Valid Inequalities and Branch-and-Cut-and-Price Algorithm for the Constrained-Routing and Spectrum Assignment Problem

Ibrahima Diarrassouba, Youssouf Hadhbi, Ali Ridha Mahjoub

Valid Inequalities and Branch-and-Cut-and-Price Algorithm for the Constrained-Routing and Spectrum Assignment Problem*

Ibrahima Diarrassouba[1], Youssouf Hadhbi[2], and Ali Ridha Mahjoub[3]

[1] Le Havre Normandie University, LMAH, FR CNRS 3335, 76600 Le Havre, France
`diarrasi@univ-lehavre.fr`

[2] Clermont Auvergne University, LIMOS, UMR CNRS 6158, 63178 Clermont Ferrand, France
`youssouf.hadhbi@uca.fr`

[3] Paris Dauphine-PSL University, LAMSADE, UMR CNRS 7243, 75775 Paris CEDEX 16, France
`ridha.mahjoub@lamsade.dauphine.fr`

Abstract. In this work, we focus on a complex variant of the so-called Routing and Spectrum Assignment problem (RSA), namely the Constrained-Routing and Spectrum Assignment (C-RSA). The C-RSA problem is a key issue when dimensioning and managing a new generation of optical networks, called spectrally flexible optical networks. It is well known to be NP-hard and can be stated as follows. Consider a spectrally flexible optical network as an undirected, loopless, and connected graph G, and an optical spectrum $\mathbb{S}$ of available contiguous frequency slots, and a multiset of traffic demands K. The C-RSA consists of assigning for each traffic demand $k \in K$ a path in G and an interval of contiguous frequency slots in $\mathbb{S}$ subject to technological constraints while optimizing some linear objective function(s). The main aim of our work is to introduce a new extended integer linear programming based on the so-called path formulation for the C-RSA. This formulation has an exponential number of variables. A column generation algorithm is then used to solve its linear relaxation. To do so, we investigate the structure and properties of the associated pricing problem. We further identify several classes of valid inequalities for the associated polytope and devise their separation procedures. Based on this, we devise Branch-and-Price (B&P) and Branch-and-Cut-and-Price (B&C&P) algorithms to solve the problem. We give at the end a detailed behavior study of these algorithms.

Keywords: Spectrally flexible optical network, network design, constrained-routing, spectrum assignment, complexity, ILP, pre-processing, valid inequality, separation, column generation, branch-and-price algorithm, branch-and-cut-and-price algorithm, conflict-graph, threshold graph, interval graph, perfect graph, intersection graph, primal heuristic, metaheuristics, heuristic, greedy-algorithm, dynamic programming algorithm, branching rules.

1 Introduction

The second decade of a new millennium saw a profound change in optical transport networks with continuous growth in bandwidth capacity due to the growth of global communication services and networking: mobile internet network (e.g., 5th generation mobile network), cloud computing (e.g., data centers), Full High-definition (HD) interactive video (e.g., TV channel, social networks) [6], etc... Therefore, a new generation of optical transport network architecture called Spectrally Flexible Optical Networks (SFONs) has been introduced as promising technology because of their flexibility and efficiency compared with the traditional Optical Wavelength Division Multiplexing (WDM)[55][56]. In SFONs the optical spectrum is divided into slots having the same frequency of 12.5 GHz where WDM uses 50 GHz as recommended by ITU-T [1]. We refer the reader to [42] for more information about the architectures, technologies, and control of SFONs.
The Routing and Spectrum Assignment (RSA) problem plays an important role when dimensioning and designing SFONs. It consists of assigning for each traffic demand k, a physical optical path, and an interval of contiguous slots (called also channels) while optimizing some linear objective(s) and satisfying the following constraints [24]:

* This work was supported by the French National Research Agency grant ANR-17-CE25-0006, project FLEXOPTIM.

1. *spectrum contiguity*: an interval of contiguous slots should be allocated to each traffic demand k with a width equals to the number of slots requested by demand k;
2. *spectrum continuity*: the interval of contiguous slots allocated to each traffic demand stills the same along the chosen path;
3. *non-overlapping spectrum*: the intervals of contiguous slots of traffic demands whose paths are not edge-disjoints in the network cannot share any slot over the shared edges.

In general, the routing and resources allocation in communication networks receives increasing attention. In particular, numerous research studies have been conducted on the RSA problem since its first appearance. From a complexity point of view, the RSA is known to be an NP-hard problem [64] [67]. Various integer linear programming formulations based on the so-called path formulation and edge-node formulation and algorithms have been proposed to solve it. A detailed survey of spectrum management techniques for SFONs is presented in [67] where authors classified variants of the RSA problem into two variants: offline and online RSA. The edge-path formulation is majorly used in the literature where variables are associated with all possible physical optical paths inducing a huge number of variables and constraints which grow exponentially and in parallel with the growth of the instance size: number of demands, number of slots, spectrum-width and topology size [24]. To the best of our knowledge, we observe that several papers which use the edge-path formulation to solve RSA problem, use a set of precomputed-paths without guaranty of optimality e.g. in [9], [35], [36], [68], [70]. On the other hand, *column generation* has been applied by Klinkowski et al. in [60] and Jaumard et al. in [28] to solve the relaxed problem of RSA taking into account all the possible paths for each traffic demand. To improve the lower bound of the RSA relaxation, Klinkowsky et al. proposed in [37] a valid inequality based on clique inequality separated by a *branch-and-bound* algorithm. On the other hand, *branch-and-cut-and-price* method has been applied to guarantee the optimality using the edge-path formulation for the RSA problem by Klinkowski et al. in [38] introducing one valid inequality based on the so-called clique inequalities. On the other hand, a compact formulation based on the so-called edge-node formulation has been introduced as an alternative for the edge-path formulation to solve routing issues in general with the guarantee of optimality. The edge-node formulation overcomes the drawbacks of the edge-path formulation usage, which holds a polynomial number of variables and constraints that grow only polynomially with the size of the instance. We found just a few works in the literature that use the edge-node formulation to solve the RSA problem e.g. [3], [68], [70]. Several edge-node formulations have been compared by Bertero et al. in [2].

However, all these proposals' formulations and exact algorithms were not able to solve large-scale instances of this problem to optimality. Therefore, several heuristics have been proposed to solve the RSA problem by Ding et al. in [15], Mesquita et al. in [48], Santos et al. in [63], and recently by He et al. in [27]. Mahala et al. proposed in [47] a greedy algorithm to solve the problem. Moreover, metaheuristics have applied also to solve the RSA problem, we found for example tabu search algorithm proposed by Goscien et al. in [21], simulated annealing algorithm by Klinkowski et al. in [38], genetic algorithms by Gong et al. in [20], Hai et al. in [25][26], and ant colony algorithms by Lezama et al. in [39]. Based on this, a hybrid meta-heuristic approach has been applied by Ruiz in [59] to solve large-sized instances of the RSA problem. Furthermore, we noticed also that artificial intelligence algorithms can be used to boost the performance of the proposals' algorithms used to solve the RSA problem, see for example the work of Liu et al. in [40] and Lohani et al. in [41]. Moreover, some techniques related to the learning aspect are also used to improve the efficiency of the algorithms proposed using deep-learning algorithms [5], and also machine-learning algorithms in [62], and recently in [69] and [23].

In this paper, we are interested in the Constrained-Routing and Spectrum Assignment (C-RSA) problem. Here we suppose that the network should also satisfy the transmission-reach constraint that is the route for each traffic demand should not exceed a certain length. Recently, Hadhbi et al. in [24] introduced a cut formulation to solve the C-RSA problem based on the so-called cut inequalities that are separable in polynomial time using network flow algorithms. It has been used by Chouman et al. in [7] and [8] to show the impact of several objective functions on the optical network state. Computational results show that their formulation solves larger instances compared with those of Velasco et al. in [68] and Cai et al. [3]. Note that the transmission-reach constraint has not been taken into account by Velasco et al. in [68], Cai et al. [3], and Bertero et al. in [2].

On the other hand, Colares et al. in [11] propose a compact formulation for the C-RSA problem based on the edge-node formulation. It can be seen as a reformulation of the cut formulation proposed by Hadhbi et al. in [24] using an oriented graph.

2 Our Contributions

Since that the exact algorithms developed were not able to solve large instances of RSA and C-RSA to optimality, it has been found appropriate to propose new tractable integer linear programming formulations for the C-RSA problem, design and develop efficient exact algorithms that may offer promise improvements over the existing methods. To the best of our knowledge, a cutting-plane-based approach has not yet been the subject of a substantial body of recent research concerning the issue of the C-RSA problem. For that, the main aim of our work is to investigate the thoroughly theoretical properties of the C-RSA problem. To this end, we aim to introduce a new integer linear programming based on the so-called path formulation for the C-RSA problem. This formulation is characterized by an exponential number of variables. We, therefore, use the column generation algorithm to solve its linear relaxation. To do so, we investigate the properties of the associated pricing problem and prove that it is equivalent to the so-called resource constrained shortest path problem, which is well known to be NP-hard. For this, we propose a pseudo-polynomial time algorithm using dynamic programming adapted to our problem. We further identify several classes of valid inequalities using some conflict graphs related to the problem: clique inequalities, odd-hole inequalities, and some cover inequalities related to the capacity constraints. We then devise their separation procedures based on exact algorithms, greedy algorithms, and heuristics [22]. Using the path formulation and the separation procedures, we develop Branch-and-Price (B&P) and Branch-and-Cut-and-Price (B&C&P) algorithms to solve the problem. Furthermore, we boost its effectiveness through some enhancements to obtain tighter primal bounds based on a warm-start algorithm based on some metaheuristics: simulated annealing and tabu search algorithms which push a feasible integral solution (if possible) in the root of our B&C&P algorithm before the start of the resolution of C-RSA, and also a primal-heuristic based on a hybrid method between a greedy algorithm and a local search algorithm to construct a feasible integral solution from a given fractionally solution in each node of the B&P and B&C&P trees.

3 Organization

The rest of this paper is organized as follows. In Section (4), we present the C-RSA problem (input and output). In Section (5), we provide some notations that are useful throughout this paper. After that, we introduce our path formulation based on the so-called path variables. It can be seen as a reformulation for our cut formulation proposed in [12],[13] and [14]. In Section (6), we thoroughly investigate the theoretical properties of the C-RSA problem by providing several valid inequalities for the associated polyhedron. Based on the results of sections (5)-(6), we give an outline of our Branch-and-Price and Branch-and-Cut-and-Price algorithms in the section (8). We close with a brief summary of results and future outlook.

4 The Constrained-Routing and Spectrum Assignment Problem

The Constrained-Routing and Spectrum Assignment Problem can be stated as follows. We consider a spectrally flexible optical networks as an undirected, loopless, and connected graph $G = (V, E)$, which is specified by a set of nodes V, and a multiset [4] E of links (optical-fibers). Each link $e = ij \in E$ is associated with a length $\ell_e \in \mathbb{R}_+$ (in kms), a cost $c_e \in \mathbb{R}_+$ such that each fiber-link $e \in E$ is divided into $\bar{s} \in \mathbb{N}_+$ slots. Let $\mathbb{S} = \{1, \ldots, \bar{s}\}$ be an optical spectrum of available frequency slots with $\bar{s} \leq 320$ given that the maximum spectrum bandwidth of each fiber-link is 4000 GHz [29], and K be a multiset [5] of demands such that each demand $k \in K$ is specified by an origin node

[4] We take into account the presence of parallel fibers such that two edges e, e' which have the same extremities i and j are independents.

[5] We take into account that we can have several demands between the same origin-node and destination-node.

$o_k \in V$, a destination node $d_k \in V \setminus \{o_k\}$, a slot-width $w_k \in \mathbb{Z}_+$, and a transmission-reach $\bar{\ell_k} \in \mathbb{R}_+$ (in kms). The C-RSA problem consists of determining for each demand $k \in K$, a (o_k,d_k)-path p_k in G such that $\sum_{e \in E(p_k)} l_e \leq \bar{l}_k$, where $E(p_k)$ denotes the set of edges belong the path p_k, and a subset of contiguous frequency slots $S_k \subset \mathbb{S}$ of width equal to w_k such that $S_k \cap S_{k'} = \emptyset$ for each pair of demands $k, k' \in K$ $(k \neq k')$ with $E(p_k) \cap E(p_{k'}) \neq \emptyset$ so the total length of the paths used for routing the demands (i.e., $\sum_{k \in K} \sum_{e \in E(p_k)} l_e$) is minimized.
Figure 2 shows the set of established paths and spectrums for the set of demands $\{k_1, k_2, k_3, k_4\}$ (Fig. 2(c) and Table 2(d)) of Table 2(b) in a graph G of 7 nodes and 10 edges (Fig. 2(a)) such that each edge e is characterized by a triplet $[l_e, c_e, \bar{s}]$, and optical spectrum $\mathbb{S} = \{1, 2, 3, ..., 8, 9\}$ with $\bar{s} = 9$.

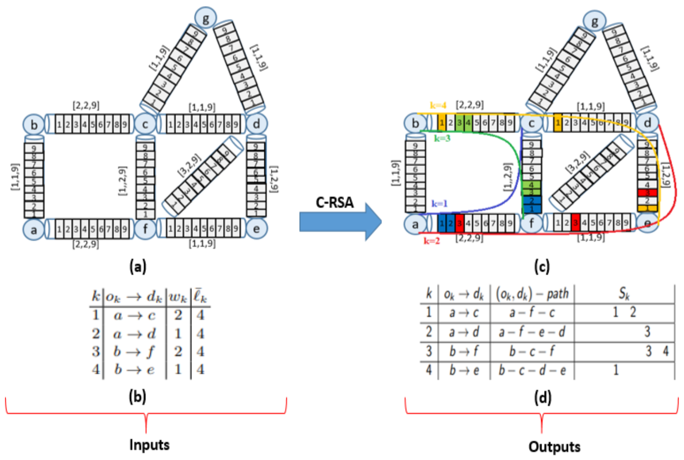

Fig. 1. Set of established paths and spectrums in graph G (Fig. 2(a)) for the set of demands $\{k_1, k_2, k_3, k_4\}$ defined in Table 2(b).

Before introducing our ILP, we proceeded to some pre-processing techniques to identify some properties due to the transmission-reach and capacity constraints as follows.
For each demand k and each node v, one can compute a shortest path between each of the pair of nodes (o_k, v), (v, d_k). If the lengths of the $(o_k, d_k)-$paths formed by the shortest paths (o_k, v) and (v, d_k) are both greater that $\bar{l}_k$ then node v cannot be in a path routing demand k, and we then say that v is a *forbidden node* for demand k due to the transmission-reach constraint. Let V_0^k denote the set of forbidden nodes for demand $k \in K$. Note that using Dijkstra's algorithm, one can identify in polynomial time the forbidden nodes V_0^k for each demand $k \in K$. On the other hand and regarding the edges, for each demand k and each edge $e = ij$, one can compute a shortest path between each of the pair of nodes (o_k, i), (j, d_k), (o_k, j) and (i, d_k). If the lengths of the $(o_k, d_k)-$paths formed by e together with the shortest (o_k, i) and (j, d_k) (resp. (o_k, j) and (i, d_k)) paths are both greater that $\bar{l}_k$ then edge ij cannot be in a path routing demand k, and we then say that ij is a *forbidden edge* for demand k due to the transmission-reach constraint. Let E_t^k denote the set of forbidden edges due to the transmission-reach constraint for demand $k \in K$. Note that using Dijkstra's algorithm, one can identify in polynomial time the forbidden edges E_t^k for each demand $k \in K$. This allows us to create in polynomial time a proper topology G_k for each demand k by deleting the forbidden nodes V_0^k and forbidden edges E_t^k from the original graph G (i.e., $G_k = G(V \setminus V_0^k, E \setminus E_t^k)$). As a result, there may exist some forbidden-nodes due to the elementary-path constraint which means that all the $(o_k, d_k)-$paths passed through a node v are not elementary-paths. This can be done in polynomial time using Breadth First Search (BFS) algorithm of complexity $O(|E \setminus E_0^k| + |V \setminus V_0^k|)$ for each demand k. Note that we did not take into account this case in our study.
Let $\delta_{G_k}(v)$ denote the set of edges incident with a node v for the demand k in G_k. Let $\delta^k(W)$ denote a cut for demand $k \in K$ in G_k such that $o_k \in W$ and $d_k \in V \setminus W$ where W is a subset of nodes in V of G_k. Let f be an edge in $\delta(W)$ such that all the edges $e \in \delta(W) \setminus \{f\}$ are forbidden

for demand k. As a consequence, edge f is an *essential edge* for demand k. As the forbidden edges, the essential edges can be determined in polynomial time using network flows as follows.

1. we create a proper topology $G_k = G(V \setminus V_0^k, E \setminus E_t^k)$ for the demand k
2. we fix a weight equals to 1 for all the edges e in $E \setminus E_t^k$ for the demand k in G_k
3. we calculate $o_k - d_k$ min-cut which separates o_k from d_k.
4. if $\delta_{G_k}(W) = \{e\}$ then the edge e is an essential edge for the demand k such that $o_k \in W$ and $d_k \in V \setminus W$. We increase the weight of the edge e by 1. Go to (3).
5. if $|\delta_{G_k}(W)| > 1$ then end of algorithm.

Let E_1^k denote the set of essential edges of demand k, and K_e denote a subset of demands in K such that edge e is an essential edge for each demand $k \in K_e$. Figure 4 shows the proper-topology G_k for a demand k between the two nodes $o_k = a$ and $d_k = g$ with $\bar{l}_k = 4$ by deleting V_0^k nodes and E_0^k edges from G.

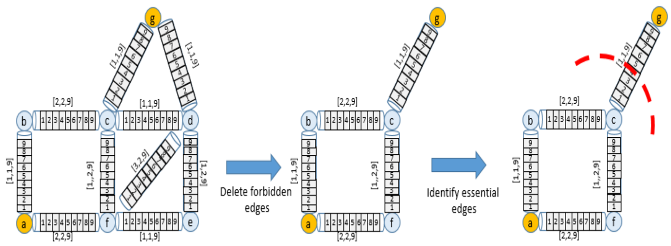

Fig. 2. Proper-topology G_k for a given demand k between $o_k = a$ and $d_k = g$ with $\bar{l}_k = 4$.

In addition to the forbidden edges thus obtained due to the transmission-reach constraints, there may exist some edges that may be forbidden because of lack of resources for demand k. This is the case when, for instance, the residual capacity of the edge in question does not allow a demand to use this edge for its routing, i.e., $w_k > \bar{s} - \sum_{k' \in K_e} w_{k'}$. Let E_c^k denote the set of forbidden edges for demand k, $k \in K$, due to the resource constraints. Note that the forbidden edges E_c^k and forbidden nodes v in V with $\delta(v) \subseteq E_t^k \cup E_c^k$, should also be deleted from the proper graph G_k of demand k, which means that G_k contains $|E| \setminus |E_t^k \cup E_c^k|$ edges and $|V| \setminus |\{v \in V, \delta(v) \subseteq E_t^k \cup E_c^k\}|$ nodes. Let $E_0^k = E_t^k \cup E_c^k$ denote the set of all forbidden edges for demand k that can be determined due to the transmission reach and resources constraints. As a result of the pre-processing stage, some non-compatibility between demands may appear due to a lack of resources as follows.

Definition 1. *For an edge e, two demands k and k' with $e \notin E_0^k \cup E_1^k \cup E_0^{k'} \cup E_1^{k'}$, are said* non-compatible demands *because of lack of resources over the edge e if and only if the the residual capacity of the edge e does not allow to route the two demands k, k' together through e, i.e., $w_k + w_{k'} > \bar{s} - \sum_{k" \in K_e} w_{k"}$. Let K_c^e denote the set of pair of demands (k, k') in K that are non-compatibles for the edge e.*

In the next section, we introduce our path formulation.

5 The C-RSA Integer Linear Programming Formulation

Let P^k denote the set of all feasible (o_k, d_k) paths in G such that for each demand $k \in K$, we have

$$\sum_{e \in E(p_k)} l_e \leq \bar{l}_k, \text{ for all } p_k \in P^k.$$

Our path formulation is based on one family of variables. We consider for $k \in K$ and $p \in P^k$ and $s \in S$, a variable $y_{p,s}^k$ which takes 1 if slot s is the last slot allocated along the path p for the routing of demand k and 0 if not, such that s represents the last slot of the interval of contiguous slots of width w_k allocated by the demand $k \in K$, with $s \in S$ and $p \in P^k$. Note that all the slots $s' \in \{s - w_k + 1, ..., s\}$ should be assigned to demand k along the path p whenever $y^k_{p,s} = 1$. Let $P^k(e)$ denote set of all admissible (o_k,d_k) paths going through the edge e in G for the demand k. The C-RSA is then equivalent to the following integer linear programming

$$\min \sum_{k\in K} \sum_{p\in P^k} \sum_{e\in E(p)} \sum_{s=w_k}^{\bar{s}} l_e y^k_{p,s}, \quad (1)$$

subject to

$$\sum_{p\in P^k} \sum_{s=1}^{w_k-1} y^k_{p,s} = 0, \forall k \in K, \quad (2)$$

$$\sum_{p\in P^k} \sum_{s=w_k}^{\bar{s}} y^k_{p,s} = 1, \forall k \in K, \quad (3)$$

$$\sum_{k\in K} \sum_{p\in P^k(e)} \sum_{s'=s}^{s+w_k-1} y^k_{p,s'} \leq 1, \forall e \in E, \forall s \in S, \quad (4)$$

$$y^k_{p,s} \geq 0, \forall k \in K, \forall p \in P^k, \forall s \in S, \quad (5)$$

$$y^k_{p,s} \in \{0,1\}, \forall k \in K, \forall p \in P^k, \forall s \in S. \quad (6)$$

Inequalities (2) express the fact that a demand $k \in K$ cannot occupy a slot s as the last slot before her slot-width w_k. Inequalities (3) express the routing and spectrum constraints such that they ensure that exactly one slot $s \in \{w_k, \dots, \bar{s}\}$ is assigned as last slot for the routing of demand k, and exactly one single path from P^k is allocated by each demand $k \in K$. Note that a slot $s \in S$ is said an allocated slot by the demand k if and only if $\sum_{p\in P^k} \sum_{s'=s}^{s+w_k-1} y^k_{p,s'} = 1$ which means that s is covered by the interval of contiguous slots allocated by demand k. Inequalities (4) ensure that a slot s over the edge e cannot be allocated to at most by one demand $k \in K$. Inequalities (5) are trivial inequalities, and constraints (6) are the integrality constraints.
Let $P(G, K, \mathbb{S}, P_K)$ be the polytope, convex hull of the solutions for our path formulation (2)-(6). In the remainder of this paper, we focus on the introduction of valid inequalities used to obtain tighter LP bounds and some symmetry-breaking inequalities that allow avoiding the equivalents sub-problems in the different enumeration trees.

6 Valid Inequalities

In what follows, we present several valid inequalities for $P(G, K, \mathbb{S}, P_K)$ such that throughout each proof, we take into account that $0 \leq y^k_{p,s} \leq 1$ for each demand $k \in K$ and path $p \in P^k$ and $s \in \mathbb{S}$, and $\sum_{p\in P^k} \sum_{s=1}^{\bar{s}} y^k_{p,s} \leq 1$ for each $k \in K$, and $0 \leq \sum_{p\in P^k} y^k_{p,s} \leq 1$ for each demand $k \in K$ and slot $s \in \mathbb{S}$. Note that a slot $s \in \mathbb{S}$ is assigned to a demand $k \in K$ if and only if $\sum_{p\in P^k} \sum_{s'=s}^{min(\bar{s},s+w_k-1)} y^k_{p,s'} = 1$. Let $\binom{n}{k}$ denote the total number of possibilities to choose a k element in a set of n elements. Let us denote by the symbole $a \preceq b$ iff b dominates a.

6.1 Edge-Interval-Cover Inequalities

Let's first introduce some valid inequalities which can be seen as cover inequalities using some notions of cover related to our problem.

Definition 2. *An interval $I = [s_i, s_j]$ represents a set of contiguous slots situated between the two slots s_i and s_j with $j \geq i + 1$ and $s_j \leq \bar{s}$.*

Definition 3. *For an interval of contiguous slots $I = [s_i, s_j]$, a subset of demands $K' \subseteq K$ is said a cover for the interval $I = [s_i, s_j]$ if and only if $\sum_{k\in \tilde{K}} w_k > |I|$ and $w_k < |I|$ for each $k \in \tilde{K}$.*

Definition 4. *For an interval of contiguous slots $I = [s_i, s_j]$, a cover $\tilde{K}$ is said a minimal cover if $\tilde{K} \setminus \{k\}$ is not a cover for interval $I = [s_i, s_j]$ for each demand $k \in \tilde{K}$, i.e., $\sum_{k' \in \tilde{K} \setminus \{k\}} w_{k'} \leq |I|$ for each demand $k \in \tilde{K}$.*

Based on these definitions, we introduce the following inequalities.

Proposition 1. *Consider an edge $e \in E$. Let $I = [s_i, s_j]$ be an interval of contiguous slots in $[1, \bar{s}]$ with $j \geq i + 1$. Let $\tilde{K}$ be a minimal cover for the interval I s.t.*

- $\sum_{k \in \tilde{K}} w_k \leq \bar{s} - \sum_{k' \in K_e \setminus \tilde{K}} w_{k'}$,
- *$e \notin E_0^k$ for each demand $k \in \tilde{K}$,*
- $\tilde{K} \geq 3$,
- *$(k, k') \notin K_c^e$ for each pair of demands (k, k') in $\tilde{K}$.*

Then, the inequality

$$\sum_{k \in \tilde{K}} \sum_{p \in P^k(e)} \sum_{s = s_i + w_k - 1}^{s_j} y_{p,s}^k \leq |\tilde{K}| - 1, \tag{7}$$

is valid for $P(G, K, \mathbb{S}, P_K)$.

