On the design of multifiber WDM networks

Afonso Ferreira, Stéphane Pérennes, A. W. Richa, Hervé Rivano, N. Stier Moses

On the design of multifiber WDM *networks*

A. Ferreira[1], S. Pérennes[1], A. W. Richa[2,*], H. Rivano[1,3] and N. Stier[4,*]

[1]MASCOTTE *Project, CNRS, I3S & INRIA Sophia Antipolis.*

[2]*Department of Computer Science and Engineering, Arizona State University.*

[3]*France Telecom Recherche & Développement Sophia Antipolis.*

[4]*MIT Operations Research Center.*

* *This research was conducted while this author was visiting the* MASCOTTE *Project.*

In this paper, we address multifiber optical networks with Wavelength Division Multiplexing (WDM). Assuming that the lightpaths use the same wavelength from source to destination, we extend the definition of the well-known Wavelength Assignment Problem (WAP) to the case where there are k fibers per link, and w wavelengths per fiber are available. This generalization is called the (k,w)-WAP. We develop a new model for the (k,w)-WAP based on *conflict hypergraphs*. Conflict hypergraphs accurately capture the lightpath interdependencies, generalizing the conflict graphs used for single-fiber networks. By relating the (k,w)-WAP with the hypergraph coloring problem, we prove that the former is $\mathcal{NP}$-complete, and present further results with respect to the complexity of that problem. We consider the two natural optimization problems that arise from the (k,w)-WAP: the problem of minimizing k given w, and that of minimizing w given k. We develop and analyze the practical performances of two methodologies based on hypergraph coloring, one for each of the two optimization problems, on existing backbone networks in Europe and in the USA. The first methodology relies on an integer programming formulation, and the second consists of a heuristic based on a randomized algorithm.

Keywords: optical networks, wavelength division multiplexing, network design, wavelength assignment problem, hypergraph coloring, integer programming, heuristics.

---

1 Introduction

Wavelength Division Multiplexing (WDM) is currently the most promising existing optical network technology, since it allows for efficient use of the high bandwidth offered by optical networks. Under WDM, wavelengths are used to implement fixed end-to-end connections — called lightpaths in this context — in the network. The major constraint imposed by this technology is that different lightpaths cannot share the same wavelength over the same link.

Our work focuses on studying WDM networks in real-life scenarios, from both theoretical and practical perspectives. Perhaps surprisingly, from the telecommunications operator viewpoint, one of the largest costs incurred while deploying an optical network stems from physically trench-digging to bury the optical fibers. Hence, it is usual to have many fibers deployed between any two points of the network, giving rise to *multifiber* WDM *networks* (or MWNs for short).

Minimizing the cost of such a network leads to the design problem known as the *wavelength assignment problem* (WAP) [RS95, BBG+97, RS98, CFK+01]. The off-line version of the WAP can be define as follows: Given a WDM network $\mathcal{N}$ and a set of lightpaths satisfying traffic requests, assign wavelengths to the lightpaths so that any two paths that cross the same link are assigned different wavelengths. Other issues on MWNs design, like optical routing, grooming, optical add-drop multiplexers placement and wavelength translation, are out of scope of this paper although they may sway the cost of the network as a whole.

Unfortunately, the existing work on single-fiber network design cannot be extended to MWNs in a straightforward manner. For instance, the model used for the WAP on single-fiber networks fails to fully capture the benefits of having more fibers per link when minimizing the total number of wavelengths used in the network in MWNs. The addition of multiple fibers to the network incur an extra degree of freedom in choosing the path wavelengths which was not present in single-fiber networks. Note that using k fibers per link immediately allows for reducing the number of wavelengths by a factor of k. In fact, multifibers may reduce the number of wavelengths required even further. For example, adding just one fiber to a single-fiber network can decrease the number of wavelengths required to route n lightpaths from n to 1 [MS00, LS00]. Unfortunately, results of this flavor, which specifically determine the impact of having multifibers either hold for very specific networks (as in [MS00, LS00]) or are very preliminary as far as modeling is concerned [ZQ98, BBGK99, HV99].

