

**Titre:** Routage et affectation de longueurs d'onde dans les réseaux optiques WDM  
Title:

**Auteur:** Hichem Ayed Harhira  
Author:

**Date:** 2005

**Type:** Mémoire ou thèse / Dissertation or Thesis

**Référence:** Harhira, H. A. (2005). Routage et affectation de longueurs d'onde dans les réseaux optiques WDM [Mémoire de maîtrise, École Polytechnique de Montréal].  
Citation: PolyPublie. <https://publications.polymtl.ca/7626/>

 **Document en libre accès dans PolyPublie**  
Open Access document in PolyPublie

**URL de PolyPublie:** <https://publications.polymtl.ca/7626/>  
PolyPublie URL:

**Directeurs de recherche:** Samuel Pierre  
Advisors:

**Programme:** Génie informatique  
Program:

UNIVERSITÉ DE MONTRÉAL

ROUTAGE ET AFFECTATION  
DE LONGUEURS D'ONDE DANS LES RÉSEAUX  
OPTIQUES WDM

HICHEM AYED HARHIRA  
DÉPARTEMENT DE GÉNIE INFORMATIQUE  
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION  
DU DIPLÔME DE MAÎTRISE ÈS SCIENCES APPLIQUÉES  
(GÉNIE INFORMATIQUE)

Mai 2005

©HARHIRA HICHEM AYED, 2005.



Library and  
Archives Canada

Bibliothèque et  
Archives Canada

Published Heritage  
Branch

Direction du  
Patrimoine de l'édition

395 Wellington Street  
Ottawa ON K1A 0N4  
Canada

395, rue Wellington  
Ottawa ON K1A 0N4  
Canada

*Your file* *Votre référence*  
*ISBN: 978-0-494-16793-9*  
*Our file* *Notre référence*  
*ISBN: 978-0-494-16793-9*

#### NOTICE:

The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

#### AVIS:

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protègent cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

---

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

  
**Canada**

UNIVERSITÉ DE MONTRÉAL  
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Ce mémoire intitulé :

ROUTAGE ET AFFECTATION  
DE LONGUEURS D'ONDE DANS LES RÉSEAUX  
OPTIQUES WDM

présenté par : HARHIRA Hichem Ayed

en vue de l'obtention du diplôme de: Maîtrise ès sciences appliquées

a été dûment acceptée par le jury d'examen constitué de:

M. ROY Robert, Ph.d., président

M. PIERRE Samuel, Ph.d., membre et directeur de recherche

M. GALINIER Philippe, Doc., membre

## Remerciements

J'aimerais remercier Mr Samuel Pierre mon directeur de recherche pour avoir cru que je pouvais mener à bien ce travail et dont l'encadrement et les conseils m'ont guidé tout au long de ma recherche.

Ma reconnaissance va aussi naturellement à mon père et à ma mère dont l'éducation, la confiance, l'appui moral et financier m'ont permis de m'épanouir, à mes frères Tarek et Mohamed et à ma sœur Fatma pour leur soutien permanent et leurs encouragements. Merci pour tout.

Ma reconnaissance va aussi à celle qui a fait énormément pour moi pendant ce travail, celle qui a supporté mes humeurs changeantes en fonction de l'évolution du projet, ma fiancée Sabine Maaroufi.

À mes amis, Ahmed, Seif, Khaled et Zied, je vous suis reconnaissant pour votre appui.

Je tiens aussi à exprimer ma profonde gratitude à tous ceux qui, de près ou de loin ont contribué, d'une manière ou d'une autre à l'élaboration de ce projet.

## RÉSUMÉ

Avec l'évolution rapide des télécommunications et l'explosion des technologies de l'information, de nouvelles habitudes ont émergé chez les utilisateurs. Au rythme de cette évolution, l'engouement des usagers croît continuellement envers ces dites technologies. Cet enthousiasme est d'autant plus important avec l'avènement et l'intégration de nouveaux services sur les réseaux actuels. Ces tendances des usagers ainsi que la « gourmandise » des nouveaux services en bande passante incitent à redéfinir les choix stratégiques des architectures mises en place (meilleure planification supportant les demandes grandissantes en bande passante) d'une part, et surtout de reconsidérer un support physique offrant la bande passante demandée. C'est dans ce contexte que la fibre optique a bénéficié d'un épanouissement rapide pour devenir le support privilégié des réseaux actuels. En outre, et face aux limites des réseaux actuels, y compris ceux à haut débit (ATM/SONET), le développement de la technologie WDM s'avère très prometteur. En fait, l'utilisation actuelle de la fibre optique est loin d'être optimale et ce surtout à cause des limitations des équipements électroniques. Le multiplexage en longueurs d'onde WDM permet une utilisation quasi-optimale de la fibre en la faisant supporter plusieurs communications concurrentes, chacune sur une longueur d'onde à part. Cela rend donc possible la transmission de plusieurs téraoctets par seconde. Néanmoins, les processus de planification, de gestion et de maintenance des réseaux optiques WDM s'avèrent complexes. Les équipements déployés par les opérateurs doivent du fait permettre de gérer la complexité des connexions entre les clients, tout en minimisant le coût global d'exploitation. Parallèlement à la croissance du trafic, la demande en qualité de service est d'autant plus forte, notamment pour la sécurisation des connexions. L'opérateur doit donc veiller à protéger chaque connexion d'un éventuel dysfonctionnement, tant au niveau matériel que logiciel.

Avec l'utilisation du multiplexage en longueurs d'onde, un nouveau problème survient lors de la planification, c'est l'affectation des longueurs d'onde. En effet, et vu

le coût élevé des dispositifs de conversion, la contrainte de continuité de longueur d'onde doit être respectée. Le problème de routage et d'affectation de longueurs d'onde RWA étant répertorié NP-Complet, le développement d'heuristiques pour la résolution est nécessaire. Le but est en général de maximiser le nombre de connexions acheminées, et donc de minimiser le taux de blocage. Une formulation équivalente existe également pour le problème RWA et vise à minimiser le nombre de longueurs d'onde nécessaires pour acheminer un certain nombre de trafics. Cette dernière formulation est surtout utile pour des fins de dimensionnement de réseaux.

Dans ce mémoire, nous exposons une formulation mathématique pour le problème max-RWA. Par la suite, un algorithme heuristique est proposé pour le routage et l'affectation de longueurs d'onde. Il est à noter qu'à des fins de simplification, nous avons adopté une approche de décomposition. En effet, les sous-problèmes de routage et d'affectation de longueurs d'onde sont traités séparément. Pour le premier sous-problème, nous cherchons un ensemble des plus courts chemins disjoints d'arcs pour chaque trafic. La sélection des routes ainsi que l'attribution des longueurs d'onde est faite par la suite grâce à des heuristiques de recherche locale, tout en tenant compte de notre objectif de départ : minimiser le nombre de longueurs d'onde pour satisfaire l'ensemble de la demande.

Pour évaluer la performance des algorithmes proposés, un plan d'expérience est construit et l'algorithme est testé pour plusieurs jeux de tests en mettant l'accent sur le nombre de longueurs d'onde nécessaires ainsi que les temps d'exécution. Il est à noter également que différents scénarios de pannes simples ont été considérés et, pour chacun, le nombre de trafics bloqués est calculé.

Ainsi, un exposé des résultats trouvés par nos algorithmes est fait en comparant ces derniers à ceux obtenus par l'algorithme *TabouN* dans un premier temps, vu que ce dernier traite les cas de demandes de connexions robustes et permanentes. Par la suite, nous avons comparé ces derniers à un algorithme rencontré dans la littérature et aux bornes supérieures calculées avec le logiciel CPLEX. Dans la plupart des cas, les résultats que nous avons obtenus sont nettement meilleurs que ceux produits par les

algorithmes de référence. Même en les comparant aux bornes supérieures obtenues par CPLEX, nos algorithmes arrivent à plusieurs coups à trouver les valeurs optimales. Ces résultats étaient quelque peu prévisibles vu que nous utilisons une heuristique de coloriage très performante surtout pour les graphes de grande taille. Pour ce qui est du temps d'exécution, il est assez raisonnable dans l'ensemble en regard du temps d'exécution des meilleurs algorithmes de routage et d'affectation de longueurs d'onde recensés dans la littérature.

## ABSTRACT

With the fast evolution of telecommunications and the increase of information technologies, new users' practices emerged. At the rhythm of this evolution, the passion of the users grows continuously towards such technologies. This enthusiasm is mainly emphasized by the advent and the integration of new services on the current networks. These trends as well as the "greediness" of the new services in bandwidth encourage to redefine the strategic choices of the architectures installation (better planning supporting the growing requirements in bandwidth) on the one hand, and especially to reconsider a physical support offering the required bandwidth, on the other hand. In this context, the optical transmissions got a fast blooming to become the privileged support of the current networks. Moreover, the development of the WDM technology proves very promising. In fact, the wavelength multiplexing WDM allows an "optimal" use of fibre by making it support several concurrent communications, each one over a separated wavelength, and thus makes possible the transmission of several terabits per second. Nevertheless, the processes of planning, management and maintenance of optical networks WDM prove to be complex. The equipment deployed by the operators must bear with the complexity of connections between the customers, while minimizing the total operating costs. Besides the traffic growth, the demand for quality of service is increasing strongly, and mainly for connections security. The operator must thus take care to protect each connection of a possible dysfunction, at hardware level as well as software level. With the use of the wavelengths' multiplexing, a new planning problem occurs; it is the wavelength assignment problem. Indeed, and considering the high cost of conversion devices, the constraint of wavelength continuity must be respected. The problem of routing and wavelength assignment is shown to be Np-Complete, so that the development of heuristics for its resolution is necessary. The goal is commonly to maximize the number of conveyed connections, and thus to minimize the blocking rate. An equivalent formulation also exists for RWA problem and aims at minimizing the

number of wavelengths needed to satisfy a given set of traffics. This last formulation is especially useful in case of network dimensioning.

In this thesis, we expose a mathematical formulation of max-RWA problem. Thereafter, heuristic algorithms are proposed for routing and wavelength assignment. For sake of simplifications, sub-problems of routing and wavelength assignment are treated separately. For the first sub-problem, we seek a set of shortest arcs' disjointed paths for each traffic. The selection of the paths as well as the wavelength assignment is made thereafter by a local research heuristic, while taking into consideration our starting objective: minimize the number of wavelengths to satisfy the whole requests. To evaluate the performance of the proposed algorithms, a plan of experiments is built and the algorithms are performed for several test sets, in order to compute the number of wavelengths as well as the running times. In addition, several simple breakdown scenarios were considered and, for each one, the number of blocked traffics is calculated.

Thus, a presentation of the results found by our algorithms is made by comparing the latter with those obtained by the *TabouN* algorithm, as it treats the cases of requests for robust and permanent connections. Thereafter, we compared our results with an algorithm met in the literature and with the high delimiters obtained with CPLEX. In general, the obtained results are definitely better than those produced by the algorithms of reference. Even while comparing them with the high delimiters obtained by CPLEX, our algorithms manage at several blows to find optimal values.

## TABLE DES MATIERES

Remerciements .....	iv
RÉSUMÉ .....	v
ABSTRACT .....	viii
TABLE DES MATIERES .....	x
LISTE DES FIGURES .....	xiii
LISTE DES TABLEAUX .....	xv
LISTE DES SIGLES ET ABRÉVIATIONS .....	xvi
CHAPITRE I INTRODUCTION.....	1
1.1 Définitions et concepts de base .....	2
1.2 Éléments de la problématique .....	6
1.3 Objectifs de recherche.....	8
1.4 Plan du mémoire .....	9
CHAPITRE II ANALYSE D'ALGORITHMES DE ROUTAGE ET D'AFFECTATION DE LONGUEURS D'ONDE .....	10
2.1 La technologie WDM.....	11
2.2 Aspects fondamentaux du problème .....	13
2.2.1 Routage optique .....	13
2.2.2 Contrôle et gestion du réseau .....	15
2.2.3 Survivabilité et gestion des pannes .....	16
2.3 Routage et affectation de longueurs d'onde.....	17
2.3.1 Formulation du problème.....	18
2.3.2 Problème RWA avec conversion .....	22
2.3.3 Routage dans les réseaux optiques WDM.....	23
2.3.4 Affectation de longueurs d'onde .....	25
2.3.4.1 Coloriage de graphe .....	26

2.3.4.2	Heuristiques d'affectation de longueurs d'onde .....	27
2.4	Autres algorithmes et modèles de résolution .....	29
2.4.1	Algorithme <i>H+</i> .....	29
2.4.2	L'algorithme <i>TabouP</i> .....	30
2.4.3	L'algorithme WCA .....	33
2.4.4	Algorithme de chemins disjoints d'arcs .....	36
2.4.5	Algorithme génétique et recuit simulé .....	37
2.4.6	Approche quasi-statique avec et sans conversion de longueurs d'onde .....	38
CHAPITRE III ALGORITHMES DE ROUTAGE ET D'AFFECTATION DE LONGUEURS D'ONDE PROPOSÉS .....		41
3.1	Formulation mathématique du problème .....	41
3.1.1	Exemple illustratif .....	45
3.1.2	Considération des scénarios de pannes simples .....	46
3.2	Méthode de résolution .....	49
3.2.1	Considérations théoriques .....	49
3.2.2	Description de l'heuristique .....	50
3.2.3	Sous-problème de routage .....	54
3.2.4	Sélection des routes et obtention du nombre minimal de longueurs d'onde... ..	56
3.2.5	Affectation de longueurs d'onde .....	60
3.2.6	Considération des scénarios de pannes .....	62
3.3	Généralisation du problème max_RWA pour les scénarios de pannes .....	64
3.3.1	Méthode de résolution proposée .....	64
3.4	Illustration de l'algorithme Lambda .....	65
3.5	Analyse de complexité de l'algorithme proposé .....	69
CHAPITRE IV IMPLÉMENTATION ET ANALYSE DES RÉSULTATS .....		73
4.1	Détails d'implémentation .....	73
4.2	Jeux de tests utilisés .....	76
4.2.1	Les Fichiers de topologie .....	77
4.2.2	Les fichiers de trafic .....	78

4.3 Évaluation de performance .....	80
4.3.1 Calibration des paramètres de l'algorithme .....	80
4.3.2 Comportement aléatoire de l'algorithme .....	82
4.3.3 Analyse des résultats de l'algorithme .....	84
CHAPITRE V CONCLUSION .....	95
5.1 Synthèse des travaux .....	95
5.2 Limitations des travaux .....	97
5.3 Travaux futurs .....	98
BIBLIOGRAPHIE .....	99

## LISTE DES FIGURES

Figure 1.1 Structure d'une fibre optique.....	2
Figure 1.2 Topologies physique et logique d'un réseau optique.....	5
Figure 1.3 Routage et affectation de longueurs d'onde.....	6
Figure 2.1 Multiplexage de longueurs d'onde.....	12
Figure 2.2 Illustration de la transformation du graphe .....	24
Figure 2.3 Algorithme $H$ .....	31
Figure 2.4 Algorithme $H+$ .....	32
Figure 2.5 Illustration de la transformation du graphe.....	34
Figure 2.6 Fonction de coût sur un lien $l$ .....	39
Figure 3.1 Illustration du problème max-RWA.....	46
Figure 3.2 Nombre de blocages pour les deux solutions A et C.....	49
Figure 3.3 Ensemble stable et sous-graphe induit.....	50
Figure 3.4 Schéma général de résolution.....	52
Figure 3.5 Construction du graphe auxiliaire $G_l$ .....	54
Figure 3.6 Cas particulier de routage.....	56
Figure 3.7 Algorithme de recherche du nombre minimal de longueurs d'onde.....	57
Figure 3.8 Schéma général de la recherche taboue.....	60
Figure 3.9 Traitement des pannes simples.....	64
Figure 3.10 Généralisation du problème max-RWA pour les scénarios de pannes simples.....	66
Figure 3.11 Illustration de la solution de routage.....	67
Figure 3.12 Construction d'un ensemble stable dans le graphe auxiliaire.....	69
Figure 3.13 Ensemble stable dans le deuxième graphe auxiliaire.....	70
Figure 3.14 Illustration de l'affectation des longueurs d'onde.....	70
Figure 3.15 Graphe auxiliaire dans le scénario de panne du lien (4,5).....	71

Figure 4.1 Nomenclature des fichiers en entrée.....	75
Figure 4.2 Topologie du réseau aléatoire à 20 nœuds.....	79
Figure 4.3 Topologie du réseau NSFNET.....	79
Figure 4.4 Effets de la variation de la graine aléatoire.....	84
Figure 4.5 Illustration du nombre de connexions bloquées dans les scénarios de pannes avec des connexions robustes.....	91
Figure 4.6 Nombre de connexions bloquées dans les scénarios de pannes.....	92
Figure 4.7 Trafics acheminés pour le réseau NSFNET.....	94
Figure 4.8 Trafics acheminés pour le réseau EONNET.....	94

## LISTE DES TABLEAUX

Tableau 4.1 Topologies physiques considérées.....	78
Tableau 4.2 Trafics considérés pour chaque topologie physique.....	80
Tableau 4.3 Nombre de longueurs d'onde nécessaires (scénarios avec et sans panne)....	85
Tableau 4.4 Nombre trafics bloqués dans les scénarios de pannes.....	86
Tableau 4.5 Nombre de trafics bloqués dans le scénario de base.....	87
Tableau 4.6 Nombre de trafics bloqués dans les scénarios de pannes.....	87
Tableau 4.7 Nombre de connexions bloquées pour le réseau à 10 nœuds.....	88
Tableau 4.8 Nombre de connexions bloquées pour le réseau NSFNET.....	88
Tableau 4.9 Nombre de connexions bloquées pour le réseau à 20 nœuds.....	89
Tableau 4.10 Nombre de connexions bloquées pour le réseau ARPANET.....	90
Tableau 4.11 Nombre de connexions bloquées pour le réseau à 30 nœuds.....	90
Tableau 4.12 Topologies et trafics considérés.....	92
Tableau 4.13 Résultats obtenus par <i>Lambda</i> pour NSFNET et EONNET.....	93
Tableau 4.14 Évaluation du temps d'exécution.....	95

## LISTE DES SIGLES ET ABRÉVIATIONS

<u>Sigles ou abréviations</u>	<u>Signification</u>
WDM	Wavelength Division Multiplexing (Multiplexage en longueurs d'onde)
RWA	Routing and Wavelength Assignment (Routage et affectation de longueurs d'onde)
RWAP	Routing and Wavelength Assignment Problem (Problème de routage et d'affectation de longueurs d'onde)
ILP	Integer Linear Program (Programme linéaire en nombres entiers)
LP	Linear Program (Programme linéaire)
CDA	Chemins Disjoints d'Arcs

# CHAPITRE I

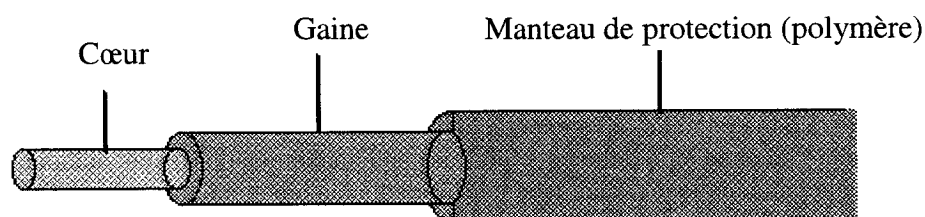
## INTRODUCTION

De nos jours, les réseaux ne sont plus exclusivement dédiés au transport de la voix mais se voient plutôt attribuer des fonctions autres. En effet, l'apparition d'Internet a engendré un surcroît très rapide du trafic de données vivement entretenu par l'intérêt dévoué des usagers. Avec l'évolution rapide des télécommunications et l'explosion des technologies de l'information, de nouvelles habitudes se sont faites sentir chez les utilisateurs. Au rythme de cette évolution, l'engouement des usagers croît continuellement envers ces dites technologies. Cet enthousiasme est d'autant plus important avec l'avènement et l'intégration de nouveaux services sur les réseaux actuels. Ainsi, la quantité croissante de trafic (voix, données, vidéo...) et le support de différents types de services sur un même réseau contraint d'assurer une certaine qualité de service, ont créé un immense besoin en bande passante. C'est dans ce contexte que la fibre optique a bénéficié d'un épanouissement rapide pour devenir le support privilégié des réseaux actuels. Néanmoins, les processus de planification, de gestion et de maintenance des réseaux optiques s'avèrent complexes. Les équipements déployés par les opérateurs doivent permettre de gérer la complexité des connexions entre les clients, tout en minimisant le coût global d'exploitation. Parallèlement à la croissance du trafic, la demande en qualité de service est d'autant plus forte, notamment pour la sécurisation des connexions. L'opérateur doit donc veiller à protéger chaque connexion d'un éventuel dysfonctionnement, tant au niveau matériel que logiciel. Dans ce mémoire, le problème de routage et d'affectation de longueur d'onde sera traité pour un modèle de trafic statique tout en tenant compte des différents cas de pannes simples.

Dans ce chapitre d'introduction, quelques concepts de base de ladite technologie seront exposés, ce qui nous permettra d'énoncer les éléments de la problématique, et de préciser nos objectifs de recherche ainsi que le plan du mémoire.

## 1.1 Définitions et concepts de base

Lorsqu'un faisceau lumineux heurte obliquement la surface qui sépare deux milieux plus ou moins transparents, il se divise en deux : une partie est réfléchiée tandis que l'autre est réfractée. L'indice de réfraction est une grandeur caractéristique des propriétés optiques d'un matériau. C'est ce principe qui est utilisé pour guider la lumière dans la fibre. La *fibre optique* comprend ainsi deux milieux : le cœur dans lequel l'énergie lumineuse se trouve confinée grâce à un second milieu, la gaine dont l'indice de réfraction est plus faible (voir Figure 1.1). Le cœur et la gaine optique sont en verre de silice de densité différente.



**Figure 1.1 Structure d'une fibre optique**

Dépendamment du diamètre du cœur, nous retrouvons la fibre monomode et la fibre multimode.

- la fibre monomode : dont le cœur est si fin que le chemin de propagation des différents modes est pratiquement direct. Elle est généralement utilisée pour de grandes distances et pour des applications demandant une grande largeur de

bande. On retrouve ces fibres essentiellement dans des réseaux WAN (*Wide Area Network*) ;

- la fibre multimode : elle est généralement utilisée dans les réseaux à étendue limitée (les réseaux LAN et MAN). Dépendamment du diamètre de la fibre et de la composition du cœur, nous retrouvons les fibres à gradient d'indice et à saut d'indice.

Les fibres optiques sont caractérisée par :

- l'atténuation qui représente l'affaiblissement du signal en fonction de la distance parcourue. Cette atténuation dépend de la longueur d'onde utilisée et peut atteindre 0.2 dB/Km aux environs de 1550 nm ;
- la dispersion qui représente l'étalement du spectre du signal à l'arrivée.

Toutefois, vu les limitations des dispositifs optiques et la lenteur relative des équipements électroniques, les transmissions sur un unique canal de la fibre ne permet de bénéficier que d'une petite fraction de la capacité de cette dernière. Afin d'exploiter le potentiel de ladite technologie, l'usage du multiplexage en longueurs d'onde WDM devient l'option de choix. Une fibre peut donc supporter plusieurs canaux en parallèle, chacun opérant sur une longueur d'onde particulière. Ces dernières doivent être suffisamment séparées pour éviter les interférences entre les signaux optiques et permettre aux nœuds en vigueur de faire un routage adéquat.

Un *réseau optique* est composé d'éléments interconnectés par des fibres optiques. L'ensemble des nœuds d'un réseau optique ainsi que les liens qui les connectent définit la *topologie physique* du réseau. Ce dernier peut être modélisé par un graphe dont les sommets correspondent aux nœuds du réseau. Un graphe est dit connexe si pour tout couple de sommets  $i$  et  $j$ , il existe au moins un chemin qui les relie, il est dit bi-connexe s'il en existe deux chemins disjoints qui relient  $i$  et  $j$ .

Dans le problème RWA, un *chemin optique* représente un ensemble de liens constituant une chaîne de l'origine à la destination. Pour une connexion, il peut y avoir plusieurs chemins optiques possibles, le fait donc de privilégier un chemin au détriment

des autres est entièrement lié à la stratégie ou à l'algorithme utilisé. Une fois le chemin optique sélectionné, une longueur d'onde lui est assignée.

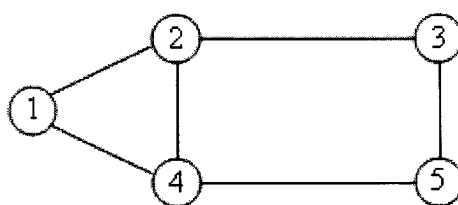
Dépendamment du budget du client, un chemin optique peut être de type permanent ou robuste. La différence réside dans la façon de faire face aux pannes. En effet, en cas de panne, seuls les chemins robustes touchés par cette dernière sont reroutés. Les chemins permanents traversant la portion défectueuse se voient automatiquement bloqués.

Dans les réseaux à routage de longueurs d'onde, deux chemins optiques traversant la même fibre doivent avoir des longueurs d'onde différentes. C'est la contrainte de *continuité de longueur d'onde* [2][3]. Pour pallier ce problème, certains équipements optiques permettent la conversion de longueurs d'onde entre les ports d'entrée et de sortie d'un nœud du réseau. La présence de tels équipements, appelés aussi convertisseurs de longueurs d'onde, va sans doute influencer les stratégies de résolution du problème RWA. Il existe deux types de conversion de longueur d'onde : la *conversion totale* dans laquelle une longueur d'onde en entrée peut être remplacée par n'importe quelle autre en sortie, et la *conversion partielle* où une longueur d'onde en entrée ne peut être convertie qu'en un ensemble restreint en sortie. Le recours à de tels dispositifs peut simplifier relativement le problème de routage et d'affectation de longueurs d'onde, ces derniers restent toutefois dispendieux, c'est ce qui limite leur utilisation. Avec la contrainte de continuité de longueur d'onde, seuls les chemins disjoints d'arcs peuvent avoir la même longueur d'onde [1].

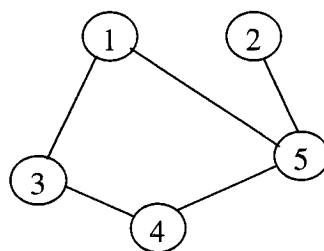
Comme dans tout réseau, un échange d'informations entre deux nœuds correspond à un trafic circulant entre ces derniers. Ce trafic est initié par une demande de communication entre le nœud demandeur ou *origine* et un nœud *destination*. Dans un modèle statique de trafic, l'ensemble des demandes de connexions à satisfaire est connu à l'avance et est modélisé par une *matrice de trafic* contenant les paires origine-destination des demandes. À partir de cette matrice, nous pouvons construire la *topologie logique* du réseau, qui est représentée par un graphe dont les sommets correspondent aux nœuds du réseau. Deux sommets dudit graphe sont reliés s'il y a une

demande de connexion entre ces derniers. La Figure 1.2 représente les topologies physique et logique d'un réseau optique dans le cas où la matrice de routage est formée par les paires  $(1,3)$ ,  $(1,5)$ ,  $(2,5)$ ,  $(4,5)$  et  $(3,4)$ .

Il est à noter que toutes les définitions précédentes supposent que les liens du réseau sont bidirectionnels. En pratique, un lien bidirectionnel correspond à une paire de fibres unidirectionnelles entre deux nœuds.



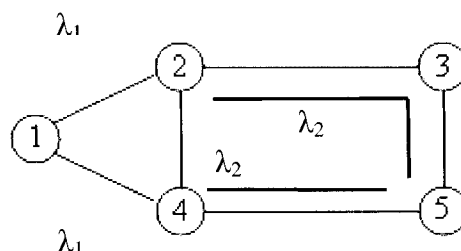
a) Topologie physique du réseau



b) Topologie logique du réseau

### Figure 1.2 Topologies physique et logique d'un réseau optique

Une solution potentielle au trafic présenté ci-haut est présentée à la Figure 1.3. Si on suppose qu'on n'a que deux longueurs d'onde disponibles (couleurs vert et bleu), et avec la solution de routage considérée, la connexion  $(4,3)$  ne peut être établie faute de longueur d'onde sur le lien  $(4,5)$ .



**Figure 1.3 Routage et affectation de longueur d'onde**

Comme tout équipement du domaine des télécommunications, les dispositifs optiques peuvent subir des pannes causées soit par un bris physique de fibre ou un dysfonctionnement matériel ou logiciel. La survivabilité des réseaux optiques représente donc un enjeu majeur pour les concepteurs de réseaux, vu la quantité immense d'informations transportée par les fibres surtout installées au niveau des réseaux cœurs [6].

