

Konrad-Zuse-Zentrum
für Informationstechnik Berlin

ZIB

Takustraße 7
D-14195 Berlin-Dahlem
Germany

ARIE M.C.A. KOSTER
ADRIAN ZYMOLKA

Provably Good Solutions for Wavelength Assignment in Optical Networks

Provably Good Solutions for Wavelength Assignment in Optical Networks

Arie M. C. A. Koster^{1,2}

Adrian Zymolka¹

Abstract

In this paper, we study the minimum converter wavelength assignment problem in optical networks. To benchmark the quality of solutions obtained by heuristics, we derive an integer programming formulation by generalizing the formulation of Mehrotra and Trick [12] for the vertex coloring problem. To handle the exponential number of variables, we propose a column generation approach. Computational experiments show that the value of the linear relaxation states a good lower bound and can often prove optimality of the best solution generated heuristically.

1 Introduction

The cost-efficient design of transparent optical networks comprises three main tasks: dimensioning, routing, and wavelength assignment. In the dimensioning, hardware devices have to be placed in the given physical topology of the network. The installation of fibers and Wavelength Division Multiplexing (WDM) systems at the links provides transmission capacity, while Optical Cross-Connects (OXC) in the nodes offer switching capacity. For every pair of nodes, a demand for a number of lightpaths to be established is specified. The lightpaths have to be routed such that their capacity consumption does not exceed the transmission and switching capacities of the installed equipment. As WDM systems are applied, each lightpath has also to be assigned an available wavelength of operation on each passed link. Along a lightpath, exchanging the operated wavelength on two consecutive links requires in the intermediate node to install a wavelength converter, which can translate any wavelength to any other for a single optical channel. The cost of the resulting network composes of the installation costs for fibers, WDM systems, OXC, and wavelength converters. The optical network design problem is to determine a hardware and lightpath configuration at minimum total cost.

The focus on optical network design in the literature has been on routing and wavelength assignment, while dimensioning was considered less frequently. To incorporate all three

¹Zuse Institute Berlin (ZIB), Takustraße 7, D-14195 Berlin, Germany. Email: {koster,zymolka}@zib.de

²Partially supported by the DFG research group "Algorithms, Structure, Randomness" (Grant number GR 883/9-3, GR 883/9-4)

Keywords: wavelength assignment, integer programming, column generation

Mathematics Subject Classification (2000): 90C10, 90B18, 90C90

issues, an alternative approach in [13] proposes the decomposition into a dimensioning and routing subproblem (without distinguishing wavelengths), and a subsequent wavelength assignment subproblem. This decomposition is beneficial in several regards. The integrated solution of dimensioning and routing as highly correlated issues from a cost-oriented perspective is, since wavelengths are neglected, very similar to non-optical network design and thus allows to employ known sophisticated methods. Moreover, optical network design combines two hard mathematical problems, an integer multicommodity flow problem (routing) and a generalized coloring problem (wavelength assignment), which are separated by the decomposition.

In this paper, we investigate the wavelength assignment subproblem. Given a properly dimensioned network and a routing of all lightpaths, the task is to assign the available wavelengths to the links of each lightpath such that the total number of required wavelength converters is minimized. Both constructive and improvement heuristics have been described and studied in [10]. For the iterative algorithms, we propose a problem extraction method to enhance the performance. Although the generated solutions typically contain a (relatively) small number of converters, the heuristics leave open whether there are better assignments. To benchmark the solution quality, it is necessary to determine the optimum or, second best, approximate the number of unavoidable converters by a lower bound. Starting from a simplified problem, we extend a successful exact approach for vertex coloring by Mehrotra and Trick [12] step-wise to a suitable integer programming formulation for the wavelength assignment problem, covering the general case. We also present a column generation algorithm to solve the linear relaxation whose value provides a lower bound for the minimum converter number. A computational study reveals that the derived formulation of the problem in fact yields good lower bounds. Using these benchmarks, we can also show that the heuristics generate provably good solutions.

The paper is structured as follows. We formally introduce the minimum converter wavelength assignment problem in Section 2. As exact solution approach, we develop the integer programming formulation in Section 3 and describe in Section 4 the associated column generation method to solve the linear relaxation. Next, we briefly review the heuristics in Section 5 and present the extraction extension. In Section 6, we report on a computational study to evaluate the quality of lower bounds and heuristic solutions. Concluding remarks in Section 7 close the paper.

2 Problem description

We consider an already dimensioned optical network and a given set of lightpaths. Each lightpath is routed on a link path in the physical topology of the network (however no specific fiber on multi-fiber links is predefined). The network is dimensioned in such a way that the number of channels consumed by all lightpaths on a link does not exceed the channel capacity of the installed fibers and WDM systems. Likewise, the switching capacity provided by optical cross-connects in the nodes is large enough to handle all traversing lightpaths. We assume that both lightpaths and all capacities are bidirectional, i.e., fibers and WDM systems are always installed in pairs, one for each direction, and a lightpath provides a virtual connection in both directions. The installed fibers and WDM systems on

each link specify the available wavelengths for operating the lightpaths. For any exchange of the operated wavelength along a lightpath, a wavelength converter has to be installed in the intermediate node. Now, the Minimum Converter Wavelength Assignment Problem (MCWAP) is the task to assign the available wavelengths to each link of each lightpath such that the total number of required wavelength converters is minimized.

From a graph theoretical point of view, MCWAP can be described as follows. Let $\mathcal{N} = (N, L)$ be an undirected graph with N representing the nodes and L the links of the physical topology. The set Λ denotes the spectrum of all wavelengths. At each link $\ell \in L$, there are κ_ℓ fibers, whereas κ_ℓ^λ defines the number of times wavelength $\lambda \in \Lambda$ is available by the installed fibers and WDM systems. Note that different WDM systems can result in different values κ_ℓ^λ for the diverse wavelengths. In total $K_\ell = \sum_{\lambda \in \Lambda} \kappa_\ell^\lambda$ optical channels are available on link $\ell \in L$.