In this paper, we generalize the WAP to the case where there are k fibers per link, and w wavelengths per fiber are available — this generalization is called the (k,w)-WAP. Two optimization problems naturally arise from the (k,w)-WAP: the problem of minimizing the number of wavelengths used, given k, and that of minimizing the number of fibers k if we are given w.

In the literature, the *efficiency* of a k-fiber network is measured in terms of k and the number of wavelengths w_k required by the (k,w)-WAP, for fixed k. For a set of lightpaths with load $\mathcal{L}$, w_k lies between $\frac{\mathcal{L}}{k}$ and $\frac{w_1}{k}$. The *efficiency* of the network is then defined as $\frac{\mathcal{L}/k}{w_k}$.

In order to build a general framework around the (k,w)-WAP, we propose a new tool for modeling conflicts arising in wavelength utilization in MWNs, based on hypergraphs. The *conflict hypergraph*, formally defined in [Riv01], is a generalization of the popular conflict graph, used for the WAP on single-fiber networks. We validate the concepts proposed in this work by considering both optimization problems (that of minimizing k with fixed c, and that of minimizing w with fixed c) in the Pan-American backbone network.

The main contributions of this work can be summarized as follows:

- We formally define the (k,w)-WAP for MWNs, where either the number of fibers, or the number of wavelengths per fiber can be optimized.
- Using this new hypergraph model, we build a bridge between coloring results for hypergraphs in the literature and the (k,w)-WAP.
- We analyze the complexity of the (k,w)-WAP in MWNs. In fact, we prove that minimizing the number of wavelengths is $\mathcal{NP}$-complete, even in the case where the number of fibers is fixed in advance, answering the open question with respect to the exact complexity of this problem. We also prove some other related results.
- We analyze the practical performances of two methodologies based on hypergraph coloring on existing backbone networks in the USA. The first relies on an integer programming formulation and the second consists on a heuristic based on a randomized approximation algorithm. We analyze the feasibility of solving real-world (k,w)-WAP with existing LP/IP solvers. The field is still open to new heuristics for hypergraph coloring.

The remainder of this paper is organized as follows. First, we present an overview of related work in Sect. 2. In Sect. 3, we present the problem formulation, recall the definition of the hypergraph model and prove that the (k,w)-WAP is $\mathcal{NP}$-complete, and presenting other results with respect to the complexity of the problems. In Sect. 4, we address the actual problem of designing a multifiber network, with respect to the optimization of either parameter. Sect. 5 discusses our prototypes and their performance evaluation. Finally, we conclude and present some future work in Sect. 6.

2 Related work

Motivated by the very large costs of deploying WDM networks, a large volume of research has targeted design issues on these networks in the past.

In single-fiber networks, it is usual to assume that two nodes are connected by one fiber of unlimited capacity (i.e. able to carry any number of wavelengths). Hence the $(1,w)$-WAP (formerly known simply as

WAP) is exactly the *path coloring problem* in standard graphs [CGK92], which has been proven equivalent to the general vertex coloring problem. Thus, there exists a fixed $\delta > 0$ such that no approximation within n^{δ} is possible unless $\mathcal{P} = \mathcal{NP}$ [Hoc97].

Therefore, a large amount of work concentrated on specific topologies and line networks, rings, trees, meshes, and so on. Specific communication patterns have also been studied like All-to-All and multicast.

The design of multifiber networks has recently been studied under different models and traffic assumptions [ZQ98, BBGK99, HV99, MS00, LS00]. For instance the $(1,w)-$WAP is $\mathcal{NP}$-complete on undirected stars but becomes polynomial with an efficiency of 1 if 2 fibers are available on each link [MS00, LS00].

Dynamic traffic — which means that lightpaths have to be established and released dynamically — has been studied in [ZQ98], where multifiber networks were shown to be more efficient than single-fiber networks with the same capacity† per link. Using multifiber links has also been shown to lead to performances equivalent to those provided by limited wavelength conversion.