## 1.2 Éléments de la problématique

Vu la complexité de planification des réseaux optiques, une approche modulaire est souvent adoptée. En effet, le nombre de paramètres et de contraintes à respecter croît exponentiellement avec la taille du réseau à considérer. Cette complexité est d'autant plus accentuée dans les réseaux à multiplexage de longueurs d'onde (WDM). Pour un ensemble de demandes de communication, la recherche d'un chemin et l'affectation de longueurs d'onde constituent le problème RWA (*Routing and Wavelength Assignment*).

Le problème de planification des réseaux optiques s'étale sur plusieurs étapes d'optimisation dont :

- caractérisation du trafic : il faut prévoir la demande à l'avance pour pouvoir dimensionner le réseau ;
- conception topologique : l'emplacement physique des nœuds ainsi que la meilleure topologie d'interconnexion doivent être déterminés pour le réseau ;

- routage et affectation de longueurs d'onde : il faut trouver les chemins pour chaque trafic ainsi que la longueur d'onde correspondante ;
- dimensionnement des liens : suite à une affectation supposée optimale, le nombre de fibres est déterminé pour chaque lien en vue de minimiser le taux de blocage.

Pendant toutes ces étapes de planification, la survivabilité des réseaux doit être rigoureusement prise en compte. En effet, ces derniers doivent avoir des mécanismes de résilience aux pannes, autant rapides qu'efficaces, afin de minimiser les pertes éventuelles.

Dans ce mémoire, le modèle statique de trafic est considéré. Cela dit que la matrice de trafic est connue à l'avance. Dans ce cas, les problèmes de routage, et d'affectation de longueur d'onde sont souvent traités séparément [4][5][6] :

- *le routage* : qui consiste à trouver une route ou un chemin à emprunter par chaque trafic, en considérant plusieurs métriques lors de l'établissement de ces chemins ;
- *l'affectation de longueur d'onde* : un fois les chemins trouvés pour chaque paire source-destination, les longueurs d'onde doivent être assignées, en fonction de la disponibilité des ressources du réseau, de la manière optimale. Les contraintes à respecter sont celles de l'unicité de longueur d'onde sur une fibre donnée, et de continuité de longueur d'onde.

Tel que mentionné précédemment, le routage optique est souvent appelé RWA dans la littérature. Pour le modèle de trafic considéré, l'objectif est souvent de maximiser les trafics acheminés, contraints au nombre de longueurs d'onde limité disponible dans chaque fibre. Le problème dual est de minimiser le nombre de longueurs d'onde nécessaires pour satisfaire l'ensemble des demandes de connexions. Plusieurs travaux traitant du problème RWA pour un modèle statique de trafic ont été recensés dans la littérature [5][6][7][9]. Cependant, rares sont ceux qui considéraient les chemins robustes et permanents ensemble. D'autre part, le problème RWA a été démontré NP-complet [3], ce qui veut dire que la mise en œuvre de méthodes exactes

nécessite des efforts de calcul considérables et la solution optimale ne peut être trouvée en un temps polynomial. En effet, la taille de l'espace de recherche croît exponentiellement avec la taille du problème (topologie du réseau et nombre de trafic demandé). Cela milite en faveur des méthodes heuristiques qui permettent de trouver des solutions acceptables mais non nécessairement optimales, dans des temps de calcul raisonnables. Dans la littérature, plusieurs approches heuristiques ont été utilisées pour aborder le problème de RWA [7][8][9][10][12][19][28][29][30][31][32]. En effet, des heuristiques constructives (méthodes gloutonnes statiques et dynamiques), génétiques et de recherche locale ont été proposées et ont abouti à des résultats plus ou moins bons.

### **1.3 Objectifs de recherche**

Dans ce mémoire, nous nous intéressons au routage et à l'affectation de longueurs d'onde dans un réseau optique. L'objectif principal est de proposer une méthode heuristique efficace pour résoudre le problème RWA statique, et ce, en maximisant le nombre de demandes satisfaites, donc en minimisant le taux de blocage.

De manière plus spécifique, ce mémoire vise à :

- étudier les différents modèles présentés pour la formulation du problème RWA et analyser subséquemment les méthodes heuristiques de résolution proposées dans la littérature ;
- proposer un modèle de résolution du problème RWA statique ainsi qu'un schéma d'affectation des longueurs d'onde basé sur des méthodes heuristiques, afin de satisfaire un maximum de demandes pouvant être robustes ou permanentes dépendamment de leur tolérance aux pannes ;
- implémenter et évaluer la performance de l'algorithme proposé en comparant ses résultats à ceux fournis par les meilleures approches de résolution identifiées dans la littérature.

### ***1.4 Plan du mémoire***

Ce mémoire s'articule en cinq chapitres. Après le bref exposé du problème présenté au chapitre d'introduction, le chapitre 2 visera à faire une présentation détaillée des réseaux optiques, de la technologie de multiplexage de longueurs d'onde WDM et de certains algorithmes et méthodes de résolution du problème RWA rencontrés dans la littérature. Dans le chapitre 3, une formulation mathématique du problème ainsi qu'un modèle de résolution et une analyse de complexité des algorithmes utilisés seront présentés. L'implémentation de ces algorithmes ainsi que l'analyse des résultats obtenus feront l'objet du chapitre 4. Finalement, dans le chapitre de conclusion, nous procéderons à une synthèse des travaux effectués, suivie de l'analyse des limites des algorithmes présentés pour terminer avec quelques indications de travaux futurs.

## **CHAPITRE II**

# **ANALYSE D'ALGORITHMES DE ROUTAGE ET D'AFFECTATION DE LONGUEURS D'ONDE**

Avec le temps, les réseaux de télécommunication ont évolué rapidement des simples communications point à point au support d'un grand nombre de transmissions simultanées voire concurrentes. Cet épanouissement grandissant des réseaux est le fruit des nouvelles habitudes des utilisateurs. En effet, l'avènement de nouvelles applications multimédia et l'expansion des services Internet, telle la vidéoconférence, les différentes applications JAVA, etc. ont fait de la planification et de l'organisation des réseaux une tâche ardue. Ces nouvelles tendances des usagers ainsi que la « gourmandise » des nouveaux services en bande passante incitent à redéfinir les choix stratégiques des architectures mises en places (meilleure planification supportant les demandes grandissantes en bande passante) d'une part, et surtout de reconsidérer un support physique offrant la bande passante demandée. Face aux limites des réseaux actuels, y compris ceux à haut débit (ATM/SONET), le développement de la technologie WDM s'avère très prometteur. En fait, l'utilisation actuelle de la fibre optique est loin d'être optimale et ce, surtout à cause des limitations des équipements électroniques. Le multiplexage en longueurs d'onde WDM permet une utilisation quasi-optimale de la fibre en la faisant supporter plusieurs communications concurrentes, chacune sur une longueur d'onde à part. Cela rend donc possible la transmission de plusieurs téraoctets par seconde. Afin de bien exploiter ce potentiel prometteur de la technologie et de saisir toutes les possibilités offertes par la dernière, plusieurs considérations, tels le routage et l'affectation de longueur d'onde, doivent être prises en compte lors de la planification des réseaux optiques. Dans ce chapitre, nous ferons d'abord un survol des réseaux optiques et de la technologie WDM. Nous passerons par la suite à une formulation du

problème de routage optique et de survivabilité (gestion des pannes), pour finir par un panorama des méthodes classiques de résolution.

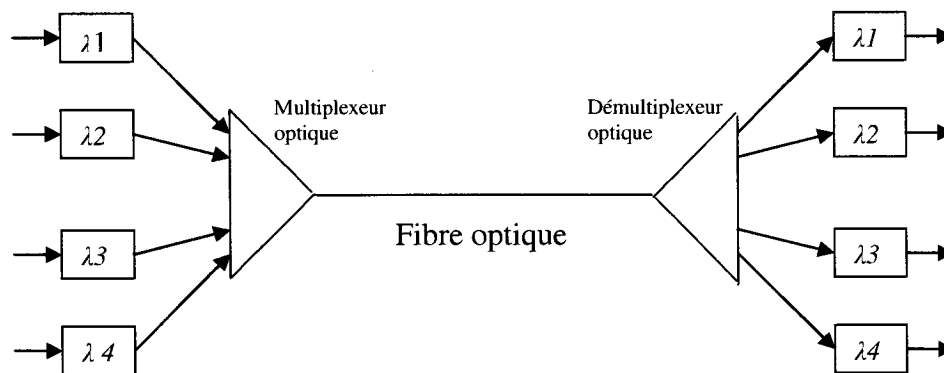
## **2.1 La technologie WDM**

Un des inconvénients les plus contraignants de l'usage de la fibre optique traditionnelle est l'énorme gaspillage de bande passante. En effet, vu les limitations des dispositifs optiques, la transmission sur un seul canal nous permet seulement d'exploiter une fraction minime de la capacité offerte par la fibre. Apparue au début des années 1990, le WDM permet de diviser la largeur de bande offerte par la fibre en canaux, opérant chacun sur une longueur d'onde indépendamment les unes des autres. Cette technologie permet donc de remédier à la grande différence entre les débits d'opération des équipements optiques et électroniques, en juxtaposant plusieurs signaux provenant de différents usagers sur une même fibre.

WDM est une technologie de multiplexage de longueurs d'onde qui permet d'accroître la largeur de bande des fibres optiques. Cette dernière utilise une vitesse de modulation commune pour tous les signaux. Le nombre de longueurs d'onde que peut transporter une fibre croît continuellement et peut atteindre quelques centaines de couleurs. Si une longueur d'onde nous offre un taux de transmission de 2.5 Gb/s, une fibre de 80 canaux nous permettra de transporter à des taux avoisinant les 200 Gb/s (voir Figure 2.1). En outre du multiplexage en longueurs d'onde, un lien peut contenir plusieurs fibres pour construire un réseau *k-fibres*.

Avec l'avènement du WDM, plusieurs équipements ont vu le jour afin de permettre d'exploiter le potentiel de ladite technologie. Cependant, une des clefs du succès du WDM est l'amplificateur de fibre dopée à l'Erbium (Erbium Doped Fiber Amplifier : EODF). En effet, ce dernier rend possible l'épanouissement de la technologie à grande échelle vu qu'il permet de compenser les pertes des signaux optiques, engendrées surtout par l'étendue de la fibre (plusieurs kilomètres) et les opérations de multiplexage et de démultiplexage, et ce, sans faire de conversion

optoélectronique [11][15]. Une présentation des différents équipements est illustrée à la Figure 2.1.



**Figure 2.1 Multiplexage de longueur d'onde**

### ***OADM***

L'OADM (*Optical Add and Drop Multiplexer*) ou multiplexeur optique d'insertion et extraction est un dispositif qui permet d'injecter ou d'extraire sélectivement un canal sans faire de conversion optoélectronique. Il consiste en un démultiplexeur suivi par un ensemble de commutateur 2x2, pour chacune des longueurs d'onde, suivis d'un multiplexeur.

### ***L'étoile passive***

C'est un dispositif qui permet de diffuser un signal présent sur l'une de ses entrées à tous ses ports de sortie. Si deux stations émettent en même temps sur une même longueur d'onde, il y a risque de collision. C'est pour cela que le déploiement d'un tel dispositif se limite aux réseaux locaux de petite taille, avec des mécanismes appropriés pour éviter les collisions.

### ***Le routeur passif***

Ce dispositif permet de diriger les canaux d'entrée vers leurs sorties respectives. La matrice de routage est prédéfinie.

### ***Le commutateur actif***

Comme pour le routeur passif, il permet de router séparément chacune des longueurs d'onde incidentes dans une même fibre à des fibres séparées. Sa matrice de routage est cependant reconfigurable électroniquement et sur commande. Certains commutateurs actifs permettent la conversion (totale ou partielle) de longueurs d'onde.

## **2.2 Aspects fondamentaux du problème**

Dans un réseau à routage de longueurs d'onde, plusieurs OXC (Optical Cross Connect) sont connectés par fibre pour former une topologie physique arbitraire. Chaque utilisateur est connecté à l'un des commutateurs à travers un lien optique, et la combinaison de l'utilisateur et du OXC correspondant forme un nœud du réseau. Un chemin optique est un canal de communication entre deux nœuds. Le défi essentiel à ce niveau est donc d'optimiser l'utilisation des installations, et ce, principalement en maximisant le nombre de connexions établies dans le réseau. Pour ce faire, trois axes d'intérêt dans les réseaux à routage de longueurs d'onde WDM émergent :

- le routage optique ;
- le contrôle et la gestion du réseau ;
- la survivabilité et la gestion des pannes.

### **2.2.1 Routage optique**

Il existe trois formes de trafic à étudier dans un réseau à routage de longueurs d'onde [1][15] :

- *Trafic statique* : dans le modèle statique, on suppose que la demande est fixe et connue au départ, le problème revient à essayer d'acheminer toutes les

connexions en minimisant l'utilisation des ressources du réseau, à savoir le nombre de longueurs d'onde et le nombre de fibres.

- *Trafic incrémental* ou *quasi-statique* : les connexions arrivent séquentiellement, un chemin optique est attribué à chaque connexion qui demeure indéfiniment.
- *Trafic dynamique* : un chemin optique est assignée à chaque demande de connexion et est relâché après un certain temps.

Pour les deux derniers modèles de trafic, l'objectif est d'acheminer les connexions et de leur attribuer des longueurs d'onde appropriées en vue de minimiser le taux de blocage des connexions, ou de maximiser le nombre de connexions établies à tout moment [13][14][31][38][39].

En somme, le routage optique consiste à chercher un chemin optique pour chacune des demandes de connexions ainsi que l'affectation de longueurs d'onde. Le problème revient donc à savoir s'il est possible d'acheminer un ensemble de connexions sous les contraintes de la limitation des ressources réseau (fibres et longueurs d'onde).

Le problème RWA pour le modèle statique de trafic peut être formulé comme une ILP (*Integer Linear Program*) mixte, qui est connue NP-complet [3][9]. Ainsi, le problème est généralement subdivisé en deux sous-problèmes : le routage et l'affectation de longueurs d'onde, et chacun de ces derniers peut être résolu séparément. Dans ce cas, on peut commencer par trouver un ensemble de chemins en appliquant une méthode de routage simple (plus court chemin par exemple), pour ensuite leur affecter des longueurs d'onde d'une manière judicieuse. Plusieurs algorithmes d'affectation de longueurs d'onde, basés généralement sur des métaheuristiques, existent dans la littérature [6][9][19][26][30]. Il est à mentionner que cette approche de résolution (soit la subdivision du problème) ne conduit nécessairement pas à une solution optimale, elle reste toutefois privilégiée, vu son degré relatif de simplicité. Finalement, les problèmes de routage et d'affectation de longueurs d'onde sont classés NP-Complets aussi [3][6]. Ainsi, les efforts de recherche dans le domaine des réseaux optiques WDM se concentrent à trouver des heuristiques qui, en des temps de calcul raisonnables, trouvent des solutions plus ou moins acceptables, qui ne sont pas nécessairement optimales.

### 2.2.2 Contrôle et gestion du réseau

Dans un réseau à routage de longueurs d'onde, un mécanisme de contrôle est indispensable pour l'établissement et la relâche des connexions. Quand une connexion arrive, ce mécanisme doit être capable de sélectionner une route, lui affecter une longueur d'onde et configurer les commutateurs correspondants [11][22]. Ce mécanisme doit aussi permettre de faire des mises à jour périodiques pour avoir une bonne vision de l'état du réseau. Un tel mécanisme peut être centralisé ou distribué. Les systèmes distribués sont souvent préférés car plus robustes. L'objectif des différents efforts de recherche à ce sujet visent à :

- minimiser la probabilité de blocage des demandes de connexions ;
- minimiser les délais d'établissement des connexions ;
- minimiser la bande passante utilisée par les mécanismes de contrôle ;
- maximiser l'évolutivité des réseaux.

Deux schémas de contrôle distribué existent en littérature [15] :

- *États des liens* (qui peut être implanté en utilisant l'algorithme OSPF): chaque nœud maintient la topologie complète du réseau, incluant les informations sur l'affectation des longueurs d'onde. Ces informations sont directement échangées entre nœuds voisins. Lors d'une demande de connexion, le nœud source utilise l'information topologique pour sélectionner une route et une longueur d'onde. Une demande de réservation est lancée dans le réseau. S'il y a blocage, la connexion est rejetée. Sinon, un accusé de réception lui est émis pour confirmer l'affectation du chemin. Des échanges de mise à jour se font donc entre les nœuds du réseau.
- *Le routage réparti* : Dans ce modèle, les routes sont sélectionnées d'une façon distribuée sans connaître la topologie globale du réseau. La table de routage de chaque nœud, qui est mise à jour à partir de celles de ses voisins, contient les informations sur les chemins les moins coûteux pour atteindre n'importe quelle destination dans le réseau ainsi que la longueur d'onde associée.

### 2.2.3 Survivabilité et gestion des pannes

Dans un réseau à routage de longueurs d'onde WDM, la panne d'un élément (fibre ou commutateur) peut engendrer la coupure de plusieurs canaux, et donc une énorme perte d'informations. C'est pour cela que la gestion des pannes devient une préoccupation majeure des concepteurs des réseaux optiques. Il existe deux types de pannes : panne des liens et panne des nœuds [6]. Il est à noter que les pannes des nœuds sont plus complexes à traiter, vu qu'ils peuvent engendrer la désactivation de plusieurs liens. Cependant, celles des liens constituent la plus grande source de pannes et sont surtout causées par les bris des fibres. Dans la littérature, plusieurs approches traitent la survivabilité des réseaux, mais considèrent surtout les pannes simples (*single link failure*) [6][16][17][18]. En effet, la probabilité d'occurrence simultanée de plus d'une panne est très faible.

Il est aussi à mentionner que, lors d'une panne, on peut, soit faire la reconfiguration totale de toutes les connexions, même celles non atteintes, ou simplement rerouter les connexions bloquées. Bien que la reconfiguration puisse permettre de trouver de meilleures solutions, il est préférable de ne traiter que les connexions touchées, de crainte de perturbations.

Il existe deux méthodes pour remédier aux pannes du réseau : approche proactive et approche réactive.

#### ***L'approche proactive***

Appelée aussi protection, cette approche prévoit et réserve à l'avance les ressources du réseau nécessaires pour ré-acheminer les connexions bloquées en cas de panne. Généralement, les «chemins de secours» considérés sont disjoints d'arcs des chemins principaux afin de garantir la restauration des connexions touchées par une panne simple d'un lien [15]. Voici quelques exemples de protection :

- *Protection optique (1+1)*: Pour chaque connexion protégée, un chemin de secours disjoint d'arcs du chemin principal lui est attribué. Les données sont envoyées simultanément sur les deux chemins ;

- *Protection optique (1:1)* : Cette technique fonctionne comme la protection (1+1), sauf que les données ne sont envoyées que sur le chemin principal. On n'émet sur le chemin de secours que si une panne survient ;
- *Protection optique (1:N)* : Afin d'optimiser la distribution des ressources du réseau, N chemins principaux peuvent partager un même chemin de secours ; il est préférable de choisir les chemins principaux disjoints d'arcs afin d'éviter le blocage en cas d'une panne simple.

### ***L'approche réactive***

Appelée aussi restauration, cette approche utilise la capacité libre dans le réseau pour restaurer dynamiquement les services affectés par la panne, au moment de son occurrence. Il existe deux types de restauration [15] [17] :

- *Restauration de lien* : Dans cette méthode, les connexions touchées par la panne du lien sont contournées autour de ce dernier. Les extrémités du lien en panne essaient de découvrir dynamiquement une route alternative pour se rejoindre afin d'acheminer le trafic par celle-ci.
- *Restauration de chemin* : Dans cette méthode, la source et la destination de chaque connexion sont informées de la panne et commencent leur découverte dynamique d'un chemin alternatif (*backup route*) pour le trafic considéré.

Généralement, la restauration dynamique est plus efficace. En effet, l'affectation du nouveau chemin ne se fait que si nécessaire. De plus, elle offre une protection contre plusieurs types de pannes (simples, doubles...). Toutefois, l'approche proactive a un meilleur délai de restauration et offre plus de garantie.

## ***2.3 Routage et affectation de longueurs d'onde***

Le problème RWA ou routage et affectation de longueurs d'onde revient à affecter des chemins et des longueurs d'ondes aux demandes de connexions, tenant compte des contraintes physiques des ressources du réseau (nombre limité de longueurs

d'ondes...). Comme mentionné précédemment, la charge peut être statique ou dynamique. Dans le premier cas, les opérations du RWA sont faites hors ligne. Cependant, dans le second cas, elles sont faites en temps réel et tiennent compte de l'état actuel du réseau (chemins affectés, longueurs d'onde disponibles, pannes éventuelles...).

### 2.3.1 Formulation du problème

Dans le modèle statique du RWA [1], les demandes de trafic sont connues à l'avance et les opérations de routage et d'affectation de longueurs d'onde sont faites hors ligne. Typiquement, l'objectif principal est de minimiser le nombre de longueurs d'onde requises à l'établissement des chemins optiques pour une topologie physique arbitraire. Une alternative à minimiser le nombre de longueurs d'onde utilisées serait de maximiser le nombre de connexions qui peuvent être établies pour un nombre de longueurs d'onde et un ensemble de demandes de connexions (qui revient à minimiser la probabilité de blocage). Comme il s'agit d'un problème de minimisation sujet à des contraintes, la programmation linéaire est souvent utilisée.

Dans un modèle de programmation linéaire, l'ensemble de contraintes est ramené à un système d'équations et d'inéquations linéaires. L'objectif est donc de trouver les valeurs positives des variables qui optimisent la fonction objectif (fonction économique).

Sous la contrainte de continuité de longueur d'onde, le modèle statique du RWA peut être formulé comme une ILP, dans laquelle la fonction objective est de minimiser le flot traversant chaque lien.

Soit  $\lambda_{sdw}$  le nombre de demandes de connexions pour une source  $s$  et une destination  $d$  avec une longueur d'onde  $w$ . On suppose un réseau « 1-fibre », c'est-à-dire que deux fibres chacune unidirectionnelle existent entre deux nœuds adjacents ( $\lambda_{sdw} \leq 1$ ). Et soit  $F_{ij}^{sdw}$  le nombre de connexions entre  $s$  et  $d$  ayant la longueur d'onde  $w$  et traversant l'arc  $(i,j)$  (avec  $F_{ij}^{sdw} \leq 1$ ). Pour une topologie physique donnée, un ensemble de longueurs d'onde sur un lien et une matrice de trafic  $A$  où  $A_{sd}$  est le nombre de connexions requises entre les nœuds  $s$  et  $d$ , le problème peut être formulé de la manière suivante :

**Minimiser** :  $F_{max}$

Tel que :

$$F_{max} \geq \sum_{s,d,w} F_{ij}^{sdw} \quad \forall i, j \quad (2.1)$$

Sous les contraintes suivantes :

$$\sum_i F_{ij}^{sdw} - \sum_k F_{jk}^{sdw} = \begin{cases} -\lambda_{sdw}, & s = j \\ \lambda_{sdw}, & d = j \\ 0, & \text{sinon} \end{cases} \quad (2.2)$$

Contrainte de conservation de flot entre la source et la destination.

$$\sum_w \lambda_{sdw} = \Lambda_{sd} \quad (2.3)$$

$$F_{ij}^{sdw} \in \{0,1\} \quad (2.4)$$

$$\sum_{s,d} F_{ij}^{sdw} \leq 1 \quad (2.5)$$

La formulation duale à celle présentée ci-haut est appelée max-RWA. Le but est donc de maximiser le nombre de demandes pouvant être établies, étant donné une matrice de connexions et un ensemble de longueurs d'onde disponibles sur chaque lien. Krishnaswamy et Sirvarajan [9] ont proposé deux formulations possibles en ILP du problème dans lesquelles ils utilisent principalement des contraintes de continuité de longueur d'onde. À la différence de toutes les formulations antérieures, le choix du chemin et de la longueur d'onde appropriée est donné par la résolution de l'ILP. Cette approche est appelée *link based ILP formulation*, c'est-à-dire que les contraintes sont placées sur les liens du réseau. La présentation de la première formulation est faite ci-dessous.

Soit  $\rho(i,j)$  le nombre de connexions à établir entre les nœuds  $i$  et  $j$ . Comme mentionné ci-haut, pour chaque connexion, une route et une longueur d'onde doivent

être affectées. Notons que deux chemins optiques ayant la même longueur d'onde ne peuvent avoir de lien physique en commun.

Voici les paramètres dont on dispose :

- $N$  : le nombre de nœuds dans le réseau ;
- $\rho$  : la matrice de trafic ;  
 $\rho(i,j)$  est donc le nombre de connexions à établir entre les nœud  $i$  et  $j$  ;
- $P_{lm}$  : l'existence d'un lien entre les nœuds  $i$  et  $j$ .  $P_{lm} = 1$  si  $(l,m)$  existe, 0 sinon ;
- $F$  : le nombre de longueurs d'onde dans une fibre.

Les notations suivantes sont utilisées dans la formulation :

- $i$  et  $j$  les nœuds source et destination d'un chemin optique ;
- $l$  et  $m$  les extrémités d'un lien physique ;
- $k$ , utilisé en exposant, correspond au numéro de la longueur d'onde  
 $k \in \{0,1,2,\dots,F-1\}$  ;
- $q$ , correspond au  $q^{\text{ème}}$  chemin optique,  $q \in \{1,2,\dots,\rho(i,j)\}$ .

Les variables du problème sont les suivantes :

- $b_q(i,j) = 1$ , s'il existe un  $q^{\text{ème}}$  chemin entre  $i$  et  $j$ , sinon ;
- $C^{(k,q)}(i,j) = 1$ , si le  $q^{\text{ème}}$  chemin optique entre  $i$  et  $j$  utilise la longueur d'onde  $k$ ,  
 $C^{(k,q)}(i,j) = 0$ , sinon ;
- $C_{l,m}^{(k,q)}(i,j) = 1$ , si le  $q^{\text{ème}}$  chemin optique entre  $i$  et  $j$  utilise la longueur d'onde  $k$  et est routé sur le lien physique  $(l,m)$ , sinon,  $C_{l,m}^{(k,q)}(i,j) = 0$ .

La formulation est la suivante :

$$\text{Maximiser : } \sum_{i,j} \sum_q b_q(i,j) \quad (2.6)$$

Sous les contraintes suivantes :

$$\sum_q b_q(i, j) \leq \rho(i, j), \text{ avec pour tout } (i, j), \sum_q b_q(i, j) \in \{0, 1\} \quad (2.7)$$

$$\sum_{k=0}^{F-1} C^{(k,q)}(i, j) = b_q(i, j), \text{ pour tout } q \text{ et } (i, j) \quad (2.8)$$

$$C_{l,m}^{(k,q)}(i, j) \leq C^{(k,q)}(i, j), \text{ pour tout } q, (i, j), (l, m) \text{ et } k \quad (2.9)$$

$$\sum_q \sum_{(i,j)} C_{l,m}^{(k,q)}(i, j) \leq 1, \text{ pour tout } k \text{ et } (l, m) \quad (2.10)$$

$$\sum_l C_{l,m}^{(k,q)}(i, j) P_{l,m} - \sum_l C_{m,l}^{(k,q)}(i, j) P_{m,l} = \begin{cases} C^{(k,q)}(i, j), m = i \\ -C^{(k,q)}(i, j), m = j \\ 0, m \neq i, j \end{cases} \quad (2.11)$$

Pour tout  $(i, j), k, q$  et  $m$ .

La relation (2.6) est la fonction objectif qui doit être maximisée et qui correspond au nombre total de connexions. Les relations (2.7) à (2.11) correspondent aux contraintes du problème qu'il faut respecter lors de la résolution. La première contrainte signifie que le nombre de connexions établies entre  $i$  et  $j$  est au plus  $\rho(i, j)$ . Les relations qui suivent sont des contraintes de continuité de longueur d'onde. La contrainte (2.8) signifie que si un chemin optique  $b_q(i, j)$  existe, une seule longueur d'onde parmi les  $F$  lui est affectée. La relation (2.9) signifie que  $C_{l,m}^{(k,q)}(i, j)$  ne peut être non nulle que si  $C^{(k,q)}(i, j)$  est non nul (en effet,  $C^{(k,q)}(i, j)$  et  $C_{l,m}^{(k,q)}(i, j)$  prennent seulement les valeurs 0 ou 1). La contrainte (2.10) veut que deux chemins traversant le même lien ne peuvent avoir la même longueur d'onde. Enfin, la dernière contrainte exprime explicitement la continuité de longueur d'onde et impose qu'un chemin optique aura la même longueur d'onde sur tous les liens qu'il traverse, cette dernière est analogue à la contrainte de conservation de flot dans le modèle présenté précédemment.