Proof. The interval $I = [s_i, s_j]$ can cover at most $|\tilde{K}| - 1$ demands given that $\tilde{K}$ is a minimal cover for interval $I = [s_i, s_j]$ over edge. It follows that if the demands $\tilde{K}$ pass together through the edge e, there is at most $|\tilde{K}| - 1$ demands that can share the interval I over edge e.
We start our proof by assuming that the inequality (7) is not valid for $P(G, K, \mathbb{S}, P_K)$. It follows that there exists a C-RSA solution S in which $\{s_i + w_k - 1, ..., s_j\} \cap S_k = \emptyset$ for a demand $k \in K'$ s.t.

$$\sum_{k' \in K' \setminus \{k\}} \sum_{p \in P^{k'}(e)} \sum_{s = s_i + w_{k'} - 1}^{s_j} y_{p,s}^{k'}(S) \geq |K'|.$$

Since $\{s_i + w_k - 1, ..., s_j\} \cap S_k = \emptyset$ for a demand $k \in K'$ this means that $\sum_{p \in P^k(e)} \sum_{s = s_i + w_k - 1}^{s_j} y_{p,s}^k(S) = 0$, and taking into account that K' is minimal cover for the interval $I = [s_i, s_j]$ over edge e, and $\sum_{p \in P^k(e)} \sum_{s = s_i + w_k - 1}^{s_j} y_{p,s}^k(S) \leq 1$ for each demand $k \in K'$, it follows that

$$\sum_{k' \in K' \setminus \{k\}} \sum_{p \in P^{k'}(e)} \sum_{s = s_i + w_{k'} - 1}^{s_j} y_{p,s}^{k'}(S) \leq |K'| - 1,$$

which contradicts what we supposed before. We conclude at the end that the inequality (7) is valid for $P(G, K, \mathbb{S}, P_K)$.

The inequality (7) can be strengthened by introducing its extended format of the minimal cover K' for the interval I over edge e as follows.

Proposition 2. *Consider an edge $e \in E$. Let $I = [s_i, s_j]$ be an interval of contiguous slots in $[1, \bar{s}]$ with $j \geq i + 1$. Let $\tilde{K}$ be a minimal cover for the interval I, and $\tilde{K}_e$ be a subset of demands in $K_e \setminus \tilde{K}$ s.t.*

- $\sum_{k \in \tilde{K}} w_k \leq \bar{s} - \sum_{k' \in K_e \setminus \tilde{K}} w_{k'}$,
- *$e \notin E_0^k$ for each demand $k \in \tilde{K}$,*
- $\tilde{K} \geq 3$,
- *$(k, k') \notin K_c^e$ for each pair of demands (k, k') in $\tilde{K}$,*
- *$w_{k'} \geq w_k$ for each $k \in \tilde{K}$ and each $k' \in \tilde{K}_e$.*

Then, the inequality

$$\sum_{k\in\tilde{K}}\sum_{p\in P^k(e)}\sum_{s=s_i+w_k-1}^{s_j} y^k_{p,s} + \sum_{k'\in\tilde{K}_e}\sum_{p\in P^{k'}(e)}\sum_{s'=s_i+w_{k'}-1}^{s_j} y^{k'}_{p,s'} \leq |\tilde{K}|-1, \tag{8}$$

is valid for $P(G,K,\mathbb{S},P_K)$.

Proof. Similar with what we did in the proof of the theorem 6.1.

More general, a strengthened inequality based on the inequality (7) can be defined using lifting procedures proposed by Nemhauser and Wolsey in [49] without modifying its right-hand side.
By inspiration of the inequality (7), and based on the set of minimal cover with cardinality equal to 2, we introduce valid inequalities defined as follows using some notions of graph theory related to conflict graphs.

6.2 Edge-Interval-Clique Inequalities

Definition 5. *Consider an edge* $e\in E$. *Let* $I=[s_i,s_j]$ *be an interval of contiguous slots in* $[1,\bar{s}]$ *with* $s_i\leq s_j-1$. *Consider the conflict graph* $\tilde{G}^e_I$ *defined as follows. For each demand* $k\in K$ *with* $w_k\leq|I|$ *and* $e\notin E^k_0$, *consider a node* v_k *in* $\tilde{G}^e_I$. *Two nodes* v_k *and* $v_{k'}$ *are linked by an edge in* $\tilde{G}^e_I$ *if* $w_k+w_{k'}>|I|$ *and* $(k,k')\notin K^e_c$. *This is equivalent to say that two linked nodes* v_k *and* $v_{k'}$ *means that the two demands* k,k' *define a minimal cover for the interval* I *over edge* e.

For an edge $e\in E$, the conflict graph $\tilde{G}^e$ is a threshold graph with threshold value equals to $t=\bar{s}-\sum_{k"\in K_e} w_{k"}$ s.t. for eachnode v_k with $e\notin E^k_0\cup E^k_1$, we associate a positive weight $\tilde{w}_{v_k}=w_k$ s.t. all two nodes v_k and $v_{k'}$ are linked by an edge if and only if $\tilde{w}_{v_k}+\tilde{w}_{v_{k'}}>t$ which is equivalent to the conflict graph $\tilde{G}^e$.

Proposition 3. *Consider an edge* $e\in E$. *Let* $I=[s_i,s_j]$ *be an interval of contiguous slots. Let* C *be a clique in the conflict graph* $\tilde{G}^e_I$ *with* $|C|\geq 3$, *and* $\sum_{v_k\in C} w_k\leq \bar{s}-\sum_{k'\in K_e\setminus C} w_{k'}$. *Then, the inequality*

$$\sum_{v_k\in C}\sum_{p\in P^k(e)}\sum_{s=s_i+w_k-1}^{s_j} y^k_{p,s}\leq 1, \tag{9}$$

is valid for $P(G,K,\mathbb{S},P_K)$.

Proof. It's trivial given the definition of a clique set in the conflict graph $\tilde{G}^e_I$.

6.3 Interval-Clique Inequalities

Note that there may exist some cases that are not covered by the inequality (9). For this, we provide the following inequality and its generalization.

Proposition 4. *Consider an interval of contiguous slots* $I=[s_i,s_j]$ *in* $\mathbb{S}$ *with* $s_i\leq s_j-1$. *Let* k,k' *be a pair of demands in* K *with* $E^k_1\cap E^{k'}_1\neq\emptyset$, *and* $w_k\leq|I|$, *and* $w_{k'}\leq|I|$, *and* $w_k+w_{k'}>|I|$. *Then, the inequality*

$$\sum_{p\in P^k}\sum_{s=s_i+w_k-1}^{s_j} y^k_{p,s} + \sum_{p'\in P^{k'}}\sum_{s'=s_i+w_{k'}-1}^{s_j} y^{k'}_{p',s'}\leq 1, \tag{10}$$

is valid for $P(G,K,\mathbb{S},P_K)$.

Proof. It is trivial given that the interval $I=[s_i,s_j]$ cannot cover the two demands k,k' shared an essential edge with total sum of number of slots exceeds $|I|$. Furthermore, the inequality (10) is a particular case of the inequality (9) for $\tilde{K}=\{k,k'\}$ over each edge $e\in E^k_1\cap E^{k'}_1$. However, it will be used for a generalized inequality as follows.

Proposition 5. *Let $I = [s_i, s_j]$ be an interval of contiguous slots in $[1, \bar{s}]$ with $s_i \leq s_j - 1$, and C be a clique in the conflict graph $\tilde{G}_I^E$ with $|C| \geq 3$. Then, the inequality*

$$\sum_{v_k \in C} \sum_{p \in P^k} \sum_{s=s_i+w_k-1}^{s_j} y_{p,s}^k \leq 1, \tag{11}$$

is valid for $P(G, K, \mathbb{S}, P_K)$.

Proof. It is trivial given the definition of clique set in the conflict graph $\tilde{G}_I^E$ s.t. for all two linked node v_k and $v_{k'}$ in $\tilde{G}_I^E$, we know from the inequality (10)

$$\sum_{s=s_i+w_k-1}^{s_j} \sum_{p \in P^k} y_{p,s}^k + \sum_{s'=s_i+w_{k'}-1}^{s_j} \sum_{p' \in P^{k'}} y_{p',s'}^{k'} \leq 1.$$

By adding the previous inequalities for all two linked node v_k and $v_{k'}$ in the clique set C, we get

$$\sum_{v_k} (|C| - 1) \sum_{s=s_i+w_k-1}^{s_j} \sum_{p \in P^k} y_{p,s}^k \leq |C| - 1$$

$$\Longrightarrow \sum_{v_k}, \sum_{s=s_i+w_k-1}^{s_j} \sum_{p \in P^k} y_{p,s}^k \leq \frac{|C| - 1}{|C| - 1} \Longrightarrow \sum_{v_k \in C} \sum_{s=s_i+w_k-1}^{s_j} \sum_{p \in P^k} y_{p,s}^k \leq 1.$$

We conclude at the end that the inequality (11) is valid for $P(G, K, \mathbb{S}, P_K)$.

6.4 Interval-Odd-Hole Inequalities

Given that the conflict graph $\tilde{G}_I^E$ is not a perfect graph, one can use the so-called odd-hole to strengthen the valid inequalities introduced previously as follows.

Proposition 6. *Let $I = [s_i, s_j]$ be an interval of contiguous slots in $[1, \bar{s}]$ with $s_i \leq s_j - 1$, and H be an odd-hole H in the conflict graph $\tilde{G}_I^E$ with $|H| \geq 5$. Then, the inequality*

$$\sum_{v_k \in H} \sum_{s=s_i+w_k-1}^{s_j} \sum_{p \in P^k} y_{p,s}^k \leq \frac{|H| - 1}{2}, \tag{12}$$

is valid for $P(G, K, \mathbb{S}, P_K)$.

Proof. It is trivial given the definition of odd-hole set in the conflict graph $\tilde{G}_I^E$. We strengthen our proof as belows. For each pair of nodes $(v_k, v_{k'})$ linked in H by an edge, we know that $\sum_{s=s_i+w_k-1}^{s_j} \sum_{p \in P^k} y_{p,s}^k + \sum_{s'=s_i+w_{k'}-1}^{s_j} \sum_{p' \in P^{k'}} y_{p',s'}^{k'} \leq 1$. Given that H is an odd-hole which means that we have $|H| - 1$ pair of nodes $(v_k, v_{k'})$ linked in H, and by doing a sum for all pairs of nodes $(v_k, v_{k'})$ linked in H, it follows that

$$\sum_{(v_k, v_{k'}) \in E(H)} \sum_{s=s_i+w_k-1}^{s_j} \sum_{p \in P^k} y_{p,s}^k + \sum_{s'=s_i+w_{k'}-1}^{s_j} \sum_{p' \in P^{k'}} y_{p',s'}^{k'} \leq |H| - 1.$$

where $E(H)$ denotes the set of edges in the sub-graph of the conflict graph $\tilde{G}_I^E$ induced by H. Taking into account that each node v_k in H has two neighbors in H, this implies that $\sum_{s=s_i+w_k-1}^{s_j} \sum_{p \in P^k} y_{p,s}^k$ appears twice in the previous inequality. As a result,

$$\sum_{(v_k, v_{k'}) \in E(H)} \sum_{s=s_i+w_k-1}^{s_j} \sum_{p \in P^k} y_{p,s}^k + \sum_{s'=s_i+w_{k'}-1}^{s_j} \sum_{p' \in P^{k'}} y_{p',s'}^{k'} = \sum_{v_k \in H} 2 \sum_{s=s_i+w_k-1}^{s_j} \sum_{p \in P^k} y_{p,s}^k$$

$$\Longrightarrow \sum_{v_k \in H} 2 \sum_{s=s_i+w_k-1}^{s_j} \sum_{p \in P^k} y_{p,s}^k \leq |H| - 1.$$

By dividing the two sides of the previous sum by 2, it follows that

$$\sum_{v_k \in H} \sum_{s=s_i+w_k-1}^{s_j} \sum_{p \in P^k} y^k_{p,s} \leq \left\lfloor \frac{|H|-1}{2} \right\rfloor = \frac{|H|-1}{2} \text{ since } |H| \text{ is an odd number.}$$

We conclude at the end that the inequality (12) is valid for $P(G, K, \mathbb{S}, P_K)$.

The inequality (12) can be strengthened without modifying its right hand side by combining the inequality (11) and (12) as follows.

Proposition 7. *Consider an interval of contiguous slots $I = [s_i, s_j] \subseteq \mathbb{S}$ with $s_i \leq s_j - 1$. Let H be an odd-hole H in the conflict graph $\tilde{G}^E_I$, and C be a clique in the conflict graph $\tilde{G}^E_I$ with*

- $|H| \geq 5$,
- *and* $|C| \geq 3$,
- *and* $H \cap C = \emptyset$,
- *and the nodes* $(v_k, v_{k'})$ *are linked in* $\tilde{G}^E_I$ *for all* $v_k \in H$ *and* $v_{k'} \in C$.

Then, the inequality

$$\sum_{v_k \in H} \sum_{s=s_i+w_k-1}^{s_j} \sum_{p \in P^k} y^k_{p,s} + \frac{|H|-1}{2} \sum_{v_{k'} \in C} \sum_{s'=s_i+w_{k'}-1}^{s_j} \sum_{p' \in P^{k'}} y^{k'}_{p',s'} \leq \frac{|H|-1}{2}, \tag{13}$$

is valid for $P(G, K, \mathbb{S}, P_K)$.

Proof. It is trivial given the definition of odd-hole set and clique set in the conflict graph $\tilde{G}^E_I$ s.t. if $\sum_{s'=s_i+w_{k'}-1}^{s_j} \sum_{p' \in P^{k'}} y^{k'}_{p',s'} = 1$ for $v_{k'} \in C$, it forces the quantity $\sum_{v_k \in H} \sum_{s=s_i+w_k-1}^{s_j} \sum_{p \in P^k} y^k_{p,s}$ to be equal to 0. Otherwise, we know from the inequality (12) that the sum $\sum_{v_k \in H} \sum_{s=s_i+w_k-1}^{s_j} \sum_{p \in P^k} y^k_{p,s}$ is always smaller than $\frac{|H|-1}{2}$. We strengthen our proof by assuming that the inequality (13) is not valid for $P(G, K, \mathbb{S}, P_K)$. It follows that there exists a C-RSA solution S in which $\{s_i + w_{k'} - 1, ..., s_j\} \notin S_{k'}$ for each demand k' with node $v_{k'}$ in the clique C s.t.

$$\sum_{v_k \in H} \sum_{s=s_i+w_k-1}^{s_j} \sum_{p \in P^k} y^k_{p,s}(S) + \frac{|H|-1}{2} \sum_{v_{k'} \in C} \sum_{s'=s_i+w_{k'}-1}^{s_j} \sum_{p' \in P^{k'}} y^{k'}_{p',s'}(S) > \frac{|H|-1}{2}.$$

Since $\{s_i + w_{k'} - 1, ..., s_j\} \notin S_{k'}$ for each node $v_{k'}$ in the clique C, this means that $\sum_{v_{k'} \in C} \sum_{s'=s_i+w_{k'}-1}^{s_j} \sum_{p' \in P^{k'}} y^{k'}_{p',s'}(S) = 0$, and taking into account the inequality (12), and that $\sum_{s=s_i+w_k-1}^{s_j} \sum_{p \in P^k} y^k_{p,s}(S) \leq 1$ for each $v_k \in H$ and $\sum_{s'=s_i+w_{k'}-1}^{s_j} \sum_{p' \in P^{k'}} y^{k'}_{p',s'}(S) \leq 1$ for each $v_{k'} \in C$, it follows that $\sum_{v_k \in H} \sum_{s=s_i+w_k-1}^{s_j} \sum_{p \in P^k} y^k_{p,s}(S) \leq \frac{|H|-1}{2}$, which contradicts that $\sum_{v_k \in H} \sum_{s=s_i+w_k-1}^{s_j} \sum_{p \in P^k} y^k_{p,s}(S) + \frac{|H|-1}{2} \sum_{v_{k'} \in C} \sum_{s'=s_i+w_{k'}-1}^{s_j} \sum_{p' \in P^{k'}} y^{k'}_{p',s'}(S) > \frac{|H|-1}{2}$. Hence $\sum_{v_k \in H} |S_k \cap I_k| + \sum_{v_{k'} \in C} |S_{k'} \cap \{s_i + w_{k'} - 1, ..., s_j\}| \leq \frac{|H|-1}{2}$.

6.5 Edge-Slot-Assignment-Clique Inequalities

On the other hand, and based on the equations (3) and non-overlapping inequalities (4), we define a new conflict graph which contains the conflict graphs $\tilde{G}^E_S$ introduced previously as a sub-graph.

Definition 6. *Let $\tilde{G}^e_S$ be a conflict graph defined as follows. For each slot $s \in \{w_k, ..., \bar{s}\}$ and demand $k \in K$ with $e \notin E^k_0$, consider a node $v_{k,s}$ in $\tilde{G}^e_S$. Two nodes $v_{k,s}$ and $v_{k',s'}$ are linked by an edge in $\tilde{G}^e_S$ if and only if*

- $k = k'$,
- *or* $\{s - w_k + 1, ..., s\} \cap \{s' - w_{k'} + 1, ..., s'\} \neq \emptyset$ *if* $k \neq k'$ *and* $(k, k') \notin K^e_c$.

The conflict graph $\tilde{G}^e_S$ is not a perfect graph given that some nodes $v_{k,s}$ and $v_{k',s'}$ are linked even if the $\{s - w_k + 1, ..., s\} \cap \{s' - w_{k'} + 1, ..., s'\} = \emptyset$, i.e., when $k = k'$.

Proposition 8. *Consider an edge $e \in E$. Let C be a clique in the conflict graph $\tilde{G}_S^e$ with $|C| \geq 3$, and $\sum_{k \in C} w_k \leq \bar{s} - \sum_{k' \in K_e \setminus C} w_{k'}$. Then, the inequality*

$$\sum_{v_{k,s} \in C} \sum_{p \in P^k(e)} y_{p,s}^k \leq 1, \tag{14}$$

is valid for $P(G, K, \mathbb{S}, P_K)$.

Proof. It is trivial given the definition of a clique set in the conflict graph $\tilde{G}_S^e$ s.t. for each two linked nodes $v_{k,s}$ and $v_{k',s'}$ in $\tilde{G}_S^e$, we know that the inequality

$$\sum_{p \in P^k(e)} y_{p,s}^k + \sum_{p' \in P^{k'}(e)} y_{p',s'}^{k'} \leq 1,$$

is valid for $P(G, K, \mathbb{S}, P_K)$. By adding the previous inequalities for all two linked nodes $v_{k,s}$ and $v_{k',s'}$ in $\tilde{G}_S^E$, we get

$$\sum_{v_{k,s}} (|C|-1) y_{p,s}^k \leq |C| - 1 \implies \sum_{v_{k,s}} y_{p,s}^k \leq \frac{|C|-1}{|C|-1} \implies \sum_{v_{k,s}} y_{p,s}^k \leq 1,$$

which ends our proof.

Remark 1. The inequality (14) associated with a clique C over edge e, it is dominated by the inequality (4) associated with the slot $\tilde{s}$ and a subset of demands $\tilde{K}$ over edge e if and only if $\tilde{s} \in \{s - w_k + 1, ..., s\} \cap \{s' - w_{k'} + 1, s'\}$ for each pair of nodes $(v_{k,s}, v_{k',s'})$ in C.

Proof. Consider an edge $e \in E$. Let $\tilde{s}$ be a slot in $\mathbb{S}$, and C be a clique in the conflict graph $\tilde{G}_S^e$, and $\tilde{K} = \{k \in K \text{ s.t. } v_{k,s} \in C\}$ be a subset of demands in K with $e \notin E_0^k$ for each $k \in \tilde{K}$.
Neccessity.
First, assume that $\tilde{s} \in \{s - w_k + 1, ..., s\} \cap \{s' - w_{k'} + 1, s'\}$ for each pair of nodes $(v_{k,s}, v_{k',s'})$ in C. Taking into account that $|\{s - w_k + 1, ..., s\}| = w_k$ for each node $v_{k,s}$ in C, and $\tilde{s} \geq s - w_k + 1$ and $\tilde{s} \leq s$ for each $k \in \tilde{K}$, it follows that $s \in \{\tilde{s}, ...\tilde{s} + w_k - 1\}$ for each node $v_{k,s}$ in C. It follows that

$$\sum_{k \in \tilde{K}} \sum_{s'=\tilde{s}}^{min(\tilde{s}+w_k-1, \bar{s})} \sum_{p \in P^k(e)} y_{p,s'}^k = \sum_{k \in \tilde{K}} \sum_{p \in P^k(e)} y_{p,s}^k + \sum_{k \in \tilde{K}} \sum_{\substack{s'=\tilde{s} \\ s' \neq s \\ v_{k,s} \in C}}^{min(\tilde{s}+w_k-1, \bar{s})} \sum_{p \in P^k(e)} y_{p,s'}^k. \tag{15}$$

Taking into account that $\tilde{K} = \{k \in K \text{ s.t. } v_{k,s} \in C\}$, this means that

$$\sum_{k \in \tilde{K}} \sum_{p \in P^k(e)} y_{p,s}^k = \sum_{v_{k,s} \in C} \sum_{p \in P^k(e)} y_{p,s}^k.$$

This implies that

$$\sum_{k \in \tilde{K}} \sum_{s'=\tilde{s}}^{min(\tilde{s}+w_k-1, \bar{s})} \sum_{p \in P^k(e)} y_{p,s'}^k = \sum_{v_{k,s} \in C} \sum_{p \in P^k(e)} y_{p,s}^k + \sum_{k \in \tilde{K}} \sum_{\substack{s'=\tilde{s} \\ s' \neq s \\ v_{k,s} \in C}}^{min(\tilde{s}+w_k-1, \bar{s})} \sum_{p \in P^k(e)} y_{p,s'}^k$$

$$\implies \sum_{v_{k,s} \in C} \sum_{p \in P^k(e)} y_{p,s}^k \preceq \sum_{k \in \tilde{K}} \sum_{s'=\tilde{s}}^{min(\tilde{s}+w_k-1, \bar{s})} \sum_{p \in P^k(e)} y_{p,s'}^k.$$

As a result, the inequality (14) is dominated by the inequality (4).
Sufficiency.
Assume that the inequality (14) associated with the clique C over edge e, it is dominated by the

inequality (4) associated with the slot $\tilde{s}$ and subset of demands $\tilde{K}$ over edge e, and taking into account that $\tilde{K} = \{k \in K \text{ s.t. } v_{k,s} \in C\}$. We have

$$\sum_{v_{k,s}\in C}\sum_{p\in P^k(e)} y^k_{p,s} \preceq \sum_{k\in\tilde{K}}\sum_{s'=\tilde{s}}^{min(\tilde{s}+w_k-1,\bar{s})}\sum_{p\in P^k(e)} y^k_{p,s'} \implies \sum_{k\in\tilde{K}}\sum_{p\in P^k(e)} y^k_{p,s} \preceq \sum_{k\in\tilde{K}}\sum_{s'=\tilde{s}}^{min(\tilde{s}+w_k-1,\bar{s})}\sum_{p\in P^k(e)} y^k_{p,s'}$$

$$\implies \sum_{p\in P^k(e)} y^k_{p,s} \preceq \sum_{s'=\tilde{s}}^{min(\tilde{s}+w_k-1,\bar{s})}\sum_{p\in P^k(e)} y^k_{p,s'} \text{ for each } k \in \tilde{K}$$

$$\implies s \in \{\tilde{s}, ..., min(\tilde{s}+w_k-1,\bar{s})\} \text{ for each } k \in \tilde{K}$$

$$\implies s \in \{\tilde{s}, ..., min(\tilde{s}+w_k-1,\bar{s})\} \text{ for each node } v_{k,s} \in C,$$

$$\implies s \geq \tilde{s} \text{ and } s \leq \tilde{s}+w_k-1 \implies s-w_k+1 \leq \tilde{s} \leq s \text{ for each node } v_{k,s} \in C,$$

$$\implies \tilde{s} \in \{s-w_k+1, ..., s\} \text{ for each node } v_{k,s} \in C.$$

It follows that $\tilde{s} \in \{s-w_k+1, ..., s\} \cap \{s'-w_{k'}+1, s'\}$ for each pair of nodes $(v_{k,s}, v_{k',s'})$ in C, that which was to be demonstrated, and which ends our proof.

Remark 2. The inequality (14) associated with a clique C over edge e, it is dominated by the inequality (9) associated with an interval $I = [s_i, s_j]$ and the subset of demands $\tilde{K}$ over edge e iff

- $\tilde{s} \in \{s-w_k+1, ..., s\} \cap \{s'-w_{k'}+1, s'\}$ for each pair of nodes $(v_{k,s}, v_{k',s'})$ in C,
- and $[\min_{v_{k,s}\in C}(s-w_k+1), \max_{v_{k,s}\in C} s] \subset I$.

Proof. Consider an edge $e \in E$, and an interval of contiguous slots $I = [s_i, s_j] \subseteq [1, \bar{s}]$. Let C be a clique in the conflict graph $\tilde{G}^e_S$, and $\tilde{K} = \{k \in K \text{ s.t. } v_{k,s} \in C\}$ be a subset of demands in K with

- $w_k + w_{k'} > |I|$ for each pair of demands $k, k' \in \tilde{K}$,
- and $\sum_{k\in\tilde{K}} w_k \leq \bar{s} - \sum_{k'\in K_e\setminus\tilde{K}} w_{k'}$,
- and $e \notin E^k_0$, and $w_k \leq |I|$ for each demand $k \in \tilde{K}$.

Necessity
First, assume that

- $\tilde{s} \in \{s-w_k+1, ..., s\} \cap \{s'-w_{k'}+1, s'\}$ for each pair of nodes $(v_{k,s}, v_{k',s'})$ in C,
- and $[\min_{v_{k,s}\in C}(s-w_k+1), \max_{v_{k,s}\in C} s] \subset I$.