In [BBGK99], an integer program and heuristics that solve the static problem are discussed. They consider path length constrained routing, wavelength assignment, wavelength conversion, and link failure restoration. The objective is to minimize the total number of fibers used in the network. Two meta-heuristic (simulated annealing and taboo-search) for MWNs design are proposed in [HV99]. Both papers show that adding fibers could improve the network efficiency.

Some theoretical properties of MWNs have been studied in [MS00, LS00]. For instance, it was proven that increasing the number of fibers per link often simplifies the optical routing problem: For all k and w, there exist a network and a set of communication requests such that exactly w wavelengths are necessary to solve the problem with k fibers per link while 1 wavelength is enough with $k+1$ fibers.

3 Problem formulation and complexity

In this section, we formally define the (k,w)-WAP, and recall the definition and complexity results of [Riv01] related to the conflict hypergraph. We then prove that the (k,w)-WAP is $\mathcal{NP}$-complete even in the case where k is fixed, and present a lower bound on the number of colors needed in a (k,c)-coloring of a (hyper)clique.

Definition 1 *The* conflict hypergraph $H = (V,E)$ *of the paths* $\mathcal{P}$ *in* $\mathcal{N}$ *is a hypergraph such that each vertex* $v \in V$ *corresponds uniquely to a path* $p \in \mathcal{P}$*, and such that for every link* $\ell \in L$*, there exists a hyperedge in* E *containing the vertices that correspond to all the paths going through* ℓ *(and these are the only hyperedges in* E*).* □

A vertex coloring of the conflict hypergraph induces a feasible wavelength assignment to the paths if and only if no hyperedge contains more than k vertices with the same color. This motivates the following definition of the (k,c)*-coloring*.

Definition 2 *Given a hypergraph* $H = (V,E)$ *and a set of colors* $\mathcal{C} = \{1\ldots c\}$*, a mapping* $f : V \to \mathcal{C}$ *is a* (k,c)-coloring *if and only if no hyperedge contains more than k vertices with the same color, that is,* $\forall e \in E,\ \forall q \in \mathcal{C},\ |\{v \in e : f(v) = q\}| \leq k.$ □

It is easy to see from Definitions 1 and 2, that there is a one-to-one correspondence between the (k,c)-colorings of the conflict hypergraph of $\mathcal{P}$ and the feasible wavelength assignments to these paths. Thus, the (k,c)-coloring problem is at least as difficult as the (k,w)-WAP. Actually, these problems are equivalent [Riv01]:

Theorem 1 *The* (k,c)*-coloring problem is polynomially equivalent to the* (k,w)*-*WAP *on* MWN*s.*

The (k,c)-coloring problem is clearly $\mathcal{NP}$-complete for general k, since it generalizes the graph coloring decision problem when $k = 1$. Therefore,

Corollary 1 *The* (k,w)*-*WAP *on a* MWN*s is* $\mathcal{NP}$*-complete for a general k.*

Moreover, we prove below that the problem remains difficult even when k is fixed.

† The capacity of a link is the sum of the capacities of each fiber in the link.

Theorem 2 *The (k,c)-coloring problem is $\mathcal{NP}$-complete for any fixed k.*

Proof: We reduce this problem to (standard) coloring on graphs. That is, given a graph G with n nodes and m edges, we have to answer "Can G be colored using c colors or less?". To prove that (k,c)-coloring is $\mathcal{NP}$-complete, we answer the graph coloring question by calling the (k,c)-coloring oracle. We can assume that $c < n$ because otherwise the answer is trivially yes.

We extend G into a hypergraph H in the following way. Let $K_{n,t}$ be a hypergraph with n nodes that contains all the possible hyperedges of rank t. We start by adding a $K_{ck,k+1}$ clique to H, which can trivially be (k,c)-colored. Fix one of the possible colorings of the clique. We will now make that coloring the only feasible one (up to permutations of the colors). For that, we add c new nodes to H, each with a different color pre-assigned and then all the possible $(k+1)$-hyperedges that do not join $k+1$ nodes of the same color. The coloring that we fixed is, of course, feasible for this structure by construction. If we vary the cardinality of nodes having a color, that is clearly infeasible because there is a clique included. If we permute the colors there will be a hyperedge preventing that to be feasible.