Les auteurs de la dernière formulation ont proposé une seconde ILP, dont les contraintes augmentent moins vite avec l'augmentation du nombre de connexions (donc

plus avantageuse pour la résolution de gros problèmes). Ils ont aussi proposé deux algorithmes de résolution du problème max-RWA basés sur leur ILP. La première fait une relaxation-LP en arrondissant les variables obtenues à la fin de la résolution de la ILP, et s'arrête lorsqu'il n'est plus possible d'arrondir sans violer la contrainte de continuité de longueur d'onde. La deuxième solution fait une relaxation linéaire sans arrondir, de cette manière on obtient forcément un chemin pour chaque demande de connexion dans la matrice de trafic. À partir des routes trouvées, un graphe auxiliaire est construit (*path-graph*) et des heuristiques lui sont appliquées pour essayer de colorier ses sommets, ou faire un  $k$ -coloriage avec  $F$  couleurs.

### 2.3.2 Problème RWA avec conversion

Dans les réseaux à routage de longueurs d'onde WDM [20][21], la contrainte de continuité de longueur d'onde peut être éliminée si on est capable de convertir les signaux optiques incidents sur une longueur d'onde en une autre longueur d'onde en sortie. Les réseaux dotés d'une telle capacité sont appelés réseaux à longueur d'onde convertible. Si la conversion est totale (*Full range conversion*), on est capable de passer de n'importe quelle longueur d'onde à une autre. Un réseau à longueur d'onde convertible avec conversion totale à chaque nœud est équivalent à un réseau à commutation de circuits classique (réseau téléphonique). Dans ce cas, seul le problème de routage doit être résolu. Une formulation en *ILP* est présentée en [1].

Dans la plupart des cas, la conversion totale à chaque nœud du réseau est évitée, vu les coûts élevés des conversions et les limitations du gain en performance. Il est cependant possible de placer quelques convertisseurs dans quelques nœuds de façon stratégique, ou d'utiliser la conversion réduite ou partielle.

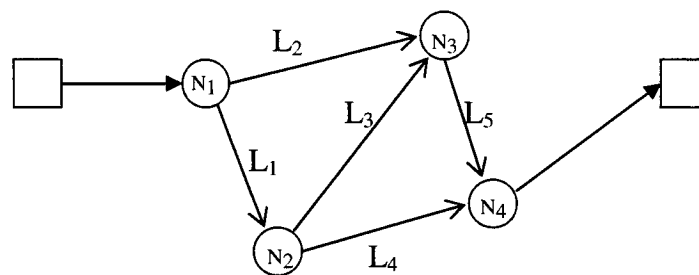
Si nous notons  $w$  le nombre de longueurs d'onde dans le système et  $k$  une constante de conversion, pour des commutateurs à capacité réduite de conversion, une longueur d'onde  $\lambda_i$  en entrée peut seulement être convertie en une longueur d'onde comprise entre  $\lambda_{max(i-k,1)}$  et  $\lambda_{min(i+k,w)}$ . Si  $k=w$ , la conversion est totale ; par contre, si  $k=0$ , il n'y a pas de conversion.

La résolution du problème de routage et affectation de longueur d'onde avec conversion réduite [23] est souvent ramené à un problème de flot dans un graphe en couches (*layered graph approach*). En effet, un graphe auxiliaire est construit tel qu'illustré à la Figure 2.2 :

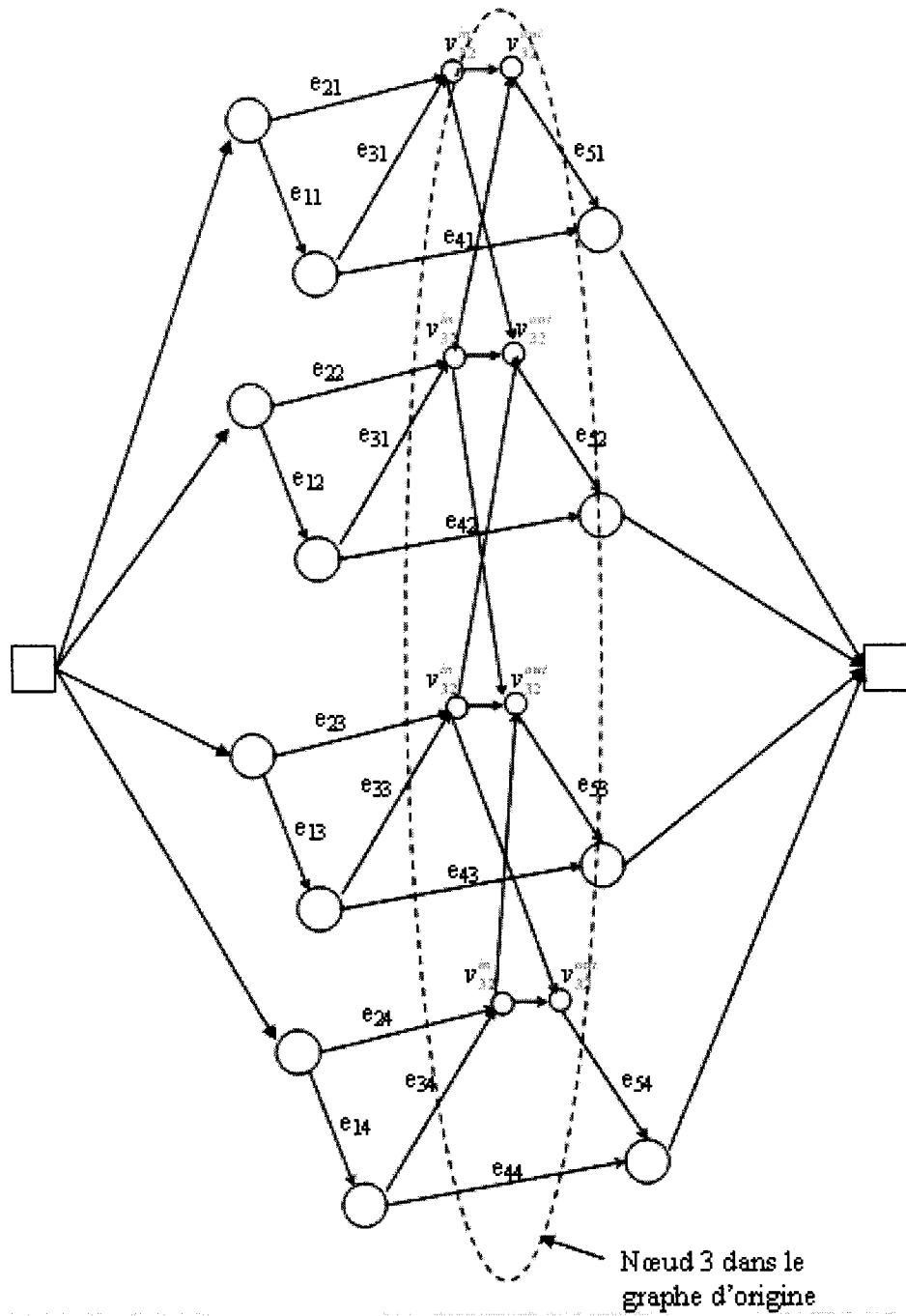
- Le graphe auxiliaire est composé de  $W$  couches, où  $W$  est le nombre de longueurs d'onde. La  $k^{\text{ème}}$  couche correspond à la  $k^{\text{ème}}$  longueur d'onde ( $\lambda_k$ ) ;
- Chaque nœud  $N_i$  est décomposé en  $W$  nœuds en entrée et  $W$  nœuds en sortie. Ces derniers sont notés comme suit :  $v_{i1}^{\text{in}}, v_{i2}^{\text{in}}, \dots, v_{i1}^{\text{out}}, v_{i2}^{\text{out}}, \dots$  ;
- Chaque arc  $L_j$  est décomposé en  $W$  arcs notés  $e_{j1}, e_{j2}, \dots, e_{jw}$  ;
- Les arcs  $w_i^{kl}$  connectent la  $k^{\text{ème}}$  et la  $l^{\text{ème}}$  couche si une conversion est possible au nœud  $N_i$  entre  $\lambda_k$  et  $\lambda_l$ .

### 2.3.3 Routage dans les réseaux optiques WDM

Comme nous l'avons mentionné précédemment, il est généralement plus commode de diviser le problème en deux sous-problèmes : 1) routage et 2) affectation de longueur d'onde. Il existe trois approches de routage citées dans la littérature [1][15] : routage fixe, routage alternatif fixe et routage dynamique.



a) Graphe original



b) Graphe en couches

Figure 2.2 Illustration de la transformation du graphe

### **Routage fixe**

Il s'agit de l'approche la plus simple. Une route fixe est attribuée à chaque paire source-destination. Les routes peuvent être trouvées en appliquant un algorithme de plus court chemin hors ligne (algorithme de Dijkstra ou Bellman-Ford). Ainsi, n'importe quelle connexion entre une paire de nœuds passerait forcément par la route spécifiée. L'inconvénient majeur de cette approche est qu'elle ne tient pas compte de la disponibilité des ressources réseau (longueur d'onde), donc engendre une plus grande probabilité de blocage.

### **Routage alternatif fixe**

Dans cette approche, chaque nœud du réseau est tenu de conserver une liste ordonnée de routes pour chaque destination. L'ensemble des routes doit être disjoint d'arcs. Si une demande de connexion arrive, la source essaie de trouver la première route où une longueur d'onde est disponible.

### **Routage dynamique**

Dans cette approche, le chemin optique de la source à la destination est choisi dynamiquement, dépendamment de l'état du réseau. Des exemples de cette approche seraient le chemin le moins coûteux (*shortest cost path*), ou le chemin le moins congestionné (*LCP : least congested path*).

#### **2.3.4 Affectation de longueurs d'onde**

Dans cette section, nous allons présenter le problème statique d'affectation de longueur d'onde. Étant donné un ensemble de chemins optiques et leurs routes correspondantes, on affecte une longueur d'onde à chaque chemin sans que deux connexions traversant le même arc n'aient la même longueur d'onde. Une approche commune de résolution est de formuler l'affectation de longueurs d'onde comme un problème de coloriage de graphe [1][6][24].

Pour se ramener à un problème de coloriage de graphe, nous devons d'abord construire un graphe auxiliaire,  $G(V, E)$ , tel que chaque chemin optique est modélisé par un nœud et que deux nœuds sont reliés par une arête si leurs chemins respectifs traversent une même fibre physique. Il est à noter que le graphe est non orienté. L'affectation qui minimiserait le nombre de longueurs d'onde sous la contrainte de continuité de cette dernière peut être obtenue en coloriant le graphe  $G$ .

### 2.3.4.1 Coloriage de graphe

#### Définition

Étant donné un graphe non orienté  $G=(V,E)$ , où  $V$  est l'ensemble des sommets et  $E$  l'ensemble des arêtes, le coloriage des sommets revient à affecter une couleur à chacun d'entre eux, tel que deux sommets adjacents n'aient pas la même couleur. Un ensemble  $S$  est dit stable s'il ne comprend aucune paire de sommets adjacents, la taille maximale d'un stable est notée  $\alpha(G)$ . Étant donné un entier  $k$ , un  $k$ -coloriage du graphe revient à le colorer en  $k$  couleurs. Si le graphe peut être coloré par  $k$  couleurs, il est dit  $k$ -chromatique ; sinon, on essaie généralement de maximiser le nombre de sommets colorés notés  $\alpha_k(G)$ . Le nombre chromatique de  $G$  est le nombre minimum de couleurs pour colorer tous les sommets, il est noté  $\chi(G)$ . Le problème de coloration de graphe est démontré NP-Complet et le nombre chromatique d'un graphe est difficile à déterminer. Cependant, on recourt généralement à trouver une borne supérieure et à appliquer des heuristiques pour obtenir des solutions acceptables. Nous présenterons par la suite deux heuristiques rencontrées dans la littérature.

#### Les heuristiques gloutonnes

Pour cette méthode de résolution, les sommets sont coloriés un par un. Il faut noter que les retours arrière ne sont pas permis. Le choix du prochain sommet à colorier dépend du critère glouton. Ce critère peut être statique ou dynamique. Dans le cas statique, un ordre des sommets est établi à l'avance, tandis que pour le cas dynamique, le critère glouton est calculé à chaque itération pour faire le choix du prochain candidat.

- *L'heuristique séquentielle (statique)* : Dans ce cas, les sommets sont classés dans un ordre bien déterminé à l'avance, suivant un critère donné (le degré le plus petit du sommet). À chaque itération, le candidat est colorié avec la plus petite couleur légale.
- *L'heuristique de saturation (dynamique)* : Comme son nom l'indique, elle est basée sur la saturation des sommets. La saturation d'un candidat correspond au nombre de couleurs utilisées par ses voisins immédiats. Le sommet ayant une saturation maximale est choisi à chaque itération.

### **Les méthodes de recherche locale**

Le principe de cette classe de métaheuristiques est très simple. Elle est basée principalement sur la notion de voisinage. Ce type d'algorithme est appliqué aux problèmes d'optimisation avec un ensemble  $S$  de configurations et une fonction de coût  $f$ . On définit aussi une fonction de voisinage et un mécanisme d'exploration de ce dernier. Comme configuration initiale, nous pouvons prendre la solution d'un algorithme séquentiel, pour procéder à l'exploration de l'espace de recherche. À chaque itération, une configuration voisine est choisie en fonction de son coût. Ce processus se poursuit tant qu'on n'a pas atteint un critère d'arrêt fixé à l'avance (nombre maximum d'itérations...). Comme exemples de méthode de recherche locale, citons la recherche taboue et le recuit simulé.

#### **2.3.4.2 Heuristiques d'affectation de longueurs d'onde**

Dans le cas où les demandes de connexions peuvent arriver en tout temps (trafic incrémental ou dynamique), des heuristiques doivent être utilisées pour affecter les longueurs d'onde [1]. Nous présentons ci-après quelques heuristiques proposées dans la littérature. Ces algorithmes doivent opérer en ligne et peuvent être combinés à différents schémas de routage. Le but général est de minimiser la probabilité de blocage pour les nouvelles connexions.

- *Random Wavelength Assignment (R)*: Ce schéma cherche d'abord l'ensemble de longueurs d'onde disponibles tout au long du chemin. Le choix parmi cet ensemble est fait d'une façon aléatoire ;
- *First Fit (FF)* : Dans ce schéma, les longueurs d'onde sont ordonnées, et la recherche se fait dans cet ordre. L'idée ici est d'essayer d'affecter le plus les premières longueurs d'onde de telle sorte que les dernières longueurs d'onde aient plus de probabilité d'être libres ;
- *Least Used (LU)/SPREAD* : Cette méthode affecte la longueur d'onde la moins utilisée dans le réseau. Cet algorithme ne donne pas de bons résultats en simulations ;
- *Most Used (MU)/PACK* : C'est l'opposé du *LU*. Ce schéma affecte la longueur d'onde la plus utilisée dans le réseau. Il a été montré qu'il est le plus performant parmi les schémas proposés ci-haut ;
- *Exhaustive* : Le plus court chemin est retenu après l'examen de toutes les longueurs d'onde. Cette méthode est très coûteuse ;
- *Min Product (MP)* : *MP* est surtout utilisé dans les réseaux multifibres. Dans sa version unifibre, le *MP* est le même que le *FF* ;
- *Least Loaded (LL)* : Comme l'heuristique *MP*, *LL* est surtout utilisée dans les réseaux multifibres. L'idée de cette heuristique est de choisir la longueur d'onde qui a la plus grande capacité résiduelle sur le lien le plus congestionné (longueurs d'onde occupée) du chemin  $p$  ;
- *Max-Sum (MS)* : *MS* est aussi bien applicable pour les réseaux monofibres que multifibres. Elle consiste à énumérer tous les chemins possibles entre la source et la destination et tend à maximiser les capacités résiduelles des liens de la route choisie après l'établissement du chemin optique.

## 2.4 Autres algorithmes et modèles de résolution

Dans cette section, nous présentons quelques modèles et algorithmes de résolution rencontrés dans la littérature, en mettant l'emphase sur les approches qui traitent du modèle statique de trafic.

### 2.4.1 Algorithme H+

Oulai [6][26] présente un modèle appelé RWAP pour *Routing and Wavelength Assignment Problem*. Pour résoudre ce problème, il propose un algorithme basé sur le routage fixe de plus court chemin et une méthode de coloriage séquentielle pour l'affectation de longueurs d'onde. L'objectif de son modèle est de minimiser la quantité de trafic bloqué dans différents scénarios de pannes simples de liens. Le scénario de base est noté  $t_0$  et correspond au cas sans panne. Il considère un schéma de trafic statique et traite deux types de demandes : permanentes et robustes. Le modèle proposé est celui de programmation mixte (certaines variables sont réelles et d'autres sont entières) basé sur les hypothèses suivantes :

- un routage fixe selon un algorithme de plus court chemin ;
- la continuité de longueur d'onde ;
- un réseau bi-connexe et des liens bidirectionnels.

Voici les ensembles utilisés par le modèle :

- $N$ , l'ensemble des nœuds du réseau ;
- $M$ , l'ensemble des liens unidirectionnels du réseau ;
- $K$ , l'ensemble des chemins tels que  $K=K^P \cup K^R$  où  $K^P$  représente l'ensemble des chemins permanents et  $K^R$ , l'ensemble des chemins robustes ;
- $O(k)$  et  $D(k)$ , l'origine et la destination du chemin  $k \in K$  ;
- $\Omega$ , l'ensemble des longueurs d'onde disponibles sur tout le réseau ;
- $\Omega_{ij}$ , l'ensemble des longueurs d'onde disponibles sur le lien  $(i,j) \in M$ , avec  $\Omega_{ij} \in \Omega$  ;

- $T$ , l'ensemble des états du réseau considéré. Le scénario sans faute est noté  $t_0$ . Chaque scénario de panne est caractérisé par une panne simple de liaisons. Il y a donc  $m+1$  scénarios possibles dans le réseau avec  $m = |M|/2$ .

Dans un premier temps, il propose un algorithme appelé  $H$ , trouve les chemins et les longueurs d'onde respectives pour tous les scénarios possibles. Le déroulement de l'algorithme  $H$  est présenté à la Figure 2.3. L'algorithme commence par traiter le scénario sans panne. Pour ce faire, il applique un algorithme de plus court chemin en fonction d'une métrique de routage définie (coût des liens ou nombre de sauts). Par la suite, un chemin est choisi pour chaque trafic pour procéder à l'affectation de longueurs d'onde. L'algorithme  $H$  utilise l'heuristique séquentielle de coloriage de graphe. Le graphe à colorier dépend du trafic à acheminer. Pour chaque scénario de panne, les trafics permanents bloqués ne sont pas ré-acheminés. Seuls les trafics robustes sont reroutés à travers le plus court chemin. Il est à noter qu'un trafic bloqué dans le scénario de base le sera pour tout scénario  $t$ .

À la fin de l'algorithme, la fonction objectif est calculée pour évaluer le nombre de connexions bloquées. Oulai a proposé un critère pour améliorer son algorithme et obtenir l'algorithme  $H+$ . Pour ce faire, il détermine un indice pour estimer le nombre de longueurs d'onde nécessaires pour établir la connexion  $k$ , noté  $n_k$ . Pour trouver cet indice, il considère le trafic  $k$  et tous les trafics avec lesquels  $k$  partage au moins un lien. L'algorithme  $H+$  est présenté à la Figure 2.4.

#### 2.4.2 L'algorithme TabouP

Certains auteurs [25][27] ont proposé une méthode de résolution du problème RWAP basée sur le coloriage de graphe. Leur approche repose sur la formulation proposée par Oulai, et décompose ledit problème en deux : le routage et l'affectation de longueurs d'onde. En effet, l'approche commence par chercher tous les chemins possibles pour router les trafics, et en sélectionne  $P$  au plus. Par la suite, l'algorithme se

déroule en deux étapes, comme pour  $H+$  : d'abord le scénario sans panne est traité, vient ensuite le traitement des scénarios de panne.

**Pour** Scénario sans panne  
 Trouver les plus courts chemins  
 Pour chaque paire  $i, j \in N$ ,  
     Trouver les plus courts chemins entre  $i$  et  $j$  en considérant la métrique de routage  
**Router** les trafics  
     Pour chaque trafic  $k \in K$ ,  
     Router  $k$  selon le plus court chemin trouvé.  
**Affecter** les longueurs d'onde aux trafics  
     Pour chaque trafic  $k \in K$ ,  
     Affecter une longueur d'onde notée  $\lambda(k)$  au chemin considéré en tenant compte de la continuité de longueur d'onde.  
     Si une longueur d'onde est trouvée,  
         Acheminer le trafic en utilisant cette longueur d'onde  
     Sinon  
         S'il existe un autre plus court chemin pour  $k$ ,  
             Essayer ce chemin  
     Sinon Bloquer le trafic pour tous le scénarios t.

**Pour** tous les scénarios de panne  
 Pour tous les trafics acheminés dans le scénario sans panne  
     Pour chaque trafic  $k \in K^P$ ,  
         Si le chemin  $k$  utilise un élément défectueux, il est bloqué  
         Sinon il reste route sur le même chemin qui lui est attribue  
     Pour chaque trafic  $k \in K^R$ ,  
         Si  $k$  n'est pas affecté par la panne,  
             Router sur le même chemin que pour le scénario sans pannes.  
         Sinon Si  $k$  est affecté par la panne,  
             Trouver un plus court chemin de l'origine à la destination.  
**Pour** tous les trafics reroutés  
     Affecter une longueur d'onde notée  $\lambda(k)$  au chemin considéré en tenant compte de la continuité de longueur d'onde et des longueurs d'onde déjà affectées.  
     Si une longueur d'onde est trouvée  
         L'attribuer au trafic  
     Sinon  
         S'il existe un autre plus court chemin  $k$ ,  
             Essayer ce chemin  
     Sinon, bloquer le trafic

**Figure 2.3** Algorithme  $H$

Calculer  $b$ (nombre de connexions bloquées) en effectuant une affectation initiale avec l'algorithme H.

Désactiver  $b/2$  connexions en désactivant à chaque fois la connexion qui a la plus grande valeur de  $n_k$ .

**Répéter**  
     Désactiver un nombre de connexions égal au pas et évaluer la solution avec l'algorithme AFF.

**Jusqu'à** ce que  $b$  connexions soient désactivées.

Retenir le nombre  $n$  de connexions supprimées au départ qui donne la meilleure solution.

Définir un espace de recherche centré autour de  $n$ .

**Pour** tout l'espace de recherche défini  
     Désactiver une connexion à la fois et évaluer la solution avec l'algorithme AFF.  
     Retenir la meilleure solution.

Réactiver tous les trafics supprimés au départ

Évaluer la solution avec l'algorithme AFF.

**Figure 2.4 Algorithme  $H+$**

Une fois les routes sélectionnées pour chaque trafic, un graphe auxiliaire est construit. Ce graphe est composé de groupes contenant chacun les  $P$  (ou moins) chemins choisis pour un trafic  $d$ . L'affectation de longueurs d'onde aux routes revient à colorer ce graphe, avec une contrainte supplémentaire : chaque groupe contient un et un seul sommet coloré. Pour ce faire, elle utilise l'heuristique taboue. Typiquement, une procédure taboue commence avec une configuration initiale  $s$  de l'ensemble des configurations. Ensuite, le processus se déroule itérativement en analysant à chaque fois le voisinage de la configuration courante  $s$  pour passer à une configuration  $s'$ , à travers un mouvement. Pour éviter le piège des extrema locaux, la notion de liste taboue est introduite. Dans l'algorithme proposé, un mouvement consiste à colorer un sommet avec une couleur  $v$  et à décolorer tous ses voisins colorés avec cette dernière. Pour calculer la performance d'un mouvement, on procède d'une manière incrémentale en évaluant le nombre de voisins colorés avec la couleur  $v$ . En somme, les simulations de la dernière approche donnent, plus ou moins, de meilleurs résultats que l'algorithme  $H+$  [27], et ce, dépendamment du paramètre  $P$  de la méthode (le nombre maximum de chemins considéré pour chaque trafic).

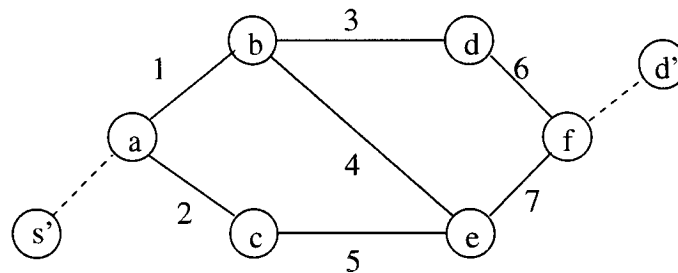
### 2.4.3 L'algorithme WCA

Zhang et al. [10] ont proposé un algorithme qui résout le problème RWA dynamique en utilisant des convertisseurs de longueurs d'onde pour minimiser le taux de blocage dans le réseau. Intitulé WCA (*Wavelength and Converter-Aware*), l'algorithme divise le problème en deux. Il résout d'abord le sous-problème de routage pour passer ensuite à l'affectation de longueurs d'onde. Dans leur approche, un graphe auxiliaire est construit et la clé de l'heuristique proposée est la manière dont on détermine les poids des liens du graphe construit. L'avantage majeur de cet algorithme est que, lors de la résolution du premier sous-problème, la disponibilité des longueurs d'onde et des convertisseurs est prise compte.

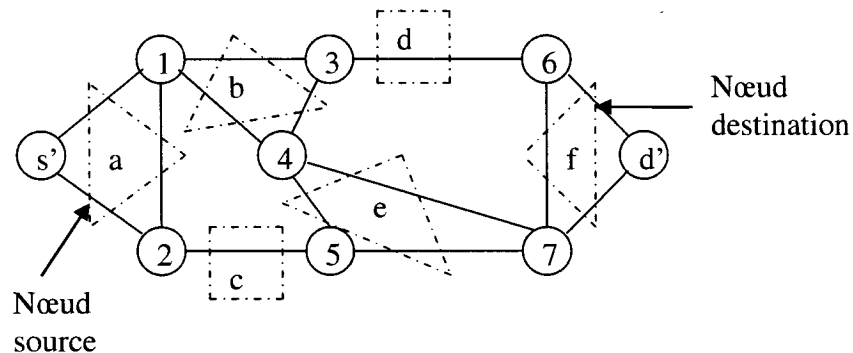
Un réseau à routage de longueurs d'onde est modélisé par un graphe orienté  $G(N,L)$ , où  $N$  et  $L$  représentent les ensembles des sommets et des arcs respectivement. La largeur de bande de chaque fibre est divisée en  $W$  longueurs d'onde et la conversion de longueurs d'onde est supposée totale. On suppose aussi que l'algorithme RWA est décentralisé, c'est-à-dire que les décisions sont prises d'une manière autonome à chaque nœud. Le routage et l'affectation de longueur d'onde sont résolus en utilisant un algorithme de routage de plus court chemin. Dans les deux cas, on recourt à la construction d'un graphe auxiliaire.

#### Algorithme de routage

Pour trouver le meilleur chemin entre les nœuds  $s$  et  $d$ , un graphe auxiliaire (graphe de ligne  $G(L)$  du graphe  $G$ ) est d'abord construit à partir du réseau original (Figure 2.5a). Avant la construction du graphe auxiliaire, deux pseudo nœuds sont ajoutés (voir Figure 2.5b). Comme le montre la Figure 2.5, à chaque nœud du graphe original correspond un ensemble d'arêtes dans le graphe auxiliaire.



a) Graphe d'origine avec pseudo nœuds



b) Graphe auxiliaire

**Figure 2.5 Illustration de la transformation du graphe**

Une fois le graphe auxiliaire construit, la métrique de routage est attribuée (le coût sur les arêtes) à chaque arc en tenant compte du nombre de chemins optiques passant par ce dernier. En effet, cette métrique est basée sur la probabilité de disponibilité d'une longueur d'onde sur l'arête. Après quelques approximations simplificatrices, cette probabilité est évaluée à :  $p_i = 1 - (1 - \frac{w_{out}}{W})^{w_{out}}$ , où  $W$  représente le nombre de longueurs d'onde sur la fibre, et  $w_{out}$  le nombre de longueurs d'onde disponibles à la sortie d'un nœud. La métrique utilisée est donc :  $\rho_i = -\log p_i$ .

Un algorithme de plus court chemin (Dijkstra) est ensuite appliqué entre les pseudo nœuds origine et destination. Il est à noter qu'un chemin trouvé ne doit pas passer par deux arêtes correspondant au même nœud dans le réseau original.

