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

Ibrahima Diarrassouba, youssouf Hadhbi, Ali Ridha Mahjoub

Valid Inequalities and Branch-and-Cut 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. The Constrained-Routing and Spectrum Assignment (C-RSA) problem arises in the dimensioning and management of a next-generation of optical transport networks, called Spectrally Flexible Optical Networks (SFONs). The C-RSA can be stated as follows. Given an SFONs as a graph G, and an optical spectrum $\mathbb{S}$ of available contiguous frequency slots, and a multiset of traffic demands K, it aims at determining for each demand $k \in K$ a path in G and an interval of contiguous slots in $\mathbb{S}$ while satisfying technological constraints, and optimizing some linear objective function(s). To the best of our knowledge, a cutting-plane-based approach has not been yet considered for the problem. For that, the main aim of our work is to introduce an integer linear programming formulation and provide several classes of valid inequalities for the associated polyhedron. We further discuss their separation problems. Using the polyhedral results and the separation procedures, we devise a Branch-and-Cut algorithm to solve the problem. We also present some computational results and show the effectiveness of our approach using real and some realistic network topologies.

1 Introduction

The global Internet Protocol (IP) traffic is expected to reach 396 exabytes per month by 2022, up from 194.4 Exabytes per month in 2020 [82]. Optical transport networks are then facing a serious challenge related to 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) [9], etc... To sustain the network operators face this trend of increase in bandwidth, a new generation of optical transport network architecture called Spectrally Flexible Optical Networks (SFONs) (called also FlexGrid Optical Networks) has been introduced as promising technology because of their flexibility, scalability, efficiency, reliability, survivability [7][9] compared with the traditional FixedGrid Optical Wavelength Division Multiplexing (WDM)[66][67]. In SFONs the optical spectrum is divided into small spectral units, called frequency slots as shown in Figure 1. They have the same frequency of 12.5 GHz where WDM uses 50 GHz as recommended by ITU-T [1]. This concept of slots was proposed initially by Jinno et al. in 2008 [36], and later explored by the same authors in 2010 [85]. This can be seen as an improvement in resource utilization. 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 a primary role when dimensioning and designing of SFONs. It can be seen as the main task for the development of this next generation of optical networks. It consists of assigning for each traffic demand, a physical optical path, and an interval of contiguous slots (called also channels) while optimizing some linear objective(s) and satisfying the following constraints [29]:

1. *spectrum contiguity*: an interval of contiguous slots should be allocated to each demand k with a width equal to the number of slots requested by demand k;

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

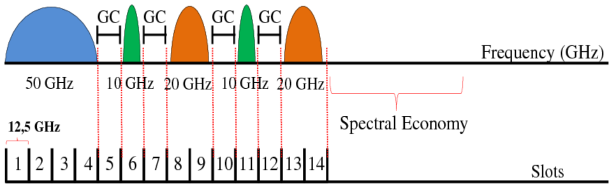


Fig. 1. Slot concept illustration in SFONs [75].

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 demands whose paths are not edge-disjoints in the network cannot share any slot over the shared edges.

1.1 Related Works

Numerous research studies have been conducted on the RSA problem since its first appearance. The RSA is known to be an NP-hard problem [78] [81], and is more complex than the historical Routing and Wavelength Assignment (RWA) problem [32]. Various integer linear programming (ILP) formulations and algorithms have been proposed to solve it. A detailed survey of spectrum management techniques for SFONs is presented in [81] where authors classified variants of the RSA problem: offline RSA which has been initiated in [61], and online or dynamic RSA which has been initiated in [86] and recently developed in [56] and [89], and an investigation of numerous aspects proposed in the tutorial [6]. This work focuses on the offline RSA problem. There exist two classes of ILP formulations used to solve the RSA problem, called edge-path and edge-node formulations. The ILP edge-path formulation is majorly used in the literature where variables are associated with all possible paths inducing huge variables and constraints that grow exponentially and in parallel with the growth of the instance size: number of demands, the total number of slots, and topology size: number of links and nodes [29]. To the best of our knowledge, we observe that several papers which use the edge-path formulation as an ILP formulation to solve the RSA problem, use a set of precomputed-paths without guaranty of optimality e.g. in [12], [61], [62], [84], [91], and recently in [73]. On the other hand, column generation techniques have been used by Klinkowski et al. in [71], Jaumard et al. in [34], and recently by Enoch in [19] to solve the relaxation of the RSA taking into account all the possible paths for each traffic demand. To improve the LP bounds of the RSA relaxation, Klinkowsky et al. proposed in [63] a valid inequality based on clique inequality separable using a branch-and-bound algorithm. On the other hand, Klinkowski et al. in [64] propose a branch-and-cut-and-price method based on an edge-path formulation for the RSA problem. Recently, Fayez et al. [21], and Xuan et al. [87], they proposed a decomposition approach to solve the RSA separately (i.e., R+SA) based on a recursive algorithm and an ILP edge-path formulation.

To overcome the drawbacks of the edge-path formulation usage, a compact edge-node formulation has been introduced as an alternative for it. It 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. [4], [84], [91].

On the other front, and due to the NP-Hardness of the C-RSA problem, we found that several heuristics [16],[49],[75], and recently in [33], and greedy algorithms [44], and metaheuristics as tabu search in [25], simulated annealing in [64], genetic algorithms in [23], [31], [32], ant colony algorithms in [39] , and a hybrid meta-heuristic approach in [70], have been used to solve large sized instances of the RSA problem. Furthermore, some resseraches start using some artificial intelligence algorithms, see for example [40] and [41], and some deep-learning algorithms [8], and also machine-learning algorithms in [74], and recently in [88] and [27] to get more perefermonce. Selvakumar et al. gives a survey in [77] in which they summarise the most contributions done for the RSA problem before 2019.

In this paper, we are interested in the resolution of a complex variant of the RSA problem, called the Constrained-Routing and Spectrum Assignment (C-RSA) problem. Here we suppose that the network should also satisfy the transmission-reach constraint for each traffic demand according to the actual service requirements. To the best of our knowledge a few related works on the RSA, to say the least, take into account this additional constraint such that the length of the chosen path for each traffic demand should not exceed a certain length (in kms). Recently, Hadhbi et al. in [29] and [30] introduced a novel tractable ILP based on the cut formulation for the C-RSA problem with a polynomial number of variables and an exponential number of constraints separable in polynomial time using network flow algorithms. Computational results show that their cut formulation solves larger instances compared with those of Velasco et al. in [84] and Cai et al. [4]. It has been used also as a basic formulation in the study of Colares et al. in [15], and also by Chouman et al. in [10] and [11] to show the impact of several objective functions on the optical network state. Bertero et al. in [3] give a comparative study between several edge-node formulations and introduce new ILP formulations adapted from the existing ILP formulations in the literature. Note that Velasco et al. in [84] and Cai et al. [4] did not take into account the transmission-reach constraint.

1.2 Our Contributions

However, so far the exact algorithms proposed in the literature could not solve large-sized instances. We believe that a cutting-plane-based approach could be powerful for the problem. To the best of our knowledge, such an approach has not been yet considered. For that, the main aim of our work is to investigate thoroughly the theoretical properties of the C-RSA problem. To this end, we aim to provide deeper theoretical analysis and design an efficient Branch-and-Cut algorithm to solve the C-RSA problem considering large-scale networks compared with what are often used. Our contribution is to introduce a new ILP formulation for the C-RSA problem which can be seen as an improved formulation for the one introduced by Hadhbi et al. in [29] and [30]. We further identify several classes of valid inequalities to obtain tighter LP bounds. Some of these inequalities are obtained by using *conflict graphs* related to the problem: clique inequalities, odd-hole, and lifted odd-hole inequalities. We also use the Chvatal-Gomory procedure to generate larger classes of inequalities. We then devise their separation procedures and use them to devise Branch-and-Cut (B&C) algorithm tree to solve the problem. Moreover, 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 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&C tree.

1.3 Organization

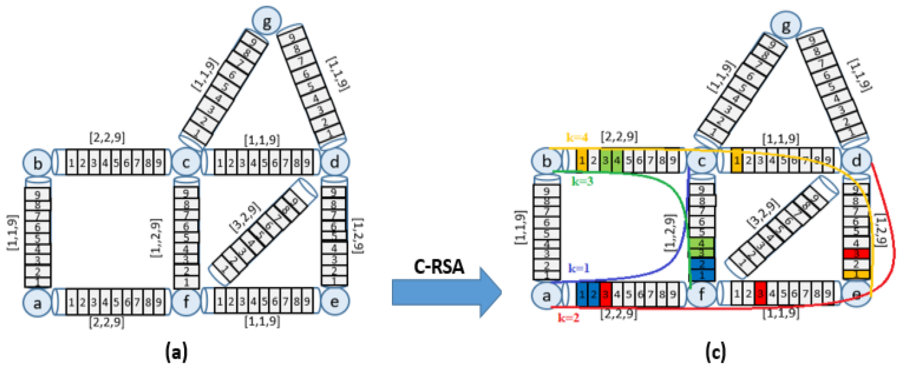
Following the introduction, the rest of this paper is organized as follows. In Section (2), we present the C-RSA problem (input and output). In Section (3), we provide the notation, then we introduce our ILP, called cut formulation based on the so-called cut inequalities. In Section (4), we thoroughly investigate the theoretical properties of the C-RSA problem by providing several valid inequalities. Based on the results of sections (3)-(4), we give an outline of our Branch-and-Cut algorithm in the section (5). We close with a brief summary of results and future outlook.

2 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

[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.

slots with $\bar{s} \leq 320$ given that the maximum spectrum bandwidth of each fiber-link is 4000 GHz [35], and K be a multiset [5] of demands such that each demand $k \in K$ is specified by an origin 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)) s.t. 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 $\bar{s} = 9$.



k	$o_k \to d_k$	w_k	$\bar{\ell}_k$
1	$a \to c$	2	4
2	$a \to d$	1	4
3	$b \to f$	2	4
4	$b \to e$	1	4

(b)

k	$o_k \to d_k$	$(o_k, d_k) - path$	S_k
1	$a \to c$	$a - f - c$	1 2
2	$a \to d$	$a - f - e - d$	3
3	$b \to f$	$b - c - f$	3 4
4	$b \to e$	$b - c - d - e$	1

(d)

Inputs Outputs

Fig. 2. 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).

3 The C-RSA Integer Linear Programming Formulation

Let's introduce some notations which will be useful throughout this paper to formulate some constraints. For any subset of nodes $X \subseteq V$ with $X \neq \emptyset$, let $\delta(X)$ denote the set of edges having one extremity in X and the other one in $\bar{X} = V \setminus X$ which is called a cut. When X is a singleton (i.e., $X = \{v\}$), we use $\delta(v)$ instead of $\delta(\{v\})$ to denote the set of edges incidents with a node $v \in V$. The cardinality of a set K is denoted by $|K|$.
Here we introduce our integer linear programming formulation based on cut formulation for the C-RSA problem which can be seen as a reformulation of the one introduced by Hadhbi et al. in [29]. For $k \in K$ and $e \in E$, let x_e^k be a variable which takes 1 if demand k goes through the edge e and 0 if not, and for $k \in K$ and $s \in \mathbb{S}$, let z_s^k be a variable which takes 1 if slot s is the last-slot allocated for the routing of demand k and 0 if not. The contiguous slots $s' \in \{s - w_k + 1, ..., s\}$ should be assigned to demand k whenever $z_s^k = 1$.
Before introducing our ILP, we proceeded to some pre-processing techniques to determine some

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

zero-one variables s.t. we are able to determine them in polynomial time using shortest-path and network flows algorithms 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. Table 1 below shows the set of forbidden edges E_0^k and forbidden nodes V_0^k for each demand k in K already given in Fig. 2(b).

k	$o_k \to d_k$	w_k	$\bar{\ell}_k$	V_0^k	E_0^k
1	$a \to c$	2	4	$\{e, d, g\}$	$\{cg, dg, de, df, cd, ef\}$
2	$a \to d$	1	4	$\{g\}$	$\{cg, dg, df\}$
3	$b \to f$	2	4	$\{e, d, g\}$	$\{cg, dg, de, df, cd, ef\}$
4	$b \to e$	1	4	$\{g\}$	$\{cg, dg, df\}$

Table 1. Topology pre-processing for the set of demands K given in Fig. 2(b).