The construction above allows us to claim that we have $k+1$ nodes having each of the c colors. Now, returning to the original graph, for each edge we add c hyperedges as follows: for each color χ, include any of the $k-1$ nodes that have color χ in the structure and the two endpoints of the edge. Every one of these hyperedges means that the two nodes cannot be colored using the same color, which is what we need for graph coloring.

If we can (k,c)-color the hypergraph, then we can also color the graph with c colors. What remains to be seen is that the transformation is polynomial on the parameters. We added $c(k+1)$ nodes which is polynomial on the input. We added less than the maximum possible number of $(k+1)$-regular hyperedges, which is equal to

$$\binom{c(k+1)}{k+1} \leq \frac{c^{k+1}(k+1)^{k+1}}{(k+1)!},$$

which is certainly polynomial on c (k is fixed). Then we added mc more hyperedges for preventing color repetitions. Therefore, recalling that $c < n$, a bound for the total number of hyperedges added is $O(n^{k+1} + mn)$, which completes the proof. □

Corollary 2 *The (k,w)-WAP on a MWNs is $\mathcal{NP}$-complete for any fixed k.*

3.1 A lower bound

Extending the notion of cliques in graphs, we can give a lower bound on the number of colors needed in a (k,c)-coloring, by using (hyper)cliques, as follows. Recall that $K_{n,t}$ is a hypergraph with n nodes that contains all the possible hyperedges of rank t.

Lemma 1 *A (k,c)-coloring of $K_{n,t}$ is feasible if and only if*

$$c \geq \begin{cases} \lceil \frac{n}{k} \rceil & \text{if } t > k, \\ 1 & \text{otherwise}. \end{cases}$$

Proof Sketch: The case where $t \leq k$ is trivial. The main argument of the case where $t > k$ is counting how many times a color can be repeated. □

The lemma above bounds the number of colors required to color any hypergraph that contains $K_{n,t}$, yielding the following generalization of the fact that the chromatic number of a graph is larger than the size of its maximum clique (just make $t = 2$ and $k = 1$).

Corollary 3 *Let H be a hypergraph containing $K_{n,t}$. If H can be (k,c)-colored, with $k < t$, then $c \geq \lceil n/k \rceil$.*

4 Tools for designing MWNS

In this section, we will present two scenarios in the design of multifi ber networks. The equivalence between solving the WAP for $\mathcal{P}$ and computing (k,c)-colorings of H allows us to concentrate on the latter. For instance, the problems we consider are the problems of fi nding the minimum k (resp. c) such that there is a feasible (k,c)-coloring of H with c (resp. k) given. We address these two problems in Sect. 4.1 and 4.2, respectively.

4.1 Minimizing the number of fibers

We consider fi rst the problem of minimizing the number of fi bers when the number of colors is given. This problem can be formulated as a Minimax Integer Program [Sri96]. For instance, we defi ne $(0,1)$-integer variables x_{ij}, for all $i \in V$ and $1 \leq j \leq c$, such that $x_{ij} = 1$ if and only if node i is colored with color j and $x_{ij} = 0$ otherwise. The variable k is a common upper bound for the constraints defi ned by each hyperedge. The optimal number of fi bers can be found by solving the following IP.