### **Algorithme d'affectation de longueurs d'onde**

Les auteurs de la méthode présentent trois façons pour affecter les longueurs d'onde aux chemins trouvés :

- *FFW (First Fit Wavelength First)* : C'est une extension de l'algorithme FF (*First Fit*) utilisé par Mokhtar et Azizoglu [37] pour résoudre le problème d'affectation de longueurs d'onde. Le nœud source trouve la première longueur d'onde disponible sur le chemin sélectionné. Cela suppose que les longueurs d'onde ont un ordre prédéfini sur les liens. Une fois la longueur d'onde trouvée, le processus se poursuit au nœud suivant dans la route. Il est possible de faire des conversions si nécessaire ;
- *LEC (LEast Converter First)* : Puisque le nombre de convertisseurs dans chaque nœud est souvent inférieurs au nombre de longueurs d'onde, il est nécessaire de tenir compte de ce fait dans l'affectation des longueurs d'onde. Le but de cette approche est donc de déterminer un chemin optique utilisant le plus petit nombre de convertisseurs. L'affectation de longueurs d'onde est traitée comme un problème de routage et utilise l'algorithme de Dijkstra pour trouver une solution convenable ;
- *LCC (Least Conversion Cost First)* : Cette approche est similaire à la précédente sauf que LCC emploie une fonction de coût non linéaire pour l'usage de convertisseurs.

Les résultats de ces algorithmes ont été comparés à ceux des algorithmes HW (*Hop-Based*) et TAW (*Total Wavelength and Available Wavelength*) présentés par Bhide et al [10]. Les simulations montrent que le WCA est plus performant.

#### 2.4.4 Algorithme de chemins disjoints d'arcs

Dans cette approche, un problème très connu dans la théorie des graphes est adapté au problème RWA [5]. L'approche revient sur la construction de chemins disjoints d'arcs pour supporter les connexions ayant la même longueur d'onde. Soit  $G=(V,E)$  le graphe de la topologie physique du réseau. Soit la paire  $(s_i,t_i)$  l'ensemble des paires origine/destination, et  $\tau$  un ensemble de connexions qui doivent être disjoints d'arcs.  $\tau$  est dit réalisable dans  $G$  s'il existe des chemins  $P_i$  mutuellement disjoints d'arcs pour chaque paire origine-destination. Le problème de chemins disjoints d'arcs maximum revient à trouver le sous-ensemble de  $\tau$  réalisable de taille maximale. Ce problème est bien connu comme étant NP-difficile.

Afin de résoudre ce problème, les auteurs [5] ont proposé une approche séquentielle pour trouver des ensembles de chemins disjoints d'arcs dont on leur affecterait une longueur d'onde à chaque itération. En effet, étant donné un graphe  $G$  et un ensemble  $\tau$ , appelons  $\tau_1, \dots, \tau_i$  une partition de  $\tau$  telle que chaque  $\tau_i$  est une solution du maximum CDA (*Chemins Disjoints d'Arcs*). Supposons que, pour la première itération, on obtienne  $\tau_1$  comme solution, on peut donc leur affecter  $\lambda_1$  comme longueur d'onde. Par la suite, une recherche se fait pour les demandes de connexions non encore servies. On répète ce processus jusqu'à ce que tous les chemins aient une longueur d'onde. Mentionnons que le nombre d'itérations représente le nombre minimum de longueurs d'onde utilisées dans le réseau. Le problème revient donc à trouver un moyen de résoudre le maximum CDA. Pour ce faire, il est proposé de fixer une borne supérieure  $d$  du nombre de liens de chaque chemin. On pose :

$$d = \max (\text{diamètre}(G), |E|^{1/2} )$$

L'algorithme ayant comme données le graphe  $G$ , l'ensemble  $\tau$  et la borne d'opération, se décrit comme suit :

- Choisir aléatoirement une connexion ;
- Trouver le plus court chemin ;

- Si la longueur du plus court chemin ne dépasse pas  $d$ , retenir la connexion et le chemin et effacer tous les liens utilisés par le chemin. Si la longueur dépasse  $d$ , la connexion n'est pas routée ;
- Répéter le processus pour chaque connexion ou jusqu'à ce qu'il ne reste plus de liens dans le réseau.

En termes de résultats, l'algorithme est comparé à une méthode basée sur la coloration de graphe et la programmation linéaire. L'algorithme étudié produit les mêmes résultats en termes de nombre de longueurs d'onde mais s'exécute beaucoup plus rapidement.

#### **2.4.5 Algorithme génétique et recuit simulé**

Dans la littérature, plusieurs méthodes basées sur ces deux algorithmes ont été proposées pour résoudre le problème d'affectation de longueurs d'onde [28][29]. Nous allons d'abord présenter une définition des deux approches.

Dans les algorithmes génétiques (AG), des opérateurs génétiques sont appliqués sur les candidats pour en produire des descendants plus performants. On doit d'abord initialiser la population des solutions qui doit être diversifiée selon des critères probabilistes afin d'aboutir à la solution optimale. On doit également avoir une fonction objectif souvent à minimiser pour évaluer les membres de la population.

Pour ce qui est du recuit simulé (RS), c'est une heuristique d'optimisation qui utilise la perturbation pour éviter le piège de l'extremum local, et ce, en acceptant de temps à autre des solutions qui détériorent la fonction objectif avec une probabilité fixée au départ.

Mikac et al. [28] considèrent pour chaque trafic un chemin principal et un chemin de secours qui sont disjoints d'arcs. On respecte la continuité de longueur d'onde. L'objectif est alors de maximiser l'utilisation des fibres. Les solutions possibles sont encodées de la même manière pour le AG et le RS. Dans le AG, cet encodage est appelé chromosome. On considère que chaque lien a 8 longueurs d'onde, il est donc possible de les coder sur 3 bits ou gènes. La longueur du chromosome est obtenue en

multipliant le nombre de bits du chemin par le nombre de connexions à établir. Les auteurs considèrent la même fonction de coût pour évaluer les solutions [28].

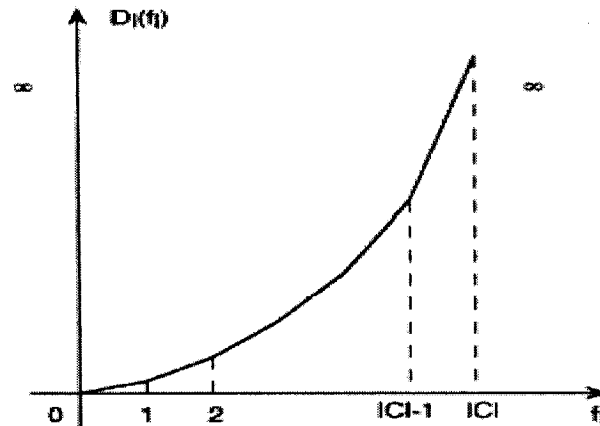
À la fin de leur étude, les auteurs ont comparé les deux algorithmes. Il est apparu qu le RS est beaucoup plus rapide pour une taille relativement petite du problème. Cependant, la qualité des solutions du AG est nettement meilleure.

#### **2.4.6 Approche quasi-statique avec et sans conversion de longueurs d'onde**

Une formulation intéressante du problème de routage et d'affectation de longueur d'onde est proposée par Ozdaglar et Bertsekas [32]. Leur approche est utilisée pour les réseaux sans conversion mais peut être facilement étendue aux cas de conversion partielle. Les auteurs considèrent un modèle de trafic dynamique à variation stochastique. Puisque les décisions sont prises en temps réel, il est important qu'elles tiennent compte des effets des variations des demandes futures et de la disponibilité des ressources. Leur approche repose sur une formulation de flot à multicommodités qui est souvent rencontrée dans la littérature. La fonction objectif est donnée par la somme des coûts des unités de flot sur chaque lien. Cependant, la fonction de coût sur chaque lien est convexe, monotone et croissante (voire Figure 2.6).

La formulation avec conversion partielle se présente comme suit. Soit :

- $W$  : l'ensemble de toutes les paires origine-destination ;
- $P_w$  : l'ensemble des chemins qu'une paire  $w$  de  $W$  peut utiliser ;
- $r_w$  : le nombre de chemin requis entre une paire  $w$  ;
- $C$  : l'ensemble des longueurs d'onde.



**Figure 2.6 Fonction de coût sur un lien  $l$**

Quelques notations :

- Si  $(l_1, l_2) \in p$ , donc les liens  $l_1$  et  $l_2$  se succèdent dans  $p$  ;
- $L_p(l)$  est le premier lien du chemin  $p$ .

Les variables du problème :

$$x_{p,l}^c = \begin{cases} 1 & \text{Si la longueur d'onde } c \text{ est utilisée sur le lien } l \text{ par la connexion } c \\ 0 & \text{Sinon} \end{cases}$$

Le flot total sur un lien  $l$  de  $L$ , est noté  $f_l$  et peut être exprimé en terme de  $x_{p,l}^c$  comme suit :

$$f_l = \sum_{p/l \in P} \sum_{c \in C} x_{p,l}^c$$

La fonction objectif du modèle correspondant est la suivante :

$$\text{Minimiser : } \sum_{l \in L} D_l(f_l) \quad (2.12)$$

À partir de la formulation précédente du problème, le fait de la conversion partielle aux nœuds du réseau peut entraîner l'infaisabilité de la solution étant donné un ensemble de demandes de connexions  $r_w$  où  $w \in W$ . Les auteurs [32] ont proposé donc d'éliminer la contrainte de conservation de flot aux nœuds sans capacité de conversion,

et alternativement d'ajouter un paramètre  $\gamma$  qui détermine le degré de pénalité si la contrainte de continuité de longueur d'onde est violée à un nœud sans capacité de conversion. La fonction objective devient donc :

$$\text{Minimiser : } \sum_{l \in L} D_l(f_l) + \gamma \sum_{c \in C} \sum_{w \in W} \sum_{p \in P_w} \sum_{(l_1, l_2) \in p} |x_{p, l_1}^c - x_{p, l_2}^c| \quad (2.18)$$

← Pour les nœuds  
sans  
conversion

Si une solution existe sans violer la contrainte de continuité de longueur d'onde, la relaxation du problème et la résolution de cette LP va donner la même solution que la formulation antérieure. Sinon, dans cette dernière formulation, on suppose que les nœuds qui physiquement n'ont pas de pouvoir de conversion peuvent en avoir au détriment du coût de pénalité  $\gamma$  (qui doit être suffisamment élevé).

La même approche est proposée par les auteurs [32] pour pallier la contrainte d'unicité de longueur d'onde qui dit que deux chemins traversant le même lien physique ne peuvent avoir la même longueur d'onde. Les détails sur la nouvelle formulation sont présentés dans [32].

# CHAPITRE III

## ALGORITHMES DE ROUTAGE ET D'AFFECTATION DE LONGUEURS D'ONDE PROPOSÉS

Dans ce chapitre, nous présentons une formulation mathématique du problème max-RWA ainsi que les méthodes de résolution proposées. Le problème de routage et d'affectation de longueurs d'onde considéré suppose un modèle statique de trafic avec des demandes robustes et permanentes et respecte la contrainte de continuité de longueurs d'onde imposée par l'absence de convertisseurs. Comme mentionné au précédent chapitre, le problème de routage et d'affectation de longueurs d'onde RWA est classé NP-Difficile [3]. Les méthodes proposées sont donc basées sur l'utilisation d'heuristiques qui permettent d'obtenir des solutions réalisables en des temps de calcul raisonnables. Dans ce chapitre, nous présenterons d'abord une formulation mathématique du problème. Suivra un exposé de la méthode de résolution proposée. Par la suite, nous tenterons de généraliser le problème max-RWA pour les scénarios de pannes, pour finir avec une illustration de l'algorithme proposé et une analyse de sa complexité.

### ***3.1 Formulation mathématique du problème***

Dans cette section, nous présentons un modèle mathématique du problème de routage et d'affectation de longueurs d'onde (RWA) dans le cas d'une demande statique de trafics, c'est-à-dire connue à l'avance. Les opérations reliées à la résolution vont alors s'exécuter hors ligne. Le but du modèle est de minimiser le nombre de longueurs d'onde occupées dans le réseau afin de satisfaire la demande de trafic pour le scénario de base,

c'est-à-dire sans panne, pour une topologie physique donnée [1]. Pour ce scénario, cette approche peut être utilisée pour obtenir le nombre minimal de longueurs d'onde nécessaires afin de satisfaire une certaine demande de connexions. Nous allons par la suite considérer les différents scénarios de pannes simples pour les deux types de trafics, robustes et permanents. En effet, il est important pour un opérateur de réseau de prévoir à l'avance le comportement de son réseau dans les différents cas de pannes pour une charge donnée. Quoique la formulation présentée ne tienne pas compte des différents scénarios de pannes, nous allons supposer sa validité pour ces derniers, et ce pour le nombre de longueurs d'onde trouvé. Une extension du modèle aux scénarios de pannes simples sera présentée par la suite.

Nous présentons un modèle de programmation linéaire en nombres entiers (ILP) [1]. Nous en faisons les hypothèses suivantes :

- continuité de longueurs d'onde ;
- réseau bi-connexe ;
- liens bidirectionnels ;
- réseau dans le scénario de base c'est-à-dire sans pannes ;
- routage selon une méthode dynamique de plus court chemin (En effet, pour chaque demande de trafic entre une source et une destination, nous considérons les  $F$  plus courts chemins disjoints d'arcs trouvés itérativement).

Nous allons maintenant définir les ensembles utilisés pour ce modèle :

- $N$ , l'ensemble des nœuds du réseau ;
- $M$ , l'ensemble des liens unidirectionnels du réseau ;
- $\Omega$ , l'ensemble des longueurs d'onde ;
- $A$ , la matrice de trafics  $k \in K = K^p \cup K^r$  qui peuvent être robustes ou permanents; une paire  $(s,d) \in N^2$  définit l'origine et la destination du trafic,  $A_{sd}$  est le nombre de demandes de trafics entre une source  $s$  et une destination  $d$ .

D'autre part, nous utilisons aussi des variables de décision :

- $x_{ij}^{k\lambda}$ , une variable binaire telle que  $x_{ij}^{k\lambda} = 1$  si et seulement si le trafic  $k$  correspondant à une paire  $(s,d)$  utilise la longueur d'onde  $\lambda$  et passe à travers le lien  $(i,j) \in M$  ;
- $x_{ij}^\lambda$ , une variable binaire telle que  $x_{ij}^\lambda = 1$  si et seulement si la longueur d'onde  $\lambda$  est occupée dans le lien  $(i,j)$ , on a donc :

$$x_{ij}^\lambda = \sum_k x_{ij}^{k\lambda} ;$$

- $e^{k(s,d)}$ , une variable binaire telle que  $e^{k(s,d)} = 1$  si et seulement si le trafic  $k$  correspond à la paire  $(s,d)$  ;
- $c_\lambda^k$ , une variable binaire telle que  $c_\lambda^k = 1$  si et seulement si le trafic  $k$  correspond à une paire  $(s,d)$  et utilise la longueur d'onde  $\lambda$  ;
- $c_\lambda^{(s,d)}$ , une variable entière qui correspond au nombre de trafics de la paire  $(s,d)$  utilisant la longueur d'onde  $\lambda$ , ainsi :

$$c_\lambda^{(s,d)} = \sum_k e^{k(s,d)} c_\lambda^k .$$

Le problème peut être formulé de la manière suivante :

**Minimiser**  $X_{MAX}$  **tel que :**

$$X_{MAX} \geq (x_{ij})_{MAX} = \max_\lambda \left( \sum_{ij} x_{ij}^\lambda \right) \quad \forall (i,j) \in M \quad (3.1)$$

**Sujet à**

*La contrainte de conservation de flot :*

$$\sum_i x_{ij}^{k\lambda} - \sum_k x_{jk}^{k\lambda} = \begin{cases} -c_\lambda^k & \text{si } s = j \\ c_\lambda^k & \text{si } d = j \\ 0 & \text{sin on} \end{cases} \quad (3.2)$$

$$\forall j \in N, \forall \lambda \in \Omega, \forall (s,d) \in N^2$$

*Les contraintes de continuité et d'unicité de longueur d'onde :*

$$\sum_k x_{ij}^{k\lambda} \leq 1 \quad \forall (i, j) \in N^2, \forall \lambda \in \Omega \quad (3.3)$$

$$\sum_\lambda c_\lambda^k \leq 1 \quad \forall k \in K \quad (3.4)$$

*La contrainte d'acheminement de l'ensemble des trafics :*

$$\Lambda_{sd} = \sum_k e^{k(s,d)}, \quad \forall (s, d) \quad (3.5)$$

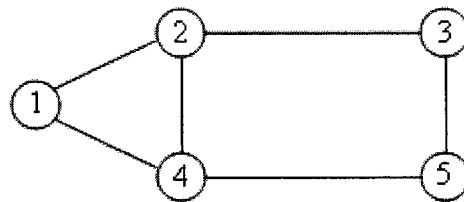
*Les contraintes d'intégrité :*

$$x \in \{0,1\}, \quad c \in \{0,1\}, \quad e \in \{0,1\} \quad (3.6)$$

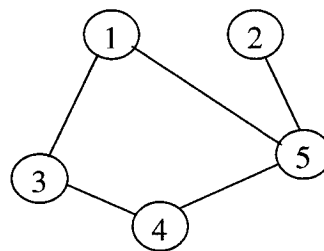
La relation (3.1) minimise la fonction objectif qui exprime le nombre maximal de longueurs d'onde occupées pour satisfaire toute la demande de trafic en minimisant le flot traversant chaque lien du réseau. La solution du problème est donc, la minimisation, sur tous les liens du réseau, du nombre de longueurs d'onde utilisées. Comme mentionné précédemment, cette formulation peut être considérée pour minimiser le nombre de longueurs d'onde utilisées dans tout le réseau et ce en appliquant l'ILP itérativement avec un nombre de longueurs d'onde décroissant tant qu'une solution réalisable est trouvée [1]. La contrainte (3.2) représente la contrainte classique de conservation de flot. Par la suite, les contraintes de continuité de longueurs d'onde, d'acheminement de l'ensemble des trafics et d'intégrité sont explicitées par les relations (3.3), (3.4) et (3.5). En effet, le problème de routage et d'affectation de longueurs d'onde considéré est supposé sans conversion. Ainsi, la relation (3.3) implique que chaque longueur d'onde sur un lien ne peut être utilisée que par un seul trafic traversant ce dernier. Cette condition est très importante du moins dans les réseaux optiques actuels n'utilisant par

de multiplexage temporel sur les canaux de la fibre. La relation (3.4) implique que la longueur d'onde est utilisée sur tout le chemin pour un trafic donné. Pour ce qui est de la contrainte (3.5), elle correspond à acheminer toute la demande, vu que l'objectif du modèle consiste à minimiser le nombre de longueurs d'onde occupées pour satisfaire l'ensemble du trafic. Dans la section qui suit, un exemple explicatif pour le problème max-RWA sera présenté.

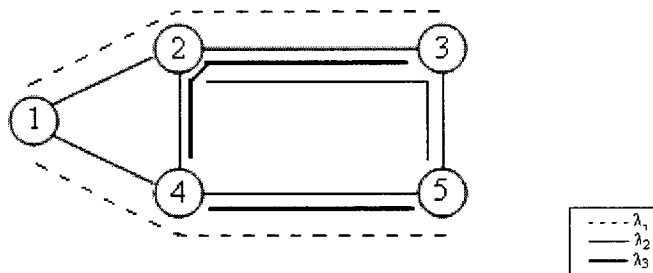
### 3.1.1 Exemple illustratif



a) Topologie physique du réseau



b) Topologie logique du réseau



c) Solution potentielle

**Figure 3.1 Illustration du problème max-RWA**

Le Figure 3.1 est une illustration du problème max-RWA énoncé ci-haut. La topologie physique du réseau est donnée à la Figure 3.1a. La matrice  $A$  de trafic est composée des paires suivantes à acheminer : (1,3), (1,5), (2,5), (3,4) et (4,5) illustrées à la Figure 3.1b. Le but est de trouver un nombre minimum de longueurs d'onde permettant de satisfaire tout le trafic. Une solution potentielle utiliserait trois longueurs d'onde  $\lambda_1$ ,  $\lambda_2$  et  $\lambda_3$  comme montré à la Figure 3.1c.

Comme le montre la Figure 3.1c, une longueur d'onde est attribuée à chaque trafic de telle sorte que, sur un lien quelconque du réseau, à une longueur d'onde n'est affectée qu'un seul trafic traversant ce dernier. Une seule longueur d'onde est également utilisée pour chaque trafic. La résolution du problème max-RWA permettrait donc d'attribuer un chemin à chaque trafic et d'en donner un schéma d'affectation de longueurs d'onde qui minimiserait le nombre de longueurs d'onde nécessaires pour acheminer la totalité de la demande sous les contraintes énoncées précédemment.

### 3.1.2 Considération des scénarios de pannes simples

Comme mentionné précédemment, le modèle présenté ne traite que le scénario de base, c'est-à-dire sans pannes. En effet, il est raisonnable de penser que, si une solution au problème max-RWA offre un schéma de routage qui ne bloque aucun trafic avec un certain nombre de longueurs d'onde  $\lambda$ , alors le nombre de trafics bloqués dans les scénarios de pannes simples serait relativement meilleur, que pour une solution qui bloque déjà un certain nombre de trafics  $n$  pour le même nombre de longueurs d'onde  $\lambda$ , vu qu'une connexion bloquée dans le scénario de base le serait dans les scénarios de pannes. Une illustration est présentée ci-dessous.

**Hypothèse :** Soient deux solutions (schémas de routage)  $A$  et  $C$  pour le problème RWA pour une même topologie physique et une même matrice de trafic, telles que :


$$\left. \begin{array}{l} B_A = 0 \\ B_C = n \end{array} \right\} \text{ avec un même nombre de longueurs d'onde } \lambda, n > 0, \text{ et où } B_X \text{ est le nombre}$$

de trafics bloqués dans le scénario sans panne pour un schéma de routage et d'affectation de longueurs d'onde  $X$ .


Le nombre de trafics bloqués dans l'ensemble des scénarios de pannes simples considérant la solution  $A$  serait inférieur à celui de la solution  $B$  pour un ensemble de trafics pouvant être robustes ou permanents et pour des schémas de recouvrement aux pannes plus ou moins identiques.

**Illustration :** Supposons que  $\bar{x}$  trafics robustes passent, en moyenne, par chaque lien du réseau pour un schéma de routage. Pour tous les scénarios de pannes simples et pour la solution  $A$  :

$$\sum_i B_A^i = 0 + (\lambda - \bar{x})|M| + \theta \bar{x}|M| \quad (3.7)$$



Pour les trafics permanents

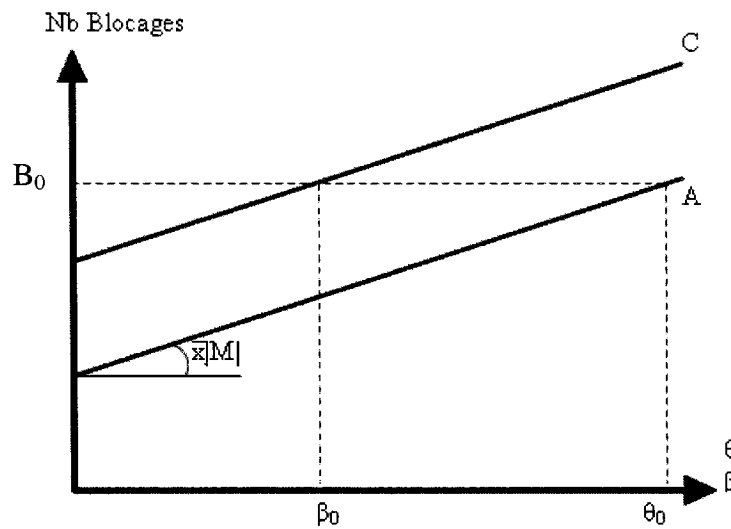


Pour les trafics robustes où seulement une fraction  $\theta$  sera bloquée

De même pour  $C$ , en supposant qu'un trafic bloqué dans le scénario de base le serait pour les scénarios sans panne :

$$\sum_i B_C^i = n \left( \frac{|M|}{2} + 1 \right) + (\lambda - \bar{x})|M| + \beta \bar{x}|M| \quad (3.8)$$

Donc, si tous les trafics sont permanents, le troisième terme disparaîtra et le nombre total de blocages sera strictement inférieur pour la solution  $A$ . Dans le cas où des trafics robustes et permanents sont considérés, le nombre de blocages a une évolution linéaire en fonction des taux moyens  $\theta$  et  $\beta$  de blocage des trafics robustes sur un lien, comme le montre la Figure 3.2



**Figure 3.2 Nombre de blocages pour les deux solutions A et C**

Ainsi, pour un même nombre de blocage  $B_0$ , le taux moyen de blocage de trafics robustes  $\theta_0$  pour la solution A doit être nettement supérieur (dépendamment du nombre de trafics robustes) à  $\beta_0$  le taux moyen de blocage des trafics robustes pour la solution C. Il en résulte donc que nous pouvons considérer que la résolution du modèle proposé, quoiqu'il ne tienne pas compte des cas de panne, optimiserait implicitement le nombre de blocages pour l'ensemble des scénarios avec et sans pannes. En effet, le nombre de connexions atteintes par une panne et pour lesquelles une seconde route légale peut être trouvée serait supérieur pour une solution qui ne bloque pas de trafics dans le scénario de base que pour une qui en bloque déjà.

Le problème max-RWA est classé NP-Difficile [3]. C'est pour cela que l'utilisation des méthodes heuristiques pour la résolution s'avère essentielle afin d'avoir des solutions acceptables (pas nécessairement optimales) dans des temps de calcul raisonnables. À cet égard, la présentation d'une méthode heuristique pour la résolution du problème fera l'objet de la prochaine section.

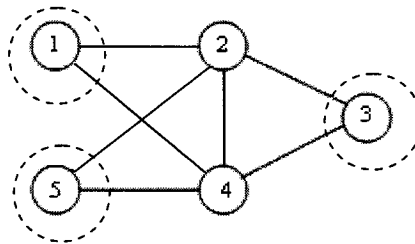
### 3.2 Méthode de résolution

Afin de simplifier le problème, les problèmes de routage et d'affectation de longueurs d'onde seront traités séparément. Ainsi, nous commençons par la sélection de chemins pour chacun des trafics. À partir des solutions de routage trouvées, un schéma d'affectation de longueurs d'onde sera proposé pour satisfaire toutes les demandes dans le scénario sans panne avec un minimum de longueurs d'onde. Une fois tout le trafic acheminé, les cas de pannes simples sont enfin traités. Ainsi, pour chacun des scénarios, les trafics affectés emprunteraient une des routes trouvées dans la première partie.

#### 3.2.1 Considérations théoriques

Soit un graphe non orienté  $G=(V,E)$ , où  $V$  est l'ensemble des sommets et  $E$  l'ensemble des arêtes. Un ensemble stable est défini comme sous-ensemble de sommets  $S$ , tel que deux sommets appartenant à  $S$  ne sont pas adjacents dans le graphe original  $G$ . D'une manière formelle :

$$\begin{aligned} \text{Soient } s_1 \in V \text{ et } s_2 \in V \\ s_1 \in S \text{ et } s_2 \in S \quad \Leftrightarrow \quad (s_1, s_2) \notin E \end{aligned}$$



a) Ensemble stable



b) Sous-graphe induit

**Figure 3.3 Ensemble stable et sous-graphe induit**

La Figure 3.3a illustre un ensemble stable dans un graphe de taille 5 (sommets encerclés). Il est clair que chacun des sommets 1, 3 et 5 ne se touchent pas deux à deux. La construction d'un ensemble stable de cardinalité maximale dans un graphe est un problème NP-Complet classique. Plusieurs méthodes heuristiques ont été développées à cet égard.

D'autre part, soit  $W$  un ensemble de sommets de  $G$ ,  $E(W)$  l'ensemble des arêtes de  $G$  ayant leurs deux extrémités dans  $W$ . Le graphe  $G' = (W, E(W))$  noté aussi  $G[W]$  est appelé sous graphe induit de  $G$ . Ainsi, en supprimant un sommet  $v$  de  $V$ ,  $G-v$  est le sous-graphe induit (c'est-à-dire que  $G-v = G[V - \{v\}]$ ). Dans l'exemple de la Figure 3.3b, le sous-graphe induit en supprimant les sommets 1, 3 et 5 est montré.