As multiple lightpaths can be routed along the same link path, we consider them cumulative. For a path p in \mathcal{N} , we denote with d_p the number of lightpaths routed along this path. All lightpaths that have to be assigned wavelengths are gathered in a multi-set \mathcal{P} where each path p is contained d_p times (a multi-set is a set with allowed element repetition). So, in total $P = |\mathcal{P}|$ lightpaths are considered. To differentiate the paths without multiplicity, the set \mathcal{P}_1 contains each path just once. Let $N(p) \subset N$ denote the intermediate nodes along the path p (the start and end node of the path are not included in $N(p)$). Similarly, $L(p) \subset L$ denotes the links of the path p , and all paths that share link $\ell \in L$ are subsumed in the set $\mathcal{P}_\ell \subset \mathcal{P}_1$.

The assumption that the channel capacities are sufficient to establish the given set of lightpaths implies that

$$\sum_{p \in \mathcal{P}_\ell} d_p \leq K_\ell \quad \forall \ell \in L \quad (1)$$

hold. Then, the existence of a feasible wavelength assignment is guaranteed by the availability of wavelength converters. Notice that by (1), lightpaths containing only a single link can always be assigned a wavelength independently from the other lightpaths (and never require converters). Therefore, such lightpaths are left out in our further considerations.

3 Mathematical formulation

In [10, 13], heuristics for the minimum converter wavelength assignment problem have been developed. In case the resulting wavelength assignment is without conversion, we can conclude that the solution is optimal. Otherwise, the best generated solution typically contains a (relatively) low number of converters, but it remains unclear whether there are better assignments with less (or even without) converters, i.e., no quality guarantee can be given. To benchmark wavelength assignments by their converter number, we have to determine the number of unavoidable wavelength converters as lower bound. The most promising approach for such a task is integer programming.

A natural modelling of MCWAP is to introduce a variable for every combination of a lightpath, each of its links, and each wavelength available on that link. By use of these

assignment variables, it is straightforward to formulate the associated integer program, which we refer to as assignment formulation. Unfortunately, such a formulation has several disadvantages, the most important one being the degeneracy of the solutions resulting from the symmetry in the wavelengths. Suppose two wavelengths $\lambda_1 \neq \lambda_2$ satisfy $\kappa_\ell^{\lambda_1} = \kappa_\ell^{\lambda_2}$ for all links $\ell \in L$. Given any wavelength assignment, the exchange of these two wavelengths results in a different solution with exactly the same number of converters. Typically, there are many such equivalent wavelengths, and thus an enormous amount of equivalent solutions exists. This not only holds for integer solutions, but also for every fractional one. For problems with a similar characteristic, like vertex coloring or frequency assignment [1], solution algorithms like linear programming based branch-and-bound on the assignment formulation have been shown to be computational intractable due to the inherent symmetry.

Since wavelength assignment in a very restrictive setting can be modeled as a vertex coloring problem, alternative approaches for the latter problem are of special interest. Mehrotra and Trick [12] have developed a so-called column generation formulation for vertex coloring which overcomes the color symmetry. In the following, we generalize this formulation step by step to MCWAP.

3.1 Single fiber links

Until further notice, we at first assume uniform WDM systems, i.e., all WDM systems provide the same set of wavelengths, Λ . In case $\kappa_\ell = 1$ for all links $\ell \in L$, the question whether or not a wavelength assignment without conversion exists then reduces to the vertex coloring problem on the so-called *path conflict graph* $G_{\mathcal{P}}$: Introduce a vertex v_p for every lightpath $p \in \mathcal{P}$ and connect two vertices by an edge if the corresponding lightpaths share at least one link. Any coloring of the vertices such that no two adjacent vertices have the same color corresponds to an assignment of wavelengths to the lightpaths. The minimum number of colors (=wavelengths) needed is denoted by the chromatic number, $\chi(G_{\mathcal{P}})$. So, a wavelength assignment without conversion exists if and only if $\chi(G_{\mathcal{P}}) \leq |\Lambda|$.

If all colors are available for all vertices of $G_{\mathcal{P}}$, the specific color of a vertex is unimportant. The only thing that matters is that adjacent vertices receive different colors. Stated otherwise, non-adjacent vertices can receive the same color. A *stable set* ϕ is a set of pairwise non-adjacent vertices and thus has the property that all vertices in ϕ can be colored with the same color. So, a coloring consists of a covering of all vertices by stable sets representing the colors (without specifying them). The less stable sets are needed, the lower the chromatic number is.

Let Φ denote the collection of all stable sets in $G_{\mathcal{P}}$. The integer programming formulation by Mehrotra and Trick [12] introduces a binary variable x_ϕ for every stable set $\phi \in \Phi$ to indicate whether or not this stable set is selected in the covering of the vertices. Now, the vertex coloring problem for $G_{\mathcal{P}}$ can be formulated as follows:

$$\chi(G_{\mathcal{P}}) = \min \sum_{\phi \in \Phi} x_\phi \tag{2}$$

$$\text{s.t.} \quad \sum_{\phi \in \Phi: v_p \in \phi} x_\phi = 1 \quad \forall p \in \mathcal{P} \tag{3}$$

$$x_\phi \in \{0, 1\} \tag{4}$$

The major advantage of this formulation in comparison to the assignment formulation is the fact that no specific colors are used anymore. In fact, all colorings that are equal up to a permutation of the colors are represented by a single solution in this formulation.

Mehrotra and Trick [12] proved that the lower bound on the number of colors provided by the linear relaxation of (2)–(4) is at least as good as the linear relaxation bound of the assignment formulation. In many cases it is indeed better.

A drawback of the formulation (2)–(4) is the number of variables which is extremely large. We have one variable for each stable set, where the number of stable sets is in general exponential in $|V|$. For non-trivial graphs, this number is too large to handle the variables explicitly. To solve the linear relaxation of (2)–(4), the technique of column generation [4, 5] can be applied: A small number of the variables is considered explicitly within a restricted program. After the linear relaxation of the restricted problem is solved, the non-explicitly handled variables are searched for those that can improve the relaxation. If such a variable exists, it is added to the restricted program, and the relaxation is solved again. If it can be proved that no such variable exists, the linear programming relaxation is solved to optimality with respect to all variables.

Column generation can be combined with branch-and-bound to solve not only the linear relaxation, but the integer problem to optimality. This method is known as *branch-and-price*. In [12], a branch-and-price method has been applied to (2)–(4) for graphs with up to several hundreds of vertices, a result that is out of the question for the assignment formulation due to its degeneracy.