Integer Program 1

$$
\begin{array}{rrcll}
\textit{minimize} & k & & & \textit{(minimize \# of fibers)} \\
\textit{s.t.} & \sum_{c} x_{ic} & = & 1 & \forall \textit{ node } i \\
 & \sum_{i \in H} x_{ic} & \leq & k & \forall \textit{ color } c, \forall \textit{ hyperedge } H \\
 & k \geq 0,\ x_{ic} & \in & \{0,1\} & \forall \textit{ color } c, \forall \textit{ node } i.
\end{array}
$$

Recently, Srinivasan showed that if the optimal solution of the LP relaxation is rounded randomly, with positive probability, a solution that is feasible and not too large can be encountered [Sri96]. A simple algorithm, discussed in [Lu98], can compute a solution that is not too far from the one proposed by Srinivasan. It is a simple randomized algorithm that takes c as input and computes a suitable k, for which it can assure, with high probability that it is not too far from the minimum possible k. Then, it proceeds with these three steps.

1. Color randomly all the nodes with $c/3$ colors.
2. Detect hyperedges whose constraints violate a (k,c)-coloring and re-color their nodes randomly with another set of $c/3$ colors.
3. Detect hyperedges whose constraints violate a (k,c)-coloring again, but now color them exhaustively with the last set of $c/3$ colors.

The values of c (given) and k (computed by the algorithm) must satisfy a certain constraint, which depends also on the maximum load $\mathcal{L}$ and on the maximum degree Δ. Indeed, for the minimum k that satisfi es these constraints, it is unlikely that we have many bad hyperedges after the second step, and thus the algorithm can be shown to run in polynomial time [Lu98]. As the k returned by the algorithm may be too large for practical purposes, we exploit this idea in Sect. 5 in order to defi ne a heuristic for the same problem. The heuristic proposed performed well in all the simulations considered, often outperforming Lu's algorithm.

4.2 Minimizing the number of wavelengths

Given the number of fi bers k, we now would like to minimize the number of colors c such that a valid (k,c)-coloring of the hypergraph exists. We present an Minimax IP formulation for this problem below. We defi ne a variable x_{ic} for each node and each color: $x_{ic} = 1$ if node i is colored with color c, and 0 otherwise. We have seen that the number of colors is bounded by $\lceil n/k \rceil$ (this bound is tight if the graph is a clique).

Integer Program 2

$$
\begin{array}{rrcll}
\min & \sum_c y_c & & & \textit{(minimize \# of colors)} \\
 & \sum_c x_{ic} & = & 1 & \forall \textit{ node } i \\
 & \sum_{i \in H} x_{ic} & \leq & k & \forall \textit{ color } c, \textit{hyperedge } H \\
 & x_{ic} & \leq & y_c & \forall \textit{ color } c, \textit{node } i \\
 & x_{ic}, y_c & \in & \{0,1\} & \forall \textit{ color } c, \textit{node } i
\end{array}
$$

There are $O(n^2)$ variables and $O(n^2m)$ constraints (we could reduce the number of constraints to $O(nm)$ if the solver generates cuts automatically).

The drawback of this IP formulation is that it is not symmetric and thus Branch-and-Bound will waste a lot of time iterating trough similar solutions [MT96]. The problem arises because after a variable is constrained by the algorithm, a permutation of them may still be feasible. This problem can be solved using automatic pruning techniques, as described in [Mar01].

5 Implementation and Performance Evaluation

To computationally evaluate the problems, we implemented the two integer programs and the approximation algorithm, described in Sect. 4. This allowed us to evaluate the tradeoff between the performance and the running time of the exact version and the approximation. As we implemented an approximation algorithm for the problem of minimizing k, and computing exact solutions is equivalent for both problems, we used that problem to compare the results. We also report our fi ndings in the experience of solving the problem of minimizing c.

We used instances based on several networks and will present the results we had using an American one which consists of 78 cities, interlinked by 102 arcs (see Fig. 1(a)).