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 s.t. $o_k \in W$ 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 $\delta(W)$ s.t. 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 s.t. $o_k \in W$ 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 s.t. edge e is an essential edge for each demand $k \in K_e$.
In addition to the forbidden edges thus obtained due to the transmission-reach constraints, there may exist 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 $\delta(v) \subseteq E_t^k$, should also be deleted from the proper graph G_k of demand k, which means that G_k contains $|E| \setminus |E_t^k|$ edges and $|V| \setminus |\{v \in V, \delta(v) \subseteq E_t^k\}|$ nodes. Let $E_0^k = E_t^k$ 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 $e = ij \notin E_0^k \cup E_1^k \cup E_0^{k'} \cup E_1^{k'}$, 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.
The C-RSA problem can hence be formulated as follows.

$$\min \sum_{k \in K} \sum_{e \in E} l_e x_e^k, \tag{1}$$

subject to

$$\sum_{e \in \delta(X)} x_e^k \geq 1, \forall k \in K, \forall X \subseteq V \text{ s.t. } |X \cap \{o_k, d_k\}| = 1, \tag{2}$$

$$\sum_{e \in E} l_e x_e^k \leq \bar{\ell}_k, \forall k \in K, \tag{3}$$

$$x_e^k = 0, \forall k \in K, \forall e \in E_0^k, \tag{4}$$

$$x_e^k = 1, \forall k \in K, \forall e \in E_1^k, \tag{5}$$

$$z_s^k = 0, \forall k \in K, \forall s \in \{1, ..., w_k - 1\}, \tag{6}$$

$$\sum_{s=w_k}^{\bar{s}} z_s^k = 1, \forall k \in K, \tag{7}$$

$$x_e^k + x_e^{k'} + \sum_{s'=s}^{min(s+w_k-1, \bar{s})} z_{s'}^k + \sum_{s'=s}^{min(s+w_{k'}-1, \bar{s})} z_{s'}^{k'} \leq 3, \forall (e, k, k', s) \in Q, \tag{8}$$

$$0 \leq x_e^k \leq 1, \forall k \in K, \forall e \in E, \tag{9}$$

$$z_s^k \geq 0, \forall k \in K, \forall s \in \mathbb{S}, \tag{10}$$

$$x_e^k \in \{0, 1\}, \forall k \in K, \forall e \in E, \tag{11}$$

$$z_s^k \in \{0, 1\}, \forall k \in K, \forall s \in \mathbb{S}. \tag{12}$$

where Q denotes the set of all the quadruples (e, k, k', s) for all $e \in E, k \in K, k' \in K$, and $s \in \mathbb{S}$ with $(k, k') \notin K_c^e$.
Inequalities (2) ensure that there is an (o_k, d_k)-path between o_k and d_k for each demand k, and guarantee that all the demands should be routed. They are called cut inequalities. By optimizing the objective function (1), and given that the capacities of all edges are strictly positives, this ensures that there is exactly one (o_k, d_k)-path between o_k and d_k which will be selected as optimal path for each demand k. We suppose that we have sufficient capacity in the network so that all the demands can be routed. This means that we have at least one feasible solution for the problem.
Inequalities (3) express the length limit on the routing paths which is called "the transmission-reach constraint". Equations (4) ensure that the variables associated to the forbidden edges for demand k are always equal to 0, and those of the essential edges are always equal to 1 for demand k. Equations (6) express the fact that a demand k cannot use slot $s \leq w_k - 1$ as the last-slot . The slots $s \in \{1, ..., w_k - 1\}$ are called forbidden last-slots for demand k. Inequalities (7) ensure that exactly one slot $s \in \{w_k, \ldots, \bar{s}\}$ must be assigned to demand k as last-slot. Inequalities (8) express the contiguity and non-overlapping constraints. Inequalities (9)-(10) are the trivial inequalities, and constraints (11)-(12) are the integrality constraints.
Note that the linear relaxation of the C-RSA can be solved in polynomial time given that inequalities (2) can be separated in polynomial time using network flows, see e.g. preflow algorithm of Goldberg and Tarjan introduced in [24] which can be run in $O(|V \setminus V_0^k|^3)$ time for each demand $k \in K$.

Proposition 1. *The formulation* (2)-(12) *is valid for the C-RSA problem.*

Proof. It is trivial given the definition of each constraint of the formulation (2)-(12) such that any feasible solution for this formulation is necessary a feasible solution for the C-RSA problem.

Proposition 2. *Every solution of our cut formulation* (1)-(12) *is a solution of multi-commodity flow problem.*

Proof. It is trivial given that any feasible solution of the C-RSA problem ensures that there is a flow of w_k slots routed along a path p_k which links between the origin-node o_k and destination-node d_k for each demand $k \in K$ while satisfying the capacity of edges which equals to $\bar{s}$.

Proposition 3. *Every solution of multi-commodity flow problem is not necessary feasible for our cut formulation* (1)-(12)*.*

Proof. It is trivial given that the solution of the multi-commodity flow problem can easily violate the contiguity and continuity constraints of our C-RSA problem. This means that the w_k slots assigned to the demand k can be not contiguous in a feasible solution of multi-commodity flow problem, and also for example when the w_k slots can be not the same along the path p_k for the demand k.

4 Valid Inequalities

An instance of the C-RSA is defined by a triplet $(G, K, \mathbb{S})$. Let $P(G, K, \mathbb{S})$ be the polytope, convex hull of the solutions for our cut formulation (1)-(12). In this section we provide several valid inequalities to obtain tighter LP bounds.
Throughout our proofs, we take into account that $x_e^k \leq 1$ for each demand $k \in K$ and edge $e \in E$, $\sum_{s=w_k}^{\bar{s}} z_s^k = 1$ for each demand k, and $z_s^k \geq 0$ 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 $\sum_{s'=s}^{min(\bar{s}, s+w_k-1)} z_{s'}^k = 1$.
In what follows, we present several valid inequalities for $P(G, K, \mathbb{S})$. Note that some proof of validity necessitates more details that may generate an overrun of the number of authorized pages. Please feel free to contact the authors for more details about each proof.
We start this section by introducing the classes of valid inequalities that can be found using Chvatal-Gomory procedures.

4.1 Edge-Slot-Assignment Inequalities

Proposition 4. *Consider an edge $e \in E$ with $K_e \neq \emptyset$. Let s be a slot in $\mathbb{S}$. Then, the inequality*

$$\sum_{k" \in K_e} \sum_{s"=s}^{\min(s+w_{k"}-1, \bar{s})} z_{s"}^{k"} \leq 1, \tag{13}$$

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

Proof. Inequality (13) ensures that the set of demands K_e cannot share the slot s over the edge e, which means that the slot s is assigned to at most one demand k from K_e over edge e.

Based on the non-overlapping inequality (8) and using the Chvatal-Gomory procedure, we define the following inequality.

Proposition 5. *Consider an edge $e \in E$. Let s be a slot in $\mathbb{S}$. Consider a triplet of demands $k, k', k" \in K$ with $e \notin E_0^k \cap E_0^{k'} \cap E_0^{k"}$. Then, the inequality*

$$x_e^k + x_e^{k'} + x_e^{k"} + \sum_{s'=s}^{\min(s+w_k-1, \bar{s})} z_{s'}^k + \sum_{s'=s}^{\min(s+w_{k'}-1, \bar{s})} z_{s'}^{k'} + \sum_{s"=s}^{\min(s+w_{k"}-1, \bar{s})} z_{s"}^{k"} \leq 4, \tag{14}$$

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

Proof. Consider an edge $e \in E$. Let s be a slot in $\mathbb{S}$. Inequality (14) ensures that if the three demands $k, k', k"$ pass through edge e, they cannot share the slot s.
Let's us show that the inequality (14) can be seen as Chvatal-Gomory cuts using Chvatal-Gomory procedure. We know from (16) that

$$x_e^k + x_e^{k'} + \sum_{s'=s}^{\min(s+w_k-1,\bar{s})} z_{s'}^k + \sum_{s'=s}^{\min(s+w_{k'}-1,\bar{s})} z_{s'}^{k'} \leq 3,$$
$$x_e^k + x_e^{k"} + \sum_{s'=s}^{\min(s+w_k-1,\bar{s})} z_{s'}^k + \sum_{s"=s}^{\min(s+w_{k"}-1,\bar{s})} z_{s"}^{k"} \leq 3,$$
$$x_e^{k'} + x_e^{k"} + \sum_{s'=s}^{\min(s+w_{k'}-1,\bar{s})} z_{s'}^{k'} + \sum_{s"=s}^{\min(s+w_{k"}-1,\bar{s})} z_{s"}^{k"} \leq 3.$$

By adding the three previous inequalities, we get the following inequality

$$2x_e^k + 2x_e^{k'} + 2x_e^{k"} + 2\sum_{s'=s}^{\min(s+w_k-1,\bar{s})} z_{s'}^k + 2\sum_{s'=s}^{\min(s+w_{k'}-1,\bar{s})} z_{s'}^{k'} + 2\sum_{s"=s}^{\min(s+w_{k"}-1,\bar{s})} z_{s"}^{k"} \leq 9$$
$$\Rightarrow x_e^k + x_e^{k'} + x_e^{k"} + \sum_{s'=s}^{\min(s+w_k-1,\bar{s})} z_{s'}^k + \sum_{s'=s}^{\min(s+w_{k'}-1,\bar{s})} z_{s'}^{k'} + \sum_{s"=s}^{\min(s+w_{k"}-1,\bar{s})} z_{s"}^{k"} \leq \left\lfloor \frac{9}{2} \right\rfloor$$
$$\Rightarrow x_e^k + x_e^{k'} + x_e^{k"} + \sum_{s'=s}^{\min(s+w_k-1,\bar{s})} z_{s'}^k + \sum_{s'=s}^{\min(s+w_{k'}-1,\bar{s})} z_{s'}^{k'} + \sum_{s"=s}^{\min(s+w_{k"}-1,\bar{s})} z_{s"}^{k"} \leq 4.$$

We conclude at the end that the inequality (14) is valid for $P(G, K, \mathbb{S})$.

The inequality (14) can then be generalized for any subset of demand $\tilde{K} \subseteq K$ under certain conditions.

Proposition 6. *Consider an edge $e \in E$, and a slot s in $\mathbb{S}$. Let $\tilde{K}$ be a subset of demands of K with $e \notin E_0^k$ for each demand $k \in \tilde{K}$, $(k, k') \notin K_c^e$ for each pair of demands (k, k') in $\tilde{K}$, and $\sum_{k\in\tilde{K}} w_k \leq \bar{s} - \sum_{k"\in K_e\setminus\tilde{K}} w_{k"}$. Then, the inequality*

$$\sum_{k\in\tilde{K}} x_e^k + \sum_{k'\in\tilde{K}} \sum_{s'=s}^{\min(s+w_{k'}-1,\bar{s})} z_{s'}^{k'} \leq |\tilde{K}| + 1, \tag{15}$$

is valid for $P(G, K, \mathbb{S})$[6].

Let $\binom{n}{k}$ denote the total number of possibilities to choose a k element in a set of n elements.

Proof. Inequality (15) ensures that if the demands $k \in \tilde{K}$ pass through edge e, they cannot share the slot s. For this, we use the Chvatal-Gomory and recurrence procedures to prove that (15) is valid for $P(G, K, \mathbb{S})$. For any subset of demands $\tilde{K} \subseteq K$ with $e \notin E_0^k$ for each demand $k \in \tilde{K}$, by recurrence procedures we get that for all demands $K' \subseteq \tilde{K}$ with $|K'| = |\tilde{K}| - 1$

$$\sum_{k\in K'} x_e^k + \sum_{k\in K'} \sum_{s'=s}^{\min(s+w_k-1,\bar{s})} z_{s'}^k \leq |K'| + 1.$$

By adding the previous inequalities for all $K' \subseteq \tilde{K}$ with $|K'| = |\tilde{K}| - 1$

$$\sum_{\substack{K'\subseteq\tilde{K}\\ |K'|=|\tilde{K}|-1}} \sum_{k\in K'} x_e^k + \sum_{\substack{K'\subseteq\tilde{K}\\ |K'|=|\tilde{K}|-1}} \sum_{k\in K'} \sum_{s'=s}^{\min(s+w_k-1,\bar{s})} z_{s'}^k \leq \sum_{\substack{K'\subseteq\tilde{K}\\ |K'|=|\tilde{K}|-1}} (|K'| + 1).$$

[6] Thanks to Prof. Hervé Kerivin for its support to have an initial idea in order to define inequalities (15) and (20).

Note that for each $k \in \tilde{K}$, the variable x_e^k and the sum $\sum_{s'=s}^{\min(s+w_k-1,\bar{s})} z_{s'}^k$ appear $(\binom{|\tilde{K}|}{|\tilde{K}|-1} - 1)$ times in the previous sum. This implies that

$$\sum_{k\in\tilde{K}}(\binom{|\tilde{K}|}{|\tilde{K}|-1} - 1)x_e^k + \sum_{k\in\tilde{K}}\sum_{s'=s}^{\min(s+w_k-1,\bar{s})}(\binom{|\tilde{K}|}{|\tilde{K}|-1} - 1)z_{s'}^k \leq \binom{|\tilde{K}|}{|\tilde{K}|-1}(|K'| + 1).$$

Given that $|K'| = |\tilde{K}| - 1$, this is equivalent to say that

$$\sum_{k\in\tilde{K}}(\binom{|\tilde{K}|}{|\tilde{K}|-1} - 1)x_e^k + \sum_{k\in\tilde{K}}\sum_{s'=s}^{\min(s+w_k-1,\bar{s})}(\binom{|\tilde{K}|}{|\tilde{K}|-1} - 1)z_{s'}^k \leq \binom{|\tilde{K}|}{|\tilde{K}|-1}|\tilde{K}|.$$

Moreover, and taking into account that $(\binom{|\tilde{K}|}{|\tilde{K}|-1} - 1) = |\tilde{K}| - 1$, we found that

$$\sum_{k\in\tilde{K}}(|\tilde{K}| - 1)x_e^k + \sum_{k\in\tilde{K}}\sum_{s'=s}^{\min(s+w_k-1,\bar{s})}(|\tilde{K}| - 1)z_{s'}^k \leq |\tilde{K}|^2.$$

By dividing the two sides of the previous sum by $|\tilde{K}| - 1$, we have

$$\sum_{k\in\tilde{K}} x_e^k + \sum_{k\in\tilde{K}}\sum_{s'=s}^{\min(s+w_k-1,\bar{s})} z_{s'}^k \leq \left\lfloor \frac{|\tilde{K}|^2}{|\tilde{K}| - 1} \right\rfloor.$$

After some simplifications, we found that

$$\sum_{k\in\tilde{K}} x_e^k + \sum_{k\in\tilde{K}}\sum_{s'=s}^{\min(s+w_k-1,\bar{s})} z_{s'}^k \leq |\tilde{K}| + \left\lfloor \frac{|\tilde{K}|}{|\tilde{K}| - 1} \right\rfloor \Rightarrow \sum_{k\in\tilde{K}} x_e^k + \sum_{k\in\tilde{K}}\sum_{s'=s}^{\min(s+w_k-1,\bar{s})} z_{s'}^k \leq |\tilde{K}| + 1,$$

given that $\left\lfloor \frac{|\tilde{K}|}{|\tilde{K}| - 1} \right\rfloor = 1$.
We conclude at the end that the inequality (15) is valid for $P(G, K, \mathbb{S})$.