3.2 Aggregating lightpaths

Given a routing of the lightpaths, it is likely that several lightpaths are routed along the same physical path $p \in \mathcal{P}_1$ through the network (i.e., $d_p > 1$). Two such lightpaths are obviously in conflict with the same set of other lightpaths. In particular, they are in conflict with each other. This means that the corresponding vertices have the same neighbors in the path conflict graph (neglecting the vertices itself) and that all parallel lightpaths are mutually adjacent, i.e., they form a clique. It is well known that in such a case we can contract all these d_p vertices to a single vertex provided that we color the new vertex with d_p different colors. In this way, the size of the path conflict graph can be reduced substantially.

For the stable set formulation (2)–(4), a contraction of d_p vertices corresponding to parallel lightpaths to a single vertex implies that the new vertex has to be covered by d_p stable sets. This simply changes constraints (3) to have a right hand side of d_p for each unique path $p \in \mathcal{P}_1$. Moreover, the variables are not longer binary but integer, since one stable set can be used for multiple colors (as long as all vertices in the stable set need more than one color). Summarized, the formulation now reads:

$$\chi(G_{\mathcal{P}}) = \min \sum_{\phi \in \Phi} x_\phi \tag{5}$$

$$\text{s.t.} \quad \sum_{\phi \in \Phi: v_p \in \phi} x_\phi = d_p \quad \forall p \in \mathcal{P}_1 \quad (6)$$

$$x_\phi \in \mathbb{Z}_0^+ \quad (7)$$

3.3 Parallel fibers

If more than $|\Lambda|$ lightpaths have to be established on the same link, multiple fibers (and WDM systems) have to be installed. One way to model parallel fibers is to include a separate link in the network \mathcal{N} for every fiber. If the fibers on which a lightpath is established are specified beforehand, the path conflict graph can be used again. However, during the dimensioning and lightpath routing, parallel links cannot be discriminated with respect to the network cost. At first during the wavelength assignment it becomes important whether some lightpaths are established on the same or on different fibers. A preliminary decision which fiber is used by which lightpath therefore restrains the space of low cost wavelength assignments unnecessarily. Consider the network in Fig. 1 where each pair of the nodes A, B, and C is connected by two lightpaths. In case we pre-allocate the dashed lightpaths to the first fiber on both links and the solid lightpaths to the second fiber on both links, two wavelength converters are needed. By allocating both lightpaths from A to B on the first fiber on both links, both lightpaths from A to C on the second fiber on both links, and both lightpaths from B to C on the second fiber between B and D and on the first fiber between D and C, no wavelength converters are necessary at all. This example makes clear that a dedication of the lightpaths to distinct fibers can prevent better wavelength assignments and thus is not a good idea.

Alternatively, we model all parallel fibers by a single link in the network and do not specify beforehand which fiber is used by which lightpath. If κ_ℓ fibers with uniform WDM systems are installed on link $\ell \in L$, then κ_ℓ lightpaths that share link ℓ can be assigned the same wavelength without being in conflict, they simply use different fibers. This extends the binary relation between the lightpaths to an integer one which cannot be represented by the edges of a simple path conflict graph anymore. In [6] a path conflict *hypergraph* is proposed to model these relations.

For our integer programming formulation, such a hypergraph is of limited interest. A slight extension of formulation (5)–(7) still models the problem, where only the mathematical nature of the sets ϕ (and by this Φ) is different. Instead of ϕ being a stable set in the path conflict graph, ϕ becomes a so-called *path packing*. A path packing is defined by a network $\mathcal{N} = (N, L)$, capacities κ_ℓ for all $\ell \in L$, and a set of paths \mathcal{P}_1 . If the multiplicity of a path

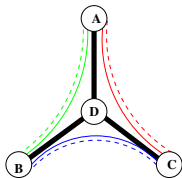


Figure 1: Wavelength assignment problem with two fibers on each link and two wavelengths per fiber

$p \in \mathcal{P}_1$ in a multi-set ϕ is denoted by t_ϕ^p , a feasible path packing is a multi-set $\phi \subset \mathcal{P}$ of paths such that

$$\sum_{p \in \phi: \ell \in L(p)} t_\phi^p \leq \kappa_\ell \quad \forall \ell \in L$$

hold, i.e., all paths in a feasible path packing ϕ can be assigned the same wavelength without conflict.

Let now Φ denote the set of all feasible path packings. To account for the path multiplicities in a path packing within the covering, we have to replace conditions (6) by

$$\sum_{\phi \in \Phi: v_p \in \phi} t_\phi^p x_\phi = d_p \quad \forall p \in \mathcal{P}_1 \quad (8)$$

Then the problem whether or not a conversion-free wavelength assignment exists can be solved by the program (5), (8), (7). If the optimum is less than or equal to $|\Lambda|$, a conflict-free assignment without converters is found, otherwise no such assignment exists.

3.4 Wavelength converters

So far, we only can answer the question whether a conflict-free assignment without conversion exists. In case no such an assignment exists, we need to install wavelength converters for the establishment of all requested lightpaths. A better objective would therefore be to minimize the number of converters needed. If the optimum is zero, we found the same answer as before, in all other cases we get an answer with added value, namely where to place converters.

The objective to minimize the number of converters involves a more substantial reformulation of (5), (8), (7). Recall that a solution for MCWAP consists of an assignment of wavelengths to the links of the lightpaths in such a way that each lightpath is assigned wavelengths on all its links and that no wavelength is assigned more than its availability on the link. Given such an assignment, the number of wavelength converters can be calculated. Looking at a lightpath that needs a converter in the solution, the wavelengths are assigned to one or more consecutive links along the path. The consecutive links with the same wavelength can be gathered to a subpath. In this view, a lightpath consists of a number of subpaths (possibly only one) that cover the path.

All subpaths that are assigned the same wavelength within the network is again a packing (of subpaths). Since we have in total $|\Lambda|$ wavelengths available, an assignment exists of $|\Lambda|$ subpath packings. The number of converters needed for a lightpath is exactly the number of subpaths to cover the lightpath minus one. Hence, minimizing the total number of converters needed is equivalent to minimizing the total number of subpaths involved in the $|\Lambda|$ subpath packings.