Given that $s - w_k + 1 \geq \min_{v_{k',s'}\in C}(s'-w_{k'}+1)$ and $s \leq \max_{v_{k',s'}\in C} s'$ for each $v_{k,s} \in C$, and that $|\{s-w_k+1, ..., s\}| = w_k$ for each $v_{k,s} \in C$, it follows that $s \in I_k$ for each $v_{k,s} \in C$ of demand $k \in \tilde{K}$. As a result, we get that

$$\sum_{k\in\tilde{K}}\sum_{s'\in I_k}\sum_{p\in P^k(e)} y^k_{p,s'} = \sum_{k\in\tilde{K}}\sum_{p\in P^k(e)} y^k_{p,s} + \sum_{k\in\tilde{K}}\sum_{s'\in I_k\setminus\{s\}}\sum_{p\in P^k(e)} y^k_{p,s'}. \tag{16}$$

Taking into account that $\tilde{K} = \{k \in K \text{ s.t. } v_{k,s} \in C\}$, this means that

$$\sum_{k\in\tilde{K}}\sum_{p\in P^k(e)} y^k_{p,s} = \sum_{v_{k,s}\in C}\sum_{p\in P^k(e)} y^k_{p,s}.$$

This implies that

$$\sum_{k\in\tilde{K}}\sum_{s'\in I_k}\sum_{p\in P^k(e)} y^k_{p,s'} = \sum_{v_{k,s}\in C}\sum_{p\in P^k(e)} y^k_{p,s} + \sum_{k\in\tilde{K}}\sum_{s'\in I_k\setminus\{s\}}\sum_{p\in P^k(e)} y^k_{p,s'}$$

$$\implies \sum_{p\in P^k(e)} y^k_{p,s} \preceq \sum_{k\in\tilde{K}}\sum_{s'\in I_k}\sum_{p\in P^k(e)} y^k_{p,s'}.$$

Hence, the inequality (14) is dominated by the inequality (9).
Sufficiency.
Assume that the inequality (14) associated with the clique C over edge e, it is dominated by the inequality (9) associated with the interval $I = [s_i, s_j]$ and the subset of demands $\tilde{K}$ over edge e. Taking into account that $\tilde{K} = \{k \in K \text{ s.t. } v_{k,s} \in C\}$, it follows that

$$\sum_{v_{k,s} \in C} \sum_{p \in P^k(e)} y^k_{p,s} \preceq \sum_{k \in \tilde{K}} \sum_{s' \in I_k} \sum_{p \in P^k(e)} y^k_{p,s'} \implies \sum_{k \in \tilde{K}} \sum_{p \in P^k(e)} y^k_{p,s} \preceq \sum_{k \in \tilde{K}} \sum_{s' \in I_k} \sum_{p \in P^k(e)} y^k_{p,s'}$$
$$\implies \sum_{p \in P^k(e)} y^k_{p,s} \preceq \sum_{s' \in I_k} \sum_{p \in P^k(e)} y^k_{p,s'} \text{ for each } k \in \tilde{K} \implies s \in I_k \text{ for each } k \in \tilde{K}$$
$$\implies s \in I_k \text{ for each node } v_{k,s} \in C \implies s - w_k + 1 \in I \text{ for each node } v_{k,s} \in C$$
$$\implies \min_{v_{k,s} \in C} (s - w_k + 1) \in I \text{ and } \max_{v_{k,s} \in C} s \in I \text{ for each node } v_{k,s} \in C$$
$$\implies [\min_{v_{k,s} \in C} (s - w_k + 1), \max_{v_{k,s} \in C} s] \subseteq I.$$

Furthermore, and given that $w_k + w_{k'} > |I|$ for each pair of demands $k, k' \in \tilde{K}$, it follows that $\{s - w_k + 1, ..., s\} \cap \{s' - w_{k'} + 1, ..., s\} \neq \emptyset$ for each $s \in I_k$ and $s' \in I_{k'}$ of each pair of demands $k, k' \in \tilde{K}$. It follows that $\{s - w_k + 1, ..., s\} \cap \{s' - w_{k'} + 1, ..., s\} \neq \emptyset$ for each pair $(v_{k,s}, v_{k',s'}) \in C$ since $s \in I_k$ and $s' \in I_{k'}$. We conclude at the end that

- $\tilde{s} \in \{s - w_k + 1, ..., s\} \cap \{s' - w_{k'} + 1, s'\}$ for each pair of nodes $(v_{k,s}, v_{k',s'})$ in C,
- and $[\min_{v_{k,s} \in C} (s - w_k + 1), \max_{v_{k,s} \in C} s] \subset I$,

which ends our proof.

6.6 Edge-Slot-Assignment-Odd-Hole Inequalities

The conflict graph $\tilde{G}^e_S$ is not a perfect graph given that some nodes $v_{k,s}$ and $v_{k',s'}$ are linked even if the $\{s-w_k+1, ..., s\} \cap \{s'-w_{k'}+1, ..., s'\} = \emptyset$, i.e., when $k = k'$. As a result, we define the following inequalities based on the so-called odd-hole inequalities that may allow us to obtain tighter LP bounds.

Proposition 9. *Let H be an odd-hole in the conflict graph $\tilde{G}^e_S$ with $|H| \geq 5$. Then, the inequality*

$$\sum_{v_{k,s} \in H} \sum_{p \in P^k(e)} y^k_{p,s} \leq \frac{|H| - 1}{2}, \tag{17}$$

is valid for $P(G, K, \mathbb{S}, P_K)$.

Proof. It is trivial given the definition of the odd-hole in the conflict graph $\tilde{G}^e_S$. We strengthen our proof as belows. For each pair of nodes $(v_{k,s}, v_{k',s'})$ linked in H by an edge, we know that $\sum_{p \in P^k(e)} y^k_{p,s} + \sum_{p' \in P^{k'}(e)} y^{k'}_{p',s'} \leq 1$. Given that H is an odd-hole which means that we have $|H| - 1$ pair of nodes $(v_{k,s}, v_{k',s'})$ linked in H, and by doing a sum for all pairs of nodes $(v_{k,s}, v_{k',s'})$ linked in H, it follows that

$$\sum_{(v_{k,s}, v_{k',s'}) \in E(H)} \sum_{p \in P^k(e)} y^k_{p,s} + \sum_{p' \in P^{k'}(e)} y^{k'}_{p',s'} \leq |H| - 1.$$

Taking into account that each node v_k in H has two neighbors in H, this implies that $\sum_{p \in P^k} y^k_{p,s}$ appears twice in the previous inequality. As a result,

$$\sum_{(v_{k,s}, v_{k',s'}) \in E(H)} \sum_{p \in P^k(e)} y^k_{p,s} + \sum_{p' \in P^{k'}(e)} y^{k'}_{p',s'} = \sum_{v_{k,s} \in H} 2 \sum_{p \in P^k(e)} y^k_{p,s} \implies \sum_{v_{k,s} \in H} 2 \sum_{p \in P^k(e)} y^k_{p,s} \leq |H| - 1$$
$$\implies \sum_{v_{k,s} \in H} \sum_{p \in P^k(e)} y^k_{p,s} \leq \left\lfloor \frac{|H| - 1}{2} \right\rfloor = \frac{|H| - 1}{2} \text{ since } |H| \text{ is an odd number.}$$

We conclude at the end that the inequality (17) is valid for $P(G, K, \mathbb{S}, P_K)$.

Note that the inequality (17) can be strengthened without modifying its right hand side by combining the inequality (17) and (14).

Proposition 10. *Let H be an odd-hole, and C be a clique in the conflict graph $\tilde{G}^e_S$ with*

- $|H| \geq 5$,
- *and* $|C| \geq 3$,
- *and* $H \cap C = \emptyset$,
- *and the nodes* $(v_{k,s}, v_{k',s'})$ *are linked in* $\tilde{G}^e_S$ *for all* $v_{k,s} \in H$ *and* $v_{k',s'} \in C$.

Then, the inequality

$$\sum_{v_{k,s} \in H} \sum_{p \in P^k(e)} y^k_{p,s} + \frac{|H|-1}{2} \sum_{v_{k',s'} \in C} \sum_{p' \in P^{k'}(e)} y^{k'}_{p',s'} \leq \frac{|H|-1}{2}, \tag{18}$$

is valid for $P(G, K, \mathbb{S}, P_K)$.

Proof. It is trivial given the definition of the odd-hole and clique in $\tilde{G}^e_S$ s.t. if $\sum_{v_{k',s'} \in C} \sum_{p' \in P^{k'}(e)} y^{k'}_{p',s'} = 1$ for a $v_{k',s'} \in C \in C$ which implies that the quantity $\sum_{v_{k,s} \in H} \sum_{p \in P^k(e)} y^k_{p,s}$ is forced to be equal to 0. Otherwise, we know from the inequality (17) that the sum $\sum_{v_{k,s} \in H} \sum_{p \in P^k(e)} y^k_{p,s}$ is always smaller than $\frac{|H|-1}{2}$. We strengthen our proof by assuming that the inequality (18) is not valid for $P(G, K, \mathbb{S}, P_K)$. It follows that there exists a C-RSA solution S in which $s' \notin S_{k'}$ for each node $v_{k',s'}$ in the clique C s.t.

$$\sum_{v_{k,s} \in H} y^k_{p,s}(S) + \frac{|H|-1}{2} \sum_{v_{k',s'} \in C} y^{k'}_{p',s'}(S) > \frac{|H|-1}{2}.$$

Since $s' \notin S_{k'}$ for each node $v_{k',s'}$ in the clique C this means that $\sum_{v_{k',s'} \in C} \sum_{p' \in P^{k'}(e)} y^{k'}_{p',s'}(S) = 0$, and taking into account the inequality (17), $\sum_{p \in P^k(e)} y^k_{p,s}(S) \leq 1$ for each $v_{k,s} \in H$, and that $\sum_{p' \in P^{k'}(e)} y^{k'}_{p',s'}(S) \leq 1$ for each $v_{k',s'} \in C$, it follows that

$$\sum_{v_{k,s} \in H} \sum_{p \in P^k(e)} y^k_{p,s}(S) \leq \frac{|H|-1}{2},$$

which contradicts that $\sum_{v_{k,s} \in H} \sum_{p \in P^k(e)} y^k_{p,s}(S) + \frac{|H|-1}{2} \sum_{v_{k',s'} \in C} \sum_{p' \in P^{k'}(e)} y^{k'}_{p',s'}(S) > \frac{|H|-1}{2}$.
Hence $\sum_{v_{k,s} \in H} |S_k \cap \{s\}| + \sum_{v_{k',s'} \in C} |S_{k'} \cap \{s'\}| \leq \frac{|H|-1}{2}$.

6.7 Slot-Assignment-Clique Inequalities

Note that there may exist some cases that are not covered by the inequalities (14)-(17). For this, we provide the following definition of a conflict graph and its associated inequality.

Definition 7. *Let $\tilde{G}^E_S$ be a conflict graph defined as follows. For all slot $s \in \{w_k, ..., \bar{s}\}$ and demand $k \in K$, consider a node $v_{k,s}$ in $\tilde{G}^E_S$. Two nodes $v_{k,s}$ and $v_{k',s'}$ are linked by an edge in $\tilde{G}^E_S$ if $k = k'$ or $E^k_1 \cap E^{k'}_1 \neq \emptyset$ and $\{s - w_k + 1, ..., s\} \cap \{s' - w_{k'} + 1, ..., s'\} \neq \emptyset$ when $k \neq k'$.*

The conflict graph $\tilde{G}^E_S$ is not a interval graph given that some nodes $v_{k,s}$ and $v_{k',s'}$ are linked even if the $\{s - w_k + 1, ..., s\} \cap \{s' - w_{k'} + 1, ..., s'\} = \emptyset$, i.e., when $k = k'$.
Based on the conflict graph $\tilde{G}^E_S$, we provide the following inequality.

Proposition 11. *Let C be a clique in conflict graph $\tilde{G}^E_S$ with $|C| \geq 3$. Then, the inequality*

$$\sum_{v_{k,s} \in C} \sum_{p \in P^k} y^k_{p,s} \leq 1, \tag{19}$$

is valid for $P(G, K, \mathbb{S}, P_K)$.

Proof. It is trivial given the definition of a clique set in the conflict graph $\tilde{G}_S^E$ s.t. for each two linked nodes $v_{k,s}$ and $v_{k',s'}$ in $\tilde{G}_S^E$, we know that the inequality

$$\sum_{p \in P^k} y_{p,s}^k + \sum_{p' \in P^{k'}} y_{p',s'}^{k'} \leq 1,$$

is valid for $P(G, K, \mathbb{S}, P_K)$. By adding the previous inequalities for all two linked nodes $v_{k,s}$ and $v_{k',s'}$ in $\tilde{G}_S^E$, we get

$$\sum_{v_{k,s}} (|C|-1) y_{p,s}^k \leq |C|-1 \implies \sum_{v_{k,s}} y_{p,s}^k \leq \frac{|C|-1}{|C|-1} \implies \sum_{v_{k,s}} y_{p,s}^k \leq 1,$$

which ends our proof.

Remark 3. The inequality (19) associated with a clique C, it is dominated by the inequality (11) associated with an interval $I = [s_i, s_j]$ and a subset of demands $\tilde{K}$ if and only if $[\min_{v_{k,s} \in C}(s - w_k + 1), \max_{v_{k,s} \in C} s] \subset I$ and $w_k + w_{k'} \geq |I| + 1$ for each $(v_k, v_{k'}) \in C$, and $2w_k \geq |I| + 1$ and $w_k \leq |I|$ for each $v_k \in C$.

Proof. Consider an interval of contiguous slots $I = [s_i, s_j] \subseteq [1, \bar{s}]$. Let C be a clique in the conflict graph $\tilde{G}_S^E$, and $\tilde{K} = \{k \in K \text{ s.t. } v_{k,s} \in C\}$ be a subset of demands in K with $\tilde{K}$ is a clique in the conflict graph $\tilde{G}_I^E$ for the interval $I = [s_i, s_j]$.

Neccessity.

First, assume that

- $\tilde{s} \in \{s - w_k + 1, ..., s\} \cap \{s' - w_{k'} + 1, s'\}$ for each pair of nodes $(v_{k,s}, v_{k',s'})$ in C,
- and $[\min_{v_{k,s} \in C}(s - w_k + 1), \max_{v_{k,s} \in C} s] \subset I$.

Given that $s - w_k + 1 \geq \min_{v_{k',s'} \in C}(s' - w_{k'} + 1)$ and $s \leq \max_{v_{k',s'} \in C} s'$ for each $v_{k,s} \in C$, and that $|\{s - w_k + 1, ..., s\}| = w_k$ for each $v_{k,s} \in C$, it follows that $s \in I_k = [s_i + w_k - 1, s_j]$ for each $v_{k,s} \in C$ of demand $k \in \tilde{K}$. As a result, we get that

$$\sum_{k \in \tilde{K}} \sum_{s' \in I_k} \sum_{p \in P^k} y_{p,s'}^k = \sum_{k \in \tilde{K}} \sum_{p \in P^k} y_{p,s}^k + \sum_{k \in \tilde{K}} \sum_{s' \in I_k \setminus \{s\}} \sum_{p \in P^k} y_{p,s'}^k. \tag{20}$$

Taking into account that $\tilde{K} = \{k \in K \text{ s.t. } v_{k,s} \in C\}$, this means that

$$\sum_{k \in \tilde{K}} \sum_{p \in P^k} y_{p,s}^k = \sum_{v_{k,s} \in C} \sum_{p \in P^k} y_{p,s}^k.$$

It follows that

$$\sum_{k \in \tilde{K}} \sum_{s' \in I_k} \sum_{p \in P^k} y_{p,s'}^k = \sum_{v_{k,s} \in C} \sum_{p \in P^k} y_{p,s}^k + \sum_{k \in \tilde{K}} \sum_{s' \in I_k \setminus \{s\}} \sum_{p \in P^k} y_{p,s'}^k.$$

Given that all the variable $y_{p,s}^k$ is positive for each $k \in K$ and $s \in \mathbb{S}$, this implies that

$$\sum_{v_{k,s} \in C} \sum_{p \in P^k} y_{p,s}^k \preceq \sum_{k \in \tilde{K}} \sum_{s' \in I_k} \sum_{p \in P^k} y_{p,s'}^k.$$

Hence, the inequality (19) is dominated by the inequality (11).

Sufficiency.

Assume that the inequality (19) is dominated by the inequality (11). It follows that

$$\sum_{v_{k,s} \in C} \sum_{p \in P^k} y_{p,s}^k \preceq \sum_{k \in \tilde{K}} \sum_{s' \in I_k} \sum_{p \in P^k} y_{p,s'}^k \implies \sum_{k \in \tilde{K}} \sum_{p \in P^k} y_{p,s}^k \preceq \sum_{k \in \tilde{K}} \sum_{s' \in I_k} \sum_{p \in P^k} y_{p,s'}^k$$

Given that the demands in $\tilde{K}$ are independants, this allows us to take that

$$\sum_{p\in P^k} y^k_{p,s} \preceq \sum_{p\in P^k}\sum_{s'\in I_k} y^k_{p,s'} \text{ for each } k\in\tilde{K}.$$

Given that the variable $\sum_{p\in P^k} y^k_{p,s}$ is positive for each $k\in K$ and $s\in\mathbb{S}$, this means that

$$s\in I_k \text{ for each } k\in\tilde{K},$$

which is equivalent to say that

$$s\in I_k \text{ for each node } v_{k,s}\in C \implies s\in\{s_i+w_k-1,...,s_j\}.$$

It follows that

$$s-w_k+1\in I \text{ for each node } v_{k,s}\in C.$$

As a result,

$$\min_{v_{k,s}\in C}(s-w_k+1)\in I \text{ and } \max_{v_{k,s}\in C} s\in I \text{ for each node } v_{k,s}\in C$$
$$\implies [\min_{v_{k,s}\in C}(s-w_k+1), \max_{v_{k,s}\in C} s]\subseteq I.$$

Furthermore, and given that $w_k+w_{k'}>|I|$ for each pair of demands $k,k'\in\tilde{K}$, it follows that $\{s-w_k+1,...,s\}\cap\{s'-w_{k'}+1,...,s\}\neq\emptyset$ for each $s\in I_k$ and $s'\in\{s_i+w_{k'}-1,...,s_j\}$ of each pair of demands $k,k'\in\tilde{K}$. Hence, $\{s-w_k+1,...,s\}\cap\{s'-w_{k'}+1,...,s\}\neq\emptyset$ for each pair $(v_{k,s},v_{k',s'})\in C$ since $s\in I_k$ and $s'\in\{s_i+w_{k'}-1,...,s_j\}$. We conclude at the end that

- $\tilde{s}\in\{s-w_k+1,...,s\}\cap\{s'-w_{k'}+1,s'\}$ for each pair of nodes $(v_{k,s},v_{k',s'})$ in C,
- and $[\min_{v_{k,s}\in C}(s-w_k+1), \max_{v_{k,s}\in C} s]\subset I$,

which ends our proof.

6.8 Slot-Assignment-Odd-Hole Inequalities

We have observed that the conflict graph $\tilde{G}^E_S$ cannot define a interval graph graph given that it contains some nodes $v_{k,s}$ and $v_{k',s'}$ that are linked even if the $\{s-w_k+1,...,s\}\cap\{s'-w_{k'}+1,...,s'\}=\emptyset$, i.e., when $k=k'$. As a result, one can strengthen the inequality (19) by introducing the following inequalities based on the so-called odd-hole inequalities.

Proposition 12. *Let H be an odd-hole in the conflict graph $\tilde{G}^E_S$ with $|H|\geq 5$. Then, the inequality*

$$\sum_{v_{k,s}\in H}\sum_{p\in P^k} y^k_{p,s}\leq\frac{|H|-1}{2}, \tag{21}$$

is valid for $P(G,K,\mathbb{S},P_K)$.

Proof. It is trivial given the definition of the odd-hole in the conflict graph $\tilde{G}^E_S$. We strengthen our proof as belows. For each pair of nodes $(v_{k,s},v_{k',s'})$ linked in H by an edge, we know that $\sum_{p\in P^k} y^k_{p,s}+\sum_{p'\in P^{k'}} y^{k'}_{p',s'}\leq 1$. Given that H is an odd-hole which means that we have $|H|-1$ pair of nodes $(v_{k,s},v_{k',s'})$ linked in H, and by doing a sum for all pairs of nodes $(v_{k,s},v_{k',s'})$ linked in H, it follows that

$$\sum_{(v_{k,s},v_{k',s'})\in E(H)}\sum_{p\in P^k} y^k_{p,s}+\sum_{p'\in P^{k'}} y^{k'}_{p',s'}\leq|H|-1.$$

Taking into account that each node v_k in H has two neighbors in H, this implies that $\sum_{p\in P^k} y^k_{p,s}$ appears twice in the previous inequality. As a result,

$$\sum_{(v_{k,s},v_{k',s'})\in E(H)} \sum_{p\in P^k} y^k_{p,s} + \sum_{p'\in P^{k'}} y^{k'}_{p',s'} = \sum_{v_{k,s}\in H} 2 \sum_{p\in P^k} y^k_{p,s} \implies \sum_{v_{k,s}\in H} 2 \sum_{p\in P^k} y^k_{p,s} \leq |H|-1$$

$$\implies \sum_{v_{k,s}\in H} \sum_{p\in P^k} y^k_{p,s} \leq \left\lfloor \frac{|H|-1}{2} \right\rfloor = \frac{|H|-1}{2} \text{ since } |H| \text{ is an odd number.}$$

We conclude at the end that the inequality (21) is valid for $P(G, K, \mathbb{S}, P_K)$.

Remark 4. The inequality (21) is dominated by the inequality (12) if and only if there exists an interval of contiguous slots $I = [s_i, s_j] \subset [1, \bar{s}]$ with

- $[\min_{v_{k,s}\in H}(s - w_k + 1), \max_{v_{k,s}\in H}] \subset I$,
- and $w_k + w_{k'} \geq |I| + 1$ for each $(v_k, v_{k'})$ linked in H,
- and $2w_k \geq |I| + 1$ and $w_k \leq |I|$ for each $v_k \in H$.

Proof. Consider an interval of contiguous slots $I = [s_i, s_j] \subseteq [1, \bar{s}]$. Let H be an odd-hole in the conflict graph $\tilde{G}^E_S$, and $\tilde{K} = \{k \in K \text{ s.t. } v_{k,s} \in H\}$ be a subset of demands in K with $\tilde{K}$ is an odd-hole in the conflict graph $\tilde{G}^E_I$ for the interval $I = [s_i, s_j]$.

Neccessity.

First, assume that

- $\tilde{s} \in \{s - w_k + 1, ..., s\} \cap \{s' - w_{k'} + 1, s'\}$ for each pair of nodes $(v_{k,s}, v_{k',s'})$ in H,
- and $[\min_{v_{k,s}\in H}(s - w_k + 1), \max_{v_{k,s}\in H} s] \subset I$.

Given that $s - w_k + 1 \geq \min_{v_{k',s'}\in H}(s' - w_{k'} + 1)$ and $s \leq \max_{v_{k',s'}\in H} s'$ for each $v_{k,s} \in H$, and that $|\{s - w_k + 1, ..., s\}| = w_k$ for each $v_{k,s} \in H$, it follows that $s \in I_k = [s_i + w_k - 1, s_j]$ for each $v_{k,s} \in H$ of demand $k \in \tilde{K}$. As a result, we get that

$$\sum_{k\in\tilde{K}} \sum_{s'\in I_k} \sum_{p\in P^k} y^k_{p,s'} = \sum_{k\in\tilde{K}} \sum_{p\in P^k} y^k_{p,s} + \sum_{k\in\tilde{K}} \sum_{s'\in I_k\setminus\{s\}} \sum_{p\in P^k} y^k_{p,s'}. \tag{22}$$

Taking into account that $\tilde{K} = \{k \in K \text{ s.t. } v_{k,s} \in H\}$, this means that

$$\sum_{k\in\tilde{K}} \sum_{p\in P^k} y^k_{p,s} = \sum_{v_{k,s}\in H} \sum_{p\in P^k} y^k_{p,s}.$$

This implies that

$$\sum_{k\in\tilde{K}} \sum_{p\in P^k} \sum_{s'\in I_k} y^k_{p,s'} = \sum_{v_{k,s}\in H} \sum_{p\in P^k} y^k_{p,s} + \sum_{k\in\tilde{K}} \sum_{p\in P^k} \sum_{s'\in I_k\setminus\{s\}} y^k_{p,s'}$$

$$\implies \sum_{v_{k,s}\in H} \sum_{p\in P^k} y^k_{p,s} \preceq \sum_{k\in\tilde{K}} \sum_{s'\in I_k} \sum_{p\in P^k} y^k_{p,s'} \implies \sum_{p\in P^k} y^k_{p,s} \preceq \sum_{s'\in I_k} \sum_{p\in P^k} y^k_{p,s'} \text{ for each } v_{k,s} \in H.$$

Hence, the inequality (21) is dominated by the inequality (12).

Sufficiency.

Assume that the inequality (21) is dominated by the inequality (12) and given that $\tilde{K} = \{k \in K \text{ s.t. } v_{k,s} \in H\}$, this means that

$$\sum_{k\in\tilde{K}} \sum_{p\in P^k} y^k_{p,s} = \sum_{v_{k,s}\in H} \sum_{p\in P^k} y^k_{p,s}.$$

It follows that

$$\sum_{v_{k,s}\in H} \sum_{p\in P^k} y^k_{p,s} \preceq \sum_{k\in\tilde{K}} \sum_{s'\in I_k} \sum_{p\in P^k} y^k_{p,s'} \implies \sum_{k\in\tilde{K}} \sum_{p\in P^k} y^k_{p,s} \preceq \sum_{k\in\tilde{K}} \sum_{s'\in I_k} \sum_{p\in P^k} y^k_{p,s'}.$$

Given that the demands in $\tilde{K}$ are independants, this implies that

$$\sum_{p\in P^k} y^k_{p,s} \preceq \sum_{s'\in I_k}\sum_{p\in P^k} y^k_{p,s'} \text{ for each } k\in\tilde{K} \implies s\in I_k \text{ for each } k\in\tilde{K} \implies s\in I_k \text{ for each node } v_{k,s}\in H.$$

As a result,

$$s-w_k+1\in I \text{ for each node } v_{k,s}\in H \implies \min_{v_{k,s}\in H}(s-w_k+1)\in I$$
$$\text{and } \max_{v_{k,s}\in H} s\in I \text{ for each node } v_{k,s}\in H \implies [\min_{v_{k,s}\in H}(s-w_k+1), \max_{v_{k,s}\in H} s]\subseteq I.$$

Furthermore, and given that $w_k + w_{k'} > |I|$ for each pair of demands $k, k' \in \tilde{K}$, it follows that $\{s - w_k + 1, ..., s\} \cap \{s' - w_{k'} + 1, ..., s\} \neq \emptyset$ for each $s \in I_k$ and $s' \in \{s_i + w_{k'} - 1, ..., s_j\}$ of each pair of demands $k, k' \in \tilde{K}$. Hence, $\{s - w_k + 1, ..., s\} \cap \{s' - w_{k'} + 1, ..., s\} \neq \emptyset$ for each pair $(v_{k,s}, v_{k',s'}) \in H$ since $s \in I_k$ and $s' \in \{s_i + w_{k'} - 1, ..., s_j\}$. We conclude at the end that

- $\tilde{s} \in \{s - w_k + 1, ..., s\} \cap \{s' - w_{k'} + 1, s'\}$ for each pair of nodes $(v_{k,s}, v_{k',s'})$ in H,
- and $[\min_{v_{k,s}\in H}(s-w_k+1), \max_{v_{k,s}\in H} s]\subset I$,

which ends our proof.

Note that the inequality (21) can be strengthened without modifying its right hand side by combining the inequality (21) and (19).