The inequality (15) can be strengthened as follows. Based on the inequalities (13) and (8), we strengthen the inequality (8) without modifying its right hand side as follows.

Proposition 7. *Consider an edge $e \in E$. Let s be a slot in $\mathbb{S}$. Consider a pair of demands $k, k' \in K$ with $e \notin E_0^k \cap E_0^{k'}$ and $(k, k') \notin K_c^e$. Then, the inequality*

$$x_e^k + x_e^{k'} + \sum_{s'=s}^{\min(s+w_k-1,\bar{s})} z_{s'}^k + \sum_{s'=s}^{\min(s+w_{k'}-1,\bar{s})} z_{s'}^{k'} + \sum_{k"\in K_e\setminus\{k,k'\}}\sum_{s'=s}^{\min(s+w_{k"}-1,\bar{s})} z_{s'}^{k"} \leq 3, \qquad (16)$$

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

Proof. Consider an edge $e \in E$, and a pair of demands $k, k' \in K$. Let s be a slot in $\mathbb{S}$. Inequality (16) ensures that if the two demands k, k' pass through edge e, they cannot share the slot s with the set of demands in $K_e \setminus \{k, k'\}$.
We start our proof by assuming that the inequality (16) is not valid for $P(G, K, \mathbb{S})$. It follows that there exists a C-RSA solution S in which $s \notin S_{k"}$ for each demand $k" \in K_e \setminus \{k, k'\}$ s.t.

$$x_e^k(S) + x_e^{k'}(S) + \sum_{s'=s}^{\min(s+w_k-1,\bar{s})} z_{s'}^k(S) + \sum_{s'=s}^{\min(s+w_{k'}-1,\bar{s})} z_{s'}^{k'}(S) + \sum_{k"\in K_e\setminus\{k,k'\}}\sum_{s"=s}^{\min(s+w_{k"}-1,\bar{s})} z_{s"}^{k"}(S) > 3.$$

Since $s \notin S_{k"}$ for each demand $k" \in K_e \backslash \{k, k'\}$ this means that $\sum_{k" \in K_e \backslash \{k,k'\}} \sum_{s"=s}^{\min(s+w_{k"}-1,\bar{s})} z_{s"}^{k"}(S) = 0$, and taking into account that $x_e^k(S) \leq 1$, $x_e^{k'}(S) \leq 1$, $\sum_{s'=s}^{\min(s+w_k-1,\bar{s})} z_{s'}^k(S) \leq 1$, and $\sum_{s'=s}^{\min(s+w_{k'}-1,\bar{s})} z_{s'}^{k'}(S) \leq 1$, it follows that

$$x_e^k(S) + x_e^{k'}(S) + \sum_{s'=s}^{\min(s+w_k-1,\bar{s})} z_{s'}^k(S) + \sum_{s'=s}^{\min(s+w_{k'}-1,\bar{s})} z_{s'}^{k'}(S) \leq 3,$$

which contradicts the inequality (16) for $\tilde{K} = \{k, k'\}$, and also what we supposed, i.e., $x_e^k(S) + x_e^{k'}(S) + \sum_{s'=s}^{\min(s+w_k-1,\bar{s})} z_{s'}^k(S) + \sum_{s'=s}^{\min(s+w_{k'}-1,\bar{s})} z_{s'}^{k'}(S) > 3$.
Hence $|E_k \cap \{e\}| + |E_{k'} \cap \{e\}| + |S_k \cap \{s\}| + |S_{k'} \cap \{s\}| + \sum_{k" \in K_e} |S_{k"} \cap \{s\}| \leq 3$.

Let's us generalize the inequality (16) for each edge e and all slot $s \in \mathbb{S}$ and any subset of demand $\tilde{K} \subseteq K$ under certain conditions.

Proposition 8. *Consider an edge $e \in E$, and a slot s in $\mathbb{S}$. Let $\tilde{K}$ be a subset of demands of K with $e \notin E_0^k$ for each demand $k \in \tilde{K}$, $(k, k') \notin K_c^e$ for each pair of demands (k, k') in $\tilde{K}$, and $\sum_{k \in \tilde{K}} w_k \leq \bar{s} - \sum_{k" \in K_e \backslash \tilde{K}} w_{k"}$. Then, the inequality*

$$\sum_{k \in \tilde{K}} x_e^k + \sum_{k \in \tilde{K}} \sum_{s'=s}^{\min(s+w_k-1,\bar{s})} z_{s'}^k + \sum_{k' \in K_e \backslash \tilde{K}} \sum_{s"=s}^{\min(s+w_{k'}-1,\bar{s})} z_{s"}^{k'} \leq |\tilde{K}| + 1, \tag{17}$$

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

Proof. Inequality (17) ensures that if the demands $k \in \tilde{K}$ pass through edge e, they cannot share the slot s with the set of demands in $K_e \setminus \tilde{K}$.
We use the Chvatal-Gomory and recurrence procedures to prove that (17) is valid for $P(G, K, \mathbb{S})$. For any subset of demands $\tilde{K} \subseteq K$ with $e \notin E_0^k$ for each demand $k \in \tilde{K}$, by recurrence procedures we get that for all demands $K' \subseteq \tilde{K}$ with $|K'| = |\tilde{K}| - 1$

$$\sum_{k \in K'} x_e^k + \sum_{k \in K'} \sum_{s'=s}^{\min(s+w_k-1,\bar{s})} z_{s'}^k + \sum_{k" \in K_e \backslash K'} \sum_{s"=s}^{\min(s+w_{k"}-1,\bar{s})} z_{s"}^{k'} \leq |K'| + 1.$$

By adding the previous inequalities for all $K' \subseteq \tilde{K}$ with $|K'| = |\tilde{K}| - 1$

$$\sum_{\substack{K' \subseteq \tilde{K} \\ |K'|=|\tilde{K}|-1}} \sum_{k \in K'} x_e^k + \sum_{\substack{K' \subseteq \tilde{K} \\ |K'|=|\tilde{K}|-1}} \sum_{k \in K'} \sum_{s'=s}^{\min(s+w_k-1,\bar{s})} z_{s'}^k + \sum_{\substack{K' \subseteq \tilde{K} \\ |K'|=|\tilde{K}|-1}} \sum_{k" \in K_e \backslash \tilde{K}} \sum_{s"=s}^{\min(s+w_{k"}-1,\bar{s})} z_{s"}^{k"}$$
$$\leq \sum_{\substack{K' \subseteq \tilde{K} \\ |K'|=|\tilde{K}|-1}} (|K'| + 1).$$

Note that for each demand $k \in \tilde{K}$, the variable x_e^k and sum $\sum_{s'=s}^{\min(s+w_k-1,\bar{s})} z_{s'}^k$ appear $(\binom{|\tilde{K}|}{|\tilde{K}|-1} - 1)$ times in the previous sum. It follows that

$$\sum_{k \in \tilde{K}} (\binom{|\tilde{K}|}{|\tilde{K}| - 1} - 1) x_e^k + \sum_{k \in \tilde{K}} \sum_{s'=s}^{\min(s+w_k-1,\bar{s})} (\binom{|\tilde{K}|}{|\tilde{K}| - 1} - 1) z_{s'}^k$$
$$+ \sum_{k" \in K_e \backslash \tilde{K}} \sum_{s"=s}^{\min(s+w_{k"}-1,\bar{s})} \binom{|\tilde{K}|}{|\tilde{K}| - 1} z_{s"}^{k"} \leq \binom{|\tilde{K}|}{|\tilde{K}| - 1} (|K'| + 1).$$

Given that $|K'| + 1 = |\tilde{K}|$ and $(\binom{|\tilde{K}|}{|\tilde{K}|-1} - 1) = |\tilde{K}| - 1$, this means that

$$\sum_{k\in\tilde{K}} (|\tilde{K}| - 1)x_e^k + \sum_{k\in\tilde{K}} \sum_{s'=s}^{\min(s+w_k-1,\bar{s})} (|\tilde{K}| - 1)z_{s'}^k + \sum_{k"\in K_e\setminus\tilde{K}} \sum_{s"=s}^{\min(s+w_{k"}-1,\bar{s})} |\tilde{K}|z_{s"}^{k"} \leq |\tilde{K}|^2.$$

By dividing the two sides of the previous sum by $|\tilde{K}| - 1$, we found that

$$\sum_{k\in\tilde{K}} x_e^k + \sum_{k\in\tilde{K}} \sum_{s'=s}^{\min(s+w_k-1,\bar{s})} z_{s'}^k + \sum_{k"\in K_e\setminus\tilde{K}} \sum_{s"=s}^{\min(s+w_{k"}-1,\bar{s})} \left\lfloor \frac{|\tilde{K}|}{|\tilde{K}|-1} \right\rfloor z_{s"}^{k"} \leq \left\lfloor \frac{|\tilde{K}|^2}{|\tilde{K}|-1} \right\rfloor.$$

After some simplifications, we found that

$$\sum_{k\in\tilde{K}} x_e^k + \sum_{k\in\tilde{K}\cup(K_e\setminus\tilde{K})} \sum_{s'=s}^{\min(s+w_k-1,\bar{s})} z_{s'}^k \leq |\tilde{K}| + \left\lfloor \frac{|\tilde{K}|}{|\tilde{K}|-1} \right\rfloor \Rightarrow \sum_{k\in\tilde{K}} x_e^k + \sum_{k\in\tilde{K}\cup(K_e\setminus\tilde{K})} \sum_{s'=s}^{\min(s+w_k-1,\bar{s})} z_{s'}^k \leq |\tilde{K}|+1,$$

given that $\left\lfloor \frac{|\tilde{K}|}{|\tilde{K}|-1} \right\rfloor = 1$. We conclude at the end that the inequality (17) is valid for $P(G, K, \mathbb{S})$.

4.2 Edge-Interval-Cover Inequalities

Let's now 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 9. *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 $K' \subseteq K_e$ be a minimal cover for interval $I = [s_i, s_j]$ over edge e. Then, the inequality*

$$\sum_{k\in K'} \sum_{s=s_i+w_k-1}^{s_j} z_s^k \leq |K'| - 1, \tag{18}$$

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

Proof. The interval $I = [s_i, s_j]$ can cover at most $|K'| - 1$ demands given that K' is a minimal cover for interval $I = [s_i, s_j]$ over edge e. We start our proof by assuming that the inequality (18) is not valid for $P(G, K, \mathbb{S})$. 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.

$$\sum_{k\in K'} \sum_{s=s_i+w_k-1}^{s_j} z_s^k(S) > |K'| - 1.$$

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

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

which contradicts what we supposed before, i.e., $\sum_{k \in K'} \sum_{s=s_i+w_k-1}^{s_j} z_s^k(S) > |K'| - 1$.
Hence $\sum_{k \in K'} |S_k \cap \{s_i + w_k - 1, ..., s_j\}| \leq |K'| - 1$.
We conclude at the end that the inequality (18) is valid for $P(G, K, \mathbb{S})$.

The inequality (18) can be strengthened using an extention of each minimal cover $K' \subset K_e$ for an interval I over edge e as follows.

Proposition 10. *Consider an edge $e \in E$. Let $I = [s_i, s_j]$ be an interval of contiguous slots in $[1, \bar{s}]$. Let $K' \subseteq K_e$ be a minimal cover for interval $I = [s_i, s_j]$ over edge e with $e \notin E_0^k$ for each demand $k \in K'$, and $\Xi(K')$ be a subset of demands in $K_e \setminus K'$ s.t. $\Xi(K') = \{k \in K_e \setminus K'$ s.t. $w_k \geq w_{k'} \quad \forall k' \in K'\}$. Then, the inequality*

$$\sum_{k \in K'} \sum_{s=s_i+w_k-1}^{s_j} z_s^k + \sum_{k' \in \Xi(K')} \sum_{s'=s_i+w_{k'}-1}^{s_j} z_{s'}^{k'} \leq |K'| - 1, \tag{19}$$

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

Proof. The interval $I = [s_i, s_j]$ can cover at most $|K'| - 1$ demands from the demands in $K' \cup \Xi(K')$ given that K' is a minimal cover for interval $I = [s_i, s_j]$ over edge e and the definition of the set $\Xi(K')$ s.t. for each pair (k, k') with $k \in K'$ and $k' \in \Xi(K')$, the set $(K' \setminus \{k\}) \cup \{k'\}$ stills defining minimal cover for the interval I over the edge e. Furthermore, for each quadruplet $(k, k', \tilde{k}, \tilde{k}')$ with $k, k' \in K'$ and $\tilde{k}, \tilde{k}' \in \Xi(K')$, the set $(K' \setminus \{k, k'\}) \cup \{\tilde{k}, \tilde{k}'\}$ stills defining minimal cover for the interval I over the edge e given that $w_k + w_{k'} \leq w_{\tilde{k}} + w_{\tilde{k}'}$.
We strengthen our proof as follows. Let's first suppose that the inequality (19) is not valid for $P(G, K, \mathbb{S})$. It follows that there exists a C-RSA solution S in which $\{s_i + w_{k'} - 1, ..., s_j\} \cap S_{k'} = \emptyset$ for each demand $k' \in \Xi(K')$ s.t.

$$\sum_{k \in K'} \sum_{s=s_i+w_k-1}^{s_j} z_s^k(S) > |K'| - 1.$$

Since $\{s_i + w_{k'} - 1, ..., s_j\} \cap S_{k'} = \emptyset$ for each demand $k' \in \Xi(K')$ this means that $\sum_{s=s_i+w_{k'}-1}^{s_j} z_s^{k'}(S) = 0$, and taking into account the inequality (18), and that K' is minimal cover for the interval $I = [s_i, s_j]$ over edge e, and $\sum_{s=s_i+w_k-1}^{s_j} z_s^k(S) \leq 1$ for each demand $k \in K'$, it follows that

$$\sum_{k \in K'} \sum_{s=s_i+w_k-1}^{s_j} z_s^k(S) \leq |K'| - 1,$$

which contradicts what we supposed before, i.e., $\sum_{k \in K'} \sum_{s=s_i+w_k-1}^{s_j} z_s^k(S) > |K'| - 1$.
Hence $\sum_{k \in K'} |S_k \cap \{s_i + w_k - 1, ..., s_j\}| + \sum_{k' \in \Xi(K')} |S_{k'} \cap \{s_i + w_{k'} - 1, ..., s_j\}| \leq |K'| - 1$.
We conclude at the end that the inequality (19) is valid for $P(G, K, \mathbb{S})$.