### 3.2.2 Description de l'heuristique

Comme mentionné précédemment, nous adoptons une approche de décomposition du problème. En effet, la méthode proposée divise le problème max-RWA en deux sous-problèmes : *le routage* et *l'affectation de longueurs d'onde*. Un schéma de résolution est proposé pour chacun de ces derniers (voir Figure 3.4). Ainsi, dans un premier temps, la sélection des routes est faite. À partir des chemins trouvés, un graphe auxiliaire initial est construit et une heuristique est appliquée pour faire un coloriage complet de ce dernier. L'algorithme de coloriage commence par trouver itérativement des ensembles stables de cardinalités maximales dans les graphes auxiliaires correspondants. Ainsi, pour le premier ensemble stable, les trafics y faisant partie sont enlevés et l'opération se poursuit ainsi de suite jusqu'à desservir l'ensemble de la demande. Il est à noter qu'aux trafics appartenant à un même ensemble stable peut être affectée une même longueur d'onde. Ceci dit qu'à priori, le nombre de longueurs d'onde nécessaires pour acheminer l'ensemble des connexions correspond au nombre d'ensembles stables construits. Le résultat ainsi trouvé ne présente qu'une borne supérieure du nombre de longueurs d'onde minimal et serait raffiné par la suite en faisant appel à une heuristique de coloriage.

Ainsi, l'algorithme pour la sélection des routes construira autant de graphes auxiliaires qu'il a besoin de longueurs d'onde pour acheminer tout le trafic. Ces graphes auxiliaires  $G_l$  sont construits à partir des solutions de routage pour chaque trafic. Pour une paire origine-destination, un ensemble de routes possibles est affecté. C'est seulement à l'étape d'affectation de longueurs d'onde qu'un chemin particulier sera choisi.

Ainsi, à partir de la topologie physique et d'une matrice de trafics donnée, l'algorithme doit :

- chercher un ensemble de plus courts chemins disjoints d'arcs pour chacun des trafics, ce qui correspond au sous-problème de *routage* ;

**Scénario sans panne :**

- Routage : trouver les  $F$  plus courts chemins disjoints d'arcs pour l'ensemble des trafics.
- Sélection des routes et affectation de longueurs d'onde : sélectionner les routes et leurs affecter un nombre minimal de longueurs d'onde.

**Scénarios avec pannes :**

- Bloquer les trafics permanents atteints par la panne.
- Essayer de re-router les trafics robustes atteints à travers le restant des chemins disponibles.

**Figure 3.4 Schéma général de résolution**

- satisfaire l'ensemble des trafics avec un minimum de longueurs d'onde en respectant les contraintes énoncées dans la formulation mathématique, ce qui correspond au sous-problème *d'affectation de longueurs d'onde* ;
- traiter les scénarios de pannes : énumérer tous les scénarios de pannes simples et pour chacun des cas :
  - un trafic non atteint par la panne garde sa route et sa longueur d'onde ;
  - un trafic permanent atteint par la panne est bloqué ;

- un trafic robuste atteint par la panne essaie les autres routes trouvées lors de l'étape de routage, dépendamment de la disponibilité des longueurs d'onde.

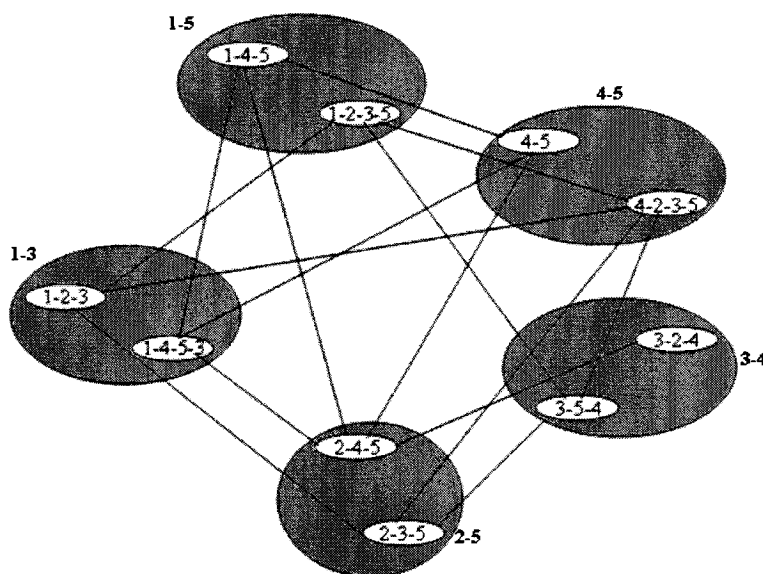
Au début, le scénario sans panne est considéré. La recherche de routes est faite une seule fois, et un ensemble des plus courts chemins disjoints d'arcs est construit pour chaque trafic. À partir des routes trouvées, les graphes auxiliaires sont construits à chaque itération pour en extraire l'ensemble stable de cardinalité maximale. Chaque ensemble stable extrait peut être colorié par une même couleur. Ainsi, le nombre de longueurs d'onde nécessaires pour satisfaire tout le trafic correspond à priori au nombre d'ensembles stables extraits au fur et à mesure de l'avancement de l'algorithme. Puisque, pour un réseau donné, le nombre de plus courts chemins disjoints d'arcs croît (linéairement au pire cas) avec la taille du réseau, seulement les  $F$  plus courts en sont retenus. Ceci permet d'éviter des graphes auxiliaires  $G_l$  de très grande taille. Ainsi, pour chacun des trafics, une classe formée des plus courts chemins disjoints d'arcs est construite. Le graphe auxiliaire est constitué de classes de trafics et est construit de la manière suivante :

- Les nœuds  $V_l$  du graphe constituent l'ensemble des routes pour chaque trafic ;
- Deux nœuds sont reliés par une arête si et seulement si les chemins correspondants ne sont pas disjoints d'arcs, c'est-à-dire qu'ils ont au moins un arc en commun au niveau de la topologie physique. Il est donc évident que deux nœuds appartenant à une même classe ne sont pas voisins vu que, par hypothèse, ils sont disjoints d'arcs ;
- Les classes de trafics contiennent forcément plus d'un sommet vu que les réseaux considérés sont supposés bi-connexes.

Nous proposons une heuristique qui permet de faire la sélection d'un chemin pour chaque trafic à partir des plus courts chemins disjoints d'arcs déjà trouvés et l'affectation d'un nombre minimum de longueurs d'onde pour les satisfaire. Cette méthode consiste à faire un coloriage complet du graphe auxiliaire initial. Pour ce faire, nous procédons en

deux étapes. La première consiste à extraire des ensembles stables de cardinalité maximale et ce, en utilisant une technique de recherche locale avec liste taboue. Cette phase est répétée tant que le graphe auxiliaire est de taille relativement grande (ensemble de sommets de cardinalité supérieure à un certain seuil). Une fois ramenée à un graphe de taille relativement petite, un coloriage complet du graphe auxiliaire résultant est fait en se basant sur une heuristique de recherche locale avec liste taboue. Le nombre de longueurs d'onde nécessaires pour satisfaire l'ensemble des connexions correspond au nombre de couleurs trouvées. Ceci fait, les cas de pannes simples de liens seront traités. Une heuristique constructive sera appliquée pour ré-acheminer les trafics atteints avec le nombre de longueurs d'onde déjà trouvé. Nous appelons cette méthode *Lambda*.

Une fois une route sélectionnée pour chaque trafic, un simple k-coloriage du graphe auxiliaire engendré par les chemins retenus pour chaque connexion est utilisé pour retrouver le nombre maximal de trafics à acheminer avec un certain nombre de longueurs d'onde dans tous les scénarios, à savoir avec et sans pannes simples.



**Figure 3.5 Construction du graphe auxiliaire  $G_l$**

La Figure 3.5 illustre le graphe auxiliaire  $G_l$  construit à partir d'une topologie physique  $G_r$  et une demande de trafics  $D$  définies comme suit :

- $V_p = \{ 1, 2, 3, 4, 5 \}$  ;
- $E_p = \{(1,2), (1,4), (2,3), (2,4), (3,5), (4,5)\}$  ;
- $D = \{(1,3), (1,5), (2,5), (3,4), (4,5)\}$ .

Dans la Figure 3.5, la classe de chemins correspondant à chaque trafic est encerclée par un trait pointillé et contient les sommets effectifs du graphe auxiliaire. La notation est faite en fonction des arcs du réseau, empruntés par chaque chemin.

Ainsi, à partir du graphe de la topologie physique  $G_p$ , nous ressortons les  $F$  plus courts chemins disjoints d'arcs pour chaque trafic. Par exemple, pour le premier trafic (1,3), on associe les chemins (1-2-3) et (1-4-5-3). Ces derniers vont former la classe du trafic (1,3). Afin de décider du chemin à prendre pour chaque trafic, nous recherchons des ensembles stables de cardinalités maximales dans chacun des graphes auxiliaires construits de telle sorte que, pour chaque classe de chemins, seulement une route est considérée. Les chemins appartenant au stable construit seront attribués aux trafics correspondants et ainsi de suite jusqu'à servir tous les trafics. Le nombre de longueurs d'onde nécessaires pour l'ensemble de la demande correspond a priori aux nombres de stables construits. Cependant, l'extraction d'ensembles stables n'étant pas une bonne technique pour une coloration optimale de graphe, une heuristique de coloriage avec recherche taboue sera appliquée par la suite.

### 3.2.3 Sous-problème de routage

L'étape de recherche des routes est faite une seule fois au début de l'algorithme et ce, pour le scénario sans pannes. En effet, à partir de la topologie physique du réseau, le graphe  $G_p$  est construit. Les demandes de trafics peuvent être robustes ou permanentes en fonction de leur tolérance aux pannes. Ainsi, dans le scénario de base, un ensemble de routes disjointes d'arcs est trouvé pour chaque trafic. L'ensemble de ces routes trouvées forme la classe du trafic correspondant.

La méthode de routage utilisée est une variante de l'algorithme de Dijkstra des plus courts chemins. La recherche des routes pour chaque trafic est faite d'une manière itérative. En effet, le plus court chemin est trouvé et les arêtes correspondantes sont ôtées du graphe  $G_p$ . Cette étape est répétée pour chaque trafic jusqu'à ce que l'origine et la destination du trafic appartiennent à des composantes connexes différentes. À chaque trafic est donc associé un ensemble de chemins disjoints d'arcs de cardinalité maximale  $F$ . Avec l'hypothèse de bi-connexité du graphe présentant la topologie physique, au moins deux chemins existent pour chaque trafic. Il est à noter que, pour certaines topologies physiques, le plus court chemin peut ne pas appartenir à l'ensemble des routes disjoints d'arcs. Dans ce cas, il est possible que le fait d'ôter les arêtes du plus court chemin engendre la division du graphe physique en deux ou plusieurs composantes connexes. Dans le cas où l'origine et la destination appartiennent à des composantes connexes différentes, aucun chemin les reliant ne peut être trouvé. Dans une telle situation, la recherche des chemins disjoints d'arcs est faite d'une manière différente. Les détails de cette recherche sont présentés à la Figure 3.6. Le nombre de chemins disjoints d'arc croît (linéairement au pire cas) avec la taille de ce dernier, c'est pour cela que nous n'en considérons que les  $F$  plus courts.

**Rechercher le plus court chemin  $c$  entre la source et la destination avec l'algorithme de Dijkstra**  
**Ôter les arêtes appartenant à  $c$  du graphe de la topologie physique.**  
**Si la source et la destination n'appartiennent plus à une même composante connexe**

- Parcourir tous les nœuds de la nouvelle topologie et mettre le poids des arêtes issues des nœuds de degré un, excepté la source et la destination, à  $M/2$ .**
- Réinsérer les arêtes de  $c$  avec un poids de  $M/2$ .**
- Refaire la recherche des plus courts chemins de la même manière.**
- Retourner au plus les  $F$  plus courts chemins.**

### **Figure 3.6 Cas particulier de routage**

Après l'étape de routage, le premier graphe auxiliaire est construit. Ce dernier est constitué d'une classe de sommets pour chaque trafic. Deux sommets dans le graphe

auxiliaire sont reliés par une arête si et seulement si les routes correspondantes passent par au moins un même arc. Il est donc clair que deux sommets appartenant à une même classe ne sont jamais reliés entre eux.

### 3.2.4 Sélection des routes et obtention du nombre minimal de longueurs d'onde

Une fois le premier graphe auxiliaire  $G_{1,0}$  construit, l'étape de sélection de routes pour chaque trafic est lancée. Nous nous ramenons donc au coloriage du graphe auxiliaire construit. Puisque ce dernier peut être de très grande taille, surtout que plusieurs routes sont trouvées pour chaque connexion, nous allons procéder à ce coloriage d'une manière différente.

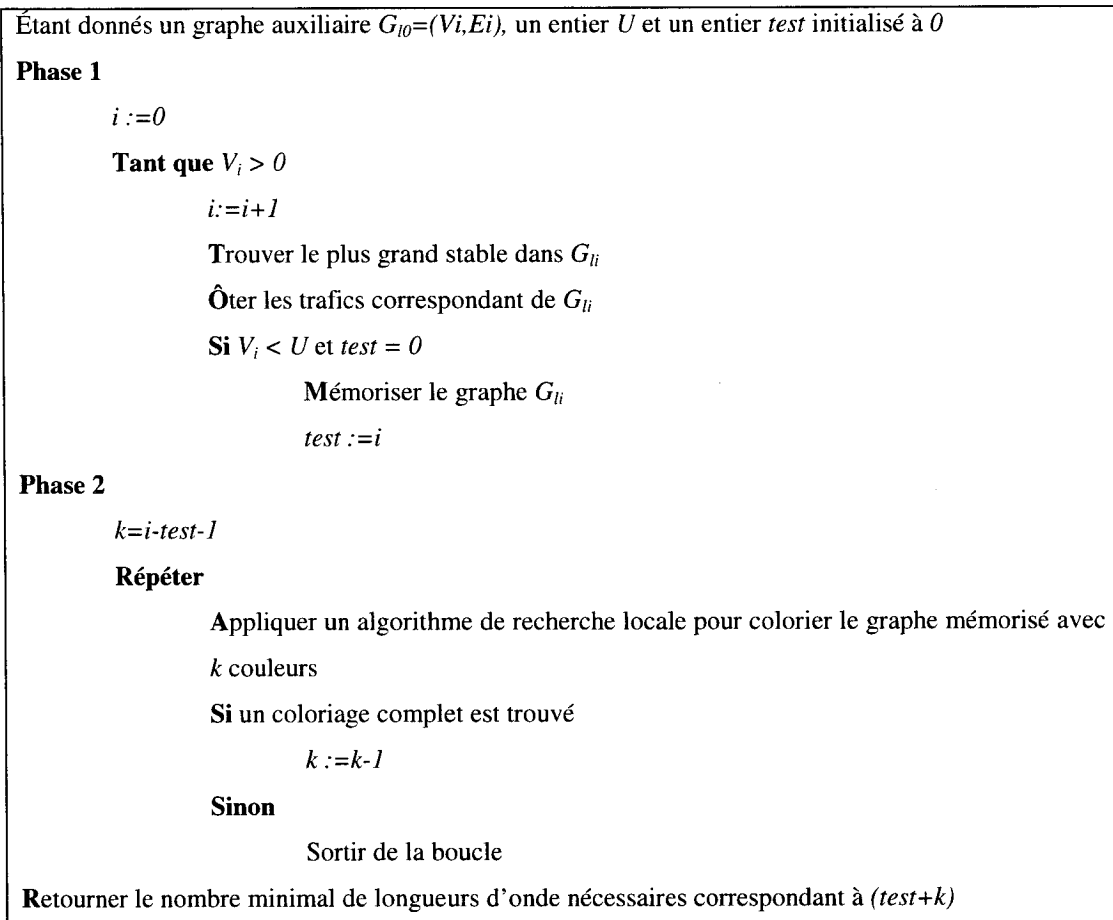


Figure 3.7 Algorithme de recherche du nombre minimal de longueurs d'onde

L'algorithme procède au coloriage du premier graphe auxiliaire en deux étapes. La description des deux phases de l'algorithme et la structure générale de la méthode de coloriage sont montrées à la Figure 3.7.

### Phase 1

L'idée est de choisir un ensemble de routes qui formerait un stable de cardinalité maximale dans le graphe auxiliaire. Cette technique distribuerait à priori les trafics uniformément à travers le réseau. Ainsi, à cet ensemble stable pourrait être attribuée une même longueur d'onde. L'obtention du stable maximum dans le graphe est faite par une heuristique de recherche locale utilisant une liste taboue. Le principe de base est que deux sommets appartenant à cet ensemble ne doivent pas avoir une arête qui les relie. À ce principe, vient s'ajouter une seconde contrainte propre à notre problème. Cette *contrainte d'unicité de route pour chaque trafic* veut qu'un seul sommet d'une classe donnée soit colorié.

Ainsi, pour le premier graphe auxiliaire  $G_{l,0}$ , l'heuristique de recherche locale est appliquée et un premier ensemble de trafics est extrait. Cet ensemble est ôté et un nouveau graphe auxiliaire  $G_{l,1}$  est construit à partir des classes de trafics restant. La même opération est donc répétée jusqu'à atteindre un graphe auxiliaire de taille relativement réduite. À ce niveau, une heuristique de coloriage est appliquée afin de satisfaire l'ensemble des trafics.

Pour trouver les ensembles stables de chacun des graphes auxiliaires  $G_l$ , nous proposons une méthode heuristique de recherche locale. Nous devons donc définir :

- *L'espace des solutions ou des configurations* : c'est l'ensemble  $R$  des partitions légales du graphe auxiliaire  $G_l$ . Ainsi, chaque solution est un ensemble stable du graphe ;
- *La configuration initiale*  $s_0$  : l'algorithme de recherche locale commence avec une solution vide ;
- *La fonction d'évaluation* : nous définissons une fonction d'évaluation  $f$  d'une solution potentielle  $s$  telle que :

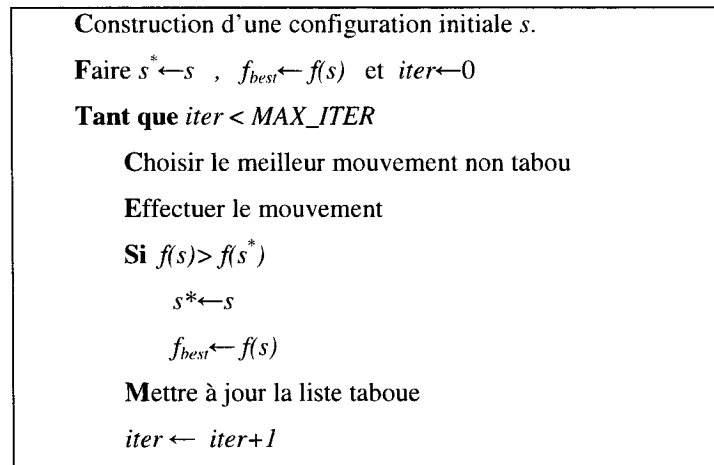
$$f : R \rightarrow N$$

$$s \mapsto |S_s|, \quad \text{où } S_s \text{ est l'ensemble stable trouvée de la solution } s.$$

Une heuristique de recherche locale part généralement d'une solution initiale pour aller explorer l'espace de recherche  $R$ . Cette exploration cherche une meilleure solution dans le voisinage de la configuration en vigueur. Pour passer d'une solution  $s$  à une autre  $s'$  dans le voisinage, un mouvement est appliqué. Une simple technique d'exploration de voisinage se trouve souvent piégée dans des minima locaux de  $R$ . Pour pallier ce problème, la notion de liste taboue est introduite. Cette liste est une mémoire à court terme qui contient les derniers mouvements effectués lors de la recherche. Ainsi, un mouvement est interdit pour un certain nombre d'itérations correspondant à la longueur de cette liste taboue. La structure générale de l'algorithme taboue est présentée à la Figure 3.8.

Pour notre cas, la procédure taboue commence avec une solution vide. Une nouvelle configuration est considérée à chaque itération. Nous proposons d'adopter une stratégie d'expulsion pour passer d'une configuration à une autre. Ainsi, pour la configuration courante (ensemble stable dans le graphe auxiliaire), un mouvement correspond à introduire un sommet  $x$  dans le stable et d'en extraire les sommets voisins déjà présents. Un sommet  $x$  est choisi selon l'évaluation de la configuration engendrée par le mouvement et seul le meilleur est considéré. Il est possible que plusieurs candidats aient le même score. Dans un tel cas, le sommet ayant le plus petit degré est choisi. Si plusieurs sommets sont encore candidats, un choix aléatoire est fait. Il est à noter qu'un sommet est considéré si et seulement si aucun de sa classe ne soit déjà dans le stable. L'évaluation des configurations est faite d'une manière incrémentale et ce, afin de réduire la complexité algorithmique. Lorsqu'un mouvement est effectué, le sommet sélectionné est introduit dans la solution courante et tous ses voisins en sont ôtés. Suite à ce mouvement, le sommet introduit, ainsi que ses voisins enlevés sont rendus tabous pendant un certain nombre d'itérations. Pour ce faire, un tableau noté *taboutenure* est introduit. Cette variable contient l'itération à partir de laquelle un sommet n'est plus tabou et peut donc être candidat. Cependant, un sommet appartenant à la liste taboue

peut être considéré dans certains cas, c'est le *critère d'aspiration*. Dans notre procédure, un sommet qui n'a aucun voisin dans le stable de la configuration courante peut être considéré malgré le fait qu'il soit tabou.



**Figure 3.8 Schéma général de la recherche taboue**

## Phase 2

Le processus de recherche locale taboue est répété tant qu'il existe un trafic non encore desservi. Le nombre de stables extraits correspond a priori au nombre de longueurs d'onde nécessaires pour acheminer tout le trafic. Afin de raffiner la solution trouvée, le processus de sélection de routes peut être fait en deux phases. En effet, à partir d'une certaine taille du graphe auxiliaire, nous pouvons procéder au coloriage du graphe [41].

Pendant la deuxième phase, le coloriage est aussi fait par une méthode de recherche locale utilisant une liste taboue. En effet, avec un nombre donné de couleurs ( $k$  dans notre cas), exactement un sommet dans chaque classe de trafic (ensemble de chemins pour chaque connexion) doit être colorié. L'algorithme explore le voisinage en ayant recours à une stratégie d'expulsion. Les sommets récemment coloriés ou décolorés sont donc mis tabous pour un certain nombre d'itérations [26].

Ainsi, l'algorithme présenté nous renvoie le nombre minimal de longueurs d'onde nécessaires afin d'acheminer l'ensemble des connexions. De plus, le choix des routes pour chaque trafic est effectué implicitement. En effet, avec la nouvelle contrainte énoncée (*contrainte d'unicité de route pour chaque trafic* qui veut qu'un seul sommet d'une classe donnée soit colorié), seule une route sera sélectionnée pour chaque demande de connexion.

Afin d'améliorer le processus de recherche taboue, une technique de diversification est introduite. Ce genre de technique fait généralement appel à des mémoires à long terme. Pour notre procédure, nous choisissons d'implémenter une variable qui contient l'itération à partir de laquelle chaque attribut (sommet-couleur) devient présent dans la solution. Périodiquement, ce tableau est consulté et le plus ancien couple sommet-couleur est ôté de la solution et est omis pour un certain nombre d'itérations.

Une fois les routes sélectionnées pour chaque trafic, nous pouvons procéder à la résolution du problème classique RWAP. Ainsi, l'heuristique taboue est encore une fois déployée pour faire l'affectation des longueurs d'onde. Le problème se ramène donc à faire le  $k$ -coloriage d'un nouveau graphe auxiliaire (construit à partir des chemins retenus pour chaque trafic) avec un certain nombre de couleurs (longueurs d'onde).

### 3.2.5 Affectation de longueurs d'onde

Après avoir sélectionné les routes pour chacun des trafics et trouvé le nombre de longueurs d'onde nécessaires afin d'acheminer l'ensemble de la demande de trafics, nous pouvons nous ramener à la résolution du problème classique RWAP qui consiste à minimiser le nombre de connexions bloquées avec un certain nombre de longueurs d'onde. À cette étape, nous devons construire un nouveau graphe auxiliaire  $G_c$  à partir des chemins choisis pour chaque trafic. Ainsi, chaque sommet du graphe  $G_c$  correspond à un trafic et une arête est présente entre deux sommets si et seulement si les chemins correspondants empruntent au moins un arc commun du réseau original.

Le but étant de maximiser le nombre de trafics acheminés avec un certain nombre de longueurs d'onde  $\lambda$ , nous procédons au coloriage du graphe auxiliaire  $G_c$ . Chaque

couleur correspond à une longueur d'onde du réseau. La recherche de la meilleure coloration partielle est faite par une heuristique taboue. L'espace des solutions est l'ensemble de tous les coloriages légaux du graphe auxiliaire. Une solution potentielle est donc définie par un ensemble de sommets et les couleurs qui leur sont associées. Le schéma général de l'heuristique de recherche taboue est présenté à la Figure 3.8.

La fonction d'évaluation  $f$  indique dans ce cas le nombre de sommets colorés dans la configuration en vigueur. Pour aller explorer le voisinage, nous adoptons encore une fois une stratégie d'expulsion. En effet, un mouvement consiste à colorier un sommet  $x$  du graphe avec une couleur  $v$  et à décolorer tous ses voisins ayant déjà cette couleur. Ce mouvement est donc noté par une paire  $\langle x, v \rangle$ . À chaque itération, le processus tabou explore l'ensemble du voisinage (les combinaisons  $\langle x, v \rangle$ ) afin d'en déduire le meilleur mouvement non tabou à exécuter. Pour évaluer la performance d'un mouvement, nous adoptons une stratégie incrémentale. Nous implémentons alors une matrice  $gm$ , où  $gm(x, v)$  contient le coût associé à colorer le sommet  $x$  par la couleur  $v$ . Cette matrice est mise à jour à chaque itération. Pour ce qui est de l'implémentation de la liste taboue, elle est faite de la même manière que pour l'heuristique de sélection de routes. Une variable *taboutenure* est donc implémentée où *taboutenure*( $x, v$ ) renvoie l'itération à partir de laquelle le mouvement n'est plus tabou. Comme mentionné précédemment, un mouvement tabou peut être effectué dans certains cas (critère d'aspiration). Dans notre procédure taboue, seul un mouvement qui permet un coloriage complet du graphe peut faire l'exception.

Voici les principales étapes de la sélection et de l'exécution d'un mouvement :

- **À chaque itération :**

- **Pour** tout sommet non colorés  $x$

- **Pour** toute couleur  $v \in \{1, \dots, \lambda\}$

Si le mouvement permet une coloration complète du graphe.

**Effectuer** le mouvement

**Retourner** la solution de coloriage

**Si**  $\langle x, v \rangle$  n'est pas tabou

Calculer le coût du mouvement ( $gm(x, v)$ ) et retenir les meilleurs.

- **Si** plusieurs mouvements  $\langle x, v \rangle$  sont trouvés

Choisir un mouvement  $\langle x, v \rangle$  aléatoirement

Effectuer le mouvement

Mettre à jour la liste taboue et la matrice de coûts  $gm$

Il est à noter que la mise à jour de la liste taboue ne concerne pas le couple sommet-couleur  $(x, v)$  en question seulement, mais aussi ses voisins colorés avec cette même couleur. Ainsi, à chaque mouvement, le sommet  $x$  ainsi que ses voisins décolorés sont mis tabous pour la couleur  $v$ .

### 3.2.6 Considération des scénarios de pannes

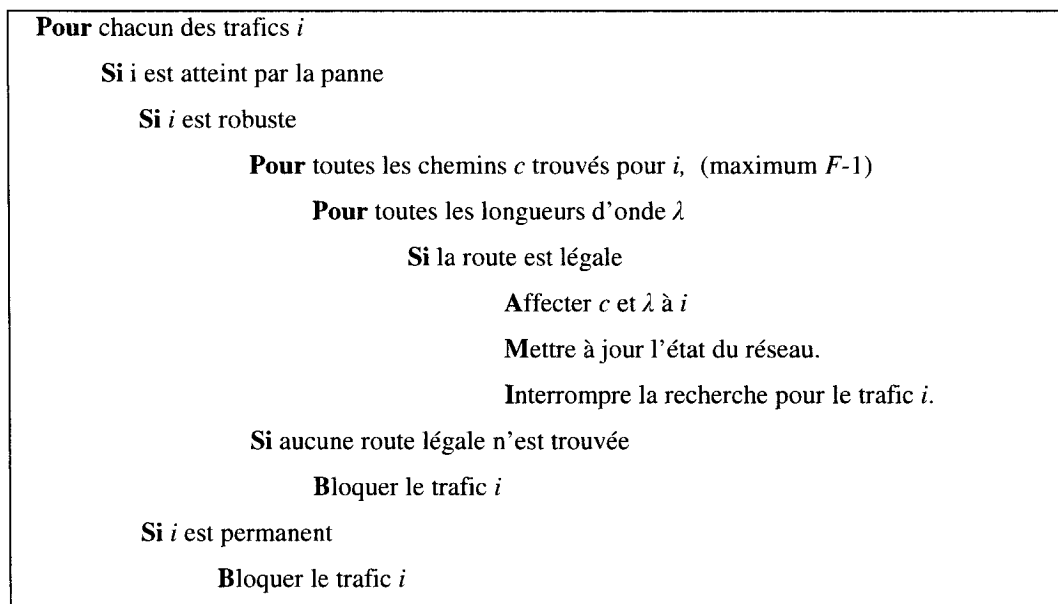
Il est très important pour un opérateur de réseau de prédire le comportement de ses installations lorsqu'un cas de panne survient. Dans notre cas, seules les pannes simples de liens sont considérées. De plus, les trafics peuvent être permanents ou robustes. Ainsi, pour chacun des cas de pannes simples, nous avons :

- les trafics non touchés par la panne gardent leurs routes et leur longueurs d'onde ;
- un trafic permanent atteint par la panne est bloqué ;
- un trafic robuste atteint par la panne essaie les autres chemins qui lui ont été trouvés. Vu que ces derniers sont disjoints d'arcs, on est sûr qu'ils ne passeront pas par le lien problématique. Donc, il ne reste qu'à lui affecter une longueur d'onde.