Proposition 13. *Let H be an odd-hole, and C be a clique in the conflict graph $\tilde{G}^E_S$ with*

- $|H| \geq 5$,
- *and* $|C| \geq 3$,
- *and* $H \cap C = \emptyset$,
- *and the nodes* $(v_{k,s}, v_{k',s'})$ *are linked in* $\tilde{G}^E_S$ *for all* $v_{k,s} \in H$ *and* $v_{k',s'} \in C$.

Then, the inequality

$$\sum_{v_{k,s}\in H}\sum_{p\in P^k} y^k_{p,s} + \frac{|H|-1}{2}\sum_{v_{k',s'}\in C}\sum_{p'\in P^{k'}} y^{k'}_{p',s'} \leq \frac{|H|-1}{2}, \tag{23}$$

is valid for $P(G, K, \mathbb{S}, P_K)$.

Proof. It is trivial given the definition of the odd-hole and clique in $\tilde{G}^E_S$ s.t. if $\sum_{v_{k',s'}\in C}\sum_{p'\in P^{k'}} y^{k'}_{p',s'} = 1$ for a $v_{k',s'} \in C \in C$ which implies that the quantity $\sum_{v_{k,s}\in H}\sum_{p\in P^k} y^k_{p,s}$ is forced to be equal to 0. Otherwise, we know from the inequality (21) that the sum $\sum_{v_{k,s}\in H}\sum_{p\in P^k} y^k_{p,s}$ is always smaller than $\frac{|H|-1}{2}$. We strengthen our proof by assuming that the inequality (23) is not valid for $P(G, K, \mathbb{S}, P_K)$. It follows that there exists a C-RSA solution S in which $s' \notin S_{k'}$ for each node $v_{k',s'}$ in the clique C s.t.

$$\sum_{v_{k,s}\in H} y^k_{p,s}(S) + \frac{|H|-1}{2}\sum_{v_{k',s'}\in C} y^{k'}_{p',s'}(S) > \frac{|H|-1}{2}.$$

Since $s' \notin S_{k'}$ for each node $v_{k',s'}$ in the clique C this means that $\sum_{v_{k',s'}\in C}\sum_{p'\in P^{k'}} y^{k'}_{p',s'}(S) = 0$, and taking into account the inequality (21), $\sum_{p\in P^k} y^k_{p,s}(S) \leq 1$ for each $v_{k,s} \in H$, and that $\sum_{p'\in P^{k'}} y^{k'}_{p',s'}(S) \leq 1$ for each $v_{k',s'} \in C$, it follows that

$$\sum_{v_{k,s}\in H}\sum_{p\in P^k} y^k_{p,s}(S) \leq \frac{|H|-1}{2},$$

which contradicts that $\sum_{v_{k,s}\in H}\sum_{p\in P^k} y^k_{p,s}(S) + \frac{|H|-1}{2}\sum_{v_{k',s'}\in C}\sum_{p'\in P^{k'}} y^{k'}_{p',s'}(S) > \frac{|H|-1}{2}$. Hence $\sum_{v_{k,s}\in H} |S_k \cap \{s\}| + \sum_{v_{k',s'}\in C} |S_{k'} \cap \{s'\}| \leq \frac{|H|-1}{2}$.

Remark 5. The inequality (23) is dominated by the inequality (13) iff there exists an interval of contiguous slots $I = [s_i, s_j] \subset [1, \bar{s}]$ with

- $[\min_{v_{k,s} \in H \cup C}(s - w_k + 1), \max_{v_{k,s} \in H \cup C}] \subset I$,
- and $w_k + w_{k'} \geq |I| + 1$ for each $(v_k, v_{k'})$ linked in H,
- and $w_k + w_{k'} \geq |I| + 1$ for each $(v_k, v_{k'})$ linked in C,
- and $w_k + w_{k'} \geq |I| + 1$ for each $v_k \in H$ and $v_{k'} \in C$,
- and $2w_k \geq |I| + 1$ and $w_k \leq |I|$ for each $v_k \in H$,
- and $2w_{k'} \geq |I| + 1$ and $w_{k'} \leq |I|$ for each $v_{k'} \in C$.

Proof. Similar with the proof of the remark 4.

6.9 Edge-Capacity-Cover Inequalities

Let's us now provide some inequalities related to the capacity constraint.

Proposition 14. *Consider an edge e in E. Then, the inequality*

$$\sum_{k \in K \setminus K_e} w_k \sum_{p \in P^k(e)} \sum_{s=1}^{\bar{s}} y_{p,s}^k \leq \bar{s} - \sum_{k' \in K_e} w_{k'}, \tag{24}$$

is valid for $P(G, K, \mathbb{S}, P_K)$.

Proof. The total number of slots allocated over the edge $e \in E$ should be less than the residual capacity of the edge e which is equal to $\bar{s} - \sum_{k' \in K_e} w_{k'}$.

Based on this, we introduce the following definitions.

Definition 8. *For an edge $e \in E$, a subset of demands $C \subseteq K$ with $e \notin E_0^k \cap E_1^k$ For each demand $k \in C$, is said a cover for the edge e if $\sum_{k \in C} w_k > \bar{s} - \sum_{k' \in K_e} w_{k'}$.*

Definition 9. *For an edge e in E, a cover C is said a minimal cover if $C \setminus \{k\}$ is not a cover for all $k \in C$, i.e., $\sum_{k' \in C \setminus \{k\}} w_{k'} \leq \bar{s} - \sum_{k" \in K_e} w_{k"}$.*

In what follows, we use these definitions to introduce the so-called cover inequalities related to the capacity constraints.

Proposition 15. *Consider an edge e in E. Let C be a minimal cover in K for the edge e. Then, the inequality*

$$\sum_{k \in C} \sum_{p \in P^k(e)} \sum_{s=1}^{\bar{s}} y_{p,s}^k \leq |C| - 1, \tag{25}$$

is valid for $P(G, K, \mathbb{S}, P_K)$.

Proof. If C is minimal cover for edge $e \in E$ this means that there is at most $|C| - 1$ demands from the set of demands in C that can use the edge e. We strengthen our proof by assuming that the inequality (25) is not valid for $P(G, K, \mathbb{S}, P_K)$. It follows that there exists a C-RSA solution S in which $e \notin E_{k'}$ for a demand $k' \in C$ s.t.

$$\sum_{k \in C} \sum_{p \in P^k(e)} \sum_{s=1}^{\bar{s}} y_{p,s}^k(S) > |C| - 1.$$

Since $e \notin E_{k'}$ for a demand $k' \in C$ this means that $\sum_{p' \in P^{k'}(e)} \sum_{s'=1}^{\bar{s}} y_{p',s'}^{k'}(S) = 0$, and taking into account that C is minimal cover for the edge e, $x_e^k(S) \leq 1$ for each $k \in C \setminus \{k'\}$ and $x_e^{k'}(S) \leq 1$, it follows that

$$\sum_{k \in C \setminus \{k'\}} \sum_{p \in P^k(e)} \sum_{s=1}^{\bar{s}} y_{p,s}^k(S) \leq |C| - 1$$

which contradicts what we supposed before, i.e., $\sum_{k \in C} \sum_{p \in P^k(e)} \sum_{s=1}^{\bar{s}} y_{p,s}^k(S) > |C| - 1$.
Hence $\sum_{k \in C} |E_k \cap \{e\}| \leq |C| - 1$.
We conclude at the end that the inequality (25) is valid for $P(G, K, \mathbb{S}, P_K)$.

Note that the inequality (25) can be easily strengthened by using its extended format which we call extended minimal cover for an edge e as follows.

Proposition 16. *Consider an edge e in E. Let C be a minimal cover in K for the edge e, and $\varXi(C)$ be a subset of demands in $K \backslash C \cup K_e$ where $\varXi = \{k \in K \backslash C \cup K_e : e \notin E_0^k$ and $w_k \geq w_{k'} \quad \forall k' \in C\}$. Then, the inequality*

$$\sum_{k \in C} \sum_{p \in P^k(e)} \sum_{s=1}^{\bar{s}} y_{p,s}^k + \sum_{k' \in \varXi(C)} \sum_{p' \in P^{k'}(e)} \sum_{s'=1}^{\bar{s}} y_{p',s'}^{k'} \leq |C| - 1, \tag{26}$$

is valid for $P(G, K, \mathbb{S}, P_K)$.

Proof. If C is minimal cover for edge $e \in E$ this means that there is at most $|C| - 1$ demands from the set of demands in $C \cup \varXi(C)$ that can use the edge e. We strengthen our proof by assuming that the inequality (26) is not valid for $P(G, K, \mathbb{S}, P_K)$. It follows that there exists a C-RSA solution S in which $e \notin E_{k'}$ for each demand $k' \in \varXi(C)$ s.t.

$$\sum_{k \in C} \sum_{p \in P^k(e)} \sum_{s=1}^{\bar{s}} y_{p,s}^k(S) > |C| - 1.$$

Since $e \notin E_{k'}$ for for each demand $k' \in \varXi(C)$this means that $\sum_{p' \in P^{k'}(e)} \sum_{s'=1}^{\bar{s}} y_{p',s'}^{k'}(S) = 0$, and taking into account that C is minimal cover for the edge e, $x_e^k(S) \leq 1$ for each $k \in C$ and $x_e^{k'}(S) \leq 1$, it follows that

$$\sum_{k \in C} \sum_{p \in P^k(e)} \sum_{s=1}^{\bar{s}} y_{p,s}^k(S) \leq |C| - 1$$

which contradicts what we supposed before, i.e., $\sum_{k \in C} \sum_{p \in P^k(e)} \sum_{s=1}^{\bar{s}} y_{p,s}^k(S) > |C| - 1$ and also the inequality (25).
Hence $\sum_{k \in C} |E_k \cap \{e\}| + \sum_{k' \in \varXi(C)} |E_{k'} \cap \{e\}| \leq |C| - 1$.
We conclude at the end that the inequality (25) is valid for $P(G, K, \mathbb{S}, P_K)$.

Furthermore, the inequality (25) can have a more generalized strengthening format using lifting procedures proposed by Nemhauser and Wolsey in [49].

7 Symmetry-Breaking Inequalities

We have noticed that several symmetrical solutions may appear given that there exist several feasible solutions that have the same value of the solution (called equivalents solutions), and they can be found by doing some permutations between the slots assigned to some demands without changing the selected paths (routing) while satisfying the C-RSA constraints. There exists several methods to break the symmetry. See, for example, perturbation method proposed by Margot in

[44], isomorphism pruning method by Margot et al. in [45] and [46], orbital branching method by Ostrowski et al. in [52] and [53], orbital fixing method by Kaibel et al. in [34], and symmetry-breaking constraints by Kaibel and Pfetsch in [31] which is applied in our study. Our aim is to introduce breaking-symmetry inequalities to remove the sub-problems in the enumeration tree that are equivalent due to the equivalency of their associated solutions. To do so, we derive the following inequalities.

Proposition 17. *Consider a demand k in K, a slot $s \in \{1, ..., \bar{s}-1\}$. Let s' be a slot in $\{s, ..., \bar{s}\}$*

$$\sum_{s"=s'}^{\min(s'+w_k-1,\bar{s})} \sum_{p \in P^k} y^k_{p,s"} - \sum_{k' \in K} \sum_{s"=s}^{\min(s+w_{k'}-1,\bar{s})} \sum_{p' \in P^{k'}} y^{k'}_{p',s"} \leq 0. \tag{27}$$

This ensures that the slot s' can be assigned to the demand k over a path $p \in P^k$ if and only if the slot s is already assigned to at least one demand k' in K over its final path $p' \in P^{k'}$.

8 Branch-and-Price and Branch-and-Cut-and-Price Algorithms

Based on the path formulation and several classes of valid inequalities previously introduced, we derive two exact algorithms: Branch-and-Price and Branch-and-Cut-and-Price to solve the C-RSA problem. In this section, we describe the framework of these algorithms. First, we give an overview of our column generation algorithm. Then, we discuss the pricing problem. We further present the different separation procedures associated with the different classes of valid inequalities useful to boost the performance of our algorithms. We give at the end some computational results and a comparative study between Branch-and-Price and Branch-and-Cut-and-Price algorithms.

8.1 Column Generation Algorithm

As it has been mentioned before, our path formulation contains a huge number of variables which can be exponentiel in the worst case due to the number of all feasible paths for each traffic demand. To manage that, we use a column generation algorithm to solve its linear relaxation. To do so, we begin our algorithm with a restricted linear program of our path formulation by considering a feasible subset of variables (columns). For that, we first generate a subset of feasible paths for each demand $k \in K$ denoted by $B^k \subset P^k$ such that the variables $y^k_{p,s}$ for each $k \in K$, $p \in B^k$ and $s \in \mathbb{S}$ induce a feasible basis for the restricted linear program. This means that there exists at least one feasible solution for the restricted linear program. Based on this, we derive the so-called restricted master problem (RMP) as follows

$$\min \sum_{k \in K} \sum_{p \in B^k} \sum_{e \in E(p)} \sum_{s=w_k}^{\bar{s}} l_e y^k_{p,s},$$

subject to

$$\sum_{p \in B^k} \sum_{s=1}^{w_k-1} y^k_{p,s} = 0, \forall k \in K,$$

$$\sum_{p \in B^k} \sum_{s=w_k}^{\bar{s}} y^k_{p,s} = 1, \forall k \in K,$$

$$\sum_{k \in K} \sum_{p \in B^k(e)} \sum_{s'=s}^{s+w_k-1} y^k_{p,s'} \leq 1, \forall e \in E, \forall s \in S,$$

$$y^k_{p,s} \geq 0, \forall k \in K, \forall p \in B^k, \forall s \in S.$$

At each iteration, our column generation algorithm checks if there exists a variable $y^k_{p,s}$ with $p \notin B^k$ for a demand k and slot s having a negative reduced cost using the solution of the dual problem, and add it to B^k. This procedure is based on the so-called "pricing problem".

8.2 Pricing Problem

As noted later, we consider an initial restricted master problem denoted by RMP_0 which is based on an initial subset of variables induced by a subset of feasible path $B^k \subset P^k$ for each demand $k \in K$. The pricing problem consists in finding a feasible path p for a demand k and slot s having a negative reduced cost using the optimal solution of the dual problem. To do so, we consider the following dual variables

- α associated with the equations (2) such that $\alpha_k \in \mathbb{R}$ for all $k \in K$,
- β associated with the equations (3) such that $\beta^k \in \mathbb{R}$ for all $k \in K$,
- μ associated with the inequalities (4) such that $\mu_s^e \leq 0$ for all $e \in E$ and $s \in \mathbb{S}$.

Th dual problem is then equivalent to

$$\max - \sum_{k \in K} \beta^k + \sum_{e \in E} \sum_{s \in \mathbb{S}} \mu_s^e, \tag{28}$$

subject to

$$\beta^k + \sum_{e \in E(p)} (l_e - \sum_{s'=s-w_k+1}^{s} \mu_{s'}^e) \geq 0, \quad \forall k \in K, \forall p \in P^k, \forall s \in \{w_k, ..., \bar{s}\}, \tag{29}$$

$$\mu_s^e \leq 0, \forall e \in E, \forall s \in \mathbb{S}. \tag{30}$$

As a result, we obtain that for all $k \in K$ and $s \in \{w_k, ..., \bar{s}\}$

$$rc_s^k = \beta^k + \min_{p \in P^k \setminus B^k} [\sum_{e \in E(p)} l_e - \sum_{s'=s-w_k+1}^{s} \mu_{s'}^e], \tag{31}$$

which defines the reduced-cost associated with each demand k and slot s. This is equivalent to the separation problem associated with the dual constraint (29). It consists in identifying a path p for a demand k and slot s s.t.

$$\beta^k + \sum_{e \in E(p)} l_e - \sum_{s'=s-w_k+1}^{s} \mu_{s'}^e < 0.$$

Based on this, and taking into account the transmission-reach constraint, the pricing problem consists in solving a Resource Constrained Shortest Path (RCSP) Problem, also called Weight Constrained Shortest Path (WCSP) Problem. This problem is well known to be an NP-hard problem [16]. Several algorithms have been proposed in the literature to solve this problem based on dynamic programming algorithms, heuristics, and some techniques related to the Lagrangian decomposition. As background references we mention [4, 17, 19, 30, 43]. In our work, we have developed an efficient algorithm based on the dynamic programming algorithm proposed which allows us to add a path p with a negative reduced cost for each pair of demand k and slot s if it exists while respecting that the length of this path p must be less than $\bar{l}_k$. We repeat this procedure in each iteration of our column generation until no new column is found (i.e., $rc_s^k \geq 0$ for all $k \in K$ and $s \in \{w_k, ..., \bar{s}_=$. As a result, the final solution is optimal for the linear relaxation of our path formulation. Furthermore, if it is integral, then it is optimal for the C-RSA problem. Otherwise, we create two subproblems called childs by branching on fractional variables $\bar{y}$ (variable branching rule) or on some constraints using the Ryan & Foster [61] branching rule (constraint branching rule).

8.3 Impact of Adding Valid Inequalities on the Structure of the Pricing Problem

Note that adding some valid inequalities can have an impact on the structure of our pricing problem but our pricing problem stills equivalent to the RCSP problem s.t. adding some valid inequalities in a certain level of our algorithms can change the calculation of the reduced-cost associated with certain demands in K and slots in $\mathbb{S}$ as follows.

Impact of Edge-Interval-Cover Inequalities let ρ the dual variable associated with the inequalities (7) such that $\rho^e_{I,\tilde{K}} \leq 0$ for all $e \in E$ and all $I = [s_i, s_j]$ in $\mathbb{S}$ and all $\tilde{K}$ in K. The associated dual program is then equivalent to

$$\max - \sum_{k\in K} \beta^k + \sum_{e\in E}\sum_{s\in\mathbb{S}} \mu^e_s + \sum_{e\in E}\sum_{I\in\tilde{I}}\sum_{\tilde{K}\in K(I)} (\tilde{K}-1)\rho^e_{I,\tilde{K}}, \tag{32}$$

subject to

$$\beta^k + \sum_{e\in E(p)} (l_e - \sum_{s'=s-w_k+1}^{s} \mu^e_{s'} - \sum_{I\in\tilde{I}} \sum_{\substack{\tilde{K}\in K(I) \\ \text{s.t. } k\in\tilde{K}, s\in I_k}} \rho^e_{I,\tilde{K}}) \geq 0,$$
$$\forall k \in K, \forall p \in P^k, \forall s \in \{w_k, ..., \bar{s}\}, \tag{33}$$

$$\mu^e_s \leq 0, \forall e \in E, \forall s \in \mathbb{S}, \tag{34}$$
$$\rho^e_{I,\tilde{K}} \leq 0, \forall e \in E, \forall I \in \tilde{I}, \forall \tilde{K} \in K(I). \tag{35}$$

From (33), we obtain that the reduced-cost for each $k \in K$ and $s \in \{w_k, ..., \bar{s}\}$, becomes equal to

$$rc^k_s = \beta^k + \min_{p\in\mathbb{P}^k} [\sum_{e\in E(p)} l_e - \sum_{s'=s-w_k+1}^{s} \mu^e_{s'}$$
$$- \sum_{I\in\tilde{I}} \sum_{\tilde{K}\in K(I)} \sum_{\text{s.t. } k\in\tilde{K}, s\in I_k} \rho^e_{I,\tilde{K}}]. \tag{36}$$

Impact of Edge-Interval-Clique Inequalities let ζ the dual variables associated with the inequalities (9) such that $\zeta^e_{I,\tilde{K}} \leq 0$ for all $e \in E$ and all $I = [s_i, s_j]$ in $\mathbb{S}$ and all $\tilde{K}$ in K. We denote by $K(I)$ the set of all the minimal cover $\tilde{K}$ for the interval I over edge e, and by $\tilde{I}$ the set of all intervals I in $\mathbb{S}$. Based on this, we define its assocoiated dual program

$$\max - \sum_{k\in K} \beta^k + \sum_{e\in E}\sum_{s\in\mathbb{S}} \mu^e_s + \sum_{e\in E}\sum_{I\in\tilde{I}}\sum_{\tilde{K}\in K(I)} \zeta^e_{I,\tilde{K}}, \tag{37}$$

subject to

$$\beta^k + \sum_{e\in E(p)} (l_e - \sum_{s'=s-w_k+1}^{s} \mu^e_{s'} - \sum_{I\in\tilde{I}} \sum_{\substack{\tilde{K}\in K(I) \\ \text{s.t. } k\in\tilde{K}, s\in I_k}} \zeta^e_{I,\tilde{K}}) \geq 0,$$
$$\forall k \in K, \forall p \in P^k, \forall s \in \{w_k, ..., \bar{s}\}, \tag{38}$$

$$\mu^e_s \leq 0, \forall e \in E, \forall s \in \mathbb{S}, \tag{39}$$
$$\zeta^e_{I,\tilde{K}} \leq 0, \forall e \in E, \forall I \in \tilde{I}, \forall \tilde{K} \in K(I). \tag{40}$$

From (38), we obtain that for all $k \in K$ and $s \in \{w_k, ..., \bar{s}\}$

$$rc^k_s = \beta^k + \min_{p\in\mathbb{P}^k} [\sum_{e\in E(p)} l_e - \sum_{s'=s-w_k+1}^{s} \mu^e_{s'}$$
$$- \sum_{I\in\tilde{I}} \sum_{\tilde{K}\in K(I)} \sum_{\text{s.t. } k\in\tilde{K}, s\in I_k} \zeta^e_{I,\tilde{K}}]. \tag{41}$$

Impact of Interval-Clique Inequalities let ϱ the dual variable associated with the inequalities (11) such that $\varrho_I^c \leq 0$ for all clique c in the conflict graph $\tilde{G}_I^E$. We denote by $C(\tilde{G}_I^E)$ the set of all clique in the conflict graph $\tilde{G}_I^E$ of the interval I. Let $\tilde{I}$ denote the set of all intervals I in $\mathbb{S}$. Our dual program is then defined as follows

$$\max - \sum_{k\in K} \beta^k + \sum_{e\in E}\sum_{s\in\mathbb{S}} \mu_s^e + \sum_{I\in\tilde{I}} \sum_{c\in C(\tilde{G}_I^E)} \varrho_I^c, \tag{42}$$

subject to

$$\beta^k - \sum_{I\in\tilde{I}} \sum_{\substack{c\in C(\tilde{G}_I^E)\\ \text{s.t. } v_k\in c, s\in I_k}} \varrho_I^c + \sum_{e\in E(p)} (l_e - \sum_{s'=s-w_k+1}^{s} \mu_{s'}^e) \geq 0,$$
$$\forall k\in K, \forall p \in P^k, \forall s \in \{w_k, ..., \bar{s}\}, \tag{43}$$

$$\mu_s^e \leq 0, \forall e\in E, \forall s \in \mathbb{S}, \tag{44}$$
$$\varrho_I^c \leq 0, \forall I \in \tilde{I}, \forall c \in C(\tilde{G}_I^E). \tag{45}$$

From (44), we obtain that for all $k \in K$ and $s \in \{w_k, ..., \bar{s}\}$

$$rc_s^k = \beta^k - \sum_{I\in\tilde{I}} \sum_{\substack{c\in C(\tilde{G}_I^E)\\ \text{s.t. } v_k\in c, s\in I_k}} \varrho_I^c + + \min_{p\in\mathbb{P}^k} [\sum_{e\in E(p)} l_e - \sum_{s'=s-w_k+1}^{s} \mu_{s'}^e]. \tag{46}$$

Impact of Interval-Odd-Hole Inequalities let η the dual variable associated with the inequalities (12) such that $\eta_I^h \leq 0$ for all odd-hole h in the conflict graph $\tilde{G}_I^E$. We denote by $H(\tilde{G}_I^E)$ the set of all odd-hole in the conflict graph $\tilde{G}_I^E$ of the interval I. Let $\tilde{I}$ denote the set of all intervals I in $\mathbb{S}$. Our dual program is then defined as follows

$$\max - \sum_{k\in K} \beta^k + \sum_{e\in E}\sum_{s\in\mathbb{S}} \mu_s^e + \sum_{I\in\tilde{I}} \sum_{h\in H(\tilde{G}_I^E)} \frac{|H|-1}{2} \eta_I^h, \tag{47}$$

subject to

$$\beta^k - \sum_{I\in\tilde{I}} \sum_{\substack{h\in H(\tilde{G}_I^E)\\ \text{s.t. } v_k\in h, s\in I_k}} \eta_I^h + \sum_{e\in E(p)} (l_e - \sum_{s'=s-w_k+1}^{s} \mu_{s'}^e) \geq 0,$$
$$\forall k\in K, \forall p \in P^k, \forall s \in \{w_k, ..., \bar{s}\}, \tag{48}$$

$$\mu_s^e \leq 0, \forall e\in E, \forall s \in \mathbb{S}, \tag{49}$$
$$\eta_I^h \leq 0, \forall I \in \tilde{I}, \forall h \in H(\tilde{G}_I^E). \tag{50}$$

From (49), we obtain that for all $k \in K$ and $s \in \{w_k, ..., \bar{s}\}$

$$rc_s^k = \beta^k - \sum_{I\in\tilde{I}} \sum_{\substack{h\in H(\tilde{G}_I^E)\\ \text{s.t. } v_k\in h, s\in I_k}} \eta_I^h + + \min_{p\in\mathbb{P}^k} [\sum_{e\in E(p)} l_e - \sum_{s'=s-w_k+1}^{s} \mu_{s'}^e]. \tag{51}$$

Impact of Edge-Slot-Assignment-Clique Inequalities let γ the dual variable associated with the inequalities (14) such that $\gamma_c^e \leq 0$ for all $e \in E$ and all clique c in the conflict graph $\tilde{G}_S^e$. We denote by $C(\tilde{G}^e_S)$ the set of all clique in the conflict graph $\tilde{G}^e_S$ of the edge e in E. The dual program is then equivalent to

$$\max - \sum_{k\in K}\beta^k + \sum_{e\in E}\sum_{s\in\mathbb{S}}\mu^e_s + \sum_{e\in E}\sum_{c\in C(\tilde{G}^e_S)}\gamma^e_c, \tag{52}$$

subject to

$$\beta^k + \sum_{e\in E(p)}(l_e - \sum_{s'=s-w_k+1}^{s}\mu^e_{s'} - \sum_{\substack{c\in C(\tilde{G}^e_S)\\ \text{s.t. } v_{k,s}\in c}}\gamma^e_c) \geq 0, \quad \forall k\in K, \forall p\in P^k, \forall s\in\{w_k,...,\bar{s}\}, \tag{53}$$

$$\mu^e_s \leq 0, \forall e\in E, \forall s\in\mathbb{S}, \tag{54}$$

$$\gamma^e_c \leq 0, \forall e\in E, \forall c\in C(\tilde{G}^e_S). \tag{55}$$

From (53), we obtain that the reduced-cost for each $k\in K$ and $s\in\{w_k,...,\bar{s}\}$ can be computed as follows

$$rc^k_s = \beta^k + \min_{p\in\mathbb{P}^k}[\sum_{e\in E(p)} l_e - \sum_{s'=s-w_k+1}^{s}\mu^e_{s'} - \sum_{\substack{c\in C(\tilde{G}^e_S)\\ \text{s.t. } v_{k,s}\in c}}\gamma^e_c]. \tag{56}$$