Moreover, the inequality (18) can be strengthened using lifting procedures proposed by Nemhauser and Wolsey in [50] without modifying its right-hand side.

Proposition 11. *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 subset of demands of K s.t.*

– $\sum_{k \in \tilde{K}} w_k \geq |I| + 1$,

– $\sum_{k\in\tilde{K}\setminus\{k'\}} w_k \leq |I|$ *for each* $k' \in \tilde{K}$,
– $\sum_{k\in\tilde{K}} w_k \leq \bar{s} - \sum_{k'\in K_e\setminus\tilde{K}} w_{k'}$,
– $e \notin E_0^k$ *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

$$\sum_{k\in\tilde{K}} x_e^k + \sum_{k\in\tilde{K}} \sum_{s=s_i+w_k-1}^{s_j} z_s^k \leq 2|\tilde{K}| - 1, \tag{20}$$

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

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 (i.e., $\sum_{k\in\tilde{K}} x_e^k = |\tilde{K}|$), 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 (20) is not valid for $P(G,K,\mathbb{S})$. 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.

$$\sum_{k\in K'} x_e^k(S) + \sum_{k'\in K'\setminus\{k\}} \sum_{s=s_i+w_{k'}-1}^{s_j} z_s^{k'}(S) \geq 2|K'|.$$

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

$$\sum_{k\in K'} x_e^k(S) + \sum_{k'\in K'\setminus\{k\}} \sum_{s=s_i+w_{k'}-1}^{s_j} z_s^{k'}(S) \leq 2|K'|-1,$$

which contradicts what we supposed before, i.e., $\sum_{k\in K'} x_e^k(S)+\sum_{k'\in K'\setminus\{k\}}\sum_{s=s_i+w_{k'}-1}^{s_j} z_s^{k'}(S)\geq 2|K'|$.
One can imagine another case also when $K'\cap K_e=\emptyset$, it follows that there exists a C-RSA solution S' in which $E_k\cap\{e\}=\emptyset$ for each demand $k\in K'$, which means that $\sum_{k\in K'} x_e^k(S')=0$ s.t.

$$\sum_{k\in K'} \sum_{s=s_i+w_k-1}^{s_j} z_s^k(S') \geq 2|K'|.$$

Given that $\sum_{s=s_i+w_k-1}^{s_j} z_s^k(S')\leq 1$ for each demand $k\in K'$, it follows that

$$\sum_{k'\in K'\setminus\{k\}} \sum_{s=s_i+w_{k'}-1}^{s_j} z_s^{k'}(S') \leq 2|K'|-1,$$

which contradicts our hypothesis, i.e., $\sum_{k\in K'}\sum_{s=s_i+w_k-1}^{s_j} z_s^k(S')\geq 2|K'|$.
Hence $\sum_{k\in K'} |E_k\cap\{e\}| + \sum_{k\in K'} |S_k\cap\{s_i+w_k-1,...,s_j\}| \leq 2|K'|-1$.
We conclude at the end that the inequality (20) is valid for $P(G,K,\mathbb{S})$.

As we did before for the inequality (18), the inequality (20) can be strengthened by introducing the extended version of the minimal cover K' for the interval I over edge e as follows.

Proposition 12. *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 subset of demands of* K, *and* $\tilde{K}_e$ *be a subset of demands in* $K_e\setminus\tilde{K}$ *s.t.*

- $\sum_{k\in\tilde{K}} w_k \geq |I|+1$,
- $\sum_{k\in\tilde{K}\setminus\{k'\}} w_k \leq |I|$ *for each* $k'\in\tilde{K}$,
- $\sum_{k\in\tilde{K}} w_k \leq \bar{s} - \sum_{k'\in K_e\setminus\tilde{K}} w_{k'}$,
- $e\notin E_0^k$ *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

$$\sum_{k\in\tilde{K}} x_e^k + \sum_{k\in\tilde{K}}\sum_{s=s_i+w_k-1}^{s_j} z_s^k + \sum_{k'\in\tilde{K}_e}\sum_{s'=s_i+w_{k'}-1}^{s_j} z_{s'}^{k'} \leq 2|\tilde{K}|-1, \tag{21}$$

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

Proof. The inequality (21) can be seen as a particular case for the inequality (20) induced by a set of demands $K'=\tilde{K}\cup\tilde{K}_e$ which stills defining a cover for the interval I over edge e.

More general, the inequality (20) can be strengthened using lifting procedures proposed by Nemhauser and Wolsey in [50] without modifying its right-hand side.

Remark 1. Consider an edge $e\in E$. Let $I=[s_i,s_j]$ be an interval of contiguous slots with $s_i+1\leq s_j$, s" be a slot in $\mathbb{S}$, and $\tilde{K}$ be a subset of demands in K satisfying the conditions of the two inequalities (17) and (20). We ensure that the inequality (17) can never dominate the inequality (20).

Let us denote by the symbole $a\preceq b$ iff b dominates a.

Proof. Assume that the inequality (17) dominates the inequality (20), this means that there exists a slot $s" \in\mathbb{S}$ s.t.

$$\sum_{k\in\tilde{K}} x_e^k + \sum_{k\in\tilde{K}}\sum_{s=s_i+w_k-1}^{s_j} z_s^k \preceq \sum_{k\in\tilde{K}} x_e^k + \sum_{k\in\tilde{K}}\sum_{s'=s"}^{\min(s"+w_k-1,\bar{s})} z_{s'}^k \leq |\tilde{K}|+1.$$

By removing the sum $\sum_{k\in\tilde{K}} x_e^k$ from the two sides of the previous comparison, we get

$$\sum_{k\in\tilde{K}}\sum_{s=s_i+w_k-1}^{s_j} z_s^k \preceq \sum_{k\in\tilde{K}}\sum_{s'=s"}^{\min(s"+w_k-1,\bar{s})} z_{s'}^k.$$

Given that the demands in $\tilde{K}$ are independants, we found that

$$\sum_{s=s_i+w_k-1}^{s_j} z_s^k \preceq \sum_{s'=s"}^{\min(s"+w_k-1,\bar{s})} z_{s'}^k \text{ for each } k\in\tilde{K}.$$

It follows that $I_k=[s_i+w_k-1,s_j]\subseteq[s", min(s"+w_k-1,\bar{s})]$ for each demand $k\in\tilde{K}$. Taking into account that $|\{s",...,min(s"+w_k-1,\bar{s})\}|\leq w_k$ for each $k\in\tilde{K}$, this means that

$$\begin{aligned}|I_k| = s_j-(s_i+w_k-1)+1\leq w_k \implies s_j-s_i+1\leq 2*w_k-1 \text{ for each } k\in\tilde{K}\\ \implies |I|\leq 2*w_k-1 \text{ for each } k\in\tilde{K} \implies |I|\leq 2*\min_{k\in\tilde{K}} w_k-1\end{aligned}$$

As a result, $w_k+w_{k'}\geq|I|$ for each pair of demand (k,k') in $\tilde{K}$ since that $w_k\geq\min_{k"\in\tilde{K}} w_{k"}$ for each $k\in\tilde{K}$. This contradicts that the set of demand $\tilde{K}$ should satisfy that $\sum_{k\in\tilde{K}\setminus\{k'\}} w_k\leq|I|$ for each $k'\in\tilde{K}$. We conclude that the inequality (17) can never dominate the inequality (20) and satisfying the conditions of validity of the inequality (20) at the same time.

4.3 Edge-Interval-Clique Inequalities

In what follows, we need to introduce some notions of graph theory to provide some valid inequalities for $P(G, K, \mathbb{S})$.

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 *$\tilde{G}_I^e$ defined as follows. For each demand $k \in K$ with $w_k \leq |I|$ and $e \notin E_0^k$, consider a node v_k in $\tilde{G}_I^e$. Two nodes v_k and $v_{k'}$ are linked by an edge in $\tilde{G}_I^e$ if $w_k + w_{k'} > |I|$ and $(k, k') \notin K_c^e$. 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 $\tilde{G}_I^e$ 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 $e \notin E_0^k \cup E_1^k$, 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 $\tilde{G}_I^e$.

Proposition 13. *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 *$\tilde{G}_I^e$ 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*

$$\sum_{v_k \in C} x_e^k + \sum_{s = s_i + w_k - 1}^{s_j} z_s^k \leq |C| + 1, \tag{22}$$

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

Proof. For each edge $e \in E$ and interval of contiguous slots $I \subseteq \mathbb{S}$, the inequality (22) ensures that if the set of demands in the clique C pass through edge e, they cannot share the interval $I = [s_i, s_j]$ over edge e. This means that there is at most one demand from the demands in C that can be totally covered by the interval I over the edge e (i.e., all the slots assigned to the demand are in I). We start our proof by assuming that the inequality (22) is not valid for $P(G, K, \mathbb{S})$. It follows that there exists a C-RSA solution S in which $\{s_i + w_k - 1, ..., s_j\} \cap S_k = \emptyset$ for each demand $v_k \in C$ s.t.

$$\sum_{v_k \in C} x_e^k(S) + \sum_{v_k \in C} \sum_{s = s_i + w_k - 1}^{s_j} z_s^k(S) > |C| + 1.$$

Since $\{s_i + w_k - 1, ..., s_j\} \cap S_k = \emptyset$ for each demand $v_k \in C$ this means that $\sum_{s = s_i + w_k - 1}^{s_j} z_s^k(S) = 0$, and taking into account that $x_e^k(S) \leq 1$ for each $v_k \in C$, it follows that

$$\sum_{v_k \in C} x_e^k(S) \leq |C| + 1,$$

which contradicts our hypothesis, i.e., $\sum_{v_k \in C} x_e^k(S) + \sum_{v_k \in C} \sum_{s = s_i + w_k - 1}^{s_j} z_s^k(S) > |C| + 1$.
On another hand, one can imagine another case also when $\{k \in K \text{ s.t. } v_k \in C\} \cap K_e = \emptyset$, it follows that there exists a C-RSA solution S' in which $E_k \cap \{e\} = \emptyset$ for each demand $v_k \in C$, which means that $\sum_{v_k \in C} x_e^k(S') = 0$ s.t.

$$\sum_{v_k \in C} \sum_{s = s_i + w_k - 1}^{s_j} z_s^k(S') > |C| + 1.$$

Given that $\sum_{s = s_i + w_k - 1}^{s_j} z_s^k(S') \leq 1$ for each demand $v_k \in C$, it follows that

$$\sum_{k' \in C \setminus \{k\}} \sum_{s = s_i + w_{k'} - 1}^{s_j} z_s^{k'}(S') \leq |C| + 1,$$

which contradicts what we supposed before, i.e., $\sum_{v_k \in C} \sum_{s = s_i + w_k - 1}^{s_j} z_s^k(S') > |C| + 1$.
Hence $\sum_{v_k \in C} |E_k \cap \{e\}| + \sum_{v_k \in C} |S_k \cap \{s_i + w_k - 1, ..., s_j\}| \leq |C| + 1$.
Furthermore, the inequality (22) can be shown as Chvatal-Gomory cuts using Chvatal-Gomory

and recurrence procedures. For any subset of demands $C \subseteq K$ with $w_k + w_{k'} > |I|$ for each pair of demands $k, k' \in C$, and $e \notin E_0^k$, $w_k \leq |I|$ for each demand $v_k \in C$, and $\sum_{v_k \in C} w_k \leq \bar{s} - \sum_{v_{k'} \in K_e \setminus C} w_{k'}$, by recurrence procedure we get that for all $K' \subseteq C$ with $|K'| = |C| - 1$

$$\sum_{v_k \in C'} x_e^k + \sum_{v_k \in C'} \sum_{s=s_i+w_k-1}^{s_j} z_s^k \leq |K'| + 1.$$

By adding the previous inequalities for all $K' \subseteq C$ with $|K'| = |C| - 1$, we get

$$\sum_{\substack{K' \subseteq C \\ |K'|=|C|-1}} \sum_{v_k \in C'} x_e^k + \sum_{\substack{K' \subseteq C \\ |K'|=|C|-1}} \sum_{v_k \in C'} \sum_{s=s_i+w_k-1}^{s_j} z_s^k \leq \sum_{\substack{K' \subseteq C \\ |K'|=|C|-1}} (|K'| + 1)$$