The number of times a subpath s can be contained in a subpath packing is restricted from two sides. On the one hand, all subpaths that share a link cannot be taken more than κ_ℓ times. On the other hand, the number of times a subpath can be used is limited by the

number of lightpaths it is part of. Formally, for each $p \in \mathcal{P}$, let \mathcal{S}_p denote the set of all subpaths s of p . Note that $|\mathcal{S}_p| = \frac{1}{2}|L(p)|(|L(p)| + 1)$. Let $\mathcal{S} = \cup_{p \in \mathcal{P}} \mathcal{S}_p$ denote the set of all subpaths. Moreover, let $\mathcal{P}_s \subset \mathcal{P}$ denote the multi-set of all lightpaths containing s as a subpath. A feasible subpath packing is now defined as a multi-set $\phi \subset \mathcal{S}$ of subpaths such that

$$\sum_{s \in \phi: \ell \in L(s)} t_\phi^s \leq \kappa_\ell \quad \forall \ell \in L$$

and

$$t_\phi^s \leq |\mathcal{P}_s| \quad \forall s \in \mathcal{S}$$

hold. Let Φ denote this time the set of all feasible subpath packings. Given $|\Lambda|$ feasible subpath packings, it is still unclear whether these build a complete wavelength assignment. Each lightpath has to be covered by subpaths on all its links. As a subpath $s \in \mathcal{S}$ can be used to cover all lightpaths in \mathcal{P}_s , we have to specify how many times a part of the path p is covered by subpath s . We denote this number by the integer variable y_p^s .

Now, we are ready to formulate MCWAP with uniform wavelength spectra as integer program:

$$\min \sum_{p \in \mathcal{P}_1} \sum_{s \in \mathcal{S}_p} y_p^s \quad \left[- \sum_{p \in \mathcal{P}_1} d_p \right] \quad (9)$$

$$\text{s.t.} \quad \sum_{s \in \mathcal{S}_p: \ell \in L(s)} y_p^s = d_p \quad \forall p \in \mathcal{P}_1, \ell \in L(p) \quad (10)$$

$$\sum_{\phi \in \Phi} t_\phi^s x_\phi = \sum_{p \in \mathcal{P}_1: s \in \mathcal{S}_p} y_p^s \quad \forall s \in \mathcal{S} \quad (11)$$

$$\sum_{\phi \in \Phi} x_\phi \leq |\Lambda| \quad (12)$$

$$y_p^s, x_\phi \in \mathbb{Z}_0^+ \quad (13)$$

As explained before, the total number of converters is given by the total number of used subpaths minus one for each lightpath, so minus the total number of lightpaths, which is expressed by (9). Note that the lightpath number is constant (and therefore put in brackets in (9)). At every link $\ell \in L$, the lightpath-multiplicity for each path p has to be satisfied, which is modeled by constraints (10). Constraints (11) state that all covering subpaths $s \in \mathcal{S}$ are provided the required number of times by the selected subpath packings. Finally, constraint (12) restricts the number of subpath packings to the size of the available spectrum Λ , and constraints (13) guarantee all variables to be integer valued.

Although we need substantially more variables to formulate MCWAP compared to the vertex coloring formulation (2)–(4), the advantages of that formulation carry over to this one. In (9)–(13), we do not consider specific wavelengths, and thus assignments that are

equivalent up to a permutation of the wavelengths are all represented by a single solution. Moreover, the linear relaxation provides a lower bound on the number of converters that is always as good as the one by the straightforward assignment formulation. In fact, where the bound by the linear relaxation of the assignment formulation is always zero (cf. [9]), the bound by the linear relaxation of the *subpath packing formulation* (9)–(13) can be positive, cf. Fig. 2. If only two wavelengths are available, we need one converter in the central node. The linear relaxation of (9)–(13) has value four (minus three as total number of lightpaths), for example by setting the variables of the two subpath packings in Fig. 2(b) to one, and thus it says that at least one converter is needed.

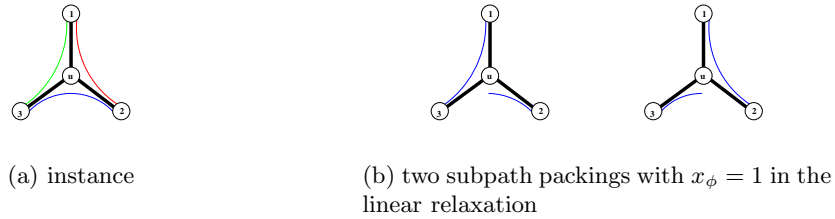


Figure 2: Star network with positive linear relaxation value

3.5 Non-uniform wavelength spectra

So far, we have discussed the case that all wavelength spectra are uniform. In practice however it can be the case that different types of WDM systems are used, providing different subsets of the spectrum Λ . As a consequence, not every wavelength is the same number of times available on a link. As a consequence, we have to refine the definition of a feasible subpath packing. A subpath packing is said to be feasible for wavelength $\lambda \in \Lambda$ if

$$\sum_{s \in \phi: \ell \in L(s)} t_\phi^s \leq \kappa_\ell^\lambda \quad \forall \ell \in L$$

and

$$t_\phi^s \leq |\mathcal{P}_s| \quad \forall s \in \mathcal{S}$$

hold. Let Φ_λ denote all λ -feasible subpath packings and let $\Phi = \cup_{\lambda \in \Lambda} \Phi_\lambda$. For each $\lambda \in \Lambda$, we now have to select one $\phi \in \Phi_\lambda$. Hence, constraint (12) are replaced by the $|\Lambda|$ constraints

$$\sum_{\phi \in \Phi_\lambda} x_\phi \leq 1 \quad \forall \lambda \in \Lambda \quad (14)$$

In principle, (14) diminishes the advantages of the subpath packing formulation, since wavelength-specific variables are introduced again. However, typically only a small number of different types of WDM systems are considered, resulting in a number of subsets of the spectrum that contain exchangeable wavelengths. Consider for example two WDM systems,

one with 40 wavelengths and one with only 20 out of the 40 in total. At every link ℓ only two different κ_ℓ^λ values can occur: one for the wavelengths that occur in both systems and one for the wavelengths that occur in the large system only. In general, the spectrum Λ can be partitioned into k subsets $\Lambda_1, \dots, \Lambda_k$ with $\kappa_\ell^{\lambda_1} = \kappa_\ell^{\lambda_2}$ for all $\ell \in L$ if $\lambda_1, \lambda_2 \in \Lambda_i$ for some $i = 1, \dots, k$. Instead of (14), only the k constraints

$$\sum_{\phi \in \Phi_\lambda} x_\phi \leq |\Lambda_i| \quad i = 1, \dots, k \quad (15)$$

replace constraint (12). If k is small, then the advantage of not specifying particular wavelengths is still kept. Notice that, in the special case that every link consists of a single fiber, the problem corresponds to a list-coloring problem (cf. [8]).