Impact of Edge-Slot-Assignment-Odd-Hole Inequalities let ξ the dual variable associated with the inequalities (17) such that $\xi^e_h \leq 0$ for all $e\in E$ and all odd-hole h in the conflict graph $\tilde{G}^e_S$. We denote by $H(\tilde{G}^e_S)$ the set of all odd-hole in the conflict graph $\tilde{G}^e_S$ of the edge e in E. The dual program is then equivalent to

$$\max - \sum_{k\in K}\beta^k + \sum_{e\in E}\sum_{s\in\mathbb{S}}\mu^e_s + \sum_{e\in E}\sum_{h\in H(\tilde{G}^e_S)}\xi^e_h, \tag{57}$$

subject to

$$\beta^k + \sum_{e\in E(p)}(l_e - \sum_{s'=s-w_k+1}^{s}\mu^e_{s'} - \sum_{\substack{h\in H(\tilde{G}^e_S)\\ \text{s.t. } v_{k,s}\in h}}\xi^e_h) \geq 0, \quad \forall k\in K, \forall p\in P^k, \forall s\in\{w_k,...,\bar{s}\}, \tag{58}$$

$$\mu^e_s \leq 0, \forall e\in E, \forall s\in\mathbb{S}, \tag{59}$$

$$\xi^e_h \leq 0, \forall e\in E, \forall h\in H(\tilde{G}^e_S). \tag{60}$$

From (58), we obtain that the reduced-cost for each $k\in K$ and $s\in\{w_k,...,\bar{s}\}$ can be computed as follows

$$rc^k_s = \beta^k + \min_{p\in\mathbb{P}^k}[\sum_{e\in E(p)} l_e - \sum_{s'=s-w_k+1}^{s}\mu^e_{s'} - \sum_{\substack{h\in H(\tilde{G}^e_S)\\ \text{s.t. } v_{k,s}\in h}}\xi^e_h]. \tag{61}$$

Impact of Slot-Assignment-Clique Inequalities let λ the dual variable associated with the inequalities (19) such that $\lambda^c \leq 0$ for all clique c in the conflict graph $\tilde{G}^E_S$. We denote by $C(\tilde{G}^E_S)$ the set of all clique in the conflict graph $\tilde{G}^E_S$. Our dual program is then defined as follows

$$\max - \sum_{k\in K}\beta^k + \sum_{e\in E}\sum_{s\in\mathbb{S}}\mu^e_s + \sum_{c\in C(\tilde{G}^E_S)}\lambda^c, \tag{62}$$

subject to

$$\beta^k - \sum_{\substack{c \in C(\tilde{G}_S^E) \\ \text{s.t. } v_{k,s} \in c}} \lambda^c + \sum_{e \in E(p)} (l_e - \sum_{s'=s-w_k+1}^{s} \mu_{s'}^e) \geq 0,$$

$$\forall k \in K, \forall p \in P^k, \forall s \in \{w_k, ..., \bar{s}\}, \quad (63)$$

$$\mu_s^e \leq 0, \forall e \in E, \forall s \in \mathbb{S}, \quad (64)$$

$$\lambda^c \leq 0, \forall c \in C(\tilde{G}_S^E). \quad (65)$$

From (63), we obtain that for all $k \in K$ and $s \in \{w_k, ..., \bar{s}\}$

$$rc_s^k = \beta^k - \sum_{\substack{c \in C(\tilde{G}_S^E) \\ \text{s.t. } v_{k,s} \in c}} \lambda^c + \min_{p \in \mathbb{P}^k} [\sum_{e \in E(p)} l_e - \sum_{s'=s-w_k+1}^{s} \mu_{s'}^e]. \quad (66)$$

Impact of Slot-Assignment-Odd-Hole Inequalities let φ the dual variable associated with the inequalities (21) such that $\varphi^h \leq 0$ for all odd-hole h in the conflict graph $\tilde{G}_S^E$. We denote by $H(\tilde{G}_S^E)$ the set of all odd-hole in the conflict graph $\tilde{G}_S^E$. Our dual program is then defined as follows

$$\max - \sum_{k \in K} \beta^k + \sum_{e \in E} \sum_{s \in \mathbb{S}} \mu_s^e + \sum_{h \in H(\tilde{G}_S^E)} \frac{|h| - 1}{2} \varphi^h, \quad (67)$$

subject to

$$\beta^k - \sum_{\substack{h \in H(\tilde{G}_S^E) \\ \text{s.t. } v_{k,s} \in h}} \varphi^h + \sum_{e \in E(p)} (l_e - \sum_{s'=s-w_k+1}^{s} \mu_{s'}^e) \geq 0,$$

$$\forall k \in K, \forall p \in P^k, \forall s \in \{w_k, ..., \bar{s}\}, \quad (68)$$

$$\mu_s^e \leq 0, \forall e \in E, \forall s \in \mathbb{S}, \quad (69)$$

$$\varphi^h \leq 0, \forall h \in H(\tilde{G}_S^E). \quad (70)$$

From (63), we obtain that for all $k \in K$ and $s \in \{w_k, ..., \bar{s}\}$

$$rc_s^k = \beta^k - \sum_{\substack{h \in H(\tilde{G}_S^E) \\ \text{s.t. } v_{k,s} \in h}} \varphi^h + \min_{p \in \mathbb{P}^k} [\sum_{e \in E(p)} l_e - \sum_{s'=s-w_k+1}^{s} \mu_{s'}^e]. \quad (71)$$

Impact of Edge-Capacity-Cover Inequalities let ϕ the dual variable associated with the inequalities (25) such that $\phi_{\tilde{K}}^e \leq 0$ for all $e \in E$ and all minimal cover $\tilde{K}$ for the edge e. We denote by $C(e)$ the set of all minimal cover $\tilde{K}$ for the edge e. The dual program can be defined as follows

$$\max - \sum_{k \in K} \beta^k + \sum_{e \in E} \sum_{s \in \mathbb{S}} \mu_s^e + \sum_{e \in E} \sum_{\tilde{K} \in C(e)} (|\tilde{K}| - 1) \phi_{\tilde{K}}^e, \quad (72)$$

subject to

$$\beta^k + \sum_{e \in E(p)} (l_e - \sum_{s'=s-w_k+1}^{s} \mu_{s'}^e - \sum_{\substack{\tilde{K} \in C(e) \\ \text{s.t. } k \in \tilde{K}}} \phi_{\tilde{K}}^e) \geq 0,$$

$$\forall k \in K, \forall p \in P^k, \forall s \in \{w_k, ..., \bar{s}\}, \quad (73)$$

$$\mu_s^e \leq 0, \forall e \in E, \forall s \in \mathbb{S}, \quad (74)$$

$$\phi_{\tilde{K}}^e \leq 0, \forall e \in E, \forall \tilde{K} \in C(e). \quad (75)$$

From (74), we obtain that for all $k \in K$ and $s \in \{w_k, ..., \bar{s}\}$

$$rc_s^k = \beta^k + \min_{p \in \mathbb{P}^k} [\sum_{e \in E(p)} l_e - \sum_{s'=s-w_k+1}^{s} \mu_{s'}^e - \sum_{\substack{\tilde{K} \in C(e) \\ \text{s.t. } k \in \tilde{K}}} \phi_{\tilde{K}}^e]. \quad (76)$$

Based on these results, we ensure that our pricing problem stills equivalent to the RCSP problem for any class of valid inequalities proposed in this paper.

8.4 Dynamic Programming Algorithm for the Pricer

We propose a pseudo-polynomial time algorithm to solve the pricing problem using dynamic programming adapated to our C-RSA problem that takes into account the transmission-reach constraint to identify a feasible path for a given pair of demand p and slot s. It is based on the dynamic programming algorithm proposed by Dumitrescu et al. in [17] to solve the RCSP problem. For each demand $k \in K$ and slot s, we associate to each node $v \in V$ in the graph G a set of labels L^v s.t. each label corresponds to differents paths from th origin node o_k to the node v, and each label p is specified by a cost equals to $\sum_{e \in E(p)} = l_e - \sum_{s'=s-w_k+1}^{s} \mu_{s'}^e$, and a weight equals to $\sum_{e \in E(p)} l_e$. We denote by T_v the set of labels on node $v \in V$. For each demand k and slot $s \in \{w_k, ..., \bar{s}\}$, the complexity of our algorithm is bounded by $\mathcal{O}(|E \setminus E_0^k|\bar{l}_k)$ [17]. Algorithm 8.4 summarizes the different steps of our dynamic programming algorithm.

8.5 Basic Columns

The basic sub-set of paths used to define the restricted master problem are generated using a brute-force search algorithm which creates a search tree that covers all the feasible paths P^k for each demand k. It is then used to pre-compute an initial subset B^k of feasible paths for each demand $k \in K$ taking into account the transmission-reach constraint which allows us to prune some non-intersecting nodes in our search tree of this algorithm.

8.6 Overwiew of Branch-and-Price and Branch-and-Cut-Price Algorithms

Based on these features, we derive a Branch-and-Price algorithm by combining a column generation algorithm with a Branch-and-Bound algorithm. The main purpose of this algorithm is to solve a sequence of linear programs using the column generation algorithm at each node of a Branch-and-Bound algorithm. At each iteration of a certain level of our algorithm, we solve our pricing problem by identifying one or more than one new column by solving an RCSP problem for each demand k and slot $s \in \{w_k, ..., \bar{s}\}$ using our dynamic programming algorithm. Furthermore, we derive Branch-and-Cut-and-Price based on our Branch-and-Price algorithm combined with a cutting-plane-based algorithm by adding several valid inequalities useful to obtain tighter bounds. Consider a fractional solution $\bar{y}$. At each iteration of our Branch-and-Price algorithm, our aim is to identify for a given class of valid inequalities the existence of one or more than one inequalities of this class that are violated by the current solution. We repeat this procedure in each iteration of our algorithm until non violated inequality is identified. Algorithm 8.6 summarizes the different steps of our Branch-and-Cut-and-Price algorithm for a given class of valid inequalities.
In what follows, we study the separation problem of each valid inequality.

8.7 Separation Procedures: Complexity and Algorithms

Separation of Edge-Interval-Cover Inequalities Let's discuss the separation problem of the inequality (7). Given a fractional solution $\bar{y}$, and an edge $e \in E$. We first construct a set of intervals of contiguous slots $I \in I_e$ s.t. each interval of contiguous slots $I = [s_i, s_j] \in I_e$ is identified using

Algorithm 1 Dynamic Programming Algorithm

Data: An undirected, loopless, and connected graph $G = (V, E)$, a spectrum $\mathbb{S}$, a multi-set K of demands, a linear program LP, a demand k and a slot $s \in \{w_k, ..., \bar{s}\}$, a set B^k of feasible paths already exists in the current LP for the demand $k \in K$ and slot s, and the optimal values of the duals variables $(\alpha^*, \beta^*, \mu^*)$

Result: Optimal path $p*$ for the demand k and slot s

Set $L^{o_k} = \{(0,0)\}$ and $L^v = \emptyset$ for each node $v \in V \setminus (V_0^k \cup \{o_k\})$;
Set $T^v = \emptyset$ for each node $v \in V \setminus V_0^k$;
Set STOP=FALSE;
Set $p^* = NULL$;
while *STOP==FALSE* **do**
    if $\cup_{v \in V}(L_v \setminus T_v) = \emptyset$ **then**
        Set STOP= TRUE;
        Set $p^* = \emptyset$;
        We select one label p from the labels L^{d_k} of destination node d_k s.t. $p \notin B^k$ with $\beta^k + \sum_{e \in E(p)} l_e - \sum_{s'=s-w_k+1}^{s} \mu_{s'}^{e} < 0$;
        if *such label exists* **then**
            Set $p^* = p$;
        end
    end
    if $\cup_{v \in V}(L_v \setminus T_v) \neq \emptyset$ **then**
        Select a node $i \in V \setminus V_0^k$ and a label $p \in L^i \setminus T^i$ having the smallest value of $\sum_{e \in E(p)} l_e$;
        for *each* $e = ij \in \delta(i) \setminus E_0^k$ *s.t.* $\sum_{e' \in E(p)} l_{e'} - \sum_{s'=s-w_k+1}^{s} \mu_{s'}^{e'} + l_e \leq \bar{l}_k$ **do**
            if $j \notin V(p)$ **then**
                Set $p' = p \cup \{e\}$;
                Update the set of label $L^j = L^i \cup \{p'\}$;
            end
        end
        Set $T^i = T^i \cup \{p\}$;
    end
end
return the best optimal path p^* for the demand k and slot s;

Algorithm 2 Branch-and-Cut-and-Price Algorithm

```
Data: An undirected, loopless, and connected graph G = (V, E), a spectrum S, a multi-set K of demands,
      a set B^k of precomputed feasible paths for each demand k ∈ K, and a given class of valid inequality
Result: Optimal solution for the C-RSA problem
LP ⟵ RMP_0;
 // Cut-and-Price Stage
 Stop= FALSE;
 while STOP==FALSE do
   // Column Generation Stage
   Solve the linear program LP;
   Let y* be the optimal solution of LP;
   Consider the optimal values of the duals variables (α*, β*, μ*);
   ADD = FALSE;
   for each demand k ∈ K do
     for each slot s ∈ {w_k, ..., s̄} do
       Compute its associated reduced cost rc_s^k;
       if rc_s^k < 0 then
         Consider the optimal path p* for the demand k and slot s with rc_s^k(p) < 0;
         Add the new variable (column) y_{p*,s}^k to the current LP;
         ADD= TRUE ;
       end
     end
   end
   if ADD==FALSE then
     // Cutting-Plane Stage
     if there exist inequalities from the given class that are violated by the current solution y* then
       Add them to LP ;
     end
     else
       STOP = TRUE;
     end
   end
end
Consider the optimal solution y* of LP;
 if y* is integer for the C-RSA then
   y* is an optimal solution for the C-RSA;
   End of our Branch-and-Cut-and-Price algorithm;
end
else
   Create two sub-problems by branching one some variables or constraints;
end
// Branching Stage
 for each sub-problem not yet solved do
   go to the Cut-and-Price stage;
end
return the best optimal solution y* for the C-RSA;
```

two slots s_i and s_j randomly generated in $\mathbb{S}$ with $s_j \geq s_i + 2\max_{k\in K\setminus \bar{K}_e} w_k$. Consider now an interval of contiguous slots $I = [s_i, s_j] \in I_e$ over an edge e. The separation problem associated with the inequality (7) is Np-Hard [33] given that it consists in identifying a cover $\tilde{K}^*$ for the interval I over the edge e, s.t.

$$\sum_{k\in \tilde{K}^*} \sum_{p\in P^k(e)} \sum_{s'=s_i+w_k-1}^{s_j} \bar{y}^k_{p,s'} > |\tilde{K}^*| - 1.$$

For that, we use a greedy algorithm introduced by Nemhauser and Sigismondi in [50] as follows. We first select a demand $k \in K$ having largest number of requested slot w_k with $\sum_{p\in P^k(e)} \sum_{s'=s_i+w_k-1}^{s_j} \bar{y}^k_{p,s'} > 0$, and assign it to $\tilde{K}^*$, i.e., $\tilde{K}^* = \{k\}$. After that, we iteratively add each demand $k' \in K \setminus \tilde{K}^*$ to $\tilde{K}^*$ with with $\sum_{p\in P^{k'}(e)} \sum_{s'=s_i+w_{k'}-1}^{s_j} \bar{y}^{k'}_{p,s'} > 0$ and while $\sum_{k\in\tilde{K}^*} w_k \leq |I|$, i.e., until a cover $\tilde{K}^*$ is obtained for the interval I over the edge e with $\sum_{k\in\tilde{K}^*} w_k > |I|$. We further derive a minimal cover from the cover $\tilde{K}^*$ by deleting each demand $k \in \tilde{K}^*$ if $\sum_{k'\in\tilde{K}^*\setminus\{k\}} w_{k'} \leq |I|$. We then add the inequality (7) induced by the minimal cover $\tilde{K}^*$ for the interval I and edge e to the current LP if it is violated, i.e.,

$$\sum_{k\in \tilde{K}^*} \sum_{p\in P^k(e)} \sum_{s'=s_i+w_k-1}^{s_j} y^k_{p,s'} \leq |\tilde{K}^*| - 1.$$

Furthermore, the inequality (7) induced by the minimal cover $\tilde{K}^*$ can be lifted by introducing an extended cover inequality (8) as follows

$$\sum_{k\in \tilde{K}^*} \sum_{p\in P^k(e)} \sum_{s'=s_i+w_k-1}^{s_j} y^k_{p,s'} + \sum_{k'\in \tilde{K}^*_e} \sum_{p\in P^{k'}(e)} \sum_{s'=s_i+w_{k'}-1}^{s_j} y^{k'}_{p,s'} \leq |\tilde{K}^*| - 1,$$

where $w_{k'} \geq w_k$ for each $k \in \tilde{K}^*$ and each $k' \in \tilde{K}^*_e$.

Separation of Edge-Interval-Clique Inequalities The separation problem related to the inequality (9) is NP-hard [54][32] given that it consists in identifying a maximal clique C^* in the conflict graph $\tilde{G}^e_I$ for a given edge e and a given interval I s.t.

$$\sum_{k\in C^*} \sum_{p\in P^k(e)} \sum_{s'=s_i+w_k-1}^{s_j} \bar{y}^k_{p,s'} > 1,$$

for a given fractional solution $\bar{y}$ of the current LP.
We start our procedure of separation by constructing a set of intervals of contiguous slots $I \in I_e$ for a given edge $e \in E$ s.t. each interval of contiguous slots $I = [s_i, s_j] \in I_e$ is identified for each slot $s_i \in \mathbb{S}$ and slot s_j with $s_j \in \{s_i + \max_{k\in K\setminus\bar{K}_e} w_k, ..., \min(\bar{s}, s_i + 2\max_{k\in K\setminus\bar{K}_e} w_k)\}$. Consider now an interval of contiguous slots $I = [s_i, s_j] \in I_e$ over an edge e, and its associated conflict graph $\tilde{G}^e_I$. We then use a greedy algorithm introduced by Nemhauser and Sigismondi in [50] to identify a maximal clique in conflict graph $\tilde{G}^e_I$ as follows. We first associate a positive weight for each node v_k in $\tilde{G}^e_I$ equals to $\sum_{p\in P^k(e)} \sum_{s'=s_i+w_k-1}^{s_j} \bar{y}^k_{p,s'}$. We then set $C^* = \{k\}$ s.t. k is a demand in K having the largest number of slots w_k and weight $\sum_{p\in P^k(e)} \sum_{s'=s_i+w_k-1}^{s_j} \bar{y}^k_{p,s'}$. After that, we iteratively add each demand k' having $\sum_{p\in P^{k'}(e)} \sum_{s'=s_i+w_{k'}-1}^{s_j} \bar{y}^{k'}_{p,s'}$ s.t. its corresponding node $v_{k'}$ is linked with all the nodes v_k with k already assigned to the current C^*. After that, we check if the inequality (9) induced by the maximal clique C^* for the interval I and edge e is violated or not. If so, we add the inequality (9) induced by the maximal clique C^* to the current LP, i.e.,

$$\sum_{k\in C^*} \sum_{p\in P^k(e)} \sum_{s'=s_i+w_k-1}^{s_j} y^k_{p,s'} \leq 1.$$

One can strengthen such inequality by adding the inequality (9) induced by the maximal clique C^* and $C^*_e \subset K_e \setminus C^*$, i.e.,

$$\sum_{k\in C^*}\sum_{p\in P^k(e)}\sum_{s'=s_i+w_k-1}^{s_j} y^k_{p,s'} + \sum_{k'\in C^*_e}\sum_{p\in P^{k'}(e)}\sum_{s'=s_i+w_{k'}-1}^{s_j} y^{k'}_{p,s'} \leq 1,$$

s.t.

- $w_{k'} + w_k \geq |I| + 1$ for each $k \in C^*$ and $k' \in C^*_e$,
- $w_{k'} + w_{k"} \geq |I| + 1$ for each $k' \in C^*_e$ and $k" \in C^*_e$,
- $w_{k'} \leq |I|$ and $2w_{k'} \geq |I| + 1$ for each $k' \in C^*_e$.

Separation of Interval-Clique Inequalities Given a fractional solution $\bar{y}$, and an interval of contiguous slots $I = [si, s_j]$. Our separation algorithm for the inequality (11) consists in identifying a maximal clique C^* in the conflict graph $\tilde{G}^E_I$ s.t.

$$\sum_{k\in C^*}\sum_{p\in P^k}\sum_{s'=s_i+w_k-1}^{s_j} \bar{y}^k_{p,s'} > 1.$$

As result, its associated sepration problem is NP-hard given that computing a maximal clique in a given graph is known to be a NP-hard problem [32]. For that, we also use the greedy algorithm introduced by Nemhauser and Sigismondi in [50] to identify a maximal clique in conflict graph $\tilde{G}^E_I$ as follows. We first generate a set of intervals of contiguous slots denoted by I_E s.t. each interval of contiguous slots $I = [s_i, s_j] \in I_E$ is defined for each slot $s_i \in \mathbb{S}$ and slot s_j with $s_j \in \{s_i + \max_{k\in K, |E^k_1|\geq 1} w_k, ..., \min(\bar{s}, s_i + 2 \max_{k\in K, |E^k_1|\geq 1} w_k)\}$. We then consider an interval of contiguous slots $I = [s_i, s_j] \in I_E$ and its associated conflict graph $\tilde{G}^E_I$. We associate a positive weight $\sum_{p\in P^k}\sum_{s'=s_i+w_k-1}^{s_j} \bar{y}^k_{p,s'}$ for each node v_k in $\tilde{G}^E_I$. We select a demand k s.t. k is a demand in K having the largest number of slots w_k and weight $\sum_{p\in P^k}\sum_{s'=s_i+w_k-1}^{s_j} \bar{y}^k_{p,s'}$, and then set $C^* = \{k\}$. After that, we iteratively add each demand k' having $\sum_{p\in P^{k'}}\sum_{s'=s_i+w_{k'}-1}^{s_j} \bar{y}^{k'}_{p,s'} > 0$ s.t. its corresponding node $v_{k'}$ is linked with all the nodes v_k with $k \in C^*$. At the end, we add the inequality (11) induced by the maximal clique C^* if it is violated, i.e., we add the following inequality to the current LP

$$\sum_{k\in C^*}\sum_{p\in P^k}\sum_{s'=s_i+w_k-1}^{s_j} y^k_{p,s'} \leq 1.$$

Moreover, this additional inequality can be strengthened as follows

$$\sum_{k\in C^*}\sum_{p\in P^k}\sum_{s'=s_i+w_k-1}^{s_j} y^k_{p,s'} + \sum_{k'\in C^*_e}\sum_{p\in P^{k'}}\sum_{s'=s_i+w_{k'}-1}^{s_j} y^{k'}_{p,s'} \leq 1,$$

where $C^*_E \subset K \setminus C^*$ s.t.

- $w_{k'} + w_k \geq |I| + 1$ and $E^k_1 \cap E^{k'}_1 \neq \emptyset$ for each $k \in C^*$ and $k' \in C^*_E$,
- $w_{k'} + w_{k"} \geq |I| + 1$ and $E^{k'}_1 \cap E^{k"}_1 \neq \emptyset$ for each $k' \in C^*_E$ and $k" \in C^*_E$,
- $w_{k'} \leq |I|$ and $2w_{k'} \geq |I| + 1$ for each $k' \in C^*_E$.

Separation of Interval-Odd-Hole Inequalities For the inequality (12), we propose a separation algorithm that consists in identifying an odd-hole H^* in the conflict graph $\tilde{G}^E_I$ for a given Interval I and a fractional solution $\bar{y}$ s.t.

$$\sum_{k\in H^*}\sum_{p\in P^k}\sum_{s'=s_i+w_k-1}^{s_j} \bar{y}^k_{p,s'} > \frac{|H^*| - 1}{2}.$$

This can be done in polynomial time as shown by Rebennack et al. in [57] and [58]. Based on this, we use the exact algorithm proposed by the same authors which consists of finding a minimum weighted odd-cycle in a graph. For that, we should first generate a set of intervals of contiguous slots I_E as we did before in the section 8.7. We then consider a conflict graph $\tilde{G}_I^E$ associated with a given interval of contiguous slots $I \in I_E$. We construct an auxiliary conflict graph $\tilde{G}_I^E$ which can be seen as a bipartite graph by duplicating each node v_k in $\tilde{G}_I^E$ (i.e., v_k and v'_k) and each two nodes are linked in $\tilde{G}_I^E$ if their original nodes are linked in $\tilde{G}_I^E$. We assign to each link (v_a, v_b) in $\tilde{G}_I^E$ a weight equals to $\frac{1-\sum_{p\in P^a}\sum_{s'=s_i+w_a-1}^{s_j}\bar{y}^a_{p,s'}-\sum_{p'\in P^b}\sum_{s'=s_i+w_b-1}^{s_j}\bar{y}^b_{p',s'}}{2}$. We then compute for each node v_k in $\tilde{G}_I^E$, the shortest path between v_k and its copy in the auxiliary conflict graph $\tilde{G}_I^E$ denoted by p_{v_k,v'_k}. After that, we check if the total sum of weight over edges belong this path is smallest than $\frac{1}{2}$,

$$\sum_{(v_a,v_b)\in E(p_{v_k,v'_k})}\frac{1-\sum_{p\in P^a}\sum_{s'=s_i+w_a-1}^{s_j}\bar{y}^a_{p,s'}-\sum_{p'\in P^b}\sum_{s'=s_i+w_b-1}^{s_j}\bar{y}^b_{p',s'}}{2}<\frac{1}{2}.$$

If so, the odd-hole H^* is composed by all the original nodes of nodes belong the computed shortest path p_{v_k,v'_k}, i.e., $V(p_{v_k,v'_k})\setminus\{v'_k\}$. We then add the inequality (12) induced by the odd-hole H^* to the current LP, i.e.,

$$\sum_{k\in H^*}\sum_{p\in P^k}\sum_{s'=s_i+w_k-1}^{s_j} y^k_{p,s'} \leq \frac{|H^*|-1}{2}.$$

It can be lifted using the greedy algorithm introduced by Nemhauser and Sigismondi in [50] to identify a maximal clique C^* in conflict graph $\tilde{G}_I^E$ s.t. s.t.