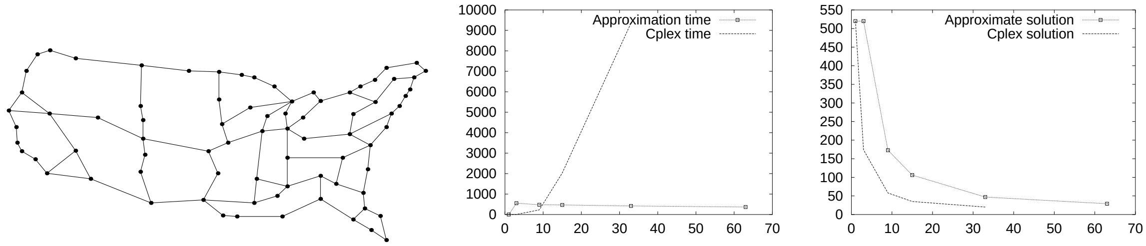
In order to generate a demand matrix, we used a gravitational model. Weights are associated to the cities. In order to basically represent the importance of every city in terms of traffi c: we made the weights proportional to the distance to 5 main population areas in the USA. Finally, the number of requests between every two cities was made proportional to the product of the two weights while keeping the outgoing number of requests from every city equal to the weight. Using different weights, we generated instances that were used for the benchmarks. We report on a relatively big instance with 2022 requests and a load of 520.

Routing was implemented through a *minimum cost disjoint path problem* for each origin-destination pair. For each origin and destination, we computed the shortest total distance of two disjoint paths linking them and distributed all the demand among the two paths. Routing the requests in such a way ensures that short paths are selected while maintaining two disjoint routes from each origin to each destination, which helps to improve the reliability.

When solving the exact version of the minimization of the number of fi bers, the solver found feasible solutions reasonably fast when restricting to small instances. Except for the biggest instances (the American network with many colors), the solver did not have diffi culties in proving optimality. It was expected, though, that when the instances grew bigger, the running time was going to degrade because the underlying problem is $\mathcal{NP}$-hard. Nevertheless, this does not seem to be an issue for the instances generated from real-world networks.

Results

Optimal and approximate computations of the number of fi bers needed, as a function of the number of colors available, are depicted in Fig. 1(b). On the left hand side, one can see that the running time of the approximate algorithm is almost a constant compared to the running time of the IP solved by CPLEX. This is of great interest when compared to right hand side, which shows that the approximate number of fi bers is close to the optimal one up to a factor lying between 2 and 3.



(a) American network

(b) Running time and number of fibers

Fig. 1: Optimal and approximate computations on American network

It is important to notice that in these instances, and often with real-world networks, the number of colors equals its lower bound, that is the load of the network divided by the number of fibers. It is known that pathological examples can be constructed although they do not usually appear in real instances.

The biggest dependency of the running time of the Integer Program 2, that optimizes the number of colors, is on the number of variables representing the colors. Initially, we used as many colors as the number of requests, because that is an upper bound. Obviously, this did not scale well when the size of the instances increased to real-world problems. Instead, we performed a binary search for the upper bound of the colors. We relied on the observation that when the bound is too small, the IP solver returns quickly that no feasible solution exists. On the other hand, when the upper bound is not tight, it takes too much time to solve the first node of the Branch-and-Bound tree because there are too many variables. With this strategy, we got IPs of the correct size that could be handled by the solver. As expected though, due to the symmetry in the formulation (the labeling of the colors can be permuted without altering the solution), the enumeration of the nodes of the Branch-and-Bound tree could not be completed in general. In any case, we had a proof of optimality. Indeed, we found that when using one less color, the LP relaxation of the problem was already not feasible. Therefore, showing a feasible solution with that many colors was enough. Indeed, it would be interesting to characterize the integrality gap of that problem.

6 Conclusion

In this paper, we have proposed a framework to model the WAP in MWNs, reducing it to a coloring problem on hypergraphs. Practically, the coloring problem appeared to be tractable since its straightforward IP formulation gave optimal solutions reasonably fast. On the other hand, in all our real-world instances we found $w_k = \lceil \mathcal{L}/k \rceil$. Hence, the efficiency gain due to multifiber flexibility was not observed. However, since ad-hoc constructions prove that this gain can be enormous, practical instances could still be found where such a gain appear. Furthermore, the heuristic that we implemented turned out to be very fast, but did not perform very well, despite the provably good asymptotic properties of the underlying randomized approximation algorithm. Therefore, we are currently working on the design of other heuristics for hypergraph coloring.