Comme mentionné précédemment, les scénarios de pannes ne sont pas considérés lors de la recherche des routes et des longueurs d'onde (première partie de l'algorithme). Cependant, les différents cas de pannes seront considérés ultérieurement et une méthode constructive sera appliquée pour essayer d'acheminer les trafics atteints.

Le choix d'une méthode aussi simple pour la considération des scénarios de pannes émane de l'hypothèse énoncée précédemment. En effet, nous nous contentions d'un ordre de grandeur (du nombre de trafics bloqués dans les scénarios de pannes) quelque peu grossier afin de valider l'hypothèse lors de la phase des tests. Il est à noter que, puisque notre objectif est de minimiser le nombre de longueurs d'onde nécessaires pour acheminer toute la demande, aucun trafic ne sera bloqué dans le scénario sans panne.

Lorsque le nombre de longueurs d'onde à disposition ne le permet pas, un trafic bloqué dans le scénario de base le sera pour les différents scénarios de pannes simples.



**Figure 3.9 Traitement des pannes simples**

Si un lien tombe en panne dans le réseau, les trafics atteints seront traités consécutivement. En effet, lors de la phase de routage, un ensemble de chemins disjoints d'arcs pour chaque trafic a été trouvé. Ces derniers sont essayés un par un avec l'ensemble des longueurs d'onde disponibles dans le réseau afin de trouver une route légale pour le trafic en question (un chemin avec une longueur d'onde disponible). Si

une route n'a pu être trouvée, le trafic est bloqué et le prochain est considéré. Le schéma de résolution est présenté à la Figure 3.9.

### 3.3 Généralisation du problème *max\_RWA* pour les scénarios de pannes

Comme mentionné précédemment, le but du modèle proposé est de minimiser le nombre de longueurs d'onde utilisées pour satisfaire toute la demande de trafics dans le scénario de base, à savoir sans panne. Une généralisation possible serait de trouver le nombre minimal de longueurs d'onde utilisées pour établir toutes les connexions dans n'importe quel scénario. Ainsi, la nouvelle ILP peut être de nouveau appliquée itérativement afin de trouver le nombre minimal de longueurs d'onde nécessaires afin de satisfaire toute la demande dans tous les scénarios. Formellement, nous nous ramenons à la formulation suivante :

$$\begin{aligned} & \text{Minimiser } X_{MAX} \quad \text{tel que :} \\ & X_{MAX} \geq (x_{ij})'_{MAX} = \max \left( \sum_{\lambda} x_{ij}^{\lambda} \right)' \quad \forall (i, j) \in M, \forall t \in T \quad (3.9) \end{aligned}$$

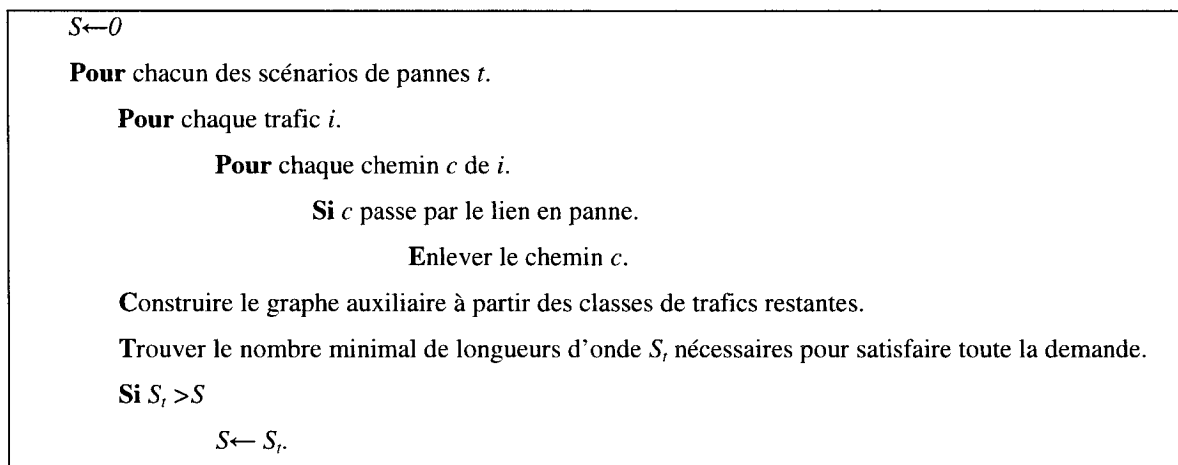
où  $T$  est l'ensemble des scénarios, i.e. avec et sans pannes simples de liens.

La relation (3.9) devient donc la fonction objectif du modèle qui garde l'ensemble des contraintes énoncées précédemment. Cette fonction objectif correspond au plus grand nombre, pour tous les scénarios, des longueurs d'onde occupées afin d'acheminer tout le trafic.

#### 3.3.1 Méthode de résolution proposée

Afin de trouver le nombre minimal de longueurs d'onde nécessaires pour acheminer tout le trafic dans n'importe quel scénario, la procédure précédente est répétée pour chacun des scénarios et la valeur maximale trouvée est retenue. En effet, pendant l'étape de routage, les  $F$  plus courts chemins disjoints d'arcs sont trouvés pour chaque trafic. L'hypothèse de bi-connexité de la topologie physique garantit qu'au moins deux

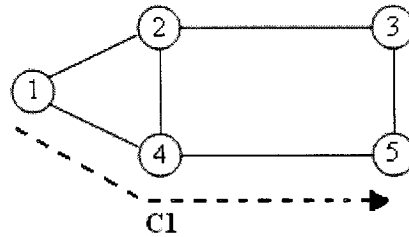
chemins existent entre n'importe quelles sources et destinations. Ainsi, et pour chaque scénario de panne simple de lien, tous les chemins trouvés pour l'ensemble des trafics et traversant le lien défectueux sont omis. Vu que les routes sont disjointes d'arcs, un chemin au plus serait ôté de chacune des classes de trafics. À partir des chemins restant, le premier graphe auxiliaire est construit et la procédure précédente de sélection de routes est répétée jusqu'à atteindre l'ensemble de la demande et du fait obtenir le nombre de longueurs d'onde nécessaires afin d'acheminer tout le trafic dans un scénario donné. Cette procédure est répétée pour chacun des scénarios et le nombre maximal de longueurs d'onde nécessaires est retenu. Le schéma de résolution est présenté à la Figure 3.10.



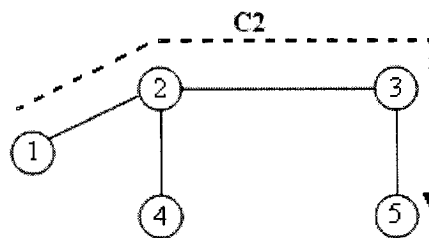
**Figure 3.10 Généralisation du problème max-RWA pour les scénarios de pannes simples**

### 3.4 Illustration de l'algorithme *Lambda*

Afin d'illustrer le fonctionnement de l'algorithme, l'exemple montré à la Figure 3.1 sera considéré. Ainsi, pour les topologies physique et logique données, nous allons illustrer les processus de routage, de sélection de routes et l'affectation des longueurs d'onde de l'algorithme *Lambda*.



a) Sélection du premier chemin C1



b) Sélection du deuxième chemin C2

**Figure 3.11 Illustration de la solution de routage**

Selon la Figure 3.11, le graphe de la topologie physique  $G_p(V_p, E_p)$  et la matrice de trafics  $D$  sont définis comme suit :

- $V_p = \{ 1, 2, 3, 4, 5 \}$  ;
- $E_p = \{(1,2), (1,4), (2,3), (2,4), (3,5), (4,5)\}$  ;
- $D = \{(1,3), (1,5), (2,5), (3,4), (4,5)\}$ .

Nous allons d'abord trouver l'ensemble des chemins disjoints d'arcs pour chacun des trafics. Pour ce faire, l'algorithme de Dijkstra est appliqué itérativement comme suit :

- un premier plus court chemin est trouvé ;
- les arcs empruntés par ce dernier sont ôtés du graphe  $G_p$  et un second chemin est trouvé ;
- de la même manière, cette itération est répétée tant que le graphe  $G_p$  reste connexe ou jusqu'à trouver les  $F$  chemins.

À la Figure 3.11, l'exemple de routage pour le trafic (1,5) est montré.

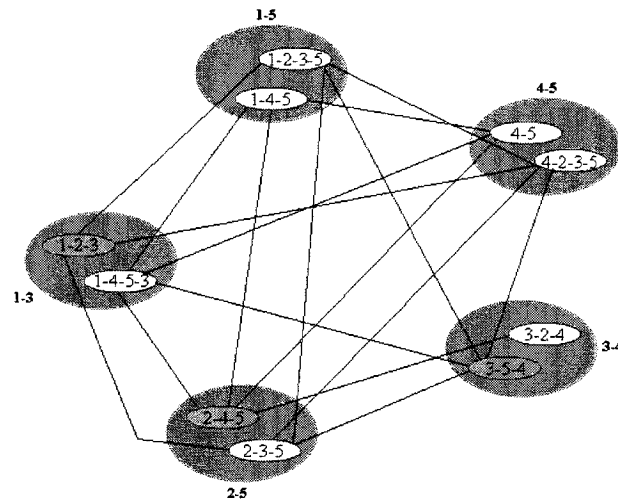
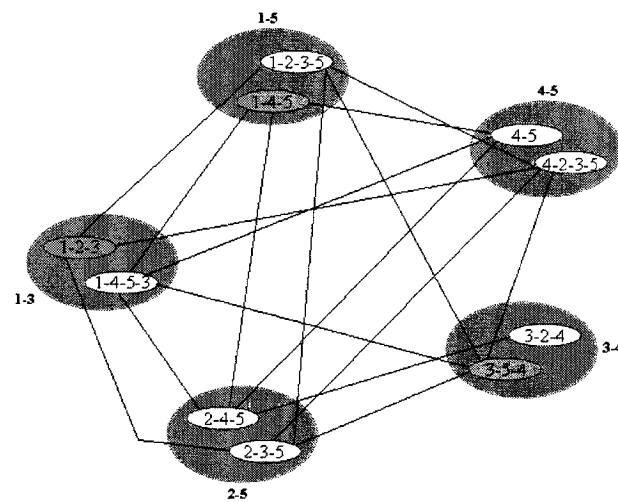
Une fois les chemins trouvés pour chaque trafic, un premier graphe auxiliaire est construit afin d'y extraire le premier ensemble stable. Il est à noter que seulement la première phase de l'algorithme de recherche du nombre minimal de longueurs d'onde nécessaires sera illustrée.

Un exemple de mouvement pendant la recherche locale est montré à la Figure 3.12. Les sommets appartenant au stable de la configuration courante sont coloriés en gris. Nous voyons que l'introduction du sommet (1-4-5) à la  $i^{\text{ème}}$  itération va engendrer l'expulsion du sommet (2-4-5) qui lui est adjacent. Les deux sommets sont par la suite considérés tabous pour un certain nombre d'itérations.

Une fois le plus grand stable construit, les trafics atteints sont enlevés, et un nouveau graphe auxiliaire est construit à partir des classes de trafics restantes.

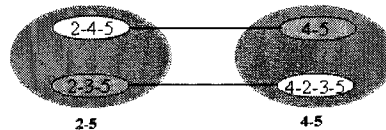
Le nombre d'ensembles stables construits correspond à priori au nombre de longueurs d'onde nécessaires pour acheminer la totalité de la demande. Une fois un chemin retenu pour chacune des connexions, une heuristique taboue est appliquée pour faire le coloriage du graphe engendré par les routes trouvées. Ainsi, avec un certain nombre de longueurs d'onde, un maximum de connexions à établir est recherché en faisant un k-coloriage partiel du nouveau graphe auxiliaire. La Figure 3.14 illustre la construction de ce graphe ainsi qu'une solution potentielle de coloriage avec deux longueurs d'onde.

Comme nous pouvons le voir, le nombre de longueurs d'onde nécessaires pour acheminer la totalité du trafic est de 2. Ceci dit que l'étape de sélection des routes faite par l'extraction des plus grands ensembles stables à partir des graphes auxiliaires nous offre plutôt une borne supérieure du nombre de longueurs d'onde nécessaire qui serait améliorée ultérieurement, si possible, par l'heuristique de coloriage (phase 2 de l'algorithme).

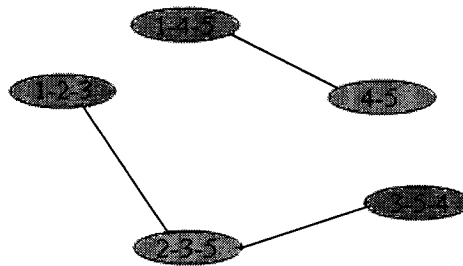
a) Graphe à la  $i^{\text{ème}}$  itérationb) Graphe à  $(i+1)^{\text{ème}}$  l'itération**Figure 3.12 Construction d'un ensemble stable dans le graphe auxiliaire**

Pour trouver le nombre minimal de longueurs d'onde nécessaires pour acheminer toute la demande dans n'importe quel scénario de panne simple, l'algorithme illustré ci-haut est répété pour chacun des scénarios. Pour ce faire, les classes de trafics sont

parcourues et dans chacune, au plus un chemin serait atteint par la panne et sera donc ôté. À la Figure 3.15 est présentée le premier graphe auxiliaire dans le scénario de panne du lien (4,5). Dans ce cas, il est par exemple facile de voir qu'au moins quatre longueurs d'onde sont nécessaires pour acheminer l'ensemble des trafics.



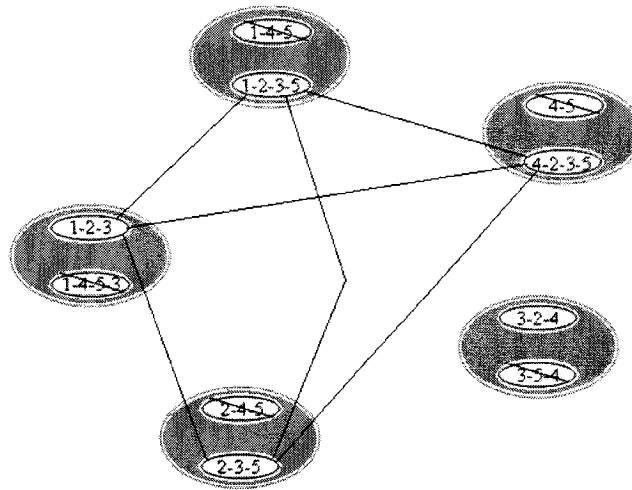
**Figure 3.13 Ensemble stable dans le deuxième graphe auxiliaire**



**Figure 3.14 Illustration de l'affectation des longueurs d'onde**

### **3.5 Analyse de complexité de l'algorithme proposé**

Afin d'évaluer la complexité de la méthode proposée, nous allons procéder itérativement en traitant chacun des sous problèmes séparément. Nous allons donc commencer par la phase de routage, puis la sélection des chemins et l'affectation des longueurs d'onde pour enfin terminer avec le traitement des scénarios de pannes. Nous allons toujours considérer les pires cas.



**Figure 3.15 Graphe auxiliaire dans le scénario de panne du lien (4,5)**

On note :

- $n$  : le nombre de nœuds dans le réseau ;
- $m$  : le nombre de liens dans le réseau, vu que ces derniers sont bidirectionnels,  $|E_p| = 2m$  ;
- $k$  : le nombre de trafics à acheminer ;
- $U$  : la taille du graphe auxiliaire à colorer dans la phase 2 de l'algorithme ;
- $\lambda$  : le nombre de longueurs d'onde nécessaires pour acheminer tout le trafic, au pire cas,  $\lambda = k$ , pour un graphe auxiliaire complet ;
- $\lambda'$  : le nombre de longueurs d'onde disponibles dans le réseau ;
- $M$  : le nombre maximal d'itérations pour l'heuristique taboue ;
- $F$  : le nombre maximal de chemins disjoints d'arcs pour chaque trafic ;
- $T$  : le nombre des scénarios de pannes, ainsi,  $T = m$ .

Dans la phase de routage, l'algorithme commence par chercher les  $F$  plus courts chemins disjoints d'arcs pour chacun des trafics. Vu que nous utilisons l'algorithme de

Dijkstra, la complexité de la recherche de chacun des chemins est  $O(m \ln(n))$ . Ainsi pour l'ensemble des trafics, la recherche des chemins coûte :

$$O(k F m \ln(n))$$

Après l'étape de routage, nous procédons à la sélection des chemins afin d'acheminer tout le trafic dans le scénario de base. Avant de commencer le processus tabou pour l'extraction des ensembles stables, nous devons à chaque fois construire le graphe auxiliaire, qui coûte au pire cas  $O(F^2 k^2 (m-1)^2)$  vu que l'on doit tester tous les  $F \times k$  chemins (qui ont une longueur ne dépassant pas  $m-1$ ) deux à deux pour vérifier leur adjacence. À chaque itération, l'algorithme doit sélectionner un sommet parmi les  $F \times k$  du graphe. Ensuite, la mise à jour de la matrice des coûts est effectuée et sa complexité est de l'ordre de  $O(F^2 k^2)$ . Ainsi, pour l'ensemble des itérations de l'algorithme de recherche locale, ce processus est répété  $M$  fois, ce qui donne une complexité de  $O(M F^2 k^2)$ . En tout, nous obtenons :

$$O(M F^2 k^2) + O(F^2 k^2 (m-1)^2)$$

La recherche d'ensembles stables est faite tant qu'il reste des trafics non encore desservis. Ainsi, l'ensemble du processus tabou sera répété  $\lambda$  fois pour avoir une complexité de :

$$O(k M F^2 k^2) + O(k F^2 k^2 (m-1)^2)$$

À ce stade, la phase 2 de l'algorithme est enclenchée. Cette étape consiste à colorer un graphe auxiliaire de taille  $U$  dans le pire cas. Pour ce faire, nous appliquons une procédure taboue de  $M$  itérations. Avant de commencer, nous devons construire le graphe auxiliaire résultant, qui nous coûte  $O((U/2)^2 (m-1)^2)$ , vu que l'on doit tester tous les  $U/2$  chemins restants au pire cas (car chaque trafic possède au moins 2 routes) pour vérifier leur adjacence. À chaque itération, l'algorithme doit sélectionner un sommet parmi les  $U/2$  du graphe et une couleur parmi  $\lambda$ . Ensuite, la mise à jour de la matrice des coûts est effectuée et sa complexité est de l'ordre de  $O((\lambda U/2)^2)$ . Ainsi, pour l'ensemble

des itérations de l'algorithme de recherche locale, ce processus est répété  $M$  fois, ce qui donne une complexité de  $O(M(\lambda U/2)^2)$ . En tout nous obtenons :

$$O(M(\lambda U/2)^2) + O((U/2)^2(m-1)^2)$$

Nous passons par la suite à l'affectation des longueurs d'onde aux chemins sélectionnés pour chaque trafic. Nous utilisons une heuristique de recherche locale qui exécute  $M$  itérations. Avant de commencer, le graphe auxiliaire est construit et, de la même manière, la complexité est de  $O(k^2(m-1)^2)$  puisque nous avons un seul chemin pour chaque trafic cette fois-ci. À chaque itération, l'algorithme considère l'ensemble des sommets du graphe ainsi que l'ensemble des couleurs disponibles, ce qui donne une complexité de  $O(\lambda' k)$ . Pour mettre à jour la matrice des coûts, on a une complexité de  $O(k^2)$  et donc pour l'ensemble du coloriage :

$$O(M(\lambda' k + k^2)) + O(k^2(m-1)^2)$$

Pour traiter les scénarios de pannes simples, nous adoptons une approche constructive qui vérifie, pour chaque trafic atteint, l'ensemble de ses  $F-1$  routes et l'ensemble des longueurs d'onde  $\lambda'$  disponibles dans le réseau. Cette vérification est faite en comparant les chemins arc par arc, ce qui donne une complexité de :

$$O(Tk^2 \lambda'(m-1)^2)$$

Ainsi, la complexité algorithmique de notre méthode est la somme des complexités des étapes de routage, sélection des routes, affectation des longueurs d'onde et traitement des scénarios de pannes. Cependant, si  $\lambda \leq \lambda'$ , l'étape d'affectation de longueurs d'onde n'est plus considérée. Dans ce cas, la complexité de  $\Lambda$  est de :

$$O(k F m \ln(n)) + O(k M F^2 k^2) + O(k F^2 k^2(m-1)^2) + O(T k^2 \lambda' (m-1)^2) + O(M(\lambda U/2)^2) + O((U/2)^2(m-1)^2)$$

# CHAPITRE IV

## IMPLÉMENTATION ET

### ANALYSE DES RÉSULTATS

Suite à la définition du modèle et à la présentation des algorithmes de résolution, nous allons passer à l'étape d'évaluation de performance. Pour ce faire, dans ce chapitre, nous allons implémenter l'algorithme *Lambda* présenté au chapitre précédent. Dans un premier temps, nous allons donner les détails d'implémentation, à savoir les structures de données utilisées et l'environnement de programmation. Par la suite, une présentation des jeux de données utilisés dans les simulations sera faite. Ces données représentent aussi bien les topologies des réseaux que les demandes de trafics utilisés. Enfin, une description du plan d'expérience sera faite pour passer ainsi à la présentation et à l'analyse des résultats obtenus.

#### ***4.1 Détails d'implémentation***

Dans cette section, nous nous intéressons aux choix d'implémentation faits afin de mener à terme notre travail et conduire les séries de tests désirées. Nous décrivons, dans un premier temps, le choix d'environnement matériel et logiciel utilisé. Ensuite, nous présentons les structures de données adoptées avant d'aborder les détails techniques de l'algorithme lui-même.

Les algorithmes proposés ont été implémentés en C++ avec le logiciel Visual C++ 6.0. Le modèle de programmation orientée objets a été adopté afin d'assurer une meilleure modularité du code surtout à des fins de réutilisations futures. Pour ce qui est

du matériel, nous avons utilisé une machine dotée d'un processeur Pentium 4 à 2.4 GHz et avec une mémoire vive de 512 Mo.

Ainsi, nous avons implémenté deux classes, *Graphe* et *Reseau*. La première implante la topologie physique du réseau et les différents traitements s'y rapportant (routage, PCCs...). Quant à la deuxième, elle implante plutôt la topologie logique du réseau. Ainsi, c'est dans cette dernière que le processus de sélection de routes et d'affectation de longueurs d'onde sera fait.

De plus, notre programme prend en entrée deux fichiers, l'un contenant la topologie physique du réseau et l'autre les demandes de connexions. Les nomenclatures de ces fichiers sont présentées à la Figure 4.1.

Origine	Destination	Quantité	Permanence
Origine	Destination	Quantité	Permanence
...			

a) Fichier des trafics

Origine	Destination	Coût du lien
Origine	Destination	Coût du lien
...		

b) Fichier présentant la topologie physique

**Figure 4.1 Nomenclature des fichiers en entrée**

### **Classe *Graphe***

Cette classe implémente des méthodes liées principalement au sous-problème de routage. En effet, la lecture du fichier d'entrée contenant la topologie physique du réseau ainsi que le calcul des plus courts chemins disjoints d'arcs entre les paires source-destination sont implantés à ce niveau. De plus, on y trouve aussi la fonction qui affecte les trafics aux liens du réseau ainsi que celle qui vérifie les adjacences entre les routes afin de construire le graphe auxiliaire. Voici les différentes méthodes de cette classe :

- *Lire\_graphe* : cette méthode prend en paramètre le nom du fichier contenant la topologie physique et permet de parcourir ce dernier afin de construire le graphe associé. Il est à noter que tous les liens du réseau ont un coût unitaire ;
- *Calcul\_pccs* : cette fonction prend en paramètre la source et la destination du trafic et trouve les  $F$  plus courts chemins disjoints d'arcs entre ces dernières. Il est à noter que, puisque les coûts des liens sont unitaires, le calcul des plus courts chemins est fait en fonction du nombre de sauts. Cette méthode fait appel à la fonction *SommetMinimum* qui retourne le sommet ayant la distance minimale de l'origine lors du déroulement de l'algorithme de *Dijkstra* ;
- *Trouver\_pccs* : cette fonction est récursive. Elle affecte les liens aux trafics à partir du tableau des précédences trouvé par l'algorithme des plus courts chemins ;
- *Communs* : cette méthode prend en paramètre deux chemins et vérifie s'ils sont adjacents ou non.

### **Classe *Reseau***

Cette classe implémente des méthodes propres à la résolution des sous-problèmes de sélection de routes et d'affectation de longueurs d'onde. Ainsi, c'est à ce niveau que les fonctions de coloriage, k-coloriage et gestion de pannes sont implantées. Voici les méthodes les plus importantes de cette classe :

- *Attribuer\_trafic* : cette méthode affecte à chaque connexion les plus courts chemins retrouvés à l'étape de routage ;
- *Generer\_graphe* : cette méthode permet de construire les graphes auxiliaires à partir des  $F$  plus courts chemins disjoints d'arcs déjà trouvés. Il est à noter que le graphe auxiliaire est construit de classes de trafics et que, dans une même classe, les sommets ne sont pas adjacents ;
- *Trouver\_stable* : cette méthode implémente la première phase de la stratégie de coloriage adoptée. En effet, elle contient l'algorithme de recherche taboue pour retrouver les ensembles stables de cardinalité maximale ;

- *Verifier* : cette méthode n'est implémentée que pour des fins de vérifications. Ainsi, à chaque fois qu'un stable est construit, cette fonction est appelée et elle vérifie les sommets de cet ensemble deux à deux ;
- *Colorer\_reste* : cette méthode implémente la deuxième phase de la stratégie de coloriage adoptée. En effet, elle contient l'algorithme de recherche taboue pour faire le coloriage complet du graphe auxiliaire généré après en avoir extrait un certain nombre d'ensembles stables ;
- *Colorer* : cette méthode traite le problème classique de routage et d'affectation de longueurs d'onde. Ainsi, elle prend en paramètre un entier correspondant au nombre de couleurs, et applique une procédure taboue afin de faire un k-coloriage du graphe auxiliaire construit à partir des chemins sélectionnés pour chaque trafic ;
- *Panne* : cette fonction traite les scénarios de pannes. Ainsi, pour le nombre de longueurs d'onde trouvé, les trafics affectés dans chacun des scénarios de pannes simples sont traités itérativement ;
- *Min\_lambda* : cette méthode permet d'implémenter la généralisation de l'algorithme *Lambda* pour les scénarios de pannes. Ainsi, pour chaque scénario de panne simple, tout le processus de coloriage est répété afin de trouver le nombre de longueurs d'onde nécessaires pour acheminer tous les trafics. La fonction retourne le nombre maximal de longueurs d'onde trouvé pour tous les scénarios.

## **4.2 Jeux de tests utilisés**

Comme nous l'avons mentionné précédemment, notre programme prend en entrée deux fichiers texte contenant respectivement la topologie physique et les demandes de connexions.

### 4.2.1 Les Fichiers de topologie

Comme topologies physiques, nous avons considéré des réseaux de tailles différentes. Le Tableau 4.1 montre les topologies physiques considérées en indiquant le nombre de liens et de nœuds pour chacune d'entre elles.