- $w_{k'} + w_k \geq |I|+1$ and $E_1^k \cap E_1^{k'} \neq \emptyset$ for each $k \in H^*$ and $k' \in C^*$,
- $w_{k'} + w_{k"} \geq |I|+1$ and $E_1^{k'} \cap E_1^{k"} \neq \emptyset$ for each $k' \in C^*$ and $k" \in C^*$,
- $w_{k'} \leq |I|$ and $2w_{k'} \geq |I|+1$ for each $k' \in C^*$.

For that, we assign a positive weight equals to the number of slots request $w_{k'}$ by the demand k' for each node $v_{k'}$ linked with all the nodes $v_k \in H^*$ in the conflict graph $\tilde{G}_I^E$. We then select the node $v_{k'}$ linked with all the nodes $v_k \in H^*$ in the conflict graph $\tilde{G}_I^E$ having the largest weight, and set C^* to $\{k'\}$. After that, we iteratively add each demand $k"$ to the current clique C^* if its associated node $v_{k"}$ is linked with all the nodes $v_k \in H^*$ and nodes $v_{k'} \in C^*$. As a result, we add the inequality (13) induced by the odd-hole H^* and clique C^* to the current LP, i.e.,

$$\sum_{k\in H^*}\sum_{s'=s_i+w_k-1}^{s_j}\sum_{p\in P^k}\sum_{s'=s_i+w_k-1}^{s_j} y^k_{p,s'} + \frac{|H^*|-1}{2}\sum_{k'\in C^*}\sum_{p'\in P^{k'}}\sum_{s"=s_i+w_{k'}-1}^{s_j} y^{k'}_{p',s"} \leq \frac{|H^*|-1}{2}.$$

Separation of Edge-Slot-Assignment-Clique Inequalities Consider an edge $e \in E$, and a fractional solution $(\bar{y})$. The separation algorithm for the inequality (14) consists in identifying a maximal clique C^* in the conflict graph $\tilde{G}_S^e$ s.t.

$$\sum_{v_{k,s}\in C^*}\sum_{p\in P^k(e)} \bar{y}^k_{p,s} > 1.$$

To do this, we use the greedy algorithm introduced by Nemhauser and Sigismondi in [50] to identify a maximal clique C^* in conflict graph $\tilde{G}_S^e$ given that computing a maximal clique in such a graph is also NP-hard problem [32]. Based on this, we first assign a positive weight $\sum_{p\in P^k(e)}\bar{y}^k_{p,s}$ to each node $v_{k,s}$ in the conflict graph $\tilde{G}_S^e$. We then select a node $v_{k,s}$ in the conflict graph $\tilde{G}_S^e$ having the largest weight compared with the other nodes in $\tilde{G}_S^e$, and set $C^* = \{v_{k,s}\}$. After that, we iteratively add each node $v_{k',s'}$ to the current C^* if it is linked with all the nodes $v_{k,s}$ already assigned to the current clique C^* and $\sum_{p' \in P^{k'}(e)} \bar{y}^{k'}_{p',s'} > 0$. At the end, we add the inequality (14) induced by the clique C^* for edge e to the current LP if it is violated, i.e., we add the following inequality

$$\sum_{v_{k,s} \in C^*} \sum_{p \in P^k(e)} y^k_{p,s} \leq 1.$$

Furthermore, it can be lifted by identifying a maximal clique N^* s.t. each $v_{k',s'} \in N^*$ is linked with all the nodes $v_{k,s} \in C^* \cup (N^* \setminus \{v_{k',s'}\})$ in $\tilde{G}^e_S$. For that, we use also the greedy algorithm introduced by Nemhauser and Sigismondi in [50] to identify the clique N^* as follows. We first set $N^* = \{v_{k',s'}\}$ with $v_{k',s'} \notin C^*$ a node in $\tilde{G}^e_S$ having the largest value of node-degree (i.e., $|\delta(v_{k',s'})|$) in $\tilde{G}^e_S$ and $v_{k',s'}$ is linked with all the nodes $v_{k,s} \in C^*$ in $\tilde{G}^e_S$ and $k' \in K_e$. Afterwards, we iteratively add each node $v_{k",s"} \notin C^* \cup N^*$ to the current N^* if it is linked in $\tilde{G}^e_S$ with all the nodes already assigned to C^* and N^* and $k" \in K_e$. At the end, we add the following inequality induced by the clique $C^* \cup N^*$ to the current LP, i.e.,

$$\sum_{v_{k,s} \in C^*} \sum_{p \in P^k(e)} y^k_{p,s} + \sum_{v_{k',s'} \in N^*} \sum_{p' \in P^{k'}(e)} y^{k'}_{p',s'} \leq 1.$$

Separation of Slot-Assignment-Odd-Hole Inequalities Consider an edge $e \in E$, the separation algorithm for inequality (17) consists in identifying an odd-hole H^* in the conflict graph $\tilde{G}^e_S$ for a given fractional solution $\bar{y}$ s.t.

$$\sum_{v_{k,s} \in H^*} \sum_{p \in P^k(e)} \bar{y}^k_{p,s} > \frac{|H^*| - 1}{2}.$$

This can be done in polynomial time as shown by Rebennack et al. in [57] and [58] by finding a minimum weighted odd-cycle in the conflict graph $\tilde{G}^e_S$. To do so, we first construct an auxiliary conflict graph $\tilde{G}^e_S$ which can be seen as a bipartite graph by duplicating each node $v_{k,s}$ in $\tilde{G}^e_S$ (i.e., $v_{k,s}$ and $v'_{k,s}$) and each two nodes are linked in $\tilde{G}^e_S$ if their original nodes are linked in $\tilde{G}^e_S$. We assign to each link $(\tilde{v}_{k,s}, \tilde{v}_{k',s'})$ in $\tilde{G}^e_S$ a weight equals to $\frac{1 - \sum_{p \in P^k(e)} \bar{y}^k_{p,s} - \sum_{p' \in P^{k'}(e)} \bar{y}^{k'}_{p',s'}}{2}$. We then compute for each node $v_{k,s}$ in $\tilde{G}^e_S$, the shortest path between $v_{k,s}$ and its copy in the auxiliary conflict graph $\tilde{G}^e_S$ denoted by $p_{v_{k,s}, v'_{k,s}}$. After that, we check if the total sum of weight over edges belonging to this path is smaller than $\frac{1}{2}$. If so, the odd-hole H^* is composed by all the original nodes of nodes belong the computed shortest path $p_{v_{k,s}, v'_{k,s}}$, i.e., $V(p_{v_{k,s}, v'_{k,s}}) \setminus \{v'_{k,s}\}$. As a result, the following inequality (17) induced by the odd-hole H^*

$$\sum_{v_{k,s} \in H^*} \sum_{p \in P^k(e)} \bar{y}^k_{p,s} \leq \frac{|H^*| - 1}{2},$$

should be added to the current LP. Moreover, one can propose a lifting procedure for the inequality (17) induced by the odd-hole H^* by using the greedy algorithm introduced by Nemhauser and Sigismondi in [50] to identify a maximal clique C^* in the conflict graph $\tilde{G}^e_S$ s.t. each node $v_{k',s'} \in C^*$ should have a link with all the nodes $v_{k,s} \in H^*$, and all the nodes $v_{k",s"} \in C^* \setminus \{v_{k',s'}\}$ in the conflict graph $\tilde{G}^e_S$. For that, we first assign a node $v_{k',s'} \notin H^*$ to the clique C^* (i.e., $C^* = \{v_{k',s'}\}$) s.t. $v_{k',s'}$ has the largest value of node-degree (i.e., $|\delta(v_{k',s'})|$) in $\tilde{G}^e_S$ and $v_{k',s'}$ is linked with all the nodes $v_{k,s} \in H^*$ in $\tilde{G}^e_S$. After that, we iteratively add each node $v_{k',s'} \notin H^* \cup C^*$ to the current clique C^* if it is linked in $\tilde{G}^e_S$ with all the nodes already assigned to the odd-hole H^* and the clique C^*. We then add the inequality (18) induced by the odd-hole H^* and clique C^*

$$\sum_{v_{k,s} \in H^*} \sum_{p \in P^k(e)} \bar{y}^k_{p,s} + \frac{|H^*| - 1}{2} \sum_{v_{k',s'} \in C^*} \sum_{p' \in P^{k'}(e)} y^{k'}_{p',s'} \leq \frac{|H^*| - 1}{2}.$$

Separation of Slot-Assignment-Clique Inequalities Now, we describe the separation algorithm for the inequality (19). It consists in identifying a maximal clique C^* in the conflict graph $\tilde{G}_S^E$ s.t.

$$\sum_{v_{k,s} \in C^*} \sum_{p \in P^k} \bar{y}_{p,s}^k > 1,$$

for a given fractional solution $\bar{y}$ of the current LP.
To do so, we use the greedy algorithm introduced by Nemhauser and Sigismondi in [50] to identify a maximal clique C^* in conflict graph $\tilde{G}_S^E$ given that computing a maximal clique in such a graph is also NP-hard problem [32]. Based on this, we first assign a positive weight $\sum_{p \in P^k} \bar{y}_{p,s}^k$ to each node $v_{k,s}$ in the conflict graph $\tilde{G}_S^E$. We then select a node $v_{k,s}$ in the conflict graph $\tilde{G}_S^E$ having the largest weight compared with the other nodes in $\tilde{G}_S^E$, and set $C^* = \{v_{k,s}\}$. After that, we iteratively add each node $v_{k',s'}$ to the current C^* if it is linked with all the nodes $v_{k,s}$ already assigned to the current clique C^* and $\sum_{p' \in P^{k'}} \bar{y}_{p',s'}^{k'} > 0$. At the end, we add the inequality (19) induced by the clique C^* to the current LP if it is violated, i.e., we add the following inequality

$$\sum_{v_{k,s} \in C^*} \sum_{p \in P^k} y_{p,s}^k \leq 1.$$

Furthermore, it can be lifted by identifying a maximal clique N^* s.t. each $v_{k',s'} \in N^*$ is linked with all the nodes $v_{k,s} \in C^* \cup (N^* \setminus \{v_{k',s'}\})$ in $\tilde{G}_S^E$. For that, we use also the greedy algorithm introduced by Nemhauser and Sigismondi in [50] to identify the clique N^* as follows. We first set $N^* = \{v_{k',s'}\}$ with $v_{k',s'} \notin C^*$ a node in $\tilde{G}_S^E$ having the largest value of node-degree (i.e., $|\delta(v_{k',s'})|$) in $\tilde{G}_S^E$ and $v_{k',s'}$ is linked with all the nodes $v_{k,s} \in C^*$ in $\tilde{G}_S^E$. Afterwards, we iteratively add each node $v_{k',s'} \notin C^* \cup N^*$ to the current N^* if it is linked in $\tilde{G}_S^E$ with all the nodes already assigned to C^* and N^*. At the end, we add the inequality (19) induced by the clique $C^* \cup N^*$ to the current LP, i.e.,

$$\sum_{v_{k,s} \in C^*} \sum_{p \in P^k} y_{p,s}^k + \sum_{v_{k',s'} \in N^*} \sum_{p' \in P^{k'}} y_{p',s'}^{k'} \leq 1.$$

Separation of Slot-Assignment-Odd-Hole Inequalities For the inequality (21), our separation algorithm consists in identifying an odd-hole H^* in the conflict graph $\tilde{G}_S^E$ for a given fractional solution $\bar{y}$ s.t.

$$\sum_{v_{k,s} \in H^*} \sum_{p \in P^k} \bar{y}_{p,s}^k > \frac{|H^*| - 1}{2}.$$

This can be done in polynomial time as shown by Rebennack et al. in [57] and [58] by finding a minimum weighted odd-cycle in the conflict graph $\tilde{G}_S^E$. To do so, we first construct an auxiliary conflict graph $\tilde{G}_S^E$ which can be seen as a bipartite graph by duplicating each node $v_{k,s}$ in $\tilde{G}_S^E$ (i.e., $v_{k,s}$ and $v'_{k,s}$) and each two nodes are linked in $\tilde{G}_S^E$ if their original nodes are linked in $\tilde{G}_S^E$. We assign to each link $(\tilde{v}_{k,s}, \tilde{v}_{k',s'})$ in $\tilde{G}_S^E$ a weight equals to $\frac{1 - \sum_{p \in P^k} \bar{y}_{p,s}^k - \sum_{p' \in P^{k'}} \bar{y}_{p',s'}^{k'}}{2}$. We then compute for each node $v_{k,s}$ in $\tilde{G}_S^E$, the shortest path between $v_{k,s}$ and its copy in the auxiliary conflict graph $\tilde{G}_S^E$ denoted by $p_{v_{k,s}, v'_{k,s}}$. After that, we check if the total sum of weight over edges belonging to this path is smaller than $\frac{1}{2}$. If so, the odd-hole H^* is composed by all the original nodes of nodes belong the computed shortest path $p_{v_{k,s}, v'_{k,s}}$, i.e., $V(p_{v_{k,s}, v'_{k,s}}) \setminus \{v'_{k,s}\}$. As a result, the following inequality (21) induced by the odd-hole H^*

$$\sum_{v_{k,s} \in H^*} \sum_{p \in P^k} \bar{y}_{p,s}^k \leq \frac{|H^*| - 1}{2},$$

should be added to the current LP. Moreover, one can strengthen the inequality (21) induced by the odd-hole H^* using the greedy algorithm introduced by Nemhauser and Sigismondi in [50] to identify a maximal clique C^* in the conflict graph $\tilde{G}_S^E$ s.t. each node $v_{k',s'} \in C^*$ should have a link with all the nodes $v_{k,s} \in H^*$, and all the nodes $v_{k",s"} \in C^* \setminus \{v_{k',s'}\}$ in the conflict graph $\tilde{G}_S^E$. For that, we first assign a node $v_{k',s'} \notin H^*$ to the clique C^* (i.e., $C^* = \{v_{k',s'}\}$) s.t. $v_{k',s'}$ has the largest value of node-degree (i.e., $|\delta(v_{k',s'})|$) in $\tilde{G}_S^E$ and $v_{k',s'}$ is linked with all the nodes $v_{k,s} \in H^*$ in $\tilde{G}_S^E$. After that, we iteratively add each node $v_{k',s'} \notin H^* \cup C^*$ to the current clique C^* if it is linked in $\tilde{G}_S^E$ with all the nodes already assigned to the odd-hole H^* and the clique C^*. We then add the inequality (23) induced by the odd-hole H^* and clique C^*

$$\sum_{v_{k,s} \in H^*} \sum_{p \in P^k} \bar{y}_{p,s}^k + \frac{|H^*|-1}{2} \sum_{v_{k',s'} \in C^*} \sum_{p' \in P^{k'}} y_{p',s'}^{k'} \leq \frac{|H^*|-1}{2}.$$

Separation of Edge-Capacity-Cover Inequalities Let's now study the separation problem of the inequality (25). Given a fractional solution $\bar{y}$, and an edge $e \in E$. The separation problem associated with the inequality (25) is Np-Hard [33] given that it consists in identifying a cover $\tilde{K}^*$ the edge e, s.t.

$$\sum_{k \in \tilde{K}^*} \sum_{p \in P^k(e)} \sum_{s \in \mathbb{S}} \bar{y}_{p,s}^k > |\tilde{K}^*| - 1.$$

To do so, we propose a separation algorithm based on a greedy algorithm introduced by Nemhauser and Sigismondi in [50]. We first select a demand $k \in K \setminus K_e$ having largest number of requested slot w_k with $\sum_{p \in P^k(e)} \sum_{s \in \mathbb{S}} \bar{y}_{p,s}^k > 0$, and set $\tilde{K}^*$ to $\tilde{K}^* = \{k\}$. After that, we iteratively add each demand $k' \in K \setminus (K_e \tilde{K}^*)$ to $\tilde{K}^*$ with $\sum_{p' \in P^{k'}(e)} \sum_{s \in \mathbb{S}} \bar{y}_{p',s}^{k'} > 0$ and while $\sum_{k \in \tilde{K}^*} w_k \leq \bar{s} - \sum_{\tilde{k} \in K_e} w_{\tilde{k}}$, i.e., until a cover $\tilde{K}^*$ is obtained for the the edge e with $\sum_{k \in \tilde{K}^*} w_k > \bar{s} - \sum_{\tilde{k} \in K_e} w_{\tilde{k}}$. We further derive a minimal cover from the cover $\tilde{K}^*$ by deleting each demand $k \in \tilde{K}^*$ if $\sum_{k' \in \tilde{K}^* \setminus \{k\}} w_{k'} \leq \bar{s} - \sum_{\tilde{k} \in K_e} w_{\tilde{k}}$. We then add the inequality (25) induced by the minimal cover $\tilde{K}^*$ for the edge e to the current LP if it is violated, i.e.,

$$\sum_{k \in \tilde{K}^*} \sum_{p \in P^k(e)} \sum_{s \in \mathbb{S}} y_{p,s}^k \leq |\tilde{K}^*| - 1.$$

Furthermore, the inequality (25) induced by the minimal cover $\tilde{K}^*$ can be lifted by introducing an extended cover inequality (26) as follows

$$\sum_{k \in \tilde{K}^*} \sum_{p \in P^k(e)} \sum_{s \in \mathbb{S}} y_{p,s}^k + \sum_{k' \in \tilde{K}_e^*} \sum_{p' \in P^{k'}(e)} \sum_{s \in \mathbb{S}} y_{p',s}^{k'} \leq |\tilde{K}^*| - 1,$$

where $w_{k'} \geq w_k$ for each $k \in \tilde{K}^*$ and each $k' \in \tilde{K}_e^*$ with $k \notin K_e$.

8.8 Primal Heuristic

Here, we propose a primal heuristic based on a hybrid method between a local search algorithm and a greedy algorithm. It is necessary to boost the performance of our algorithms, obtain tighter bounds, accelerate our algorithm, and reduce the memory consumed by the tree of B&P and B&C&P by pruning certain nodes that are not interesting. Given a feasible fractional solution $\bar{y}$, our primal heuristic consists of constructing an integral "feasible" solution from this fractional solution. To do so, we propose a local search algorithm that consists of generating at each iteration a sequence of demands L (order) enumerated with $L = 1', 2', ..., |K|' - 1, |K|'$. Based on this sequence of demands, our greedy algorithm selects a path p and a slot s for each demand $k' \in L$ with $y_{p,s}^{k'} \neq 0$ while respecting the non-overlapping constraint with the set of demands that precede the demand k' in the list L (i.e., the demands $1', 2, ..., k'-1$). However, if there does not exist such pair of path p and slot s for the demand k', we then select a path p and a slot s for the demand $k' \in L$ with $y_{p,s}^{k'} = 0$ and $s \in \{w_{k'}, ..., \bar{s}\}$ while respecting the non-overlapping constraint with the set of demands that precede the demand k' in the list L. Algorithm 8.8 summarizes the different

Algorithm 3 Greedy-Algorithm for the B&P and B&C&P Algorithms

Data: A set of edges E, a spectrum $\mathbb{S}$, a multi-set K of demands, a set B_s^k of precomputed feasible paths for each demand $k \in K$ and slot $s \in \mathbb{S}$, an optimal solution y^* of the current LP, set FIX_0 of fixed variables to 0, a set FIX_1 of fixed variables to 1 in the current node in the tree of B&P or B&C&P, and a sequence of demands $L = 1', 2', ..., |K|' - 1, |K|'$

Result: integral solution

```
Set E_k = ∅, P_k = ∅, and S_k = ∅ for each demand k ∈ K  for each demand k' ∈ L do
    Set SERVED = FALSE  for each slot s ∈ {w_k', ..., s̄} do
        if SERVED = FALSE then
            Order the set of paths in B_s^k' in increasing order according to the total length of the paths
             p ∈ B_s^k', and let B'_s^k' denote the set of ordered paths in B_s^k'  for each path p ∈ B'_s^k' do
                if y_{p,s}^k' ∈ FIX_1 then
                    Set E_k' = E(p), S_k' = {s}, P_k' = {p}, and SERVED = TRUE
                end
                else
                    if y_{p,s}^k' ∉ FIX_0 and 0 < y*_{p,s}^k' ≤ 1 then
                        Set FEASIBLE= TRUE  for each demand k ∈ {1, ..., k' − 1} do
                            Let s_k denote the last-slot already selected for the demand k with s_k ∈ S_k  if
                             E(p) ∩ E_k ≠ ∅ and {s − w_k' + 1, ..., s} ∩ {s_k − w_k + 1, ...s_k} ≠ ∅ then
                                Set FEASIBLE= FALSE
                            end
                        end
                        if FEASIBLE = TRUE then
                            Set E_k' = E(p), S_k' = {s}, P_k' = {p}, and SERVED = TRUE
                        end
                    end
                end
            end
        end
    end
    if SERVED = FALSE then
        for each slot s ∈ {w_k', ..., s̄} do
            if SERVED = FALSE then
                for each path p ∈ B'_s^k' do
                    if y_{p,s}^k' ∉ FIX_0 and y*_{p,s}^k' = 0 then
                        Set FEASIBLE= TRUE  for each demand k ∈ {1, ..., k' − 1} do
                            Let s_k denote the last-slot already selected for k with s_k ∈ S_k  if E(p) ∩ E_k ≠ ∅
                             and {s − w_k' + 1, ..., s} ∩ {s_k − w_k + 1, ...s_k} ≠ ∅ then
                                Set FEASIBLE= FALSE
                            end
                        end
                        if FEASIBLE = TRUE then
                            Set E_k' = E(p), S_k' = {s}, P_k' = {p}, and SERVED = TRUE
                        end
                    end
                end
            end
        end
    end
end
```

Let $\mathcal{S} = (\{P_k \text{ for all } k \in K\}, \{S_k \text{ for all } k \in K\})$ be the final solution obtained by our greedy-algorithm. It is feasible for the C-RSA iff $E_k \neq \emptyset$ and $S_k \neq \emptyset$ for each demand $k \in K$ **return** integral solution $\mathcal{S}$ for current node in the tree of our B&P and B&C&P algorithms steps of our greed algorithm for a given sequence of demands.

After that, we compute the associated total length of the paths selected for the set of demands K in the final solution $\mathcal{S}$ given by the greedy algorithm. Our local search algorithm generates a new sequence by doing some permutation of demands in the last sequence of demands if the value of the solution given by the greedy algorithm is smaller than the value of the best solution found until the current iteration. Otherwise, we stop our algorithm, and we give in the output the best solution found during our primal heuristic induced by the best sequence of demands having the smallest value of the total length of the selected path compared with the other generated sequences. Algorithm 8.8 summarizes the different steps of our local search algorithm which calls our greedy-algorithm 8.8 at each iteration.

Algorithm 4 Primal Heuristic Based on a Hybrid Algorithm Between a Local Search Algorithm and Greedy-Algorithm for the B&P and B&C&P Algorithms.

Data: A set of edges E, a spectrum $\mathbb{S}$, a multi-set K of demands, a set B_s^k of precomputed feasible paths for each demand $k \in K$ and slot $s \in \mathbb{S}$, a maximum number of iterations $iter$, maximal size of neighborhood n

Result: integral solution

Let y^* be the optimal solution of the current LP Let FIX_0 be the fixed variables to 0 in the current node in the tree of B&P or B&C&P Let FIX_1 be the fixed variables to 1 in the current node in the tree of B&P or B&C&P Set $val^* = INF$, and best solution $\mathcal{S}^* = \emptyset$ Consider a sequence of demands $L = 1', 2', ..., |K|' - 1, |K|'$ Call the greedy-algorithm 8.8 based on the sequence L Let $\mathcal{S}$ be the final solution obtained by our greedy-algorithm 8.8 for the sequence L Compute its associated cost by summing the total length of the paths selected to route the demands K in the solution $\mathcal{S}$, denoted by VAL **if** *$\mathcal{S}$ is feasible* **then**

  Set $val^* = VAL$ Set $\mathcal{S}^* = \mathcal{S}$

end

Set $i = 1$ **while** $i \leq iter$ **do**

  Set $val_i^* = INF$ Construct n sequences denoted by $N(L)$ from the sequence L by doing some permutations between some demands selected randomly in the sequence L **for** *each neighbour $L_j \in N(L)$* **do**

    Call the greedy-algorithm 8.8 based on the sequence L_j Let $\mathcal{S}_j$ be the final solution obtained by our greedy-algorithm 8.8 for the sequence L_j Compute its associated cost by summing the total length of the paths selected to route the demands K in the solution $\mathcal{S}_j$, denoted by val_j **if** *$\mathcal{S}_j$ is feasible and $val_i^* > val_j$* **then**

      Set $val_i^* = val_j$ Set $\tilde{\mathcal{S}}_i^* = \mathcal{S}_j$

    end

  end

  if $val^* > val_i^*$ **then**

    Set $val^* = val_i^*$ Set $\mathcal{S}^* = \tilde{\mathcal{S}}_i^*$

  end

  Set $i = i++$

end

return integral solution $\mathcal{S}^*$ for current node in the tree of our B&P and B&C&P algorithms

In the next section, we will show the effectiveness of our approach.

9 Computational Results

9.1 Implementation's Feature

Our B&P and B&C&P algorithms have been implemented in C++ under Linux using the "Solving Constraint Integer Programs" (SCIP 7.0) framework [66]. For the resolution of the linear relaxation at each node in the B&P and B&C&P trees, SCIP uses CPLEX 12.9 [10]. These have been tested on LIMOS high-performance servers with a memory size limited to 64 Gb while benefiting from parallelism by activating 8 threads, and with a CPU time limited to 5 hours (18000 s).

9.2 Description of Instances

We further proposed a deep study of the behavior of our algorithl using two types of instances: random and real, and 14 graphs (topologies). They are composed of two types of graphs: real, and other realistics. They are composed of two types of graphs: real, and other realistics from SND-Lib [51] with a number of links $21 \leq |E| \leq 166$, and a number of nodes $14 \leq |V| \leq 161$ as shown in the table of Table 1. Note that we tested 4 instances for each triplet $(G, K, \bar{s})$ with $|K| \in \{10, 20, 30, 40, 50, 100, 150, 200, 250, 300\}$, and $\bar{s}$ up to 320 slots.