4 Column Generation Algorithm

In this paper, we would like to solve the linear programming relaxation of the subpath packing formulation (9)–(13) to obtain a lower bound on the number of needed converters for a conflict-free wavelength assignment. The number of variables and thus columns of the formulation (9)–(13) complicates such a computation, since it is tremendously large: the y -variables plus a variable for every subpath packing. The program size can be reduced by relaxing constraints (11) to

$$\sum_{\phi \in \Phi} t_\phi^s x_\phi \geq \sum_{p \in \mathcal{P}_1: s \in \mathcal{S}_p} y_p^s \quad \forall s \in \mathcal{S} \quad (16)$$

So, we allow the number of times a subpath is provided in the selected subpath packings to be larger or equal to the desired number. By the spectrum constraint (12), the number of selected subpath packings will not increase. This relaxation provides the advantage to restrict the set of subpath packings to those that are maximal. That is, for every subpath packing $\phi \in \Phi$, it holds that there is no $\phi' \in \Phi$ with $t_{\phi'}^s \geq t_\phi^s$ for all $s \in \mathcal{S}$ and $t_{\phi'}^s > t_\phi^s$ for at least one s .

Although, in theory, the replacement of (11) by (16) reduces the number of columns substantially, this number is still too large to be explicitly considered in practice. Therefore, like in Mehrotra and Trick [12], we propose a column generation approach, where only a subset of the columns are stored explicitly in a restricted program. As explained in Section 3.1, after computation of the linear relaxation for this restricted program, other profitable columns are searched for. Due to the exponential number of the implicitly handled columns, an enumeration of these columns is not possible. Therefore, we formulate a so-called pricing problem to find the most profitable column to add to the restricted program. Such a column is selected based on the values of the dual variables, like in the simplex method. In case no profitable columns can be generated anymore, the linear relaxation including all columns has been solved (cf. [4, 5] for further details).

For MCWAP, we apply column generation for the x_ϕ variables, whereas all y_p^s variables are taken into account explicitly. Let $\bar{\Phi} \subset \Phi$ denote the actual subset of subpath packings

included in the restricted program. To formulate the pricing problem for MCWAP, we introduce the dual variables π_ℓ^p , π^s , and π^Λ for respectively the constraints (10), (11), and (12). From linear programming, we know that a primal-dual pair $((x, y), \pi)$ is optimal for the linear programming relaxation of (9)–(13), whenever $c - A^T \pi \leq 0$, with c the primal objective function, and A the coefficient matrix. For a subpath packing $\phi \in \Phi$, we have $c_\phi = 0$, and the coefficients of A corresponding to (10) equal zero as well. So, the optimality condition reads

$$-\sum_{s \in \mathcal{S}} t_\phi^s \pi^s \leq \pi^\Lambda \quad (17)$$

Note that, by (16), $\pi^s \leq 0$, whereas $\pi^\Lambda \geq 0$. By optimality of $((x, y), \pi)$, (17) holds for every $\phi \in \bar{\Phi}$. To verify whether (17) holds for all $\phi \in \Phi$, we search for a subpath packing ϕ that maximizes the left hand side of (17). If the maximum is less than or equal to π^Λ then no improving columns exist, and the linear relaxation is solved optimally, i.e., all still not contained columns are proven to have value 0 in the optimal solution. Otherwise, a subpath packing ϕ that violates (17) is found and can be added to improve the restricted program.

Maximizing $-\sum_{s \in \mathcal{S}} t_\phi^s \pi^s$ can be formulated as an optimization problem as well. We introduce the integer variables t_s for all $s \in \mathcal{S}$ representing the multiplicity function of a subpath packing. Then the pricing problem reads

$$z = \max \sum_{s \in \mathcal{S}} -\pi^s t_s \quad (18)$$

$$\text{s.t.} \quad \sum_{s \in \mathcal{S}: \ell \in L(p)} t_s \leq \kappa_\ell \quad \forall \ell \in L \quad (19)$$

$$t_s \in \mathbb{Z}_0^+ \quad (20)$$

In case $\kappa_\ell = 1$ for all $\ell \in L$, the pricing problem reduces to a maximum weighted set packing (or stable set) problem which is well studied (cf. Borndörfer [3]). Moreover, instead of solving the pricing problem optimally in every iteration, it suffices to find a solution with value larger than π^Λ . Only if such a solution cannot be found heuristically, the integer program (18)–(20) must be solved to optimality. However, our experience shows that this is most often only required to prove that no further improving column exists. Having optimally solved the linear relaxation of (9)–(13), the objective value provides a lower bound on the number of unavoidable converters. As we will see in Section 6, the formulation yields often a non-zero bound, serving best to benchmark the wavelength assignments generated by our heuristics.

5 Heuristics

In [10], MCWAP has been proven to be \mathcal{NP} -hard, even on networks as simple as star graphs with single fiber links. As a consequence, we cannot expect to find efficient solution algorithms in the general case. Therefore, a couple of heuristics for MCWAP has been proposed and evaluated in [10]. All of these heuristics process the lightpaths sequentially.

In each step, the (remaining) available wavelengths are assigned to the actual lightpath such that, beginning with the first (unassigned) link, we repeatedly select a wavelength that can be assigned as far as possible, i.e., to the maximum number of consecutive links, until the end of the lightpath is reached. It is easy to verify that this assignment strategy is locally optimal, i.e., places a minimum number of converters on the actual lightpath, but does not guarantee to end up with a globally optimal MCWAP solution.

Basically, we distinguish constructive and iterative methods. Where the constructive methods try to generate good assignments from scratch, the iterative algorithms start with an assignment (or processing order) and try to reduce the converter number by clever transformations, exploiting the information about the former placement of converters. Since the computational experiments documented in [10] revealed that the iterative methods performed best, we focus on these algorithms in the following.