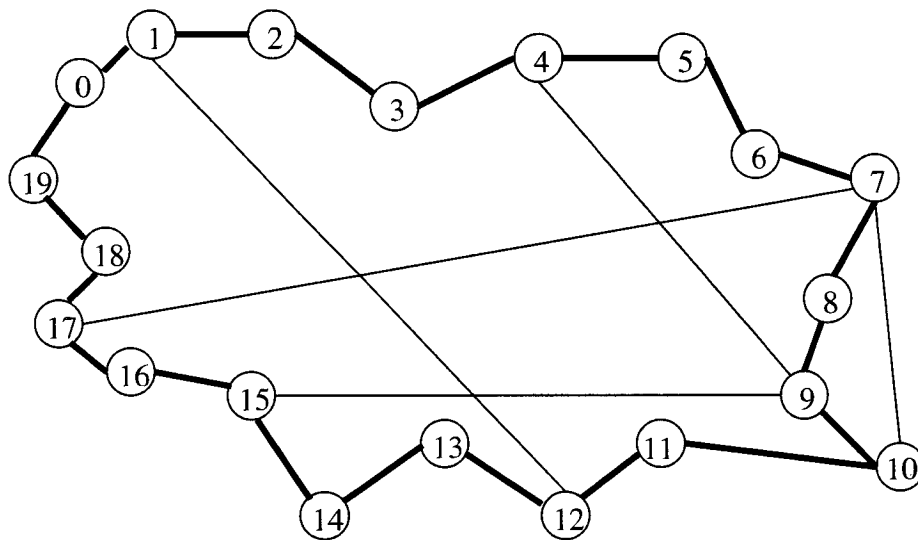
**Tableau 4.1 Topologies physiques considérées**

	Nombre des nœuds	Nombre des liens
<b>Aléatoire 10 nœuds</b>	10	30
<b>Aléatoire 20 nœuds</b>	20	25
<b>NSFNET</b>	14	31
<b>ARPANET</b>	20	31
<b>Aléatoire 30 nœuds</b>	30	74

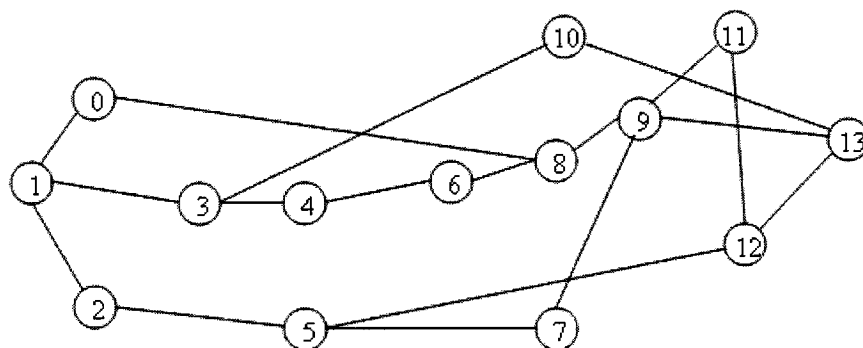
Pour ce qui est des réseaux aléatoires, ce sont des exemplaires générés aléatoirement et proposés par Oulai [26] pour présenter ses résultats. Cette construction est faite à partir d'un programme qui requiert comme variables d'entrées le nombre de nœuds  $n$ , le nombre de liens  $L$ , qui doit être supérieur au nombre de nœuds, et les bornes inférieures et supérieures du degré d'adjacence des nœuds du réseau. Pour assurer la bi-connexité des réseaux, le programme construit itérativement au début un cycle hamiltonien. À cette étape, le réseau contient déjà  $n$  liens. Le reste des liens, soit  $L-n$ , est construit aléatoirement en respectant les contraintes de degré d'adjacence. La topologie du réseau aléatoire à 20 nœuds est montrée à la Figure 4.2; le cycle hamiltonien, construit au début, y est en gras.

Il est à noter que les coûts des liens dans les réseaux considérés sont unitaires. C'est ainsi que la métrique considérée lors de la recherche des plus courts chemins est le nombre de sauts (*hops*).

Les deux autres réseaux considérés sont ARPANET et NSFNET. Pour ce qui est de la topologie ARPANET, nous avons utilisé celle à 20 nœuds et 38 liens. Nous avons choisi ces deux réseaux afin de comparer nos résultats avec ceux de Dzungang [26]. Il est à noter que lesdites topologies sont très utilisées dans la littérature. La topologie du réseau NSFNET est présentée à la Figure 4.3.



**Figure 4.2 Topologie du réseau aléatoire à 20 nœuds**



**Figure 4.3 Topologie du réseau NSFNET**

#### 4.2.2 Les fichiers de trafic

Tout comme les topologies des réseaux aléatoires, les fichiers de trafic ont été générés par un programme en fonction du nombre de nœuds du réseau. Les différents fichiers de trafics utilisés pour chaque topologie sont présentés dans le Tableau 4.2.

**Tableau 4.2 Trafics considérés pour chaque topologie physique**

	trafics robustes	trafics permanents	total
<b>Aléatoire 10 nœuds</b>	42	0	42
	0	42	42
	20	22	42
<b>Aléatoire 20 nœuds</b>	152	0	152
	0	152	152
	68	70	152
<b>ARPANET</b>	152	0	152
	0	152	152
	68	70	152
<b>NSFNET</b>	131	0	131
	0	131	131
	70	61	131
<b>Aléatoire 30 nœuds</b>	329	0	329
	0	329	329
	169	160	329

Afin de trouver le nombre minimal de longueurs d'onde nécessaires pour acheminer l'ensemble de la demande dans tous les scénarios de pannes, nous avons considéré que toutes les demandes de connexions étaient robustes. Cependant, pour trouver le nombre de connexions bloquées dans les scénarios de pannes simples avec un certain nombre de longueurs d'onde, différents schémas de trafics seront considérés. En effet, afin d'appuyer l'hypothèse énoncée au Chapitre 3 selon laquelle l'impact en terme de blocages d'une panne sur un schéma de routage qui satisfait l'ensemble des connexions dans le scénario de base serait moins important que sur un schéma qui en bloque déjà, l'algorithme constructif présenté pour traiter les scénarios de pannes sera évalué pour différents trafics. Pour ce faire, nous avons considéré trois types ou modèles de trafics pour chaque topologie. Les deux premiers modèles consistent à considérer toutes les connexions robustes ou permanentes. Pour ce qui est du troisième modèle,

certaines connexions sont robustes et d'autres permanentes comme le montre le Tableau 4.2.

Il est à noter que les fichiers de trafics des première, deuxième et cinquième topologies ont été générés aléatoirement. Afin de les obtenir, une probabilité d'avoir une demande permanente et une demande robuste est fixée pour chaque paire source-destination. Pour notre cas, nous prenons des fichiers tels que  $pP=pR=0.2$ .

### **4.3 Évaluation de performance**

Après avoir donné les détails d'implémentation et les données utilisées, nous allons passer à l'évaluation de performance des algorithmes proposés. Nous allons d'abord commencer par calibrer les paramètres de notre algorithme pour enfin présenter une analyse comparative des résultats obtenus.

#### **4.3.1 Calibration des paramètres de l'algorithme**

Afin d'évaluer les performances des algorithmes proposés, nous allons d'abord identifier les paramètres pertinents susceptibles d'avoir un impact sur le comportement des heuristiques ainsi que sur les résultats des simulations. Nous allons d'abord faire des tests préliminaires afin de faire une bonne calibration de l'algorithme en variant les niveaux des paramètres considérés. Nous nous concentrerons surtout sur l'algorithme *Lambda* qui cherche le nombre minimal de longueurs d'onde nécessaires pour satisfaire l'ensemble des connexions. Comme mentionné au Chapitre 3, ce dernier est divisé en deux phases, une pour la recherche d'ensembles stables de cardinalité maximale et l'autre pour le coloriage complet du graphe auxiliaire résultant. Puisque l'algorithme utilise des techniques de recherche locale avec liste taboue, les paramètres variables qui pourraient influencer cette recherche sont :

- La taille du graphe auxiliaire à partir de laquelle la phase 2 est amorcée : comme montré à la Figure 3.7, l'algorithme prend en entrée une variable  $U$  correspondant à la taille maximale du graphe auxiliaire sur lequel le coloriage

sera appliqué. Pour notre cas, cette valeur sera fixée à 100 et c'est d'ailleurs très proche de celle suggérée dans la littérature pour la méthode de coloriage adoptée [41] ;

- La longueur de la liste taboue pour la recherche d'ensembles stables : comme la taille des graphes auxiliaires diminue au fur et à mesure de l'avancement de l'algorithme, nous devons adapter la longueur de la liste taboue. Pour ce faire, nous avons considéré une longueur dynamique en fonction de la taille du graphe auxiliaire ;
- La longueur de la liste taboue pour le coloriage ;
- Le nombre maximal  $F$  de routes considérées pour chaque connexion : dans notre cas, nous avons fixé ce dernier à 5. Nous constatons que pour tous les exemplaires de problèmes traités, l'algorithme de routage n'arrive pas à les trouver et se comporte donc comme si  $F = \infty$  ;
- Le nombre d'itérations pour la recherche des ensembles stables : pour faire la recherche d'ensembles stables, nous nous contentons d'un nombre restreint d'itérations. Il est à noter qu'avec un nombre très petit d'itérations, la recherche locale se comporte comme une méthode constructive ;
- Le nombre d'itérations maximal au cours du coloriage.

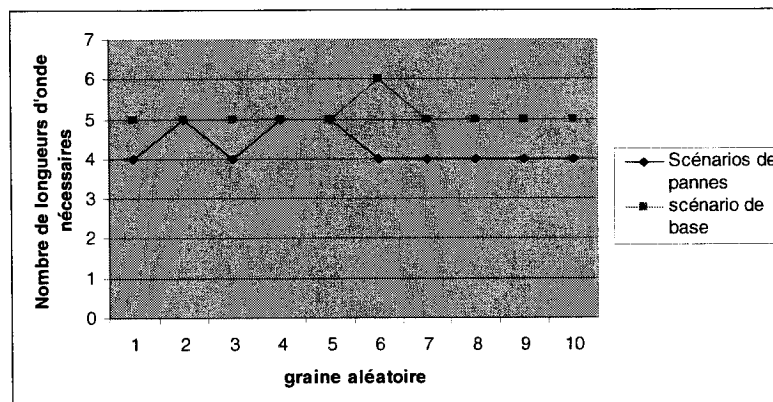
Après avoir défini les paramètres les plus importants, nous devons définir leurs valeurs respectives à considérer. Il est à noter que le nombre d'itérations a une influence directe sur le temps d'exécution et que la taille de la liste taboue influe plutôt sur la qualité des solutions, à savoir sur la robustesse de la recherche locale.

Pour ce faire, nous avons essayé différentes valeurs de la longueur de la liste taboue ainsi que le nombre d'itérations, tout en tenant compte du temps d'exécution du programme ainsi que de la qualité des solutions obtenues. Nous avons choisi un nombre d'itérations égal à 10000 et une liste taboue de longueur 20 pour l'heuristique de coloriage.

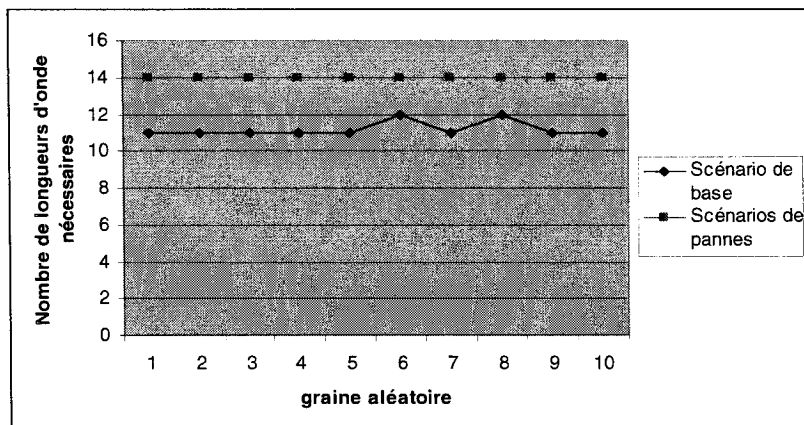
### 4.3.2 Comportement aléatoire de l'algorithme

Aux cours des deux phases de l'algorithme, plusieurs choix aléatoires sont effectués. En effet, à l'étape de recherche des stables, l'algorithme cherche à chaque itération le meilleur sommet à introduire dans la configuration courante. Il est possible que plusieurs candidats aient le même score. Dans un tel cas, l'algorithme privilégie le candidat ayant le degré inférieur. Si plusieurs sommets se trouvent ex-æquo, un choix aléatoire est fait. De la même manière, lors de la phase de coloriage, le meilleur mouvement est considéré à chaque itération. Lorsque plusieurs candidats ont le même score, un choix aléatoire est fait.

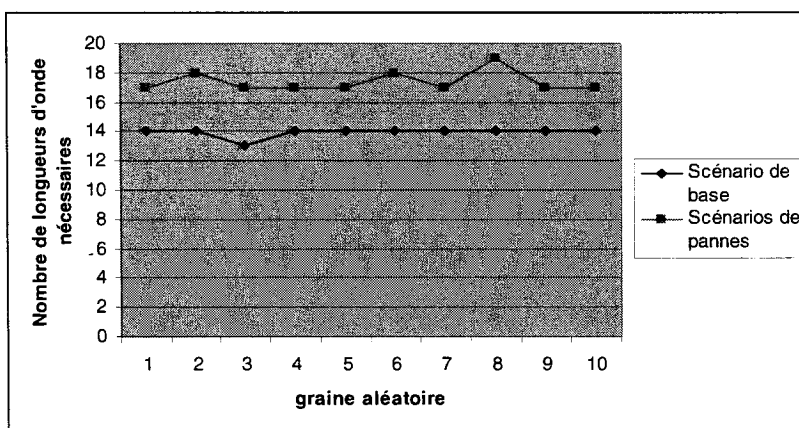
Pour ce faire, nous avons utilisé la fonction *rand()* qui renvoie un entier pseudo-aléatoire. Afin d'éviter de générer les mêmes séquences de nombres à chaque exécution, nous devons faire varier la graine aléatoire en la faisant passer au début de l'exécution au générateur pseudo-aléatoire grâce à la fonction *srand()*. La Figure 4.4 illustre les résultats de l'algorithme pour différentes graines aléatoires pour certaines topologies physiques considérées.



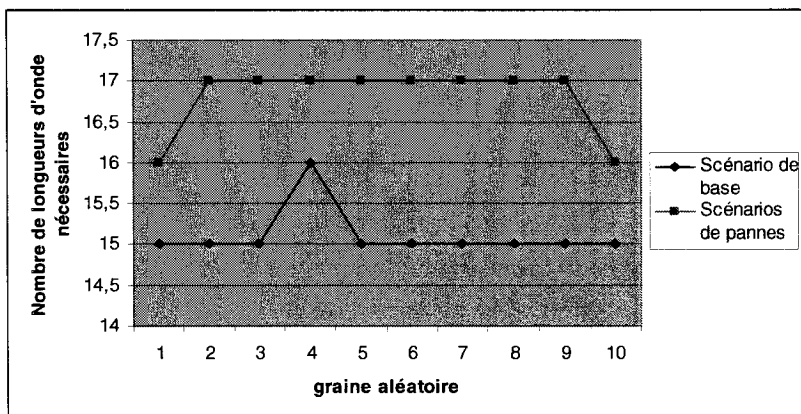
a) Topologie physique à 10 nœuds



b) Topologie physique NSFNET



c) Topologie physique à 20 nœuds



d) Topologie physique ARPANET

Figure 4.4 Effets de la variation de la graine aléatoire

D'après la Figure 4.4, nous voyons bien qu'en changeant la graine aléatoire passée au générateur pseudo-aléatoire, la qualité des résultats obtenus change par conséquent. Il est à noter que pour des graines aléatoires différentes donnant une même qualité de solutions (même nombre de longueurs d'onde nécessaires), les schéma de routage et d'affectation de longueurs d'onde ne sont pas les mêmes.

### 4.3.3 Analyse des résultats de l'algorithme

Après avoir calibré les paramètres de notre algorithme, nous pouvons passer à la présentation des résultats obtenus. Ainsi, nous allons dans un premier temps les comparer à ceux obtenus avec l'algorithme *TabouN* [26], vu qu'il traite le cas des trafics robustes et permanents. Ensuite, nous allons comparer nos résultats pour le scénario sans panne avec ceux obtenus par l'algorithme *KS* [9], déjà présenté dans le Chapitre 2, et avec les bornes supérieures obtenues par CPLEX.

#### 4.3.3.1 Résultats de l'algorithme *Lambda*

Dans cette section, une présentation des résultats obtenus par *Lambda* sera faite, et ce, par rapport au nombre de longueurs d'onde nécessaires pour acheminer l'ensemble des connexions aussi bien dans les scénarios de base que pour les scénarios de pannes simples. Le Tableau 4.3 montre, pour chaque topologie physique étudiée, le nombre minimal de longueurs d'onde nécessaires pour pouvoir acheminer toute la demande.

**Tableau 4.3** Nombre de longueurs d'onde nécessaires(scénarios avec et sans pannes)

	10 Nœuds	20 Nœuds	NSFNET	ARPANET	30 Nœuds
Nombre de longueurs d'onde nécessaires dans le scénario de base	4	13	11	15	46
Nombre de longueurs d'onde nécessaires dans tous les scénarios	5	17	14	16	54

Comme mentionné précédemment, les valeurs de la deuxième ligne du Tableau 4.3 sont obtenues en résolvant le problème max-RWA pour chacun des scénarios de pannes, et la valeur maximale de longueurs d'onde nécessaires est retenue. Ainsi, pour le réseau à 10 nœuds et pour la matrice de trafic illustrée, il faut avoir au moins 4 longueurs d'onde pour satisfaire toutes les connexions dans le scénario de base, et 5 pour les satisfaire dans tous les scénarios.

Le Tableau 4.4 montre le nombre de connexions bloquées dans les scénarios de pannes simples avec le nombre de longueurs d'onde déjà trouvé pour chaque topologie physique. Ainsi, pour les trois modèles de trafic considérés, nous avons trouvé le nombre de connexions bloquées dans les scénarios de pannes simples. Nous voyons bien que le taux de blocage augmente avec le nombre de connexions permanentes considérées. Ceci est tout à fait trivial, puisque ces dernières sont automatiquement bloquées lorsqu'elles sont atteintes par une panne. Les résultats obtenus à ce niveau vont nous être utiles par la suite afin de valider l'hypothèse énoncée au Chapitre 3.

**Tableau 4.4 Nombre de trafics bloqués dans les scénarios de pannes**

		10 Nœuds	20 Nœuds	NSFNET	ARPANET	30 Nœuds
<b>Nombre de longueurs d'onde nécessaires dans le scénario de base</b>		4	13	11	15	46
<b>Nombre de trafics bloqués dans les scénarios de pannes</b>	<b>1<sup>er</sup> modèle de trafic</b>	33	135	19	13	139
	<b>2<sup>ème</sup> modèle de trafic</b>	87	512	300	502	1973
	<b>3<sup>ème</sup> modèle de trafic</b>	56	344	129	295	819

#### 4.3.3.2 Analyse comparative des résultats

Dans cette section, nous allons comparer les résultats de nos algorithmes avec ceux obtenus par l'algorithme *TabouN* [26]. Le choix de ce dernier émane du fait qu'il considère des demandes de connexions robustes et permanentes.

Le Tableau 4.5 illustre le nombre de connexions bloquées dans le scénario de base pour les deux algorithmes *TabouN* et *Lambda*. Le nombre de longueurs d'onde considéré pour chaque réseau est trouvé par l'algorithme *Lambda*. Ainsi, aucune connexion ne sera bloquée par ce dernier dans le scénario de base. Nous pouvons voir que *TabouN* n'arrive pas à acheminer la totalité des connexions et en bloque un certain nombre, dépendamment de la taille de l'exemplaire considéré.

**Tableau 4.5 Nombre de trafics bloqués dans le scénario de base**

	20 Nœuds	10 Nœuds	NSFNET	ARPANET	30 Nœuds
$\lambda$	13	4	11	15	46
<i>TabouN</i>	3	1	1	2	9
<i>Lambda</i>	0	0	0	0	0

Les résultats de la généralisation de l'algorithme *Lambda* pour les scénarios de pannes sont montrés au Tableau 4.6. En effet, le nombre de longueurs d'onde considéré est celui trouvé précédemment pour acheminer l'ensemble des connexions dans tous les scénarios. Il est donc normal que *Lambda* ne bloque aucune connexion. De même que pour le scénario de base, il est évident que *TabouN* n'arrive pas à acheminer l'ensemble de la demande dans tous les scénarios et requiert donc un nombre plus élevé de longueurs d'onde. Il est à noter que les trafics considérés ici sont tous robustes. En effet, avec le nombre de longueurs d'onde trouvé pour chacune des topologies, toutes les connexions sont acheminées dans tous les scénarios.

**Tableau 4.6 Nombre de trafics bloqués dans les scénarios de pannes**

	20 Nœuds	10 Nœuds	NSFNET	ARPANET	30 Nœuds
$\lambda$	17	5	14	16	54
<i>TabouN</i>	487	76	282	420	1501
<i>Lambda</i>	0	0	0	0	0

Une fois la sélection des routes est faite pour chaque connexion, nous allons passer à la résolution du problème RWAP dont l'objectif est de minimiser le nombre de connexions bloquées avec un certain nombre de longueurs d'onde dans tous les scénarios. C'est à ce niveau que l'heuristique de coloriage est appliquée pour l'affectation des longueurs d'onde. Les Tableaux 4.7, 4.8, 4.9, 4.10 et 4.11 illustrent le nombre de connexions bloquées pour les différentes topologies considérées avec un certain nombre de longueurs d'onde dans tous les scénarios.

**Tableau 4.7 Nombre de connexions bloquées pour le réseau à 10 nœuds**

10 nœuds	Scénario de base		Scénarios de pannes					
			TabouN			Lambda		
	TabouN	Lambda	mod 1	mod 2	mod 3	mod 1	mod 2	mod 3
1	12	12	235	241	236	219	234	227
2	1	0	39	90	63	33	87	56

**Tableau 4.8 Nombre de connexions bloquées pour le réseau NSFNET**

NSFNET	Scénario de base		Scénarios de pannes					
			TabouN			Lambda		
	TabouN	Lambda	mod 1	mod 2	mod 3	mod 1	mod 2	mod 3
1	76	76	1755	1755	1755	1646	1676	1669
2	45	45	1138	1147	1144	1076	1125	1085
4	11	8	352	493	416	280	468	355
11	1	0	94	301	187	19	300	129

Dans l'ensemble, les résultats obtenus par notre algorithme sont meilleurs que ceux de *TabouN*, surtout pour les scénarios de pannes simples et ce, pour les trois modèles de trafics considérés. En effet, nous pouvons voir par exemple pour le réseau aléatoire à 20 nœuds que le nombre de connexions bloquées dans les scénarios de pannes avec 2 longueurs d'onde est inférieur d'environ 4% à celui obtenu par l'algorithme *TabouN* pour les trois modèles de trafics. Cette différence peut être d'autant plus importante si le nombre de connexions bloquées dans le scénario de base est aussi inférieur. Ceci est tout

à fait normal vu qu'une connexion bloquée dans le scénario de base le sera pour tous les scénarios de pannes simples de liens. Ainsi, pour la même topologie physique et avec 12 longueurs d'onde, cette différence s'élève à plus de 35% lorsque tous les trafics sont robustes. La Figure 4.5 montre un sommaire des résultats obtenus pour les différents fichiers de topologies considérés.

**Tableau 4.9 Nombre de connexions bloquées pour le réseau à 20 nœuds**

20 Nœuds	Scénario de base		Scénarios de pannes					
			TabouN			Lambda		
	TabouN	Lambda	mod 1	mod 2	mod 3	mod 1	mod 2	mod 3
1								
2	107	107	2868	2869	2869	2753	2762	2757
4	76	76	2150	2157	2155	2041	2082	2056
8	34	34	1160	1217	1182	1074	1200	1145
12	7	4	448	638	522	275	592	446

Au Chapitre 3, nous avons énoncé une hypothèse quant au nombre de connexions bloquées dans les scénarios de pannes avec deux schémas de routage différents pour le scénario de base. Nous allons montrer sa validité en s'appuyant sur les résultats obtenus par notre algorithme. En effet, ce dernier cherche le nombre minimal de longueurs d'onde nécessaires pour acheminer toute la demande. Pour ce même nombre de longueurs d'onde, nous allons évaluer le nombre de connexions bloquées dans les scénarios de pannes pour l'algorithme *TabouN*. Il est à noter que ce dernier en bloque déjà un certain nombre dans le scénario de base, comme le montre le Tableau 4.5.

Dans ce contexte, nous allons évaluer le nombre de connexions bloquées par l'algorithme *TabouN* et par notre méthode constructive pour les trois modèles de trafics. Il est à noter que le cas le plus critique se présente avec des connexions robustes, comme montré à la Figure 3.2. En effet, la pente des droites présentant le nombre de connexions bloquées sera très grande. Des résultats comparatifs pour certaines topologies physiques sont illustrés à la Figure 4.6.

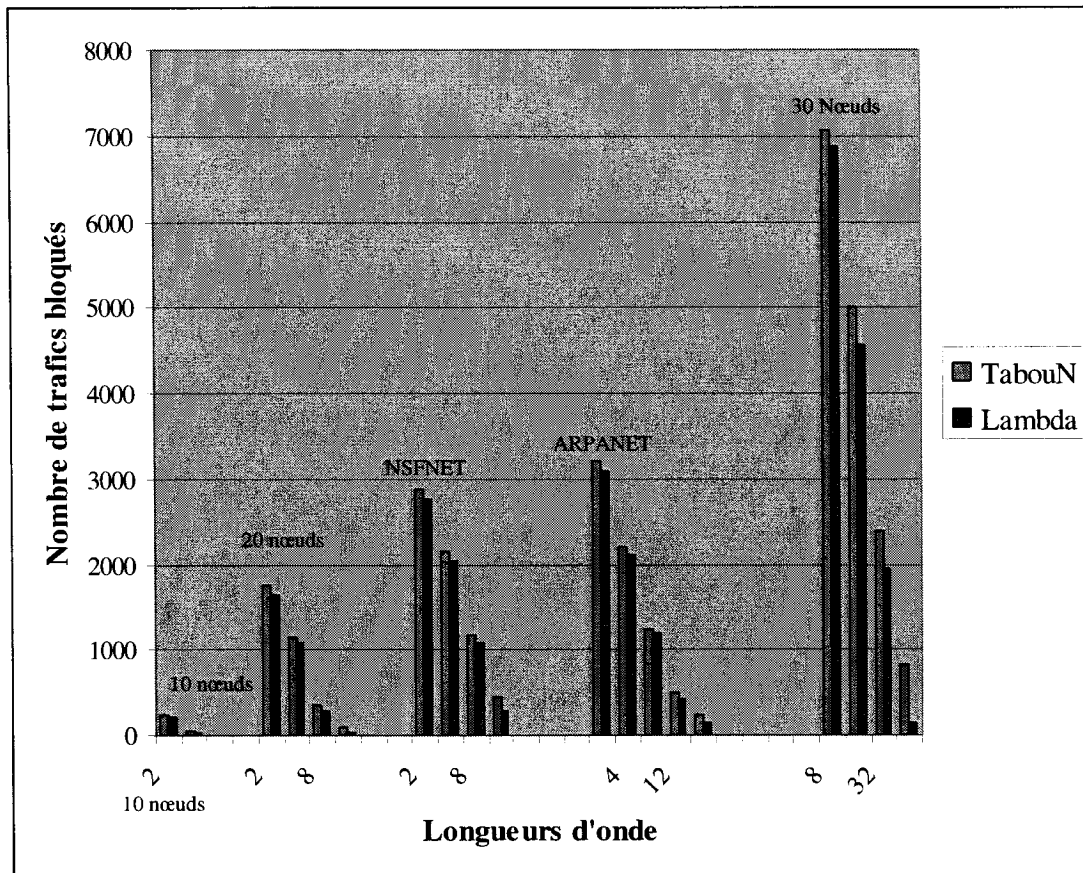
**Tableau 4.10 Nombre de connexions bloquées pour le réseau ARPANET**

ARP ANET	scénario de base		scénarios de pannes					
			TabouN			Lambda		
	TabouN	Lambda	mod 1	mod 2	mod 3	mod 1	mod 2	mod 3
1								
2	97	97	3193	3204	3198	3095	3109	3101
4	64	64	2201	2243	2220	2114	2182	2151
8	35	35	1243	1415	1330	1196	1405	1314
12	11	11	482	733	617	423	778	628
14	4	3	245	535	390	143	576	399

**Tableau 4.11 Nombre de connexions bloquées pour le réseau à 30 nœuds**

30 nœuds	scénario de base		scénarios de pannes					
			TabouN			Lambda		
	TabouN	Lambda	mod 1	mod 2	mod 3	mod 1	mod 2	mod 3
1								
8	177	177	7066	7178	7103	6867	7004	6962
16	121	117	5002	5352	5111	4573	5113	4838
32	50	50	2382	3086	2694	1949	3113	2504
46	9	0	826	1793	1255	139	1973	819

Ainsi, il est clair que le nombre de connexions bloquées dans les scénarios de pannes est toujours inférieur avec notre algorithme pour toutes les topologies considérées. En effet, pour le réseau aléatoire à 30 nœuds par exemple, le nombre de trafics bloqués est inférieur de plus de 80% avec notre algorithme. Ces résultats auraient pu être nettement meilleurs si une méthode plus puissante (recherche locale par exemple) serait considérée pour traiter les scénarios de pannes.