Topology		Number of Nodes	Number of Links	Max Node Degree	Min Node Degree	Average Node Degree
Real Topology	German	17	25	5	2	2.94
	Nsfnet	14	21	4	2	3
	Spain	30	56	6	2	3.73
	Conus75	75	99	5	2	2.64
	Coronet100	100	136	5	2	2.72
Realistic Topology	Europe	28	41	5	2	2.92
	France	25	45	10	2	3.6
	German50	50	88	5	2	3.52
	Brain161	161	166	37	1	2.06
	Giul39	39	86	8	3	4.41
	India35	35	80	9	2	4.57
	Pioro40	40	89	5	4	4.45
	Ta65	65	108	10	1	3.32
	Zib54	54	80	10	1	2.96

Table 1. Characteristics of different topologies used for our experiments.

9.3 Impact of Valid Inequalities

We first studied the efficiency of each family of valid inequalities introduced before to strengthen the linear relaxation of our B&P algorithm. To do so, we consider a subset of instances with a number of demands ranges in $\{10, 20, 30, 40, 50\}$ and $\bar{s}$ up to 50, while using three topologies (German, Nsfnet, and Spain). The results show that introducing each family of valid inequalities improves the effectiveness of our B&P algorithm considering 5 criteria, the average number of nodes in the enumeration tree (Nb_Nd), average gap (Gap) which represents the relative error between the lower bound gotten at the end of the resolution and best upper bound, average CPU time computation (T_Cpu), the average number of columns added during the pricing procedure (Ncols_Add), the average number of violated inequalities added (Ineq_Add). In fact, the results show that introducing each family of valid inequalities enables reducing the average number of nodes in the B&C&P tree, and also the average CPU time for several instances. Furthermore, we observe that adding valid inequalities decreases the average number of added columns for several instances. On the other hand, the results show that the cover-based inequalities (25) and (7) are efficient compared with those of clique-based inequalities (19), (14) and (9). In fact, our B&C&P algorithm is very efficient when adding the cover-based inequalities (25) and (7). We notice that adding these families of valid inequalities reduces the average gap, average number of nodes, average CPU time, and also the number of generated columns. Moreover, the results show also that several inequalities of the cover-based inequalities (25) and (7), and clique-based inequalities (19), (14) and (9), they are generated along our B&C&P algorithm. However, the number of clique-based inequalities (19) generated is very less high for the instances tested such that they have not generated for several instances. Based on these results, we conclude that our valid inequalities are very interesting to obtain tighter bounds and strengthen the linear relaxation. On the other hand, the different families of odd-hole inequalities are shown to be not efficient for the instances used such that the number of their violated inequalities generated is very less and equals to 0 for several instances. As a result, we combine these families of valid inequalities such that their separation is performed along with the B&C&P algorithm in the following order

1. edge-capacity-cover inequalities (25),
2. edge-interval-cover inequalities (7),
3. edge-slot-assignment-clique inequalities (14),
4. edge-interval-clique inequalities (9),
5. slot-assignment-clique inequalities (19).

We further provide a comparative study between B&P (without additional valid inequalities) and B&C&P (with additional valid inequalities) algorithms. To do so, we evaluate the impact of valid inequalities used together within our B&C&P algorithm. For this, we present some computational results using several instances with a number of demand ranges in $\{10, 20, 30, 40, 50, 100, 150, 200, 250, 300\}$ and $\bar{s}$ up to 320 slots. We classify instances in two classes: small-sized instances with number of demands $\{10, 20, 30, 40, 50\}$ and $\bar{s}$ up to 180, and ones of large-sized instances with number of demands ranges in $\{100, 150, 200, 250, 300\}$ and $\bar{s}$ up to 320. We use two types of topologies: real, and realistic ones from SND-LIB already described in Table 1.
Tables 2 and 3 respectively, present a comparison between B&P and B&C&P using small-scale and large-scale instances based on real graphs. In Tables 4 and 5 respectively, we give some numerical results obtained for small-sized instances and large-instances with additional valid inequalities. Note that the gap values given in red, represent the instances solved to optimality.
As reported in the Tables 2...5, the results show that adding several families of valid inequalities improves the effectiveness of our B&C&P algorithm compared with the classical approach when adding just one family of valid inequalities. In fact, we first notice that introducing valid inequalities allows solving several instances to optimality that are not solved to optimality using the B&P algorithm. Furthermore, they enabled reducing the average number of nodes in the B&C&P tree, and also the average CPU time for several instances. On the other hand, and when the optimality is not guaranteed, adding valid inequalities decreases the average gap for several instances. However, there exist few instances very rare in which adding valid inequalities does not improve the results of the B&P algorithm. Based on these results, we ensure that using our valid inequalities strengthens the linear relaxation of our path formulation.

9.4 Impact of Symmetry-Breaking Inequalities

Here we show the impact of our symmetry-breaking inequalities already introduced on the effectiveness of our B&P and B&C&P algorithms. To do this, we consider a subset of instances with a number of demands ranges in $\{10, 20, 30, 40, 50\}$ and $\bar{s}$ up to 50, while using three real topologies (German, Nsfnet, and Spain), and two realistic topologies (Ta65 and Zib54). The results are reported in the following Tables 9.4 and 9.4. As reported in these Tables, we notice that adding these symmetry-breaking inequalities allows solving to optimality some instances that are not solved to optimality using the B&C&P algorithm (without additional symmetry-breaking inequalities). Furthermore, they allow reducing the average gap, average number of nodes for several instances. However, there exist some cases in which adding these inequalities makes the problem hard for solving to optimality. As a result, we observe in the Tables 9.4 and 9.4, the B&P and B&C&P algorithms (without additional symmetry-breaking inequalities) are able to solve to optimality some instances that are not solved to optimality when adding our symmetry-breaking inequalities.

Instances			B&P_SCIP				B&C&P_SCIP				
Topology	$\|K\|$	$\|S\|$	Nb_Nd	Gap	T_Cpu	Ncols_Add	Nb_Nd	Gap	T_Cpu	Ncols_Add	Ncons_Add
Conus75	10	40	1	0,00	1,26	0	1	0,00	1,45	0	0
Conus75	20	40	1	0,00	2,57	0	1	0,00	2,72	0	0
Conus75	30	40	3694	0,33	18000	124,25	2648	0,26	18000	497,50	430
Conus75	40	40	2282	0,31	13500,73	233,50	954	0,28	13507,02	401,25	824
Conus75	50	80	1	0,00	15,02	0	1	0,00	14,46	0	0
Coronet100	10	40	1	0,00	32,58	1,75	1	0,00	29,82	1,75	0
Coronet100	20	40	1	0,00	42,94	0	1	0,00	37,95	0	0
Coronet100	30	40	1922,50	0,47	13501,82	232,50	634,50	1,64	14331,94	612,25	325,75
Coronet100	40	40	1818,50	0,40	13504,14	267	807	0,33	13513,17	976,50	413,75
Coronet100	50	80	2	0,00	51,97	0,75	189	0,04	4522,96	20	36,75
German	10	15	28	0,00	1,56	1,50	3,50	0,00	0,36	1,25	9,25
German	20	45	227,50	0,00	74,68	0	1	0,00	0,57	0	2,25
German	30	45	1,50	0,00	1,01	0	2,50	0,00	2,27	0	1,25
German	40	45	1002,50	0,37	4498,37	68,25	1107	0,17	4502,33	55	223
German	50	55	4243,50	0,35	18000	76,50	3132,75	0,17	13505,87	52,50	429
Nsfnet	10	15	1	0,00	0,05	0	1	0,00	0,06	0	0
Nsfnet	20	20	120,50	0,00	12,52	11	137	0,00	18,20	0	3
Nsfnet	30	30	1434	0,00	749,96	1	1292	0,00	391,26	0	0
Nsfnet	40	35	2030,50	0,21	5184,20	21,50	828	0,18	4527,57	0	1,50
Nsfnet	50	50	4305	0,45	13478,99	6,50	3926	0,45	13497,69	0	7,75
Spain	10	15	1	0,00	0,18	0	1	0,00	0,19	0	0
Spain	20	20	1	0,00	0,55	0,50	1	0,00	0,87	0,50	3
Spain	30	25	30,50	0,00	28,48	5,25	1	0,00	1,53	0,25	1,75
Spain	40	30	1912,50	0,07	4495,04	30,75	314,50	0,00	489,39	3,25	48,50
Spain	50	35	2506,25	0,11	13485,82	24,25	1818,50	0,10	5445,88	28,75	320,50

Table 2. Efficiency of a combination of valid inequalities using real topologies for small-scale instances : B&P Vs B&C&P.

Instances			B&P_SCIP				B&C&P_SCIP				
Topology	$\|K\|$	$\|S\|$	Nb_Nd	Gap	T_Cpu	Ncols_Add	Nb_Nd	Gap	T_Cpu	Ncols_Add	Ncons_Add
Conus75	100	120	51	0,00	1460,58	1,50	46	0,00	1745,17	1	21
Conus75	150	200	1	0,00	705,35	0	1	0,00	555,89	0	0
Conus75	200	240	6	0,01	5215,75	0,75	1,25	0,01	2119,99	0,75	7,25
Conus75	250	320	1	0,00	2101,99	0,50	1	0,00	2111,54	0,50	4,25
Conus75	300	320	1	0,00	6087,75	1,25	1	0,00	2828,78	1,25	0
Coronet100	100	120	1	0,00	139,61	0	1	0,00	158,51	0	0
Coronet100	150	200	1	0,00	461,86	0	1	0,00	429,36	0	0
Coronet100	200	280	12,50	0,02	5845,43	1	1,25	0,02	4142,16	1	3
Coronet100	250	320	12,25	0,01	18000	1,25	1	0,01	3769,06	1,25	12,50
Coronet100	300	320	3,50	0,08	9933,59	9	1	0,08	4414,07	9	0
German	100	140	1	0,00	11,25	0	1	0,00	11,84	0	0
German	150	210	1	0,00	53,03	0	1	0,00	74,89	0	0
German	200	260	99	0,01	4621,81	0	1	0,02	1079,52	0	3,25
German	250	320	1	0,00	847,06	0	1	0,00	764,77	0	0
German	300	320	1	0,00	1386,45	0	1	0,00	1496,42	0	0
Nsfnet	100	120	1	0,00	6,37	0	1	0,00	7,88	0	0
Nsfnet	150	160	1	0,00	29,08	0	1	0,00	44,37	0	0
Nsfnet	200	210	232	0,01	5497,21	0	196,50	0,01	5911,91	0	0,50
Nsfnet	250	285	1	0,00	375,21	0	1	0,00	484,91	0	0
Nsfnet	300	320	6,50	0,00	5125,24	0	7,50	0,00	5196,10	0	0,25
Spain	100	120	1	0,00	22,05	0	1	0,00	22,28	0	0
Spain	150	160	1	0,00	86,47	0,25	1	0,00	121,71	0,25	0,25
Spain	200	200	1	0,00	359,46	0,25	1	0,00	451,46	0,25	7,25
Spain	250	240	1	0,00	958,33	1,25	1	0,00	1779,34	1,25	11,25
Spain	300	280	1	0,00	1864,60	1,50	1	0,03	2274,13	1,50	6,25

Table 3. Efficiency of a combination of valid inequalities using real topologies for large-scale instances : B&P Vs B&C&P.

Instances			B&P_SCIP				B&C&P_SCIP				
Topology	$\|K\|$	$\|S\|$	**Nb_Nd**	**Gap**	**T_Cpu**	**Ncols_Add**	**Nb_Nd**	**Gap**	**T_Cpu**	**Ncols_Add**	**Ncons_Add**
Brain161	10	40	1	0,00	3,17	0	1	0,00	3,12	0	0
Brain161	20	40	1	0,00	6,25	0	1	0,00	6,15	0	0
Brain161	30	40	1	0,00	9,45	0	1	0,00	9,40	0	0
Brain161	40	40	1635,50	0,01	18000	0	1579	0,01	18000	0	205,50
Brain161	50	40	1932,25	0,73	18000	6,50	1774,50	0,36	18000	12,75	487,25
Europe	10	25	7	0,00	1,45	1,25	12,50	0,00	2,47	2,75	1,75
Europe	20	60	1196	0,14	4499,56	77	1857,50	0,00	4496,18	75,50	220,25
Europe	30	80	3861,50	0,24	13490,79	480	1649,50	0,20	8997,69	814,75	363
Europe	40	80	1998	0,46	9024,29	434,50	806,50	0,31	9139,18	428,50	866,50
Europe	50	180	1	0,00	13,83	0	1	0,00	11,96	0	0
France	10	50	409,50	0,00	341,88	115,75	61,50	0,00	47,28	37	11,50
France	20	60	3345	0,73	12010,59	88	176	0,39	4511,73	0	100,50
France	30	80	3549	0,74	18000	244,50	1090,50	0,55	18000	0,25	666,50
France	40	100	1932,50	0,33	13548,11	81,25	545,75	0,23	14633,48	0	332
France	50	120	1679,50	0,25	13499,67	78,75	543	0,23	13520,17	0	843,50
German50	10	35	1	0,00	0,88	0	1	0,00	0,97	0	0
German50	20	40	1	0,00	2,09	0	1	0,00	2,31	0	0
German50	30	50	1	0,00	4,05	0	1	0,00	4,53	0	0
German50	40	50	1	0,00	5,62	0	1	0,00	6	0	0
German50	50	50	1	0,00	7,26	0	1	0,00	7,62	0	0
Giul39	10	40	1	0,00	1,06	0	1	0,00	1,17	0	0
Giul39	20	40	1	0,00	2,17	0	1	0,00	2,31	0	0
Giul39	30	40	1	0,00	3,20	0	1	0,00	3,37	0	0
Giul39	40	40	1	0,00	4,45	0	1	0,00	4,74	0	0
Giul39	50	40	683,75	0,02	4502,95	0,50	1	0,00	9,40	0	4,50
India35	10	40	16,50	0,00	13,95	0	1	0,00	1,05	0	2
India35	20	40	1	0,00	1,67	0	1	0,00	2,10	0	14,25
India35	30	40	2456,50	0,10	8996,21	96,50	917,50	0,12	4665,90	84,75	115,50
India35	40	40	1830,50	0,50	13505,63	533	375	0,52	7073,75	82,25	219,50
India35	50	80	544	0,00	4509,40	0	368,50	0,00	4478,10	0	242,25
Pioro40	10	40	1	0,00	1,22	0,25	1	0,00	1,28	0,25	0
Pioro40	20	40	1	0,00	2,42	1,25	1	0,00	2,50	1,25	0
Pioro40	30	40	1	0,00	3,38	0	1	0,00	3,54	0	0
Pioro40	40	40	907,50	0,02	4503,98	14,75	1	0,00	20,20	1,25	11,25
Pioro40	50	80	388,50	0,00	4515,13	21	1	0,00	31,62	0,75	17,75
Ta65	10	40	1	0,00	1,58	0	1	0,00	1,71	0	0
Ta65	20	40	1	0,00	4,13	1,25	1	0,00	4,52	1,25	0
Ta65	30	40	706	0,01	4509,13	16,50	483	0,01	4513,46	28,75	173,25
Ta65	40	40	2098	0,09	13503,83	130,50	401,50	0,04	4526,49	63	156,50
Ta65	50	40	2339,50	0,05	18000	90,75	1573,50	0,01	13661,46	68,50	381,50
Zib54	10	40	1	0,00	1,30	0	1	0,00	1,25	0	301,75
Zib54	20	40	1	0,00	2,23	0,75	1	0,00	2,29	0,75	1583,75
Zib54	30	40	1	0,00	5,97	3,50	1	0,00	9,73	3,50	4999,25
Zib54	40	40	757,50	0,10	4503,19	28,25	66,50	0,00	478,74	11	13375,75
Zib54	50	40	2835,50	0,31	18000	92,25	2376	0,31	18000	92,50	143067

Table 4. Efficiency of a combination of valid inequalities using realistic topologies: B&P Vs B&C&P.

Instances			B&P_SCIP				B&C&P_SCIP				
Topology	$\|K\|$	$\|S\|$	**Nb_Nd**	**Gap**	**T_Cpu**	**Ncols_Add**	**Nb_Nd**	**Gap**	**T_Cpu**	**Ncols_Add**	**Ncons_Add**
Brain161	100	80	521,25	0,14	18000	0	402,50	0,22	18000	0	187,50
Brain161	150	160	131,50	0,01	18000	0	101	0,07	18000	0	49,25
Brain161	200	200	16,50	0,29	18000	0	2	0,10	16500,15	0	24,50
Brain161	250	240	1,50	0,32	18000	0	1	0,54	15984,66	0	12,75
Brain161	300	320	1,50	0,02	9439,62	0	1	0,08	6411,93	0	0
Europe	100	320	1	0,00	1086,09	0	1	0,00	1233,02	0	229,25
Europe	150	320	3	0,26	18000	0,75	1	0,12	16134,92	0	289,75
Europe	200	320	1	0,00	4045,59	0	1	0,00	2635,73	0	421
Europe	250	320	1	0,00	9118,67	0	1	0,00	10000,62	0	592,50
Europe	300	320	1	0,37	18000	0	1	0,00	16824,42	5,50	771,25
France	100	320	211,50	0,20	18000	5,75	32,50	0,32	18000	0	36,50
France	150	320	3,50	2,77	18000	21	1	4,42	18000	0	30,25
France	200	320	14	0,96	18000	3,25	2,50	2,51	18000	0	15,50
France	250	320	1	3,52	18000	0	1	4,34	18000	0	6,75
France	300	320	1	4,65	18000	0	1	7,55	18000	0	0
German50	100	100	1	0,00	48,10	0	1	0,00	35,63	0	0
German50	150	140	1	0,00	224,72	0	1	0,00	196,15	0	0
German50	200	140	44	0,21	8488,93	0	43	0,12	9239,41	39,25	0
German50	250	180	1	0,28	5897,62	0	1,50	0,43	7648,19	12	0
German50	300	180	1	0,57	18000	0	1	0,86	13968,59	12,25	0
Giul39	100	40	2290	0,16	18000	185,75	1253,75	1,13	16076,79	114,50	1598,75
Giul39	150	120	1	0,00	152,94	1,25	1	0,00	430,16	1,25	15
Giul39	200	120	82,50	0,00	6098,96	3,25	59	0,05	6188,09	3	90,25
Giul39	250	160	34,50	0,01	5339,55	1	26	0,00	5852,20	1	30,25
Giul39	300	200	38,50	0,08	18000	6	3,50	0,08	11583,66	5,25	17,25
India35	100	120	641	0,02	18000	0	225	0,01	9320,12	0	608
India35	150	200	31	0,00	4909,68	0	1,25	0,00	3378,23	0	19,50
India35	200	280	1	0,00	7046,13	0	1	0,05	2556,02	0	3
India35	250	280	1	0,01	8389,92	0	1	0,09	9262,44	0	15,25
India35	300	320	1	0,00	5521,26	0	1	0,00	6359,41	0	0
Pioro40	100	80	874,75	0,02	13712,83	15	432,50	0,01	9287,22	13,50	209,25
Pioro40	150	160	8,50	0,00	936,64	16,25	1	0,00	820,48	16,25	12,75
Pioro40	200	280	11	0,04	5780,53	16,50	1	0,04	3719,52	14,50	11,50
Pioro40	250	280	34,50	0,07	12054,33	26,25	1	0,13	7040,90	24,50	8,25
Pioro40	300	320	14	0,25	18000	34,25	1	0,27	9744,06	34,25	3
Ta65	100	80	628,75	0,02	13548,86	31,75	542,25	0,02	13847,85	38	326,25
Ta65	150	160	18,50	0,00	1460,38	3,75	24	0,00	2101,63	3,25	21
Ta65	200	200	1	0,00	1623,90	9	1	0,00	2190,52	8	14,25
Ta65	250	240	20	0,03	10794,63	12,75	1	0,07	5241	12,75	14
Ta65	300	280	10,75	0,10	15819,23	28,50	1	0,11	9440,85	6,75	11,75
Zib54	100	80	849,75	0,13	13770,26	80,25	524,75	0,52	13854,91	36,75	84096
Zib54	150	160	54	0,02	4799,19	15,25	6	0,19	2074,75	13	40639
Zib54	200	200	94	0,00	13018,71	7,75	3,25	0,15	4450,02	5,75	39934,25
Zib54	250	240	24,50	0,20	18000	6,75	1	0,22	5215,45	6,75	32673,50
Zib54	300	280	1,50	0,52	15029,79	26,25	1	0,63	9736,63	26,25	20400

Table 5. Efficiency of a combination of valid inequalities using realistic topologies for large-scale instance: B&P Vs B&C&P.

Instances			B&P_SCIP_Without_Symmetry_Breaking_Ineq				B&P_SCIP_With_Symmetry_Breaking_Ineq			
Topology	$\|K\|$	$\|S\|$	**Nbr_Nd**	**Gap**	**T_Cpu**	**Ncols_Add**	**Nbr_Nd**	**Gap**	**T_Cpu**	**Ncols_Add**
German	10	15	28,00	0,00	1,56	1,50	28,50	0,00	2,46	2,25
German	20	45	227,50	0,00	74,68	0,00	763,50	0,15	4504,82	0,00
German	30	45	1,50	0,00	1,01	0,00	1,00	0,00	27,35	0,00
German	40	45	1002,50	0,37	4498,37	68,25	1064,50	0,37	4677,83	60,25
German	50	55	4243,50	0,35	18000	76,50	3461,00	0,35	18000	85,50
Nsfnet	10	15	1,00	0,00	0,05	0,00	1,00	0,00	0,19	0,00
Nsfnet	20	20	120,50	0,00	12,52	11,00	254,50	0,00	94,80	3,25
Nsfnet	30	30	1434,00	0,00	749,96	1,00	1267,00	0,00	720,21	0,00
Nsfnet	40	35	2030,50	0,21	5184,20	21,50	836,00	0,21	4716,39	7,00
Nsfnet	50	50	4305,00	0,45	13478,99	6,50	1179,00	0,14	13928,73	0,00
Spain	10	15	1,00	0,00	0,18	0,00	1,00	0,00	0,32	0,00
Spain	20	20	1,00	0,00	0,55	0,50	1,00	0,00	2,14	0,25
Spain	30	25	30,50	0,00	28,48	5,25	65,00	0,00	179,56	5,25
Spain	40	30	2549,67	0,07	5993,17	41,00	1282,33	0,07	6164,94	44,33
Spain	50	35	2506,25	0,11	13485,82	24,25	2551,50	0,10	9611,75	27,25
Ta65	10	40	1,00	0,00	1,58	0,00	1,00	0,00	1,72	0,00
Ta65	20	40	1,00	0,00	4,13	1,25	1,00	0,00	9,41	0,75
Ta65	30	40	706,00	0,01	4509,13	16,50	665,50	0,01	4559,66	19,50
Ta65	40	40	2098,00	0,09	13503,83	130,50	1848,25	0,08	13534,41	165,50
Ta65	50	40	2339,50	0,05	18000	90,75	973,25	0,04	18000	74,75
Zib54	10	40	1,00	0,00	1,30	0,00	1,00	0,00	3,08	0,00
Zib54	20	40	1,00	0,00	2,23	0,75	1,00	0,00	13,54	0,25
Zib54	30	40	1,00	0,00	5,97	3,50	1,00	0,00	38,72	0,00
Zib54	40	40	757,50	0,10	4503,19	28,25	342,00	0,09	4611,08	34,00
Zib54	50	40	2835,50	0,31	18002,35	92,25	763,75	0,31	18000	44,25

Table 6. Efficiency of symmetry-breaking inequalities for the B&P algorithm using small-scale instances.

Instances			B&C&P_SCIP_Without_Symmetry_Breaking_Ineq					B&C&P_SCIP_With_Symmetry_Breaking_Ineq				
Topology	$\|K\|$	$\|S\|$	**Nbr_Nd**	**Gap**	**T_Cpu**	**Ncuts_Add**	**Ncols_Add**	**Nbr_Nd**	**Gap**	**T_Cpu**	**Ncuts_Add**	**Ncols_Add**
German	10	15	3,50	0,00	0,36	9,25	1,25	1,00	0,00	0,31	4,75	0,25
German	20	45	1,00	0,00	0,57	2,25	0,00	1,00	0,00	15,24	1,50	0,00
German	30	45	2,50	0,00	2,27	1,25	0,00	1,00	0,00	70,31	0,75	0,00
German	40	45	1107,00	0,17	4502,33	223,00	55,00	362,50	0,21	5007,95	68,25	35,00
German	50	55	3132,75	0,17	13505,87	429,00	52,50	175,00	0,16	12892,57	65,25	8,50
Nsfnet	10	15	1,00	0,00	0,06	0,00	0,00	1,00	0,00	0,19	0,00	0,00
Nsfnet	20	20	137,00	0,00	18,20	3,00	0,00	42,50	0,00	27,60	17,25	0,00
Nsfnet	30	30	1292,00	0,00	391,26	0,00	0,00	139,50	0,00	620,51	11,25	0,00
Nsfnet	40	35	828,00	0,18	4527,57	1,50	0,00	53,50	0,18	1002,96	12,25	0,00
Nsfnet	50	50	3926,00	0,45	13497,69	7,75	0,00	924,00	0,62	6558,79	51,00	0,00
Spain	10	15	1,00	0,00	0,20	0,00	0,00	1,00	0,00	0,32	0,00	0,00
Spain	20	20	1,00	0,00	0,87	3,00	0,50	1,00	0,00	1,86	0,00	0,25
Spain	30	25	1,00	0,00	1,53	1,75	0,25	1,00	0,00	6,11	3,75	0,25
Spain	40	30	419,00	0,00	652,29	64,67	4,33	383,00	0,00	2475,74	54,33	12,33
Spain	50	35	1818,50	0,10	4655,84	320,50	28,75	211,50	0,10	4490,18	142,25	9,00
Ta65	10	40	1,00	0,00	1,71	424,25	0,00	1,00	0,00	3,48	0,00	0,00
Ta65	20	40	1,00	0,00	4,52	2280,75	1,25	1,00	0,00	17,51	0,00	0,75
Ta65	30	40	483,00	0,01	4513,47	14585,25	28,75	1,00	0,00	279,41	8,75	2,25
Ta65	40	40	401,50	0,04	4526,49	17037,25	63,00	193,75	0,05	6909,24	182,50	70,25
Ta65	50	40	1573,50	0,01	13661,46	107263,50	68,50	378,25	0,00	10576,28	76,75	14,75
Zib54	10	40	1,00	0,00	1,25	301,75	0,00	1,00	0,00	3,68	0,00	0,00
Zib54	20	40	1,00	0,00	2,29	1583,75	0,75	1,00	0,00	11,52	0,00	0,25
Zib54	30	40	1,00	0,00	9,73	4999,25	3,50	114,50	0,08	2421,37	44,75	21,25
Zib54	40	40	66,50	0,00	478,74	13375,75	11,00	265,00	0,16	2343,81	49,50	8,75
Zib54	50	40	2376,00	0,31	18000	143067,00	92,50	526,00	0,13	13499,72	202,50	34,75

Table 7. Efficiency of symmetry-breaking inequalities for the B&C&P algorithm using small-scale instances.