5.1 Iterative improvement

Having finished a sequential processing of all lightpaths, we obtain a feasible wavelength assignment together with the placement of required converters. If no converters are needed, the generated MCWAP solution is obviously optimal. Otherwise, we know for which lightpaths converters are needed. The key idea is that those lightpaths probably have just been processed 'too late' in the sequence. So, we put these lightpaths at the beginning and proceed with the reordered sequence. By iterating the sequential assignment algorithm and the reordering, better assignments are hopefully found. This general method allows for some variants concerning the reordering mechanism, e.g., moving only single or multiple converted lightpaths to the beginning, the latter in the same or reversed order, and others. For further details, we refer to [10] and restrict in the following on a brief summary of the computational results documented there.

We have observed that it is most favorable to put, in each iteration, all converted lightpaths at the beginning of the sequence, either in the former or in reversed order. Both algorithms typically generate a series of assignments whose number of converters decreases rapidly in the beginning, soon finding good solutions, and have then to work more and more for further improvements. While many converter-free solutions could be found this way, some instances did not allow to find assignments with less than a certain number of converters, starting to yo-yo at some point. As long as it is unclear whether the best solution found so far is provably optimal, the iterative method does not terminate (without setting a time limit). In such a situation, even a lower bound on the converter number is not helpful unless it already matches the optimum (and an appropriate assignment is known). However, the best found assignments typically have a low number of converters. So, it might be helpful to reduce the problem on those lightpaths (or wavelengths) that cause trouble.

5.2 Extraction

To improve the iterative heuristic performance, we propose a problem extraction method. Assume the algorithm has generated a solution with $z < \frac{1}{2}|\Lambda|$ converters. Since any converter affects exactly two different wavelengths, the solution contains some wavelengths

that are never converted to or from. Stated otherwise, such wavelengths are assigned only to complete lightpaths. Nevertheless, these lightpaths are always reprocessed in the next iteration, too, and typically similar assignments with some unconverted wavelengths are produced.

The key idea now is to extract all unconverted wavelengths together with the associated lightpaths from the best known solution and to continue with the reduced instance. On the one hand, this clearly reduces the search space of complete assignments and can, in the worst case, avoid to find an optimal solution. On the other hand, such an extraction does not only improve the computation time for each iteration and thus allows to examine many more solutions, but opens also the chance to find better assignments by focusing on the 'critical part' of the problem. As long as the newly generated solutions contain further unconverted wavelengths, the extraction method can also be applied repeatedly.

Finally, we remark that the extracted problem states a complete MCWAP instance itself. So, the size reduction can make the remaining instance tractable for exact approaches, too, and is not limited to be applied within the iterative algorithms (but note that even an optimal solution of such a reduced instance need not to prove optimality for the original problem). However, we experienced that the extraction is already successful in improving the heuristics. In particular in combination with a good lower bound computed by the column generation algorithm, we obtain in many cases provably optimal wavelength assignments, as shown next.

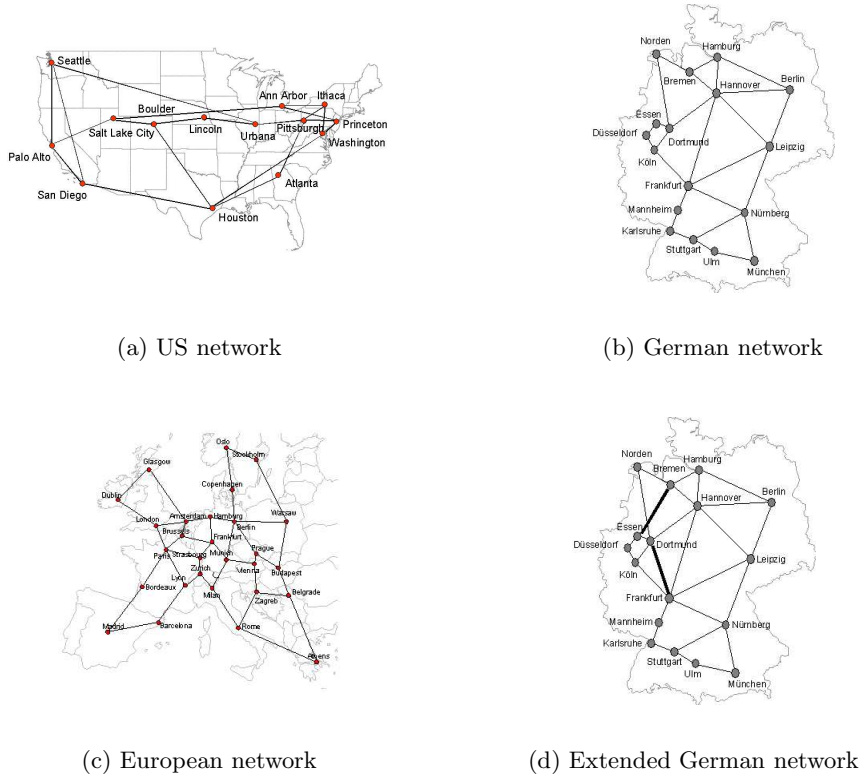
6 Computational experiments

In this section, we present a computational study to evaluate the methods proposed in this paper. For this, we restrict on the case of MCWAP with uniform wavelength spectra. We describe the used optical network design instances, present the results obtained by our methods, and discuss their impact.

6.1 Instances

For the design of optical transport networks, three reference scenarios have been defined within the MultiTeraNet project [2]. Every scenario consists of a network topology and a traffic matrix which specifies the demand for each pair of nodes as number of lightpaths to establish. The networks represent an US network based on the NSF topology with 14 nodes and 21 links, a hypothetical German network with 17 nodes and 26 links, and a European network with 28 nodes and 41 links. All networks have a meshed topology and can be equipped with the same set of devices, including fibers, WDM systems providing $|\Lambda| = 40$ wavelengths, OXCs, and wavelength converters.