Note that for each demand k with $v_k \in C$, the variable x_e^k and the sum $\sum_{s=s_i+w_k-1}^{s_j} z_s^k$ appear $(\binom{|C|}{|C|-1} - 1)$ times in the previous sum. It follows that

$$\sum_{v_k \in C} (\binom{|C|}{|C|-1} - 1) x_e^k + \sum_{v_k \in C} \sum_{s=s_i+w_k-1}^{s_j} (\binom{|C|}{|C|-1} - 1) z_s^k \leq \binom{|C|}{|C|-1} |C|.$$

Given that $(\binom{|C|}{|C|-1} - 1) = |C| - 1$, we found that

$$\sum_{v_k \in C} (|C| - 1) x_e^k + \sum_{v_k \in C} \sum_{s=s_i+w_k-1}^{s_j} (|C| - 1) z_s^k \leq |C|^2.$$

By dividing the two sides of the previous sum by $|C| - 1$, we have

$$\sum_{v_k \in C} x_e^k + \sum_{v_k \in C} \sum_{s=s_i+w_k-1}^{s_j} z_s^k \leq \left\lfloor \frac{|C|^2}{|C|-1} \right\rfloor \Rightarrow \sum_{v_k \in C} x_e^k + \sum_{v_k \in C} \sum_{s=s_i+w_k-1}^{s_j} z_s^k \leq \left\lfloor |C| \frac{|C|}{|C|-1} \right\rfloor$$
$$\Rightarrow \sum_{v_k \in C} x_e^k + \sum_{v_k \in C} \sum_{s=s_i+w_k-1}^{s_j} z_s^k \leq \left\lfloor |C| \frac{|C|-1+1}{|C|-1} \right\rfloor$$

By doing the following simplification

$$\sum_{v_k \in C} x_e^k + \sum_{v_k \in C} \sum_{s=s_i+w_k-1}^{s_j} z_s^k \leq \left\lfloor |C| \frac{|C|-1}{|C|-1} + \frac{|C|}{|C|-1} \right\rfloor \Rightarrow \sum_{v_k \in C} x_e^k + \sum_{v_k \in C} \sum_{s=s_i+w_k-1}^{s_j} z_s^k \leq \left\lfloor |C| + \frac{|C|}{|C|-1} \right\rfloor,$$

we found that

$$\sum_{v_k \in C} x_e^k + \sum_{v_k \in C} \sum_{s=s_i+w_k-1}^{s_j} z_s^k \leq |C| + \left\lfloor \frac{|C|}{|C|-1} \right\rfloor \Rightarrow \sum_{v_k \in C} x_e^k + \sum_{v_k \in C} \sum_{s=s_i+w_k-1}^{s_j} z_s^k \leq |C| + 1$$
$$\text{given that } \left\lfloor \frac{|C|}{|C|-1} \right\rfloor = 1.$$

We conclude at the end that the inequality (22) is valid for $P(G, K, \mathbb{S})$.

Remark 2. Consider an edge e and an interval of contiguous slots $I = [s_i, s_j]$. Let $\tilde{K}$ be a subset of demands in K satisfying the conditions of validity of the inequalities (17) and (22). Then, the inequality (22) is dominated by the inequality (17) associated with slot $s" = s_i + \min_{k \in \tilde{K}} w_k + 1$ if and only if $|\{s_i + w_k, ., s_j\}| \leq w_k$ for each demand $k \in \tilde{K}$.

Proof. We know from inequalities (17) and (22) that

$$\sum_{k\in\tilde{K}} x_e^k + \sum_{k\in\tilde{K}} \sum_{s'=s"}^{\min(s"+w_k-1,\bar{s})} z_{s'}^k \leq |\tilde{K}|+1 \text{ and } \sum_{k\in\tilde{K}} x_e^k + \sum_{k\in\tilde{K}} \sum_{s=s_i+w_k-1}^{s_j} z_s^k \leq |\tilde{K}|+1.$$

Sufficiency.
First, assume that the inequality (17) dominates the inequality (22), this means that there exists a slot $s" \in \mathbb{S}$ s.t.

$$\sum_{k\in\tilde{K}} x_e^k + \sum_{k\in\tilde{K}} \sum_{s=s_i+w_k-1}^{s_j} z_s^k \preceq \sum_{k\in\tilde{K}} x_e^k + \sum_{k\in\tilde{K}} \sum_{s'=s"}^{\min(s"+w_k-1,\bar{s})} z_{s'}^k \leq |\tilde{K}|+1.$$

By removing the sum $\sum_{k\in\tilde{K}} x_e^k$ from the two sides of the previous comparison

$$\sum_{k\in\tilde{K}} \sum_{s=s_i+w_k-1}^{s_j} z_s^k \preceq \sum_{k\in\tilde{K}} \sum_{s'=s"}^{\min(s"+w_k-1,\bar{s})} z_{s'}^k.$$

Given that the demands $\tilde{K}$ are independants, we found that

$$\sum_{s=s_i+w_k-1}^{s_j} z_s^k \preceq \sum_{s'=s"}^{\min(s"+w_k-1,\bar{s})} z_{s'}^k \text{ for each } k \in \tilde{K}.$$

It follows that $I_k = [s_i + w_k - 1, s_j] \subseteq [s", min(s" + w_k - 1, \bar{s})]$ for each demand $k \in \tilde{K}$. Taking into account that $|\{s", ..., min(s" + w_k - 1, \bar{s})\}| \leq w_k$ for each $k \in \tilde{K}$, this means that

$$|I_k| = s_j - (s_i + w_k - 1) + 1 \leq w_k \text{ for each } k \in \tilde{K},$$

that which was to be demonstrated.
Neccessity.
Assume that $|I_k| \leq w_k$ for each demand $k \in \tilde{K}$. Given that $I_k = [s_i + w_k - 1, s_j]$ and $s_i + w_k - 1 \geq s_i + \min_{k'\in\tilde{K}} w_{k'} - 1$ for each demand $k \in \tilde{K}$, this means that $[s_i + w_k - 1, s_j] \subseteq [s_i + \min_{k'\in\tilde{K}} w_{k'} - 1, s_j]$ for each demand $k \in \tilde{K}$.
Let $\tilde{k}$ be a demand in $argmin\{k \in \tilde{K}, w_k = \min_{k'\in\tilde{K}} w_{k'}\}$. We know that $|I_{\tilde{k}}| \leq w_{\tilde{k}}$, i.e., $|\{s_i + \min_{k'\in\tilde{K}} w_{k'} - 1, s_j\}| = s_j - (s_i + \min_{k'\in\tilde{K}} w_{k'} - 1) + 1 \leq w_k$ for each demand $k \in \tilde{K}$. This implies that $(s_i + \min_{k'\in\tilde{K}} w_{k'} - 1) + w_k - 1 \geq s_j$ for each demand $k \in \tilde{K}$. It follows that $[s_i + \min_{k'\in\tilde{K}} w_{k'} - 1, s_j] \subseteq [s_i + \min_{k'\in\tilde{K}} w_{k'} - 1, s_i + \min_{k'\in\tilde{K}} w_{k'} + w_k - 2]$ for each demand $k \in \tilde{K}$. As a result, we obtain that for each demand $k \in \tilde{K}$

$$\begin{aligned} I_k = [s_i + w_k - 1, s_j] &\subseteq [s_i + \min_{k'\in\tilde{K}} w_{k'} - 1, s_j] \\ \text{and } [s_i + \min_{k'\in\tilde{K}} w_{k'} - 1, s_j] &\subseteq [s_i + \min_{k'\in\tilde{K}} w_{k'} - 1, s_i + \min_{k'\in\tilde{K}} w_{k'} + w_k - 2] \\ \Longrightarrow I_k = [s_i + w_k - 1, s_j] &\subseteq [s_i + \min_{k'\in\tilde{K}} w_{k'} - 1, s_i + \min_{k'\in\tilde{K}} w_{k'} + w_k - 2]. \end{aligned}$$

By giving $s" = s_i + \min_{k'\in\tilde{K}} w_{k'} - 1$, it is equivalent to say that

$$I_k = [s_i + w_k - 1, s_j] \subseteq [s", s" + w_k - 1] \text{ for each } k \in \tilde{K}$$

We know from (17) that

$$\sum_{k\in\tilde{K}} x_e^k + \sum_{k\in\tilde{K}} \sum_{s'=s"}^{\min(s"+w_k-1,\bar{s})} z_{s'}^k \leq |\tilde{K}|+1.$$

Taking into account that $[s", s" + w_k - 1] = [s", s_i + w_k - 2] \cup [s_i + w_k - 1, s_j] \cup [s_j + 1, s" + w_k - 1]$ for each $k \in \tilde{K}$, it follows that

$$\sum_{k\in\tilde{K}} x_e^k + \sum_{k\in\tilde{K}} \sum_{s'=s"}^{\min(s"+w_k-1,\bar{s})} z_{s'}^k = \sum_{k\in\tilde{K}} x_e^k + \sum_{k\in\tilde{K}} [\sum_{s'=s"}^{s_i+w_k-2} z_{s'}^k + \sum_{s'=s_i+w_k-1}^{s_j} z_{s'}^k + \sum_{s'=s_j+1}^{\min(s"+w_k-1,\bar{s})} z_{s'}^k] \leq |\tilde{K}| + 1$$

$$\implies \sum_{k\in\tilde{K}} x_e^k + \sum_{k\in\tilde{K}} \sum_{s'=s"}^{\min(s"+w_k-1,\bar{s})} z_{s'}^k = \sum_{k\in\tilde{K}} x_e^k + \sum_{k\in\tilde{K}} \sum_{s'=s"}^{s_i+w_k-2} z_{s'}^k + \sum_{s'\in I_k} z_{s'}^k + \sum_{s'=s_j+1}^{\min(s"+w_k-1,\bar{s})} z_{s'}^k \leq |\tilde{K}| + 1$$

$$\implies \sum_{k\in\tilde{K}} x_e^k + \sum_{k\in\tilde{K}} \sum_{s'=s"}^{\min(s"+w_k-1,\bar{s})} z_{s'}^k = \sum_{k\in\tilde{K}} x_e^k + \sum_{k\in\tilde{K}} \sum_{s'\in I_k} z_{s'}^k + \sum_{s'=s"}^{s_i+w_k-2} z_{s'}^k + \sum_{s'=s_j+1}^{\min(s"+w_k-1,\bar{s})} z_{s'}^k \leq |\tilde{K}| + 1,$$

which shows that the inequality (17) dominates the inequality (22)

$$\sum_{k\in\tilde{K}} x_e^k + \sum_{k\in\tilde{K}} \sum_{s\in I_k} z_s^k \preceq \sum_{k\in\tilde{K}} x_e^k + \sum_{k\in\tilde{K}} \sum_{s'=s"}^{\min(s"+w_k-1,\bar{s})} z_{s'}^k \leq |\tilde{K}| + 1.$$

Remark 3. Consider an edge e and an interval of contiguous slots $I = [s_i, s_j]$. Let $\tilde{K}$ be a subset of demands in K satisfying the conditions of validity of the inequalities (17) and (22). Then, the inequality (22) dominates the inequality (17) associated with each slot $s" \in I$ if and only if $|\{s_i + w_k - 1, ..., s_j\}| \geq w_k$ for each demand $k \in \tilde{K}$ and $s" \in \{s_i + \max_{k'\in\tilde{K}} w_k - 1, ..., s_j - \max_{k\in\tilde{K}} w_k + 1\}$.

Proof. We know from inequalities (17) and (22) that

$$\sum_{k\in\tilde{K}} x_e^k + \sum_{k\in\tilde{K}} \sum_{s'=s}^{\min(s+w_k-1,\bar{s})} z_{s'}^k \leq |\tilde{K}| + 1 \text{ and } \sum_{k\in\tilde{K}} x_e^k + \sum_{k\in\tilde{K}} \sum_{s\in I_k} z_s^k \leq |\tilde{K}| + 1.$$

Neccessity.
First, assume that $|I_k| \geq w_k$ and $s" \in \{s_i + \max_{k'\in\tilde{K}} w_k - 1, ..., s_j - \max_{k\in\tilde{K}} w_k + 1\}$ for each demand $k \in \tilde{K}$, this means that

$$s" \geq s_i + w_k - 1 \text{ and } s" \leq s_j - w_k + 1 \text{ for each } k \in \tilde{K}$$

$$\implies s" \geq s_i + w_k - 1 \text{ and } s" + w_k - 1 \leq s_j \text{ for each } k \in \tilde{K}$$

$$\implies [s", s + w_k - 1] \subseteq [s_i + w_k - 1, s_j] \text{ for each } k \in \tilde{K}$$

$$\implies [s", s + w_k - 1] \subseteq I_k \text{ with } |I_k| \geq w_k \text{ for each } k \in \tilde{K}.$$

This means that I_k can be written as unions of sub-intervals, i.e., $I_k = [s_i + w_k - 1, s" - 1] \cup [s", s" + w_k - 1] \cup [s" + w_k - 1, s_j]$. As a result,

$$\sum_{s\in I_k} z_s^k = \sum_{s=s_i+w_k-1}^{s"-1} z_{s'}^k + \sum_{s'=s"}^{s"+w_k-1} z_{s'}^k + \sum_{s'=s"+w_k}^{s_j} z_{s'}^k \text{ for each } k \in \tilde{K}.$$

By doing a sum over all the demands in $\tilde{K}$, it follows that

$$\sum_{k\in\tilde{K}} \sum_{s\in I_k} z_s^k = \sum_{k\in\tilde{K}} \sum_{s=s_i+w_k-1}^{s"-1} z_{s'}^k + \sum_{s'=s"}^{s"+w_k-1} z_{s'}^k + \sum_{s'=s"+w_k}^{s_j} z_{s'}^k.$$