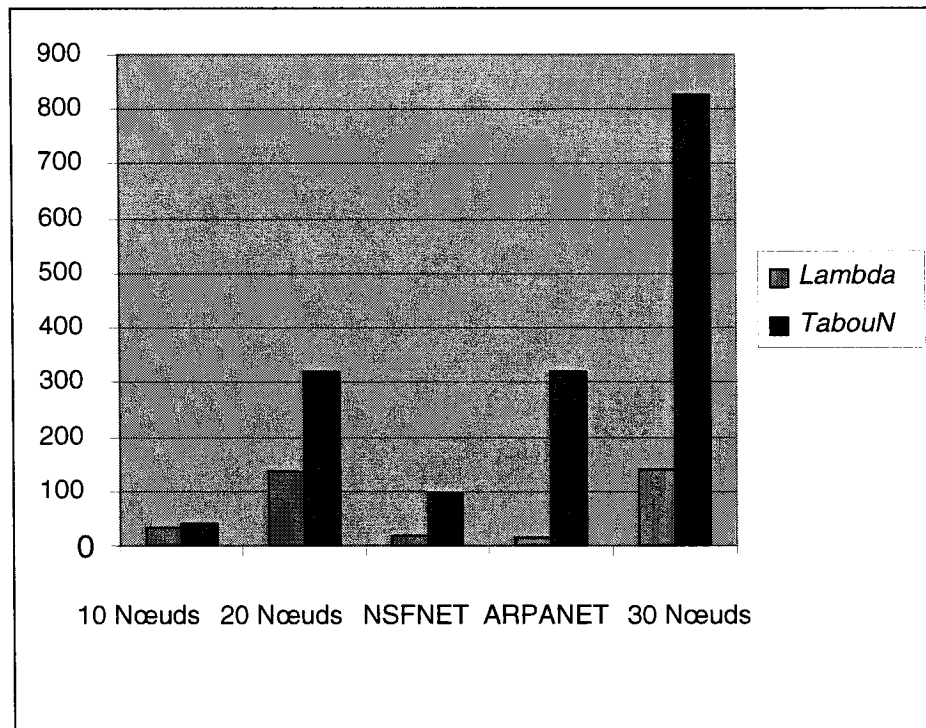


**Figure 4.5 Illustration du nombre de connexions bloquées dans les scénarios de pannes avec des connexions robustes**

#### 4.3.3.3 Résultats comparatifs pour le scénario de base

Krishnaswamy et Sivarajan ont proposé dans [9] des algorithmes qui permettent de maximiser le nombre de connexions acheminées dans le scénario de base. Ces algorithmes se fondent sur la formulation du problème sous la forme d'un problème linéaire et sont décrites au Chapitre 2. Les algorithmes proposés par les auteurs résolvent la relaxation continue du problème et arrondissent les solutions selon différentes techniques détaillées dans [9]. Afin d'évaluer la performance des algorithmes proposés,

les auteurs ont utilisé des topologies physiques et des matrices de connexions différentes, présentées au Tableau 4.12.



**Figure 4.6** Nombre de connexions bloquées dans les scénarios de pannes

Les résultats obtenus par *Lambda*, *KS*, *TabouN* ainsi que les bornes inférieures trouvées par CPLEX sur ces jeux de tests sont rapportés dans le Tableau 4.13.

**Tableau 4.12** Topologies et trafics considérés

	Nb nœuds	Nb liens	Connexions
NSFNET	14	21	268
EONNET	20	39	374

Le Tableau 4.13 montre le nombre de connexions acheminées avec un certain nombre de longueurs d'onde variant de 16 à 24 pour les différents algorithmes étudiés. Nous remarquons que *Lambda* arrive souvent à l'optimal pour les deux topologies

physiques étudiées. Nous pouvons aussi voir que notre algorithme trouve la solution optimale, c'est-à-dire le nombre minimal de longueurs d'onde nécessaires pour acheminer l'ensemble du trafic. Les Figures 4.7 et 4.8 illustrent bien ces résultats.

**Tableau 4.13 Résultats obtenus par *Lambda* pour NSFNET et EONNET**

$\lambda$	NSFNET				EONNET			
	BS	KS	TabouN ( $P=\infty$ )	Lambda	BS	KS	TabouN ( $P=\infty$ )	Lambda
16	258	239	252	258	350	319	345	346
17	263	250	258	262	356	333	352	354
18	267	255	261	266	362	339	356	360
19	<b>268</b>	263	264	<b>268</b>	367	340	361	365
20	<b>268</b>	265	265	<b>268</b>	370	343	366	369
21	<b>268</b>	267	266	<b>268</b>	373	347	370	372
22	<b>268</b>	267	267	<b>268</b>	<b>374</b>	355	372	<b>374</b>
23	<b>268</b>	268	268	<b>268</b>	<b>374</b>	361	374	<b>374</b>
24	<b>268</b>	268	268	<b>268</b>	<b>374</b>	367	374	<b>374</b>
Total	2396	2342	2369	2394	3300	3104	3270	3288

#### 4.3.3.3 Temps d'exécution

Il est évident que le temps d'exécution représente un paramètre crucial dans l'évaluation de tout algorithme. Il est ainsi intéressant de pouvoir atteindre nos résultats dans les délais les plus brefs possibles.

Dans l'ensemble, l'algorithme *Lambda* est assez rapide pour toutes les topologies considérées. La partie la plus complexe de l'algorithme est la phase 2, c'est-à-dire le coloriage complet du graphe auxiliaire. En effet, le nombre d'itération pour la recherche d'ensembles stables est très réduit, ce qui lui confère un caractère glouton. De plus, une approche constructive a été adoptée pour traiter les scénarios de pannes simples.

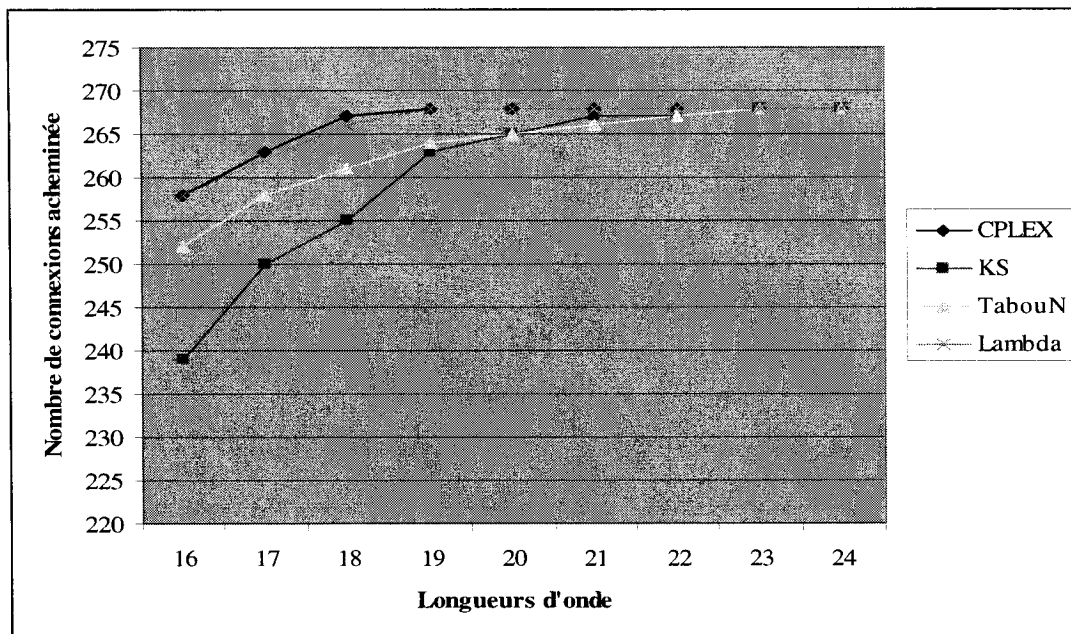


Figure 4.7 Trafics acheminés pour le réseau NSFNET

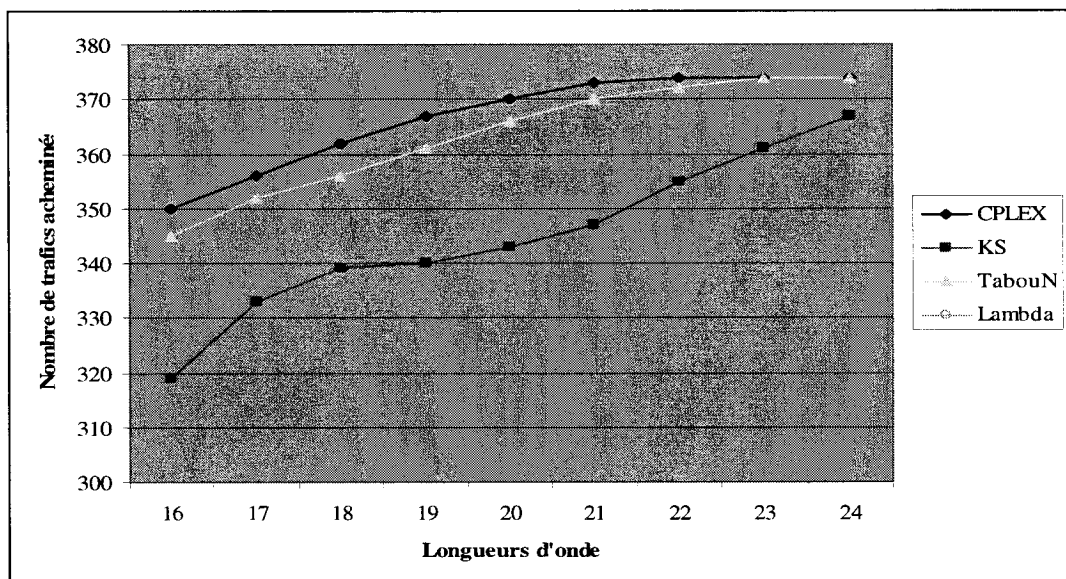


Figure 4.8 Trafics acheminés pour le réseau EONNET

Ainsi, la taille des problèmes traités n'aurait pas une grande incidence sur le temps d'exécution, vu que ce dernier ne croît pas linéairement par rapport à la taille des exemplaires considérés. En effet, la taille du graphe auxiliaire engendré est plus ou moins la même pour tous les réseaux et les matrices de trafics considérés, vu que l'algorithme cherche d'abord des ensembles stables qu'il extrait du graphe jusqu'à atteindre une borne  $U$  fixée (voir Figure 3.7).

Le Tableau 4.14 montre les temps d'exécution de l'algorithme *Lambda* pour les différentes topologies physiques considérées. Afin de comparer ces temps avec ceux de l'algorithme *TabouN* et vu que ce dernier dépend du nombre de longueurs d'onde dans le réseau, nous avons évalué le temps d'exécution de *TabouN* avec le nombre de longueurs d'onde trouvé par notre algorithme.

Enfin, il est à mentionner que le temps n'est pas aussi capital dans notre cas parce que nous sommes dans le cas statique et que cette opération est effectuée hors ligne et avant la mise en service du réseau.

**Tableau 4.14 Évaluation du temps d'exécution**

	Temps (s)	
	TabouN	Lambda
10 nœuds	1,90	1,95
NSFNET	11,24	4,34
20 nœuds	17,62	6,19
ARPANET	22,67	9,72
30 nœuds	87,33	58,80

# CHAPITRE V

## CONCLUSION

Dans ce mémoire, nous avons traité le problème de routage et d'affectation de longueurs d'onde max-RWA dans les réseaux optiques WDM. Nous avons considéré le cas statique de trafic et l'objectif était principalement de minimiser le nombre de longueurs d'onde nécessaires afin d'acheminer l'ensemble des connexions dans les scénarios avec et sans pannes simples de liens. Vu le caractère difficile du problème, nous avons développé une heuristique générique de résolution basée sur le coloriage de graphe et utilisant la recherche locale avec liste taboue. Dans le présent chapitre, nous allons faire une synthèse des travaux réalisés. Comme pour tout travail de recherche, nous présenterons par la suite les limites des méthodes proposées ainsi que quelques indications des travaux futurs.

### ***5.1 Synthèse des travaux***

Le problème de routage et d'affectation de longueurs d'onde consiste à trouver une route à chaque connexion et à lui affecter une longueur d'onde en respectant les contraintes d'unicité et de continuité de longueurs d'onde. La première contrainte interdit que deux chemins optiques traversant la même fibre aient la même longueur d'onde. Quant à la deuxième, elle est imposée par l'absence de conversion et veut qu'une connexion garde la même longueur d'onde tout au long de son chemin optique. Dans ce travail, nous avons aussi traité les différents scénarios de pannes simples considérant des trafics robustes et permanents.

Dans le présent mémoire, nous avons traité la variante du problème max-RWA consistant à minimiser le nombre de longueurs d'onde nécessaires afin d'acheminer l'ensemble des connexions. Nous avons donc exposé une formulation mathématique du

problème en répertoriant l'ensemble des contraintes énoncées et en mettant l'emphase sur l'acheminement de l'ensemble des connexions. Afin de résoudre le problème, nous avons opté pour une approche de décomposition. Ainsi, les sous-problèmes de routage et d'affectation de longueurs d'onde étaient traités séparément. En effet, un ensemble d'au plus  $F$  routes disjointes d'arcs était trouvé pour chaque connexion en utilisant une version améliorée de l'algorithme de *Dijkstra* des plus courts chemins avec pour métrique de routage le nombre de sauts. Par la suite, nous avons abordé le problème d'affectation de longueurs d'onde en transformant ce dernier en un problème de coloriage complet de graphe. Une heuristique est alors proposée à cette fin. Cette heuristique, nommée *Lambda*, s'inspire d'une méthode de coloriage efficace [41] et est divisée en deux phases. La première consiste à extraire des ensembles stables du graphe auxiliaire jusqu'à atteindre une certaine taille fixée de ce dernier. À ce stade, une heuristique de recherche locale taboue est appliquée afin de colorier au complet le graphe auxiliaire résiduel. Le nombre de couleurs trouvées correspond au nombre de longueurs d'onde nécessaires afin d'acheminer l'ensemble des connexions. Il est à noter que cette heuristique, en plus d'affecter une longueur d'onde à chaque connexion, sélectionne la route à adopter parmi l'ensemble des  $F$  chemins disjointes d'arcs qui ont été trouvés auparavant.

Afin de traiter les scénarios de pannes, nous avons développé une heuristique constructive qui énumère tous les scénarios de pannes simples de liens et qui, pour le nombre de longueurs d'onde calculé précédemment, essaie de ré-acheminer les trafics robustes atteints à travers les autres routes qui lui ont été trouvées lors de la résolution du sous-problème de routage. Le choix d'une méthode à priori peu efficace pour traiter les scénarios de pannes est surtout influencé par l'hypothèse que nous avons énoncée au Chapitre 3.

Dans un deuxième temps, nous avons essayé de généraliser la formulation mathématique présentée pour les scénarios de pannes simples de liens. Ainsi, l'objectif devient de minimiser le nombre de longueurs d'onde nécessaires afin d'acheminer l'ensemble des connexions dans tous les scénarios. Afin de résoudre ce nouveau

problème, nous avons énuméré tous les états du réseau (à la suite d'une panne simple de lien), et pour chacun, nous avons appliqué l'algorithme *Lambda*. Le nombre de longueurs d'onde ainsi nécessaire correspond au nombre maximal pour tous les scénarios. Avec un tel nombre de longueurs d'onde, on est donc sûr qu'aucun trafic ne sera bloqué quel que soit l'état du réseau (sans ou avec pannes simples de liens).

Pour évaluer la performance de nos algorithmes, nous avons comparé nos résultats à ceux obtenus avec l'algorithme *TabouN* [27]. Nous avons considéré cinq topologies physiques et des modèles de trafics différents. Dans la majorité des cas, les résultats que nous avons obtenus étaient meilleurs que ceux de *TabouN*. De plus, nous avons comparé nos résultats pour le scénarios de base à l'algorithme *KS* [6] et aux bornes supérieures calculées avec CPLEX par les auteurs de la méthode, pour les réseaux NSFNET et EONNET et pour de nouvelles matrices de trafics. Dans tous les cas, nos résultats étaient nettement meilleurs que ceux de *KS* et nous arrivons, pour ces deux topologies, à trouver l'optimum, c'est-à-dire le nombre minimal de longueurs d'onde nécessaires afin d'acheminer toute la demande. Enfin, les temps d'exécution de nos algorithmes étaient très raisonnables surtout que ceci n'est pas un élément aussi crucial, vu que le cas statique est considéré.

## **5.2 Limitations des travaux**

En dépit des résultats intéressants obtenus par nos algorithmes, ces derniers présentent certaines limitations. En effet, nous avons considéré des réseaux avec des liens monofibres, c'est-à-dire que chaque lien est constitué de deux fibres unidirectionnelles. Cependant, les réseaux actuels deviennent de plus en plus k-fibres, vu que les coûts associés à l'installation de plusieurs fibres optiques sur un même lien sont nettement inférieurs à ceux engendrés par l'installation d'une fibre supportant un plus grand nombre de longueurs d'onde. De plus, nous n'avons pas considéré le cas de conversion de longueurs d'onde. En effet, de plus en plus de technologies permettent les conversions tout optiques (sans recourir à des conversions optoélectroniques). Enfin,

pour traiter les scénarios de pannes, nous avons considéré une technique constructive qui pourrait être améliorée.

### **5.3 Travaux futurs**

Le problème de routage et d'affectation de longueurs d'onde est un problème ouvert nécessitant encore un énorme effort de recherche. Premièrement, pour résoudre le problème de routage, nous avons considéré les plus courts chemins disjoints d'arcs pour chaque trafic. Ce choix pourrait s'avérer non optimal et une option serait de considérer un plus grand nombre d'alternatives de routes sous-optimales. De cette manière, une plus grande flexibilité pour l'affectation de longueurs d'onde serait donnée. De plus, et dans le même contexte, il serait peut être plus intéressant de recalculer les routes pour chaque connexion dans chaque scénario. Enfin, il serait également intéressant d'étendre le problème en introduisant des convertisseurs de longueurs d'onde dans le réseau. Ainsi, le problème de placement stratégique des ces équipement viendra compléter le problème routage et d'affectation de longueurs d'onde RWA traité dans ce mémoire.

## BIBLIOGRAPHIE

- [1] Zang H., Jue J. P., Mukherjee B., "A Review of Routing and Wavelength Assignment Approaches for Wavelength-Routed Optical WDM Networks", *Optical Network Magazine*, January 2000, pp. 47-60.
- [2] Gangxiang S., Bose S.; Hiang C., "Low Delay Heuristic Algorithm for Wavelength Assignment in WDM Networks", *Proceedings, IEEE International Conference on Networks (ICON '99)*, 1999, Brisbane (Australia), pp. 3-10.
- [3] Harder E., Lee S., Choi H., "On Wavelength Assignment in WDM Optical Networks", *Proceedings of the Fourth International Conference on Massively Parallel Processing Using Optical Interconnections*, 1997, pp. 32-38.
- [4] Banerjee S., Mukherjee B., "A practical approach for routing and wavelength assignment in large wavelength routed optical networks", *IEEE J.Sel. Areas in Communications*, Vol. 14, No. 5, June 1996, pp. 903-908.
- [5] Manohar P., Manjunath D., Shevgaonar R., "Routing and Wavelength Assignment in Optical Networks From Edge Disjoint Path Algorithm", *IEEE Communications Letters*, Vol. 6, No. 5, May 2002, pp. 211-213.
- [6] Chamberland S., Oulai D., Pierre S., "Routing and Wavelength Assignment for Permanent and Reliable Wavelength Path in WDM Networks", *IEEE CCECE 2002, Canadian Conference on Electrical and Computer Engineering*, 2002. vol. 1, Winnipeg (MB), pp. 301-306.

- [7] Datta R., Ghose S., Sengupta I., “An Algorithm for Optimal Assignment of a Wavelength in a Tree Topology and Its Application in WDM Networks”, *IEEE Journal On Selected Areas in Communications*, Vol. 22, No. 9, November 2004, pp. 1589-1600.
- [8] Baskiotis N., Pérennes S., Rivano H., “Dimensionnement Heuristiques des Réseaux Optiques WDM Multifibres par Arrondi Aléatoire de Multiflot”, *Rapport de recherche Projet MASCOTTE*, Université Nice Sophia Antipolis, mars 2002.
- [9] Krishnaswamy R. M., K. N. Sivarajan, “Algorithms for Routing and Wavelength Assignment Based on Solutions of LP-Relaxations”, *IEEE Communications Letters*, Vol. 5, No. 10, Octobre 2001, pp. 435-437.
- [10] Zhang Y., Taira K., Takagi H., Das S. K., “An Efficient heuristic for Routing and Wavelength Assignment in Optical WDM Networks”, *IEEE International on Communications*, ICC 2002, Vol. 5, May 2002, pp. 2734-2739.
- [11] Zyane A., “Les réseaux optiques sous la technologie WDM : Principes et perspectives”, Rapport de stage, École Polytechnique de Montréal, 2001.
- [12] Choi J. S., Golmie N., Lapeyrere F., Mouveaux F., Su D., “A Functional Classification of Routing and Wavelength Assignment Schemes in DWDM networks : Static Case”, *7th International Conference on Optical Communications and Networks*, January 2000.
- [13] Mokhtar A., Azizoglu M., “Adaptive Wavelength Routing in All-Optical Networks”, *IEEE/ACM Transactions on Networking*, Vol. 6, No. 2, April 1998, pp. 197-206.

- [14] Mokhtar A., Azizoglu M., "Adaptive Techniques for Routing and Wavelength Assignment in All-Optical WANs", *IEEE 39<sup>th</sup> Midwest symposium on Circuits and Systems*, Vol. 3, No. 3, August 1996, pp. 1195-1198.
- [15] Mukherjee, B., "WDM optical communication networks: progress and challenges", *IEEE Journal on Selected Areas in Communications*, Vol. 18, No. 10, October 2000, pp. 1810-1824.
- [16] Modiano E., Narula-Tam A., "Designing Survivable Networks using Effective Routing and Wavelength Assignment (RWA)", *Optical Fiber Communication Conference and Exhibit*, vol. 2, 2001, Dallas (Texas), pp. TuG5-T1-3.
- [17] Lima E., Stubkjaer K., "An algorithm for link restoration of wavelength routing optical networks", *IEEE International Conference on Communications, ICC '99*, Vol. 3, 1999, Vancouver (AB), pp. 2055-2061.
- [18] Shiefeng L., Jun T., Guanqun G., "Routing and Wavelength Assignment in All-Optical Networks to Establish Survivable Lightpaths", *IEEE Region 10 Conference on Computers, Communications, Control and Power Engineering*, Vol. 2, October 2002, pp. 1193-1196.
- [19] Zang Z., Acampora A., "A Heuristic Wavelength Assignment Algorithm for Multihop WDM Networks with Wavelength Routing and Wavelength Re-Use", *IEEE/ACM Transactions on Networking*, vol. 3, No. 3, June 1995, pp. 281 -288.
- [20] Ramamurth B., Mukherjee B., "Wavelength conversion in WDM Networking", *IEEE Journal on Selected Areas in Communications*, Vol. 16, No. 7, September 1998, pp. 1061-1073.

- [21] Chu X., Li B., Sohraby K., Zhang Z., "Routing and Wavelength Assignment Issues in the Presence of Wavelength Conversion for All-Optical Networks", *IEEE Global Telecommunications Conference, 2002. GLOBECOM '02*, Vol.3, November 2002, pp. 2787-2791.
- [22] Sohraby K., Zhang Z., Chu X., "Resource Management in An Integrated Optical Network", *IEEE Journal On Selected Areas in Communications*, Vol. 21, No. 7, September 2003, pp. 1052-1062.
- [23] Harai H., Murata M., Miyahara H. "Performance analysis of wavelength assignment policies in all-optical networks with limited-range wavelength conversion", *IEEE Journal on Selected Areas in Communications*, Vol. 16, No. 7, September 1998, pp. 1051-1060.
- [24] Mukherjee B., Ramamurthy S., Banerjee D., Mukherjee A., "Some principles for designing a wide-area optical network", *INFOCOM '94 Networking for Global Communications., 13th Proceedings IEEE*, vol.1, 1994, Toronto (Canada), pp. 110-119.
- [25] Dzongang C., Galinier P, Pierre S., "A Tabu Search Heuristic for the Routing and Wavelength Assignment Problem in Optical Networks", *IEEE Communications Letters*, Vol. 9, No.5, April 2005, pp. 426 - 428.
- [26] Oulaï D., "Routage et Affectation de Longueurs D'onde dans les Réseaux Optiques WDM", Mémoire de maîtrise, École Polytechnique de Montréal, Décembre 2002.

- [27] Dzungang C., “Heuristiques Taboues Pour le Routage et l’Affectation de Longueurs d’Onde Dans les Réseaux Optiques”, Mémoire de maîtrise, École Polytechnique de Montréal, Décembre 2003.
- [28] Mikac B., Inkret R., Podnar I., “An Application of Heuristic search Techniques in Telecommunication System Design”, *Proceedings Second International Conference on University Knowledge-Based Intelligent Electronic Systems*, Vol. 1, 1998, Adelaide (Australia), pp. 311-317.
- [29] Qui H, Liu Z, Zhang S., Wen A., “Routing and Wavelength Assignment Based on Genetic Algorithm”, *IEEE Communications Letters*, Vol. 6, No. 10, October 2002, pp. 455-457.
- [30] Li N., Chu X., “Routing And Wavelength Assignment vs. Wavelength Converter Placement In All-Optical Networks”, *IEEE Optical Communications*, August 2003, pp. S22-S28.
- [31] Chu X., Li B., Zhang Z., “A dynamic RWA algorithm in a wavelength-routed all-optical network with wavelength converters” *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies*. Vol. 3, 30 March-3 April 2003, pp. 1795-1804.
- [32] Ozdaglar A. E., Bertsekas D. P., “Routing and Wavelength Assignment in Optical Networks”, *IEEE/ACM Transactions on Networking*, Vol. 11, No. 2, April 2003, pp. 259-272.
- [33] Beauquier B., Bermond J. C., Gargano L., Hell P., Perennes S., Vaccaro U., “Graph Problems Arising from Wavelength-Routing in All-Optical Networks”, *Rapport de recherche de l’INRIA-Sofia Antipolis*, Mai 1997.

- [34] Wilson B. J., Stoffel N. G., Pastor J. L., Post M. J., Lui K. H., Li T., Walsh K. A., Wei J. Y., Tsai Y., “Multiwavelength Optical Networking Management and Control”, *IEEE Journal of Lightwave Technology*, Vol. 18, No. 12, December 2000, pp. 2038-2057.
- [35] Coudert D., Rivano H., “Routage optique dans les réseaux WDM multifibres avec conversion partielle”, *AlgoTel'02*, Mèze. France, Mai 2002.
- [36] Kuchar A., “All-Optical Routing – Progress and Challenges”, *IEEE International Conference on Transparent Optical Networks, ICTON 2002*.
- [37] Mokhtar A., Azizoglu M., “Dynamic Route selection and Wavelength Assignment in All-Optical Networks”, *Lasers and Electro-Optics Society Annual Meeting*, Vol. 1, 1995, San Francisco (CA), pp. 220-221.
- [38] Taechan L., Kyungsik L., Sungsoo P., “Optimal Routing and Wavelength Assignment in WDM Ring Networks”, *IEEE Journal on Selected Areas in Communications*, Vol. 18, No. 10, October 2000, pp 2146-2154.
- [39] Zhiang X., Qiao C., “Wavelength Assignment for Dynamic traffic in Multi-fiber WDM Networks”, *Proceedings 7th International Conference on Computer Communications and Networks*, 1998, Lafayette (Louisiana), pp. 479-485.
- [40] Chen B., Wang J., “Efficient Routing and Wavelength Assignment for Multicast in WDM Networks”, *IEEE Journal on Selected Areas in Communications*, Vol. 20, No. 1, January 2002, pp. 97-109.

- [41] Galinier P., Hertz A., "A Survey of Local Search Methods for Graph Coloring",  
*Les Cahiers du Gerad*, April 2004.