In each scenario, we consider four survivability specifications which differ in the fraction $p \in [0, 1]$ of the traffic that has to be protected against any single link or node failure. In the UNPROTECTED case ($p = 0$), no survivability is provided, while FULL PROTECTED ($p = 1$) means that all lightpaths have to be protected. The other two cases, $\frac{1}{3}$ -PROTECTED and $\frac{2}{3}$ -PROTECTED, require to protect a fraction of $p = \frac{1}{3}$ and $p = \frac{2}{3}$ of each demand,



(a) US network

(b) German network

(c) European network

(d) Extended German network

Figure 3: Networks of the computation scenarios.

respectively, such that for a demand of d lightpaths, $\lceil p \cdot d \rceil$ of them have to be protected. The specified survivability is realized by additional establishment of backup lightpaths according to the concept Demand-wise Shared Protection (DSP) proposed in [11]. To reduce the number of needed backup lightpaths, DSP spreads the lightpath routing of each demand by exploiting the network’s connectivity. For evaluation of the concept, we also extended the German network by two links to increase the connectivity which allowed to save further backup lightpaths for the high survivability requirements in the $\frac{2}{3}$ -PROTECTED and FULL PROTECTED cases. These instances yielded non-trivial wavelength assignment subproblems and are therefore included in our test set.

So, we consider in total 14 instances on four networks which are depicted in Fig. 3. For each instance, we have computed an optical network design using the tool OND (Optical Network Design) described in [13]. Besides the dimensioning and routing, the tool offers also an initial wavelength assignment generated by a short run of an iterative heuristic. Some instance characteristics are listed in Table 1. Subsequently, the columns list the number of nodes ($|N|$) and links ($|L|$) in the network as well as the total number of established lightpaths ($|\mathcal{P}|$). The last column contains the number of converters in the initial wavelength assignment provided by OND.

instance	statistics			initial solution
	$ N $	$ L $	$ \mathcal{P} $	
US network				
UNPROTECTED	14	21	2710	0
$\frac{1}{3}$ -PROTECTED	14	21	2710	0
$\frac{2}{3}$ -PROTECTED	14	21	3166	1
FULL PROTECTED	14	21	4655	14
German network				
UNPROTECTED	17	26	686	0
$\frac{1}{3}$ -PROTECTED	17	26	699	0
$\frac{2}{3}$ -PROTECTED	17	26	836	12
FULL PROTECTED	17	26	1193	0
Extended German network				
$\frac{2}{3}$ -PROTECTED	17	28	796	6
FULL PROTECTED	17	28	1122	8
European network				
UNPROTECTED	28	41	1008	15
$\frac{1}{3}$ -PROTECTED	28	41	1148	3
$\frac{2}{3}$ -PROTECTED	28	41	1480	75
FULL PROTECTED	28	41	1855	39

Table 1: Characteristics of the test instances.

6.2 Results

Clearly, for all instances without converters, we can already conclude optimality of the initial wavelength assignment. So, we focus in the following on the remaining instances for which a converter-free solution of the associated MCWAP was not found within the network design procedure. All reported computations have been run on a Linux-operated PC with an Intel Pentium 4 3,2 GHz HT processor.

At first, we have determined a lower bound on the number of unavoidable converters by solving the linear relaxation of the MCWAP integer programming formulation. For this, we have implemented the column generation method in C++, using CPLEX 9.0 [7] as (integer) linear programming solver with the C++ interface of ILOG's Concert Technology. The restricted program was initialized with a single column representing a subpath packing that contains only subpaths with a single link (each with multiplicity κ_ℓ). Note that this column suffices to guarantee for a feasible solution of the restricted program, but with the worst possible value (placing a converter in each intermediate node of each lightpath). Then we iteratively generated improving columns, as described in Section 4, until the linear relaxation was solved optimally.

These computations are documented in Table 2. For each instance, the first column recalls the number of converters in the initial solution (init. sol.). Next, the columns list subsequently the final optimal LP value as lower bound (LB), the total number of generated columns (cols), and the total CPU time in seconds (time). To illustrate the size of the linear program, the last two columns display the total numbers of involved subpaths ($|\mathcal{S}|$)

instance	init. sol.	column generation method				
		LB	cols	time	$ \mathcal{S} $	y-vars
US network						
$\frac{2}{3}$ -PROTECTED	1	0	176	5.3	252	1739
FULL PROTECTED	14	0	156	5.1	252	1685
German network						
$\frac{2}{3}$ -PROTECTED	12	12	112	2.1	259	1096
Extended German network						
$\frac{2}{3}$ -PROTECTED	6	6	154	4.0	311	1220
FULL PROTECTED	8	6	154	3.3	256	1067
European network						
UNPROTECTED	15	2	376	75.7	606	4243
$\frac{1}{3}$ -PROTECTED	3	0	746	1165.7	1102	13218
$\frac{3}{3}$ -PROTECTED	75	0	1630	4306.0	1185	15398
FULL PROTECTED	39	0	1557	4067.0	1350	15902

Table 2: Lower bound computations.

instance	LB	heur. sol.	extraction		
			sol.	$ \Lambda_r $	$ \mathcal{P}_r $
US network					
$\frac{2}{3}$ -PROTECTED	0	0			
FULL PROTECTED	0	0			
Extended German network					
FULL PROTECTED	6	6			
European network					
UNPROTECTED	2	9	8	13	329
$\frac{1}{3}$ -PROTECTED	0	1	0	2	49
$\frac{3}{3}$ -PROTECTED	0	54	54	33	1213
FULL PROTECTED	0	15	5	13	606

Table 3: Iterative heuristic results with 6000 seconds CPU time limit.

and of y_p^s -variables (y-vars).

Remarkably, we obtain a non-zero lower bound in roughly half of the cases, i.e., these problems are proven to have no converter-free assignments. Moreover, in two instances the lower bound additionally indicates that the solution at hand is indeed optimal. Hence, these instances need not to be considered further.

So, there are seven unsolved instances left. As next step, we have tried to find better solutions with a longer run of the iterative heuristic for which we have set a limit of 6000 seconds CPU time. The results are shown in the left part of Table 3, listing the lower bound (LB) and the number of converters in the best solution found (heur. sol.) during that run. By this, advanced assignments have been found for all instances, and three more solutions match the lower bound, i.e., are provably optimal.