As a result,

$$\sum_{k\in\tilde{K}} x_e^k + \sum_{k\in\tilde{K}} \sum_{s\in I_k} z_s^k = \sum_{k\in\tilde{K}} x_e^k + \sum_{k\in\tilde{K}} \sum_{s=s_i+w_k-1}^{s"-1} z_{s'}^k + \sum_{s'=s"}^{s"+w_k-1} z_{s'}^k + \sum_{s'=s"+w_k}^{s_j} z_{s'}^k \leq |\tilde{K}| + 1$$

$$\implies \sum_{k\in\tilde{K}} x_e^k + \sum_{k\in\tilde{K}} \sum_{s'=s"}^{s"+w_k-1} z_{s'}^k \leq \sum_{k\in\tilde{K}} x_e^k + \sum_{k\in\tilde{K}} \sum_{s\in I_k} z_s^k \leq |\tilde{K}| + 1.$$

As a result, the inequality (22) dominates the inequality (17).
Sufficiency.
We assume that the inequality (22) dominates the inequality (17)

$$\sum_{k\in\tilde{K}} x_e^k + \sum_{k\in\tilde{K}} \sum_{s'=s"}^{s"+w_k-1} z_{s'}^k \preceq \sum_{k\in\tilde{K}} x_e^k + \sum_{k\in\tilde{K}} \sum_{s\in I_k} z_s^k.$$

By removing the sum $\sum_{k\in\tilde{K}} x_e^k$ from two sides of the previous comparison, we found

$$\sum_{k\in\tilde{K}} \sum_{s'=s"}^{s"+w_k-1} z_{s'}^k \preceq \sum_{k\in\tilde{K}} \sum_{s\in I_k} z_s^k.$$

Taking into account that the demands in $\tilde{K}$ are indepedants, it follows that

$$\sum_{s'=s"}^{s"+w_k-1} z_{s'}^k \preceq \sum_{s\in I_k} z_s^k \text{ for each demand } k \in \tilde{K}.$$

Hence, $[s", s" + w_k - 1] \subseteq I_k$ for each $k \in \tilde{K}$. This means that

$$\begin{aligned} |I_k| \geq w_k \text{ and } s" \geq s_i + w_k - 1 \text{ and } s" + w_k - 1 \leq s_j \text{ for each } k \in \tilde{K} \\ \implies s" \geq s_i + \max_{k\in\tilde{K}} w_k - 1 \text{ and } s" \leq s_j - \max_{k\in\tilde{K}} w_k + 1 \\ \implies s" \in \{s_i + \max_{k\in\tilde{K}} w_k - 1, ..., s_j - \max_{k\in\tilde{K}} w_k + 1\} \end{aligned}$$

As a result, $|I_k| \geq w_k$ for each demand $k \in \tilde{K}$, and $s" \in \{s_i + \max_{k\in\tilde{K}} w_k - 1, ..., s_j - \max_{k\in\tilde{K}} w_k + 1\}$ that which was to be demonstrated, and which ends our proof.

Moreover, the inequality (22) can be strengthened as follows.

Proposition 14. *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 *$\tilde{G}_I^e$ with $|C| \geq 3$, and $\sum_{v_k\in C} w_k \leq \bar{s} - \sum_{k'\in K_e\setminus C} w_{k'}$. Let $C_e \subseteq K_e \setminus C$ be a clique in the* conflict graph *$\tilde{G}_I^e$ s.t. $w_k + w_{k'} \geq |I| + 1$ for each $v_k \in C$ and $v_{k'} \in C_e$. Then, the inequality*

$$\sum_{v_k\in C} x_e^k + \sum_{v_k\in C} \sum_{s=s_i+w_k-1}^{s_j} z_s^k + \sum_{v_{k'}\in C_e} \sum_{s'=s_i+w_{k'}-1}^{s_j} z_{s'}^{k'} \leq |C| + 1, \quad (23)$$

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

Proof. For each edge $e \in E$ and interval of contiguous slots $I \subseteq \mathbb{S}$, the inequality (23) ensures that if the set of demands in the clique C pass through edge e, they cannot share the interval $I = [s_i, s_j]$ over edge e with a subset of demands in C_e. We first suppose that the inequality (23) is not valid for $P(G, K, \mathbb{S})$. It follows that there exists a C-RSA solution S in which $S_{k'} \cap \{s_i + w_{k'} - 1, ..., s_j\} = \emptyset$ for each demand $k' \in C_e$ s.t.

$$\sum_{v_k\in C} x_e^k(S) + \sum_{v_k\in C} \sum_{s=s_i+w_k-1}^{s_j} z_s^k(S) > |C| + 1.$$

Since $S_{k'} \notin I$ for each demand $k' \in C_e$ this means that $\sum_{v_{k'}\in C_e} \sum_{s'=s_i+w_{k'}-1}^{s_j} z_{s'}^{k'}(S) = 0$, and taking into account inequality (22) and that $x_e^k(S) \leq 1$ for each demand $v_k \in C$ and $\sum_{s=s_i+w_k-1}^{s_j} z_s^k(S)(S) \leq 1$ for each demand $v_k \in C$, it follows that

$$\sum_{v_k\in C} x_e^k(S) + \sum_{v_k\in C} \sum_{s=s_i+w_k-1}^{s_j} z_s^k(S) \leq |C| + 1,$$

which contradicts what we supposed before.
On another hand and when $C \cap K_e = \emptyset$, it follows that there exists a C-RSA solution S' in which $E_k \cap \{e\} = \emptyset$ and $S_{k'} \cap \{s_i + w_{k'} - 1, ..., s_j\} = \emptyset$ for each demand $k' \in C$ s.t.

$$\sum_{v_k \in C} \sum_{s=s_i+w_k-1}^{s_j} z_s^k(S') > |C| + 1.$$

Given that $\sum_{s=s_i+w_k-1}^{s_j} z_s^k(S') \leq 1$ for each demand $k \in C$, it follows that

$$\sum_{v_k \in C} \sum_{s=s_i+w_k-1}^{s_j} z_s^k(S') \leq |\tilde{K}| + 1,$$

which contradicts what we supposed before, i.e., $\sum_{v_k \in C} \sum_{s=s_i+w_k-1}^{s_j} z_s^k(S') > |C| + 1$.
As a result,

$$\sum_{v_k \in C} |E_k \cap \{e\}| + \sum_{v_k \in C} |S_k \cap \{s_i + w_k - 1, ..., s_j\}| + \sum_{k' \in C_e} |S_{k'} \cap \{s_i + w_{k'} - 1, ..., s_j\}| \leq |C| + 1.$$

Looking at the definition of the inequality (22), we detected that there may exist some cases that we can face that are not covered by the inequality (22). For this, we provide the following inequality and its generalization.

4.4 Interval-Clique Inequalities

Proposition 15. *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 $E_1^k \cap E_1^{k'} \neq \emptyset$, and $w_k \leq |I|$, and $w_{k'} \leq |I|$, and $w_k + w_{k'} > |I|$. Then, the inequality*

$$\sum_{s=s_i+w_k-1}^{s_j} z_s^k + \sum_{s'=s_i+w_{k'}-1}^{s_j} z_{s'}^{k'} \leq 1, \tag{24}$$

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

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 (24) is a particular case of the inequality (22) for $\tilde{K} = \{k, k'\}$ over each edge $e \in E_1^k \cap E_1^{k'}$. However, it will be used for a generalized inequality using the following conflict graph.

Definition 6. *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 *$\tilde{G}_I^E$ defined as follows. For each demand $k \in K$ with $w_k \leq |I|$, consider a node v_k in $\tilde{G}_I^E$. Two nodes v_k and $v_{k'}$ are linked by an edge in $\tilde{G}_I^E$ if $w_k + w_{k'} > |I|$ and $E_1^k \cap E_1^{k'} \neq \emptyset$.*

Proposition 16. *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*

$$\sum_{v_k \in C} \sum_{s=s_i+w_k-1}^{s_j} z_s^k \leq 1, \tag{25}$$

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

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 (24)

$$\sum_{s=s_i+w_k-1}^{s_j} z_s^k + \sum_{s'=s_i+w_{k'}-1}^{s_j} z_{s'}^{k'} \leq 1.$$

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

$$\sum_{v_k} (|C| - 1) \sum_{s=s_i+w_k-1}^{s_j} z_s^k \leq |C| - 1 \implies \sum_{v_k} \sum_{s=s_i+w_k-1}^{s_j} z_s^k \leq \frac{|C| - 1}{|C| - 1} \implies \sum_{v_k \in C} \sum_{s=s_i+w_k-1}^{s_j} z_s^k \leq 1.$$

We conclude at the end that the inequality (25) is valid for $P(G, K, \mathbb{S})$.

4.5 Interval-Odd-Hole Inequalities

Proposition 17. *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*

$$\sum_{v_k \in H} \sum_{s=s_i+w_k-1}^{s_j} z_s^k \leq \frac{|H|-1}{2}, \tag{26}$$

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

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 $\sum_{s=s_i+w_k-1}^{s_j} z_s^k + \sum_{s'=s_i+w_{k'}-1}^{s_j} z_{s'}^{k'} \leq 1$. 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

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

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 $\sum_{s=s_i+w_k-1}^{s_j} z_s^k$ appears twice in the previous inequality. As a result,

$$\sum_{(v_k, v_{k'}) \in E(H)} \sum_{s=s_i+w_k-1}^{s_j} z_s^k + \sum_{s'=s_i+w_{k'}-1}^{s_j} z_{s'}^{k'} = \sum_{v_k \in H} 2 \sum_{s=s_i+w_k-1}^{s_j} z_s^k, \sum_{v_k \in H} 2 \sum_{s=s_i+w_k-1}^{s_j} z_s^k \leq |H| - 1.$$

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

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

We conclude at the end that the inequality (26) is valid for $P(G, K, \mathbb{S})$.

The inequality (26) can be strengthened without modifying its right-hand side by combining the inequality (25) and (26) as follows.

Proposition 18. *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 $\tilde{G}_I^E$, and C be a clique in the conflict graph $\tilde{G}_I^E$ with*

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

Then, the inequality

$$\sum_{v_k \in H} \sum_{s=s_i+w_k-1}^{s_j} z_s^k + \frac{|H|-1}{2} \sum_{v_{k'} \in C} \sum_{s'=s_i+w_{k'}-1}^{s_j} z_{s'}^{k'} \leq \frac{|H|-1}{2}, \tag{27}$$

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

Proof. It is trivial given the definition of odd-hole set and clique set in the conflict graph $\tilde{G}_I^E$ s.t. if $\sum_{s'=s_i+w_{k'}-1}^{s_j} z_{s'}^{k'} = 1$ for $v_{k'} \in C$, it forces the quantity $\sum_{v_k \in H} \sum_{s=s_i+w_k-1}^{s_j} z_s^k$ to be equal to 0. Otherwise, we know from the inequality (26) that the sum $\sum_{v_k \in H} \sum_{s=s_i+w_k-1}^{s_j} z_s^k$ is always smaller than $\frac{|H|-1}{2}$. We strengthen our proof by assuming that the inequality (27) is not valid for

$P(G, K, \mathbb{S})$. 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.

$$\sum_{v_k \in H} \sum_{s=s_i+w_k-1}^{s_j} z_s^k(S) + \frac{|H|-1}{2} \sum_{v_{k'} \in C} \sum_{s'=s_i+w_{k'}-1}^{s_j} z_{s'}^{k'}(S) > \frac{|H|-1}{2}.$$

Since $\{s_i+w_{k'}-1, ..., s_j\} \notin S_{k'}$ for each node $v_{k'}$ in the clique C, this means that $\sum_{v_{k'} \in C} \sum_{s'=s_i+w_{k'}-1}^{s_j} z_{s'}^{k'}(S) = 0$, and taking into account the inequality (26), and that $\sum_{s=s_i+w_k-1}^{s_j} z_s^k(S) \leq 1$ for each $v_k \in H$ and $\sum_{s'=s_i+w_{k'}-1}^{s_j} z_{s'}^{k'}(S) \leq 1$ for each $v_{k'} \in C$, it follows that $\sum_{v_k \in H} \sum_{s=s_i+w_k-1}^{s_j} z_s^k(S) \leq \frac{|H|-1}{2}$, which contradicts that $\displaystyle\sum_{v_k \in H} \sum_{s=s_i+w_k-1}^{s_j} z_s^k(S) + \frac{|H|-1}{2} \sum_{v_{k'} \in C} \sum_{s'=s_i+w_{k'}-1}^{s_j} z_{s'}^{k'}(S) > \frac{|H|-1}{2}$.
Hence $\sum_{v_k \in H} |S_k \cap I_k| + \sum_{v_{k'} \in C} |S_{k'} \cap \{s_i + w_{k'} - 1, ..., s_j\}| \leq \frac{|H|-1}{2}$.

4.6 Edge-Slot-Assignment-Clique Inequalities

Taking into account the non-overlapping inequalities (8), we define another conflict graph differently compared with the conflict graphs introduced previously.

Definition 7. *Let $\tilde{G}_S^e$ be a conflict graph defined as follows. For each slot $s \in \{w_k, ..., \bar{s}\}$ and demand $k \in K$ with $e \notin E_0^k$, consider a node $v_{k,s}$ in $\tilde{G}_S^e$. Two nodes $v_{k,s}$ and $v_{k',s'}$ are linked by an edge in $\tilde{G}_S^e$ 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 $(k, k') \notin K_c^e$.*

The conflict graph $\tilde{G}_S^e$ is not an 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 $k = k'$.

