *IEEE Communications Surveys & Tutorials*, 12(4):441 – 458, 2010.

[123] Z. Zhang, W. Zhong et S. B. Bose. Dynamically survivable WDM network design with *p*-cycle-based PWCE. *IEEE Communications Letters*, 9(8):756–758, 2005.

[124] Z. Zhang, W.D. Zhong et B. Mukherjee. A heuristic method for design of survivable WDM networks with *p*-cycles. *IEEE Communications Letters*, 8(7):467–469, 2004.

[125] J. Zheng et H.T. Mouftah. *Optical WDM Networks : Concepts and Design principles*. Wiley-Interscience/IEEE press, 2004.

[126] W.D. Zhong et Z. Zhang. *p*-Cycles-based dynamic protection provisioning in optical WDM networks. *IEICE Transactions on Communications*, 88(5):1921–1926, 2005.

[127] D. Zhou et S. Subramaniam. Survivability in Optical Networks. *IEEE Network*, 6(14):16–23, 2000.

[128] H. Zhu, H. Zang, K. Zhu et B. Mukherjee. A Novel Generic Graph Model for Traffic Grooming In Heterogeneous WDM Mesh Networks. *IEEE/ACM Transactions on Networking*, 11(2):285–299, 2003.

[129] K. Zhu et B. Mukherjee. Traffic grooming in an optical WDM mesh network. *IEEE Journal on Selected Areas in Communications*, 20(1):122–133, 2002.

[130] K. Zhu et B. Mukherjee. A review of traffic grooming in WDM optical networks : Architectures and challenges. *Optical Networks Magazine*, 4(2):55–64, 2003.

[131] K. Zhu, H. Zhu et B. Mukherjee. Traffic Engineering in Multigranularity Heterogeneous Optical WDM Mesh Networks through Dynamic Traffic Grooming. *IEEE Network*, 17(2):8–15, 2003.

[132] K.Z. Zhu, H. Zang et B. Mukherjee. A comprehensive study on next generation optical grooming. *IEEE Journal on Selected Areas in Communications*, 21(7): 1173–1186, 2003.