For further improvement of the heuristic search, we have applied the extraction method

instance	LB	2. extraction			3. extraction		
		sol.	$ \Lambda_r $	$ \mathcal{P}_r $	sol.	$ \Lambda_r $	$ \mathcal{P}_r $
European network							
UNPROTECTED	2	5	8	201	3	7	174
$\frac{2}{3}$ -PROTECTED	0	58	28	1035	54	26	969
FULL PROTECTED	0	0	7	324			

Table 4: Results for repeated extractions.

which removes all unconverted wavelengths and the associated lightpaths from the best solution known so far. As listed in the right part of Table 3, we have generated new solutions (sol.) by running the iterative algorithm, again with a time limit of 6000 seconds, on the extracted instances with a reduced set of wavelengths (Λ_r) and lightpaths (\mathcal{P}_r). The extraction allowed to find assignments with less converters in three of four cases, one optimal solution among them.

In addition, the newly computed assignments for the other three cases have also been structurally improved by generating further unconverted wavelengths. This enables to repeat the extraction method, step-wise removing those wavelengths and lightpaths after each run of the iterative heuristic (of at most 6000 seconds). In the same form as in Table 3, the results of two further extraction iterations are shown in Table 4. For the $\frac{2}{3}$ -PROTECTED case, no further improvement was achievable, with an even worse solution in the second iteration. For the two remaining instances, the repeated extraction was more successful. While a converter-free assignment was already found in the second extraction for FULL PROTECTED, the UNPROTECTED case could iteratively reduce the number of required converters down to three, a fairly good solution as indicated by the lower bound of two.

6.3 Discussion

The computational results point up some interesting aspects of the models and methods proposed in this paper. First of all, we have in total found provably optimal wavelength assignments for twelve out of 14 optical network designs, i.e., 85 % of the instances. Only five of the instances have already been solved within the optical network design procedure, while the remaining nine instances state more difficult problems and required additional effort. For their solution, the derived lower bound turned out as valuable information to benchmark the quality of the generated wavelength assignments and to guide the heuristic search.

By knowledge of such a benchmark, we could apply the iterative algorithm purposive to improve the assignments for those instances that have not been solved so far. Thereby, the extraction method was helpful to enhance the heuristic performance and to approach a minimum number of converters foreshadowed by the lower bound. In fact, the computed lower bounds match the optimum for at least seven of the nine instances and proved in four cases that no converter-free assignment exists. Moreover, in one of the two non-solved instances, the benchmark of two shows that the best found assignment with three converters already represents a nearly optimal solution. So, we can conclude that for nearly

all instances, provably good solutions for the minimum converter wavelength assignment problem have been generated by our methods.

7 Concluding remarks

In this paper, we studied the Minimum Converter Wavelength Assignment Problem (MCWAP) which states an important task in the design of optical networks. We developed a suitable integer programming formulation for the general case, the subpath packing formulation, and derived a column generation method to solve the linear relaxation whose value states a lower bound on the number of required converters. We also revisited heuristics to compute wavelength assignments and proposed an advancement by a problem extraction method.

The computational study revealed the quality of the lower bounds provided by the subpath packing formulation. In contrast to a standard assignment formulation, the linear relaxation optimum can take positive values and thus indicates how many converters are unavoidable for any feasible wavelength assignment. The obtained benchmarks have in fact verified optimality in most cases and were helpful to guide the heuristic search which was substantially enhanced by the extraction method. Finally, only the additional information of the lower bound allows to state to have found provably good solutions.

A direction for further research consists in the development of exact solution algorithms, i.e., branch-and-price or branch-cut-and-price methods, for the subpath packing formulation. Moreover, further performance improvements for the heuristics are of interest.

References

- [1] K. I. Aardal, C. P. M. van Hoesel, A. M. C. A. Koster, C. Mannino, and A. Sassano. Models and solution techniques for the frequency assignment problem. *4OR*, 1(4):261–317, 2003.
- [2] A. Betker, C. Gerlach, R. Hülsermann, M. Jäger, M. Barry, S. Bodamer, J. Späth, C. Gauger, and M. Köhn. Reference transport network scenarios. Technical report, BMBF MultiTeraNet, July 2003. http://www.ikr.uni-stuttgart.de/IKRSimLib/Referenz_Netze_v14_full.pdf.
- [3] R. Borndörfer. *Aspects of Set Packing, Partitioning, and Covering*. PhD thesis, Technische Universität Berlin, published at Shaker Verlag Aachen, 1998.
- [4] V. Chvátal. *Linear Programming*. W.H. Freeman and Company, New York, 1983.
- [5] J. Desrosiers and M.E. Lübbecke. A primer in column generation. In G.Desaulniers, J.Desrosiers, and M.M. Solomon, editors, *Column Generation*. Kluwer Academic Publishers, Boston, MA, 2005.
- [6] A. Ferreira, S. Pérennes, A. W. Richa, H. Rivano, and N. Stier Moses. Models, Complexity and Algorithms for the Design of Multifiber WDM Networks. *Telecommunication Systems*, 2003. To be published.

- [7] ILOG. CPLEX version 9.0, 2003. <http://www.ilog.com/products/cplex>.
- [8] T. R. Jensen and B. Toft. *Graph Coloring Problems*. Series in Discrete Mathematics and Optimization. Wiley-Interscience, 1995.
- [9] A. M. C. A. Koster and A. Zymolka. Linear programming lower bounds for minimum converter wavelength assignment in optical networks. ZIB-report 04–XY, Konrad-Zuse-Zentrum für Informationstechnik Berlin, Berlin, Germany, 2004.
- [10] A. M. C. A. Koster and A. Zymolka. Minimum converter wavelength assignment in all-optical networks. In *Proceedings of ONDM 2004*, pages 517–535, Ghent, Belgium, 2004. The 8th IFIP Working Conference on Optical Network Design & Modelling.
- [11] A. M. C. A. Koster, A. Zymolka, M. Jäger, R. Hülsermann, and C. Gerlach. Demand-wise shared protection for meshed optical networks. In *Proceedings of the 4th International Workshop on Design of Reliable Communication Networks (DRCN 2003)*, pages 85–92, Banff, Canada, 2003. IEEE.
- [12] A. Mehrotra and M. A. Trick. A column generation approach for graph coloring. *INFORMS Journal on Computing*, 8:344–354, 1996.
- [13] A. Zymolka, A. M. C. A. Koster, and R. Wessäly. Transparent optical network design with sparse wavelength conversion. In *Proceedings of ONDM 2003*, pages 61–80, Budapest, Hungary, 2003. The 7th IFIP Working Conference on Optical Network Design & Modelling.