Proposition 19. *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*

$$\sum_{v_{k,s} \in C} (x_e^k + z_s^k) \leq |C| + 1, \tag{28}$$

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

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 have

$$x_e^k + x_e^{k'} + z_s^k + z_{s'}^{k'} \leq 3.$$

This can be generalized for a triplet of linked nodes $v_{k,s}$ and $v_{k',s'}$ and $v_{k'',s''}$ with $w_k + w_{k'} + w_{k"} \leq \bar{s} - \sum_{\tilde{k} \in K_e \setminus \{k,k',k"\}} w_{\tilde{k}}$, such that for each linked nodes $(v_{k,s}, v_{k',s'})$ and $(v_{k,s}, v_{k'',s''})$ and $(v_{k',s'}, v_{k'',s''})$, we have

$$x_e^k + x_e^{k'} + z_s^k + z_{s'}^{k'} \leq 3,$$
$$x_e^k + x_e^{k"} + z_s^k + z_{s"}^{k"} \leq 3,$$
$$x_e^{k'} + x_e^{k"} + z_{s'}^{k'} + z_{s"}^{k"} \leq 3.$$

By adding the three previous inequalities, we get the following inequality using the chvatal gomory procedure

$$2x_e^k + 2x_e^{k'} + 2x_e^{k"} + 2z_s^k + 2z_{s'}^{k'} + 2z_{s"}^{k"} \leq 9$$
$$\Rightarrow x_e^k + x_e^{k'} + x_e^{k"} + z_s^k + z_{s'}^{k'} + z_{s"}^{k"} \leq 4 \text{ given that } \left\lfloor \frac{9}{2} \right\rfloor = 4.$$

This can be generalized for each clique C with $|C| \geq 4$ while showing that the inequality (28) can be seen as Chvatal-Gomory cuts. For that, and using the Chvatal-Gomory and recurrence procedures, we get that for all $C' \subset C$ with $|C'| = |C| - 1$ and $|C'| \geq 3$

$$\sum_{v_{k,s} \in C'} x_e^k + z_s^k \leq |C'| + 1.$$

By adding the previous inequalities for all $C' \subset C$ with $|C'| = |C| - 1$, and doing then some simplification, we get at the end that

$$\sum_{v_{k,s} \in C} x_e^k + z_s^k \leq \left\lfloor |C| + \frac{|C|}{|C|-1} \right\rfloor \Rightarrow \sum_{v_{k,s} \in C} x_e^k + z_s^k \leq |C| + 1$$

given that $\left\lfloor \frac{|C|}{|C|-1} \right\rfloor = 1$. We conclude that the inequality (28) is valid for $P(G, K, \mathbb{S})$.

This gives us an idea about new non-overlapping inequalities defined as follows.

Proposition 20. *Consider an edge e, and a pair of demands $k, k' \in K$ with $e \notin E_0^k \cup E_0^{k'}$. Let s be a slot in $\{w_k, ..., \bar{s}\}$. Then, the inequality*

$$x_e^k + x_e^{k'} + z_s^k + \sum_{s"=s-w_k+1}^{\min(s+w_{k'}-1, \bar{s})} z_{s"}^{k'} \leq 3, \tag{29}$$

is valid for $P"(G, K, \mathbb{S}) = \{(x, z) \in P(G, K, \mathbb{S}) : \sum_{s=w_k}^{\bar{s}} z_s^k = 1 \ \& \sum_{s=w_{k'}}^{\bar{s}} z_s^{k'} = 1\}$.

Remark 4. The inequality (29) is a particular case of inequality (28) for a clique $C = \{v_{k,s}\} \cup \{v_{k',s'} \in \tilde{G}_c^e \text{ s.t. } \{s' - w_k' + 1, ..., s'\} \cap \{s - w_k + 1, ..., s\} \neq \emptyset\}$.

Remark 5. The inequality (28) associated with a clique C over edge e, it is dominated by the inequality (22) 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 and an interval of contiguous slots $I = [s_i, s_j] \subseteq [1, \bar{s}]$. Let C be a clique in the conflict graph $\tilde{G}_S^e$, 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 $I = [s_i, s_j]$.
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

$$\sum_{k \in \tilde{K}} x_e^k + \sum_{k \in \tilde{K}} \sum_{s' \in I_k} z_{s'}^k = \sum_{k \in \tilde{K}} x_e^k + \sum_{k \in \tilde{K}} z_s^k + \sum_{k \in \tilde{K}} \sum_{s' \in I_k \setminus \{s\}} z_{s'}^k \tag{30}$$

$$\Longrightarrow \sum_{k \in \tilde{K}} \sum_{s' \in I_k} z_{s'}^k = \sum_{k \in \tilde{K}} z_s^k + \sum_{k \in \tilde{K}} \sum_{s' \in I_k \setminus \{s\}} z_{s'}^k. \tag{31}$$

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

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

This implies that

$$\sum_{k\in\tilde{K}}\sum_{s'\in I_k} z^k_{s'} = \sum_{v_{k,s}\in C} z^k_s + \sum_{k\in\tilde{K}}\sum_{s'\in I_k\setminus\{s\}} z^k_{s'} \implies \sum_{v_{k,s}\in C} z^k_s \preceq \sum_{k\in\tilde{K}}\sum_{s'\in I_k} z^k_{s'}.$$

Given that the demands are independants, it follows that

$$z^k_s \preceq \sum_{s'\in I_k} z^k_{s'} \text{ for each } v_{k,s}\in C.$$

Hence, the inequality (28) is dominated by the inequality (22).
Sufficiency: Assume that the inequality (28) is dominated by the inequality (22). It follows that

$$\sum_{v_{k,s}\in C} x^k_e z^k_s \preceq \sum_{k\in\tilde{K}} x^k_e + \sum_{s'\in I_k} z^k_{s'} \implies \sum_{v_{k,s}\in C} z^k_s \preceq \sum_{k\in\tilde{K}}\sum_{s'\in I_k} z^k_{s'} \implies \sum_{k\in\tilde{K}} z^k_s \preceq \sum_{k\in\tilde{K}}\sum_{s'\in I_k} z^k_{s'}$$

$$\implies z^k_s \preceq \sum_{s'\in I_k} z^k_{s'} \text{ for each } k\in\tilde{K} \implies s\in I_k \text{ for each } k\in\tilde{K} \implies s\in I_k \text{ for each node } v_{k,s}\in C$$

$$\implies s-w_k+1\in I \text{ for each node } v_{k,s}\in C \implies \min_{v_{k,s}\in C}(s-w_k+1)\in I$$

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

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 $k,k'\in\tilde{K}$. 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 $s'\in I_{k'}$. 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.

4.7 Slot-Assignment-Clique Inequalities

On the other hand, we detected that there may exist some cases that are not covered by the inequalities (17) and (28). For this, we provide the following definition of a *conflict graph* and its associated inequality.

Definition 8. *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 $\tilde{G}^E_S$ iff*

- *$k=k'$,*
- *or $E^k_1\cap E^{k'}_1\neq\emptyset$ and $\{s-w_k+1,...,s\}\cap\{s'-w_{k'}+1,...,s'\}\neq\emptyset$.*

The conflict graph $\tilde{G}^E_S$ cannot define an 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 $k=k'$.

Proposition 21. *Let C be a clique in* conflict graph *$\tilde{G}^E_S$ with $|C|\geq 3$. Then, the inequality*

$$\sum_{v_{k,s}\in C} z^k_s \leq 1, \tag{32}$$

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

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

$$z^k_s + z^{k'}_{s'} \leq 1,$$

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

$$\sum_{v_{k,s}}(|C|-1)z^k_s \le |C|-1 \implies \sum_{v_{k,s}} z^k_s \le \frac{|C|-1}{|C|-1} \implies \sum_{v_{k,s}} z^k_s \le 1,$$

which ends our proof.

Remark 6. The inequality (32) associated with a clique C, it is dominated by the inequality (25) associated with an interval $I = [s_i, s_j]$ and the 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 $w_k + w_{k'} \ge |I| + 1$ for each $(v_k, v_{k'}) \in C$, and $2w_k \ge |I| + 1$ and $w_k \le |I|$ 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 $\tilde{G}^E_S$, 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 $\tilde{G}^E_I$ 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 C,
- and $[\min_{v_{k,s}\in C}(s - w_k + 1), \max_{v_{k,s}\in C} s] \subset I$.

Given that $s - w_k + 1 \ge \min_{v_{k',s'}\in C}(s' - w_{k'} + 1)$ and $s \le \max_{v_{k',s'}\in C} s'$ 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

$$\sum_{k\in\tilde{K}}\sum_{s'\in I_k} z^k_{s'} = \sum_{k\in\tilde{K}} z^k_s + \sum_{k\in\tilde{K}}\sum_{s'\in I_k\setminus\{s\}} z^k_{s'}. \tag{33}$$

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

$$\sum_{k\in\tilde{K}} z^k_s = \sum_{v_{k,s}\in C} z^k_s.$$

It follows that

$$\sum_{k\in\tilde{K}}\sum_{s'\in I_k} z^k_{s'} = \sum_{v_{k,s}\in C} z^k_s + \sum_{k\in\tilde{K}}\sum_{s'\in I_k\setminus\{s\}} z^k_{s'}.$$

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

$$\sum_{v_{k,s}\in C} z^k_s \preceq \sum_{k\in\tilde{K}}\sum_{s'\in I_k} z^k_{s'}.$$

Hence, the inequality (32) is dominated by the inequality (25).
Sufficiency.
Assume that the inequality (32) is dominated by the inequality (25). It follows that

$$\sum_{v_{k,s}\in C} z^k_s \preceq \sum_{k\in\tilde{K}}\sum_{s'\in I_k} z^k_{s'} \implies \sum_{k\in\tilde{K}} z^k_s \preceq \sum_{k\in\tilde{K}}\sum_{s'\in I_k} z^k_{s'}$$

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

$$z^k_s \preceq \sum_{s'\in I_k} z^k_{s'} \text{ for each } k \in \tilde{K}.$$

Given that the variable z_s^k 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.

4.8 Slot-Assignment-Odd-Hole Inequalities

Proposition 22. *Let H be an odd-hole in the* conflict graph $\tilde{G}_S^E$ *with $|H| \geq 5$. Then, the inequality*

$$\sum_{v_{k,s} \in H} z_s^k \leq \frac{|H| - 1}{2}, \tag{34}$$

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

Proof. It is trivial given the definition of the odd-hole in the conflict graph $\tilde{G}_S^E$. 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 $z_s^k + z_{s'}^{k'} \leq 1$. 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

$$\sum_{(v_{k,s}, v_{k',s'}) \in E(H)} z_s^k + z_{s'}^{k'} \leq |H| - 1.$$

Taking into account that each node v_k in H has two neighbors in H, this implies that z_s^k appears twice in the previous inequality. As a result,

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

We conclude at the end that the inequality (34) is valid for $P(G, K, \mathbb{S})$.

Remark 7. The inequality (34) is dominated by the inequality (26) if and only if there exists an interval of contiguous slots $I = [s_i, s_j] \subset [1, \bar{s}]$ with

– $[\min_{v_{k,s} \in H \cup C}(s - w_k + 1), \max_{v_{k,s} \in H \cup C}] \subset I$,
– and $w_k + w_{k'} \geq |I| + 1$ for each $(v_k, v_{k'})$ linked in H,
– and $2w_k \geq |I| + 1$ 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 $\tilde{G}_S^E$, 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 $\tilde{G}_I^E$ 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 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

$$\sum_{k \in \tilde{K}} \sum_{s' \in I_k} z_{s'}^k = \sum_{k \in \tilde{K}} z_s^k + \sum_{k \in \tilde{K}} \sum_{s' \in I_k \setminus \{s\}} z_{s'}^k. \tag{35}$$

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

$$\sum_{k \in \tilde{K}} z_s^k = \sum_{v_{k,s} \in H} z_s^k.$$

This implies that

$$\sum_{k \in \tilde{K}} \sum_{s' \in I_k} z_{s'}^k = \sum_{v_{k,s} \in H} z_s^k + \sum_{k \in \tilde{K}} \sum_{s' \in I_k \setminus \{s\}} z_{s'}^k$$
$$\implies \sum_{v_{k,s} \in H} z_s^k \preceq \sum_{k \in \tilde{K}} \sum_{s' \in I_k} z_{s'}^k \implies z_s^k \preceq \sum_{s' \in I_k} z_{s'}^k \text{ for each } v_{k,s} \in H.$$

Hence, the inequality (34) is dominated by the inequality (26).
Sufficiency.
Assume that the inequality (34) is dominated by the inequality (26) and given that $\tilde{K} = \{k \in K \text{ s.t. } v_{k,s} \in H\}$, this means that

$$\sum_{k \in \tilde{K}} z_s^k = \sum_{v_{k,s} \in H} z_s^k.$$

It follows that

$$\sum_{v_{k,s} \in H} z_s^k \preceq \sum_{k \in \tilde{K}} \sum_{s' \in I_k} z_{s'}^k \implies \sum_{k \in \tilde{K}} z_s^k \preceq \sum_{k \in \tilde{K}} \sum_{s' \in I_k} z_{s'}^k.$$

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

$$z_s^k \preceq \sum_{s' \in I_k} z_{s'}^k \text{ for each } k \in \tilde{K} \implies s \in I_k \text{ for each } k \in \tilde{K} \implies s \in I_k \text{ for each node } v_{k,s} \in H.$$

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.

The inequality (34) can be strengthened without modifying its right hand side by combining the inequality (34) and (32) as follows.