Another interesting research direction is to address the design of MWNs in the case where the routing is not fixed in advance. In such a case the lightpaths are not given, and one needs to design both the routing and the wavelength assignment at once. We believe that, as soon as k is large enough, this problem can be practically solved to optimality.

References

[BBG+97] B. Beauquier, J-C. Bermond, L. Gargano, P. Hell, S. Pérennes, and U. Vaccar. Graph problems arising from wavelength-routing in all-optical networks. In *2nd IEEE Workshop on Optics and Computer Science, part of IPPS'97*, Genova,Switzerland, April 1997. IEEE Press.

[BBGK99] S. Baroni, P. Bayvel, R. J. Gibbens, and S. K. Korothy. Analysis and design of resilient multifiber wavelength-routed optical transport networks. *Journal of Lightwave Technology*, 17(5):743–754, May 1999.

[CFK+01] I. Caragiannis, A. Ferreira, C. Kaklamanis, S. Pérennes, and H. Rivano. Fractional path coloring with applications to WDM networks. In *28th International Colloquium on Automata, Languages and Programming (ICALP'01)*, Lecture Notes in Computer Science, pages 732–743, Crete, Greece, July 2001. Springer-Verlag.

[CGK92] I. Chlamtac, A. Ganz, and G. Karmi. Lightpath communications: An approach to high bandwidth optical WAN's. *IEEE Transactions on Communications*, 40(7):1171–1182, July 1992.

[Hoc97] D. S. Hochbaum, editor. *Approximation Algorithms for NP-Hard Problems*. PWS Publishing Compagny, 1997.

[HV99] E. Hyytiä and J. Virtamo. Wavelength assignment in multifibre in WDM-networks. Technical Report COST257TD(99)04, Helsinki University of Technology, February 1999.

[LS00] G. Li and R. Simha. On the Wavelength Assignement Problem in Multifiber WDM Star and Ring Networks. *IEEE Infocom*, 3:1771–1780, 2000.

[Lu98] C. Lu. Deterministic hypergraph coloring and its applications. In *Proceedings of the 2nd International Workshop on Randomization and Approximation Techniques in Computer Science*, volume 1518 of *Lecture Notes in Computer Science*, pages 35–46. Springer-Verlag, October 1998.

[Mar01] F. Margot. Pruning by isomorphism in Branch-and-Cut. In K. Aardal and B. Gerards, editors, *Lecture Notes in Computer Science 2081: 8th IPCO Conference*, pages 304–317. Springer, June 2001.

[MS00] L. Margara and J. Simon. Wavelength assignment problem on all-optical networks with k fibres per link. In *27rd International Colloquium on Automata, Languages and Programming*, pages 768–779, 2000.

[MT96] A. Mehrotra and M. A. Trick. A column generation approach for graph coloring. *INFORMS Journal on Computing*, 8(4):344–354, 1996.

[Riv01] H. Rivano. Planification de réseaux optiques wdm k-fibres. In *AlgoTel'01 - 3èmes Rencontres Françaises sur les Aspects Algorithmiques des Télécommunications*, pages 41–46, Saint Jean de Luz, France, mai 2001. INRIA.

[RS95] R. Ramaswami and K. N. Sivarajan. Routing and wavelength assignment in all-optical networks. *IEEE/ACM Transactions on Networking*, 3(5):489–500, 1995.

[RS98] R. Ramaswami and K. N. Sivarajan. *Optical Networks: a practical perspective.* Morgan-Kaufman, 1998.

[Sri96] A. Srinivasan. An extension of the Lovász Local Lemma, and its applications to integer programming. In *ACM-SIAM Symposium on Discrete Algorithms*, pages 6–15, 1996.

[ZQ98] X. Zhang and C. Qiao. Wavelength assignment for dynamic traffic in multi-fiber WDM networks. In *ICCCN'98*, pages 479–585, 1998.