10 Conclusion

In this paper, we studied the Constrained-Routing and Spectrum Assignment problem. We introduced integer linear programming based on the so-called path formulation for the problem. We further derive several valid inequalities for the associated polytope that have been shown to be efficient within the Branch-and-Cut-and-Price algorithm. As a result, we notice that the Branch-and-Cut-and-Price algorithm was very efficient compared with the Branch-and-Price algorithm using several instances. Some instances are still difficult to solve with both B&P and B&C&P algorithms. Our next step is to study the impact of the following branching strategies on the effectivness of the B&P and B&C&P algorithms.

10.1 Demand-Path-Slot Classical Variable Branching Strategy

Here, we use the classical branching schemes. We select a variable $y^k_{p,s}$ induced by a demand $k \in K$ and slot $s \in \{w_k, ..., \bar{s}\}$ and path $p \in B^k$ having the largest value ${y^*}^k_{p,s}$ with $0 < {y^*}^k_{p,s} < 1$. Then, our branching algorithm generates two nodes by selecting or not the slot s as last-slot along path p for the demand k, i.e., $y^k_{p,s} = 0$ or $y^k_{p,s} = 1$ which induces two new sub-problems. This has no impact on the structure of our pricing problem.

10.2 Demand-Path Constraint Branching Strategy

We propose a new branching scheme based on the Ryan and Foster branching scheme. It consists in branching on a constraint $0 \leq \sum_{s=1}^{\bar{s}} y^k_{p,s} \leq 1$ for a demand k and path $p \in B^k$ which is valid for the C-RSA problem given that $\sum_{p\in P^k} \sum_{s=1}^{\bar{s}} y^k_{p,s} = 1$. To do so, we select a demand $k \in K$ and a path $p \in B^k$ having the largest value of $\sum_{s=1}^{\bar{s}} {y^*}^k_{p,s}$ with $0 < \sum_{s=1}^{\bar{s}} {y^*}^k_{p,s} < 1$. Then, we generate two nodes by imposing the usage of the path p to route the demand k or no, i.e., we create two sub-problem with $\sum_{s=1}^{\bar{s}} y^k_{p,s} = 0$ or $\sum_{s=1}^{\bar{s}} y^k_{p,s} = 1$. However, if such pair of demand k and path p does not identified in a certain level of our algorithm, we select a variable $y^k_{p,s}$ induced by a demand $k \in K$ and slot $s \in \{w_k, ..., \bar{s}\}$ and path $p \in B^k$ having the largest value ${y^*}^k_{p,s}$ with $0 < {y^*}^k_{p,s} < 1$, and then generate two nodes by imposing that $y^k_{p,s} = 0$ or $y^k_{p,s} = 1$. Note that, branching in certain constraint $\sum_{s=1}^{\bar{s}} y^k_{p,s} = 0$ or $\sum_{s=1}^{\bar{s}} y^k_{p,s} = 1$, it has no impact on the structure of our pricing problem.

10.3 Demand-Slot Constraint Branching Strategy

Let us present now another branching scheme based on the Ryan and Foster branching scheme. It consists in branching on a constraint $0 \leq \sum_{p\in B^k} y^k_{p,s} \leq 1$ for a demand k and slot s which is valid for the C-RSA problem given that $\sum_{p\in P^k} \sum_{s=1}^{\bar{s}} y^k_{p,s} = 1$. To do so, we select a demand $k \in K$ and a slot s having the largest value of $\sum_{p\in B^k} {y^*}^k_{p,s}$ with $0 < \sum_{p\in B^k} {y^*}^k_{p,s} < 1$. Then, we generate two nodes by imposing the assignment of slot s as last-slot for the demand k or no, i.e., $\sum_{p\in P^k} y^k_{p,s} = 0$ or $\sum_{p\in P^k} y^k_{p,s} = 1$. However, if such pair of demand k and slot s does not exist in a certain level of our algorithm, we select a variable $y^k_{p,s}$ induced by a demand $k \in K$ and slot $s \in \{w_k, ..., \bar{s}\}$ and path $p \in B^k$ having the largest value ${y^*}^k_{p,s}$ with $0 < {y^*}^k_{p,s} < 1$, and then generate two nodes by imposing that $y^k_{p,s} = 0$ or $y^k_{p,s} = 1$. Note that, branching in certain constraint $\sum_{p\in P^k} y^k_{p,s} = 0$ or $\sum_{p\in P^k} y^k_{p,s} = 1$, it changes the calculation of the reduced-cost associated with the demand k and slot s in each child node of the current node as follows.

$$rc^k_s = \beta^k + \lambda^k_s + \min_{p\in P^k \setminus B^k} [\sum_{e\in E(p)} l_e - \sum_{s'=s-w_k+1}^{s} \mu^e_{s'}], \tag{77}$$

where $\lambda^k_s \in \mathbb{R}$ is the dual variable associated with this branching constraint induced by demand k and slot s. However, it has no impact on the structure of our pricing problem given that it consists in solving RCSP problem.

10.4 Demand-Edge Constraint Branching Strategy

In what follows, we introduce a new branching scheme based on the Ryan and Foster branching scheme. It consists in branching on a constraint $0 \leq \sum_{s=1}^{\bar{s}} \sum_{p \in B^k(e)} y_{p,s}^k \leq 1$ for a demand k and an edge $e \notin E \setminus (E_1^k \cup E_0^k)$ which is valid for the C-RSA problem given that $\sum_{p \in P^k} \sum_{s=1}^{\bar{s}} y_{p,s}^k = 1$. For this, we select a demand $k \in K$ and an edge $e \notin E \setminus (E_1^k \cup E_0^k)$ having the largest value of $\sum_{s=1}^{\bar{s}} \sum_{p \in B^k(e)} {y^*}_{p,s}^k$ with $0 < \sum_{s=1}^{\bar{s}} \sum_{p \in B^k(e)} {y^*}_{p,s}^k < 1$. Then, we generate two nodes by imposing the usage of edge e to route the demand k or no, i.e., $\sum_{s=1}^{\bar{s}} \sum_{p \in P^k(e)} y_{p,s}^k = 0$ or $\sum_{s=1}^{\bar{s}} \sum_{p \in P^k(e)} y_{p,s}^k = 1$. However, if such pair of demand k and edge e does not exist in a certain level of our algorithm, we select a variable $y_{p,s}^k$ induced by a demand $k \in K$ and slot $s \in \{w_k, ..., \bar{s}\}$ and path $p \in B^k$ having the largest value ${y^*}_{p,s}^k$ with $0 < {y^*}_{p,s}^k < 1$, and then generate two nodes by imposing that $y_{p,s}^k = 0$ or $y_{p,s}^k = 1$. Note that, branching in certain constraint $\sum_{s=1}^{\bar{s}} \sum_{p \in P^k(e)} y_{p,s}^k = 0$ or $\sum_{s=1}^{\bar{s}} \sum_{p \in P^k(e)} y_{p,s}^k = 1$, it changes the calculation of the reduced-cost associated with the demand k and slot s in each child node of the current node as follows.

$$rc_s^k = \beta^k + \min\Big(\min_{\substack{p \in P^k \setminus B^k, \\ e \in E(p)}} [\lambda_e^k + \sum_{e' \in E(p) \setminus \{e\}} l_{e'} - \sum_{s'=s-w_k+1}^{s} \mu_{s'}^{e'}], \min_{\substack{p \in P^k \setminus B^k, \\ e \notin E(p)}} [\sum_{e' \in E(p)} l_{e'} - \sum_{s'=s-w_k+1}^{s} \mu_{s'}^{e'}] \Big), \tag{78}$$

where $\lambda_e^k \in \mathbb{R}$ is the dual variable associated with this branching constraint induced by demand k and edge e. However, it has no impact on the structure of our pricing problem given that it consists in solving RCSP problem.

10.5 Edge-Slot Constraint Branching Strategy

We further present another constraint branching scheme which consists in branching on the non-overlapping constraint $0 \leq \sum_{p \in P^k(e)} \sum_{s'=s}^{\min(s+w_k-1,\bar{s})} y_{p,s'}^k \leq 1$. For this, we select an edge e and a slot s having the largest value of $\sum_{p \in P^k(e)} \sum_{s'=s}^{\min(s+w_k-1,\bar{s})} {y^*}_{p,s'}^k$ with $0 < \sum_{p \in P^k(e)} \sum_{s'=s}^{\min(s+w_k-1,\bar{s})} {y^*}_{p,s'}^k <$ 1. Then, we generate two nodes by imposing the usage of edge e to route the demand k or no, i.e., $\sum_{p \in P^k(e)} \sum_{s'=s}^{\min(s+w_k-1,\bar{s})} y_{p,s'}^k = 0$ or $\sum_{p \in P^k(e)} \sum_{s'=s}^{\min(s+w_k-1,\bar{s})} y_{p,s'}^k = 1$. However, if such pair of edge e and slot s does not exist, we select a variable $y_{p,s}^k$ induced by a demand $k \in K$ and slot $s \in \{w_k, ..., \bar{s}\}$ and path $p \in B^k$ having the largest value ${y^*}_{p,s}^k$ with $0 < {y^*}_{p,s}^k < 1$, and then generate two nodes by imposing that $y_{p,s}^k = 0$ or $y_{p,s}^k = 1$. Note that, branching in certain constraint $\sum_{s=1}^{\bar{s}} \sum_{p \in P^k(e)} y_{p,s}^k = 0$ or $\sum_{s=1}^{\bar{s}} \sum_{p \in P^k(e)} y_{p,s}^k = 1$, it changes the calculation of the reduced-cost associated with the demand k and each slot s' with $s \in \{s' - w_k + 1, ..., s'\}$ in each child node of the current node as follows.

$$\begin{aligned} rc_{s'}^k = \beta^k + \min\Big(& \min_{\substack{p \in P^k \setminus B^k, \\ e \in E(p) \text{ and } s \in \{s'-w_k+1,\dots,s'\}}} [\lambda_s^e + \sum_{e' \in E(p) \setminus \{e\}} l_{e'} - \sum_{s"=s'-w_k+1}^{s'} \mu_{s"}^{e'}], \\ & \min_{\substack{p \in P^k \setminus B^k, \\ e \notin E(p) \text{ or } s \notin \{s'-w_k+1,\dots,s'\}}} [\sum_{e' \in E(p)} l_{e'} - \sum_{s"=s'-w_k+1}^{s'} \mu_{s"}^{e'}] \Big), \end{aligned} \tag{79}$$

where $\gamma_s^e \in \mathbb{R}$ is the dual variable associated with this branching constraint induced by edge e and slot s. However, it has no impact on the structure of our pricing problem given that it consists in solving RCSP problem.

References

1. Amar, D. : Performance assessment and modeling of flexible optical networks. Theses, Institut National des Télécommunications 2016.

2. Bertero, F., and Bianchetti, M., and Marenco, J. : Integer programming models for the routing and spectrum allocation problem. In: Official Journal of the Spanish Society of Statistics and Operations Research 2018, pp. 465-488.
3. Cai, A. , Shen, G. , Peng, L. , and Zukerman, M. : Novel Node-Arc Model and Multiiteration Heuristics for Static Routing and Spectrum Assignment in Elastic Optical Networks. In: Journal of Lightwave Technology 2013, pp. 3402-3413.
4. Carlyle, W.M., Royset, J.O., and Wood, R.K.: Lagrangian relaxation and enumeration for solving constrained shortest-path problems. In: Networks Journal 2008, pp. 256-270.
5. Chen, X., and Guo, J., and Zhu, Z., and Proietti, R., and Castro, A., and Yoo, S.J.B. : Deep-RMSA: A Deep-Reinforcement-Learning Routing, Modulation and Spectrum Assignment Agent for Elastic Optical Networks. In: Optical Fiber Communications Conference and Exposition (OFC) 2018, pp. 1-3.
6. Cheng, B., and Hang, C., and Hu, Y., and Liu, S., and Yu, J., and Wang, Y., and Shen, J. : Routing and Spectrum Assignment Algorithm based on Spectrum Fragment Assessment of Arriving Services. In: 28th Wireless and Optical Communications Conference (WOCC) 2019, pp. 1-4.
7. Chouman, H., and Gravey, A., and Gravey, P., and Hadhbi, Y., and Kerivin, H., and Morvan, M, and Wagler, A.: Impact of RSA Optimization Objectives on Optical Network State. In: https://hal.uca.fr/hal-03155966.
8. Chouman, H., and Luay, A., and Colares, R., and Gravey, A, and Gravey, P., and Kerivin, H., and Morvan, M., and Wagler, A.: Assessing the Health of Flexgrid Optical Networks. In: https://hal.archives-ouvertes.fr/hal-03123302.
9. Christodoulopoulos, K., Tomkos, I., and Varvarigos, E.A. : Elastic Bandwidth Allocation in Flexible OFDM-Based Optical Networks. In: Lightwave Technology 2011, pp. 1354-1366.
10. Cplex, I.I., 2020. V12. 9: User's Manual for CPLEX. International Business Machines Corporation, 46(53), pp. 157.
11. Colares, R., Kerivin, H., and Wagler, A. : An extended formulation for the Constraint Routing and Spectrum Assignment Problem in Elastic Optical Networks. In: https://hal.uca.fr/hal-03156189, 2021.
12. Diarassouba, I., Hadhbi, Y., and Mahjoub, A.R.: Valid Inequalities and Branch-and-Cut Algorithm for the Constrained-Routing and Spectrum Assignment Problem. In: Research Report, July 2021, hal-03287146.
13. Diarassouba, I., Hadhbi, Y., and Mahjoub, A.R.: On the Facial Structure of the Constrained-Routing and Spectrum Assignment Polyhedron: Part I. In: Research Report in Hal UCA, July 2021, hal-03287174.
14. Diarassouba, I., Hadhbi, Y., and Mahjoub, A.R.: On the Facial Structure of the Constrained-Routing and Spectrum Assignment Polyhedron: Part II. In: Research Report in Hal UCA, July 2021, hal-03287205.
15. Ding, Z., and Xu, Z., and Zeng, X., and Ma, T., and Yang, F. : Hybrid routing and spectrum assignment algorithms based on distance-adaptation combined coevolution and heuristics in elastic optical networks. In: Journal of Optical Engineering 2014, pp. 1-10.
16. Dror, M. : Note on the Complexity of the Shortest Path Models for Column Generation in VRPTW. In: Journal of Operations Research 1994, pp. 977-978.
17. Dumitrescu, I., and Boland, N.: Algorithms for the weight constrained shortest path problem. In: International Transactions in Operational Research, pp. 15-29.
18. Enoch, J. : Nested Column Generation decomposition for solving the Routing and Spectrum Allocation problem in Elastic Optical Networks. In: http://arxiv.org/abs/2001.00066, 2020.
19. Eppstein, D. : Finding the k shortest paths. In: 35th Annual Symposium on Foundations of Computer Science, pp. 154-165.
20. Gong, L., and Zhou, X., and Lu, W., and Zhu, Z. : A Two-Population Based Evolutionary Approach for Optimizing Routing, Modulation and Spectrum Assignments (RMSA) in O-OFDM Networks. In: IEEE Communications Letters 2012, pp. 1520-1523.
21. Goscien, R., and Walkowiak, K., and Klinkowski, M. : Tabu search algorithm, Routing, Modulation and spectrum allocation, Anycast traffic, Elastic optical networks. In: Journal of Computer Networks 2015, pp. 148-165.
22. Grötschel, M. , Lovász, L., and Schrijver, A. : Geometric Algorithms and Combinatorial Optimization. In: Springer 1988.
23. Gu, R., Yang, Z., and Ji, Y.: Machine Learning for Intelligent Optical Networks: A Comprehensive Survey. In : Journal CoRR 2020, pp. 1-42.
24. Hadhbi, Y., Kerivin, H., and Wagler, A. : A novel integer linear programming model for routing and spectrum assignment in optical networks. In: Federated Conference on Computer Science and Information Systems (FedCSIS) 2019, pp. 127-134.
25. Hai, D.H., and Hoang, K.M. : An efficient genetic algorithm approach for solving routing and spectrum assignment problem. In: Journal of Recent Advances in Signal Processing 2017.

26. Hai, D.H., and Morvan, M., and Gravey, P.: Combining heuristic and exact approaches for solving the routing and spectrum assignment problem. In: Journal of Iet Optoelectronics 2017, pp. 65-72.
27. He, S., Qiu, Y., and Xu, J. : Invalid-Resource-Aware Spectrum Assignment for Advanced-Reservation Traffic in Elastic Optical Network. In: Sensors 2020.
28. Jaumard, B. , and Daryalal, M. : Scalable elastic optical path networking models. In: 18th International Conference Transparent Optical Networks (ICTON) 2016, pp. 1-4.
29. Jiang, R., and Feng, M., and Shen, J. : An defragmentation scheme for extending the maximal unoccupied spectrum block in elastic optical networks. In: 16th International Conference on Optical Communications and Networks (ICOCN) 2017, pp. 1-3.
30. Joksch, H.C. : The shortest route problem with constraints. In: Journal of Mathematical Analysis and Applications, pp. 191-197.
31. Kaibel, V., and Pfetsch, M. E.: Packing and partitioning orbitopes. In: Mathematical Programming 2008, pp. 136.
32. Karp, R.M.: Reducibility among Combinatorial Problems. In: Complexity of Computer Computations: Proceedings of a symposium on the Complexity of Computer Computations, held March 20–22, 1972, at the IBM Thomas J. Watson Research Center 1972, pp. 85-94.
33. Klabjan, D., Nemhauser, G.L., and Tovey, C. : The complexity of cover inequality separation. In: Journal of Operations Research Letters 1998, pp. 35-40.
34. Kaibel, V., Peinhardt, M., and Pfetsch, M. E. : Orbitopal fixing. In: Discrete Optimization 2011, pp. 595-610.
35. Klinkowski, M., Pedro, J., Careglio, D., Pioro, M., Pires, J., Monteiro, P., and Sole-Pareta, J. : An overview of routing methods in optical burst switching networks. In: Optical Switching and Networking 2010, pp. 41-53.
36. Klinkowski, M., and Walkowiak, K. : Routing and Spectrum Assignment in Spectrum Sliced Elastic Optical Path Network. In: IEEE Communications Letters 2011, pp. 884-886.
37. Klinkowski, M., Pioro, M., Zotkiewicz, M., Ruiz, M., and Velasco, L. : Valid inequalities for the routing and spectrum allocation problem in elastic optical networks. In: 16th International Conference on Transparent Optical Networks (ICTON) 2014, pp. 1-5.
38. Klinkowski, M., Pioro, M., Zotkiewicz, M., Ruiz, M., and Velasco, L. : A Simulated Annealing Heuristic for a Branch and Price-Based Routing and Spectrum Allocation Algorithm in Elastic Optical Networks. In: Intelligent Data Engineering and Automated Learning – IDEAL 2015, Springer International Publishing, pp. 290-299.
39. Lezama, F., Martinez-Herrera, A.F., Castanon, G., Del-Valle-Soto, C., Sarmiento, A.M., Munoz de Cote, A. : Solving routing and spectrum allocation problems in flexgrid optical networks using precomputing strategies. In: Journal of Photon Netw Commun 41, pp. 17-35.
40. Liu, L., and Yin, S., and Zhang, Z., and Chu, Y., and Huang, S. : A Monte Carlo Based Routing and Spectrum Assignment Agent for Elastic Optical Networks. In: Asia Communications and Photonics Conference (ACP) 2019, pp. 1-3.
41. Lohani, V., Sharma, A., and Singh, Y.N. : Routing, Modulation and Spectrum Assignment using an AI based Algorithm. In: 11th International Conference on Communication Systems & Networks (COMSNETS) 2019, pp. 266-271.
42. Lopez, V., and Velasco, L.: Elastic Optical Networks: Architectures, Technologies, and Control. In: Springer Publishing Company, Incorporated 2016.
43. Lozano, L., and Medaglia, A.L. : On an exact method for the constrained shortest path problem. In: Journal of Computers & Operations Research, pp. 378-384.
44. Margot, F. : Symmetry in integer linear programming. In: 50 Years of Integer Programming 1958-2008, Springer, 2010, pp. 647-686.
45. Margot, F. : Pruning by isomorphism in branch-and-cut. In: Mathematical Programming 2002, pp. 71-90.
46. Margot, F. : Exploiting orbits in symmetric ilp. In: Mathematical Programming 2003, pp. 3-21.
47. Mahala, N., and Thangaraj, J. : Spectrum assignment technique with first-random fit in elastic optical networks. In : 4th International Conference on Recent Advances in Information Technology (RAIT) 2018, pp. 1-4.
48. Mesquita, L.A.J., and Assis, K., and Santos, A.F., and Alencar, M., and Almeida, R.C. : A Routing and Spectrum Assignment Heuristic for Elastic Optical Networks under Incremental Traffic. In: SBFoton International Optics and Photonics Conference (SBFoton IOPC) 2018, pp. 1-5.
49. Nemhauser, G.L., and Wolsey, L.A. : Integer and Combinatorial Optimization. In: John Wiley 1988.
50. Nemhauser, G. L., and Sigismondi, G.: A Strong Cutting Plane/Branch-and-Bound Algorithm for Node Packing. In: The Journal of the Operational Research Society 1992, pp. 443-457.
51. Orlowski, S., Pióro, M., Tomaszewski, A. , and Wessäly, R.: SNDlib 1.0-Survivable Network Design Library. In: Proceedings of the 3rd International Network Optimization Conference (INOC 2007), Spa, Belgium, http://www.zib.de/orlowski/Paper/OrlowskiPioroTomaszewskiWessaely2007-SNDlib-INOC.pdf.gz.

52. Ostrowski, J., Anjos, M. F., and Vannelli, A. : Symmetry in scheduling problems. In: Citeseer 2010.
53. Ostrowski, J., Linderoth, J., Rossi, F., and Smriglio, S.: Orbital branching. In: Mathematical Programming 2011, pp. 147-178.
54. Padberg, M.W. : On the facial structure of set packing polyhedra. In: Journal of Mathematical Programming 1973, pp. 199-215.
55. Ramaswami, R. : Optical Networks: A Practical Perspective, 3rd Edition. In: Morgan Kaufmann Publishers Inc. 2009.
56. Ramaswami, R., Sivarajan, K., and Sasaki, G. : Multiwavelength lightwave networks for computer communication. In: IEEE Communications Magazine 1993, pp. 78-88.
57. Rebennack, S. : Stable Set Problem: Branch & Cut Algorithms. In: Encyclopedia of Optimization Book 2009.
58. Rebennack, S., and Reinelt, G., and Pardalos, P.M. : A tutorial on branch and cut algorithms for the maximum stable set problem. In: Journal of International Transactions in Operational Research 2012, pp. 161-199.
59. Ruiz, M., Pioro, M., Zotkiewicz, M., Klinkowski, M., Velasco, L. : A hybrid meta-heuristic approach for optimization of routing and spectrum assignment in Elastic Optical Network (EON). In: Journal of Enterprise Information Systems 2020, pp. 11-24.
60. Ruiz, M., Pioro, M., Zotkiewicz, M., Klinkowski, M., Velasco, L. : Column generation algorithm for RSA problems in flexgrid optical networks. In: Photonic Network Communications 2013, pp. 53-64.
61. Ryan, D. M. and Foster, B. A.: An integer programming approach to scheduling. In A. Wren (editor), Computer Scheduling of Public Transport Urban Passenger Vehicle and Crew Scheduling, North-Holland, Amsterdan, 1981, pp. 269-280.
62. Salani, M., and Rottondi, C., and Tornatore, M. : Routing and Spectrum Assignment Integrating Machine-Learning-Based QoT Estimation in Elastic Optical Networks. In: IEEE INFOCOM - IEEE Conference on Computer Communications 2019, pp. 173846.
63. Santos, A.F.D, and Assis, K., and Guimarães, M.A., and Hebraico, R.: Heuristics for Routing and Spectrum Allocation in Elastic Optical Path Networks. In: 2015, Journal Of Modern Engineering Research (IJMER), pp. 1-13.
64. Shirazipourazad, S., Zhou, C. , Derakhshandeh, Z., and Sen, A. : On routing and spectrum allocation in spectrum-sliced optical networks. In: Proceedings IEEE INFOCOM 2013, pp. 385-389.
65. Schrijver, A. : Combinatorial Optimization - Polyhedra and Efficiency. In: Springer-Verlag 2003.
66. Gamrath, G., Anderson, D., Bestuzheva, K., Chen, W.K., Eifler, L., Gasse, M., Gemander, P., Gleixner, A., Gottwald, L., Halbig, K., and Hendel, G., and Hojny, C., Koch, T., Bodic, L., Maher, P. J., Matter, F., Miltenberger, M., Mühmer, E., Müller, B., Pfetsch, M.E., Schlösser, F., Serrano, F., Shinano, Y., Tawfik, C., Vigerske, S., Wegscheider, F., Weninger, D., and Witzig, J.: The SCIP Optimization Suite 7.0. In: http://www.optimization-online.org/DB_HTML/2020/03/7705.html, March 2020.
67. Talebi, S. , Alam, F. , Katib, I. , Khamis, M. , Salama, R. , and Rouskas, G. N. : Spectrum management techniques for elastic optical networks: A survey. In: Optical Switching and Networking 2014, pp. 34-48.
68. Velasco, L. , Klinkowski, M. , Ruiz, M. , and Comellas, J. : Modeling the routing and spectrum allocation problem for flexgrid optical networks. In: Photonic Network Communications 2012, pp. 177-186.
69. Zhang, Y., Xin, J., and Li, X., and Huang, S. : Overview on routing and resource allocation based machine learning in optical networks. In: Journal of Optical Fiber Technology, pp. 1-21.
70. Zotkiewicz, M. , Pioro, M. , Ruiz, M. ,Klinkowski, M. , and Velasco, L. : Optimization models for flexgrid elastic optical networks. In: 15th International Conference on Transparent Optical Networks (ICTON) 2013, pp. 1-4.