Proposition 23. *Let H be an odd-hole, and C be a clique in the conflict graph $\tilde{G}_S^E$ 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 $\tilde{G}_S^E$ for all $v_{k,s} \in H$ and $v_{k',s'} \in C$.*

Then, the inequality

$$\sum_{v_{k,s} \in H} z_s^k + \frac{|H|-1}{2} \sum_{v_{k',s'} \in C} z_{s'}^{k'} \leq \frac{|H|-1}{2}, \tag{36}$$

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

Proof. It is trivial given the definition of the odd-hole and clique in $\tilde{G}_S^E$ s.t. if $\sum_{v_{k',s'} \in C} z_{s'}^{k'} = 1$ for a $v_{k',s'} \in C \in C$ which implies that the quantity $\sum_{v_{k,s} \in H} z_s^k$ is forced to be equal to 0. Otherwise, we know from the inequality (34) that the sum $\sum_{v_{k,s} \in H} z_s^k$ is always smaller than $\frac{|H|-1}{2}$. We strengthen our proof by assuming that the inequality (36) is not valid for $P(G, K, \mathbb{S})$. 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.

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

Since $s' \notin S_{k'}$ for each node $v_{k',s'}$ in the clique C this means that $\sum_{v_{k',s'} \in C} z_{s'}^{k'}(S) = 0$, and taking into account the inequality (34), $z_s^k(S) \leq 1$ for each $v_{k,s} \in H$, and that $z_{s'}^{k'}(S) \leq 1$ for each $v_{k',s'} \in C$, it follows that

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

which contradicts that $\sum_{v_{k,s} \in H} z_s^k(S) + \frac{|H|-1}{2} \sum_{v_{k',s'} \in C} z_{s'}^{k'}(S) > \frac{|H|-1}{2}$.
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 8. The inequality (36) is dominated by the inequality (27) if and only if there exists an interval of contiguous slots $I = [s_i, s_j] \subset [1, \bar{s}]$ with

- $[\min_{v_{k,s} \in H \cup C}(s - w_k + 1), \max_{v_{k,s} \in H \cup C}] \subset I$,
- 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 to the proof of the remark 7.

4.9 Non-Compatibility-Clique Inequalities

Let us now introduce some valid inequalities that are related to the routing sub-problem due to the transmission-reach constraint.

Definition 9. *For a demand k, two edges $e = ij \notin E_0^k \cap E_1^k, e' = lm \notin E_0^k \cap E_1^k$ are said* non-compatible edges *if and only if the lengths of (o_k, d_k)-paths formed by $e = ij$ and $e' = lm$ together are greater that $\bar{l}_k$.*

Note that we are able to determine the non-compatible edges for each demand k in polynomial time using shortest-path algorithms.

Proposition 24. *Consider an edge $e \in E$. Let (k, k') be a pair of non-compatible demands for the edge e. Then, the inequality*

$$x_e^k + x_e^{k'} \leq 1, \tag{37}$$

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

Proof. It is trivial given the definition of non-compatible demands for the edge e.

Proposition 25. *Consider a demand $k \in K$. Let (e, e') be a pair of non-compatible edges for the demand k. Then, the inequality*

$$x_e^k + x_{e'}^k \leq 1, \tag{38}$$

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

Proof. It is trivial given the definition of non-compatible edges for a demand k.

Based on the inequalities (37) and (38), we introduce the following *conflict graph.*

Definition 10. *Let $\tilde{G}_E^K$ be a* conflict graph *defined as follows. For each demand k and edge $e \notin E_0^k \cup E_1^k$, consider a node v_e^k in $\tilde{G}_E^K$. Two nodes v_e^k and $v_{e'}^{k'}$ are linked by an edge in $\tilde{G}_E^K$*

- *if $k = k'$: e and e' are non compatible edges for demand k.*
- *if $k \neq k'$: k and k' are non compatible demands for edge e.*

Proposition 26. *Let C be a clique in $\tilde{G}_E^K$. Then, the inequality*

$$\sum_{v_e^k \in C} x_e^k \leq 1, \tag{39}$$

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

Proof. It is trivial given the definition of a clique set in the conflict graph $\tilde{G}_E^K$ s.t. by adding the inequalities (38) for all pairs of nodes $(v_e^k, v_{e'}^{k'})$ in the clique C in $\tilde{G}_E^K$

$$\sum_{v_e^k \in C} (|C| - 1) x_e^k \leq (|C| - 1) \implies \sum_{v_e^k \in C} x_e^k \leq \frac{|C| - 1}{|C| - 1} \implies \sum_{v_e^k \in C} x_e^k \leq 1,$$

which ends our proof.

4.10 Non-Compatibility-Odd-Hole Inequalities

Proposition 27. *Let H be an odd-hole in the* conflict graph *$\tilde{G}_E^K$ with $|H| \geq 3$. Then, the inequality*

$$\sum_{v_e^k \in H} x_e^k \leq \frac{|H| - 1}{2}, \tag{40}$$

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

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

$$\sum_{(v_e^k, v_{e'}^{k'}) \in E(H)} x_e^k + x_{e'}^{k'} \leq |H| - 1.$$

Taking into account that each node v_e^k in H has two neighbors in H, this implies that x_e^k appears twice in the previous inequality. As a result,

$$\sum_{(v_e^k, v_{e'}^{k'}) \in E(H)} x_e^k + x_{e'}^{k'} = \sum_{v_e^k \in H} 2x_e^k \implies \sum_{v_e^k \in H} 2x_e^k \leq |H| - 1$$

$$\implies \sum_{v_e^k \in H} x_e^k \leq \left\lfloor \frac{|H| - 1}{2} \right\rfloor = \frac{|H| - 1}{2} \text{ since } |H| \text{ is an odd number.}$$

We conclude at the end that the inequality (40) is valid for $P(G, K, \mathbb{S})$.

The inequality (40) can be strengthened without modifying its right hand side by combining the inequality (40) and (39) as follows.

Proposition 28. *Let H be an odd-hole in the conflict graph $\tilde{G}_E^K$, and C be a clique in the conflict graph $\tilde{G}_E^K$ with*

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

Then, the inequality

$$\sum_{v_e^k \in H} x_e^k + \frac{|H| - 1}{2} \sum_{v_{e'}^{k'} \in C} x_{e'}^{k'} \leq \frac{|H| - 1}{2}, \tag{41}$$

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

Proof. It is trivial given the definition of the odd-hole and clique in $\tilde{G}_E^K$ s.t. if $\sum_{v_{e'}^{k'} \in C} x_{e'}^{k'} = 1$ for a $v_{e'}^{k'} \in C$, which implies that the quantity $\sum_{v_e^k \in H} x_e^k$ is forced to be equal to 0. Otherwise, we know from the inequality (40) that the sum $\sum_{v_e^k \in H} x_e^k$ should be smaller than $\frac{|H|-1}{2}$. We strengthen our proof by assuming that the inequality (41) is not valid for $P(G, K, \mathbb{S})$. It follows that there exists a C-RSA solution S in which $e' \notin E_{k'}$ for each node $v_{e'}^{k'}$ in the clique C s.t.

$$\sum_{v_e^k \in H} x_e^k(S) + \frac{|H| - 1}{2} \sum_{v_{e'}^{k'} \in C} x_{e'}^{k'}(S) > \frac{|H| - 1}{2}.$$

Since $e' \notin E_{k'}$ for each node $v_{e'}^{k'}$ in the clique C this means that $\sum_{v_{e'}^{k'} \in C} x_{e'}^{k'}(S) = 0$, and taking into account the inequality (40), and that $x_e^k(S) \leq 1$ for each $v_e^k \in H$ and $x_{e'}^{k'}(S) \leq 1$ for each $v_{e'}^{k'} \in C$, it follows that

$$\sum_{v_e^k \in H} x_e^k(S) \leq \frac{|H| - 1}{2},$$

which contradicts what we supposed before, i.e., $\sum_{v_e^k \in H} x_e^k(S) + \frac{|H|-1}{2} \sum_{v_{e'}^{k'} \in C} x_{e'}^{k'}(S) > \frac{|H|-1}{2}$.

Hence $\sum_{v_e^k \in H} |E_k \cap \{e\}| + \sum_{v_{e'}^{k'} \in C} |E_{k'} \cap \{e'\}| \leq \frac{|H| - 1}{2}$.

We conclude at the end that the inequality (41) is valid for $P(G, K, \mathbb{S})$.

On the other hand, let's us now provide some inequalities related to the capacity constraint.

4.11 Edge-Capacity-Cover Inequalities

Proposition 29. *Consider an edge e in E. Then, the inequality*

$$\sum_{k \in K \setminus K_e} w_k x_e^k \leq \bar{s} - \sum_{k' \in K_e} w_{k'}, \tag{42}$$

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

Proof. The number of slots allocated in 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 11. *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 12. *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., $\sum_{k' \in C \setminus \{k\}} w_{k'} \leq \bar{s} - \sum_{k" \in K_e} w_{k"}$.*

Proposition 30. *Consider an edge e in E. Let C be a minimal cover in K for the edge e. Then, the inequality*

$$\sum_{k \in C} x_e^k \leq |C| - 1, \tag{43}$$

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

Proof. If C is a minimal cover for edge $e \in E$ this means that there are 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 (43) is not valid for $P(G, K, \mathbb{S})$. It follows that there exists a C-RSA solution S in which $e \notin E_{k'}$ for a demand $k' \in C$ s.t.

$$\sum_{k \in C} x_e^k(S) > |C| - 1.$$

Since $e \notin E_{k'}$ for a demand $k' \in C$ this means that $x_e^{k'}(S) = 0$, 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

$$\sum_{k \in C \setminus \{k'\}} x_e^k(S) \leq |C| - 1$$

which contradicts what we supposed before, i.e., $\sum_{k \in C} x_e^k(S) > |C| - 1$.
Hence $\sum_{k \in C} |E_k \cap \{e\}| \leq |C| - 1$.
We conclude at the end that the inequality (43) is valid for $P(G, K, \mathbb{S})$.

We verified that the inequality (43) can be easily strengthened by using its extended format which we call extended minimal cover for an edge e as follows.

Proposition 31. *Consider an edge e in E. Let C be a minimal cover in K for the edge e, and $\Xi(C)$ be a subset of demands in $K \setminus C \cup K_e$ where $\Xi = \{k \in K \setminus C \cup K_e : e \notin E_0^k$ and $w_k \geq w_{k'} \quad \forall k' \in C\}$. Then, the inequality*

$$\sum_{k \in C} x_e^k + \sum_{k' \in \Xi(C)} x_e^{k'} \leq |C| - 1, \tag{44}$$

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

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 $C \cup \Xi(C)$ that can use the edge e. We strengthen our proof by assuming that the inequality (44) is not valid for $P(G, K, \mathbb{S})$. It follows that there exists a C-RSA solution S in which $e \notin E_{k'}$ for each demand $k' \in \Xi(C)$ s.t.

$$\sum_{k \in C} x_e^k(S) > |C| - 1.$$

Since $e \notin E_{k'}$ for for each demand $k' \in \Xi(C)$this means that $x_e^{k'}(S) = 0$, 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

$$\sum_{k \in C} x_e^k(S) \leq |C| - 1$$

which contradicts what we supposed before, i.e., $\sum_{k \in C} x_e^k(S) > |C| - 1$ and also the inequality (43).
Hence $\sum_{k \in C} |E_k \cap \{e\}| + \sum_{k' \in \Xi(C)} |E_{k'} \cap \{e\}| \leq |C| - 1.$
We conclude at the end that the inequality (43) is valid for $P(G, K, \mathbb{S})$.

Furthermore, the inequality (43) can have a more generalized strengthening format using lifting procedures proposed by Nemhauser and Wolsey in [50].

5 Branch-and-Cut Algorithm

Based on the theoretical results presented in this paper, we devise a Branch-and-Cut algorithm to solve the C-RSA problem. We aim to study the effectiveness of our algorithm and assess the impact of each valid inequality on the effectiveness of our algorithm. First, we give an overview of our algorithm. Then, we describe the separation procedure used for each valid inequality based on exact algorithms, greedy algorithms, and heuristics. In the end, we provide a detailed behavior study of our Branch-and-Cut algorithm using two types of topologies: real, and realistic, and two types of instances: random, and realistic ones.

5.1 Overview

In what follows, we describe our Branch-and-Cut algorithm. Consider an undirected, loopless, and connected graph $G = (V, E)$, which is specified by a set of nodes V, and a multiset E of links. Each link $e = ij \in E$ is associated with a length $\ell_e \in \mathbb{R}_+$ (in kms), a cost $c_e \in \mathbb{R}_+$ s.t. each 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 $\bar{s} \leq 320$, and K be a multiset of demands s.t. each demand $k \in K$ is specified by an origin 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). We first consider a restricted linear problem denoted by LP_0 given by the inequalities (3)-(7) and (9)-(12) s.t. the cut inequalities (2) and non-overlapping inequalities (8) are not included in LP_0. LP_0 is so equivalent to

$$\min \sum_{k \in K} \sum_{e \in E} l_e x_e^k$$

$$\sum_{e \in E} l_e x_e^k \leq \bar{\ell}_k, \forall k \in K,$$

$$x_e^k = 0, \forall k \in K, \forall e \in E_0^k,$$

$$x_e^k = 1, \forall k \in K, \forall e \in E_1^k,$$

$$z_s^k = 0, \forall k \in K, \forall s \in \{1, ..., w_k - 1\},$$

$$\sum_{s=w_k}^{\bar{s}} z_s^k = 1, \forall k \in K,$$

$$0 \leq x_e^k \leq 1, \forall k \in K, \forall e \in E,$$

$$0 \leq z_s^k \leq 1, \forall k \in K, \forall s \in \mathbb{S}.$$

Given an optimal solution $(\bar{x}, \bar{z})$ for the relaxation of LP_0. It is feasible for the C-RSA problem iff $(\bar{x}, \bar{z})$ is integral and it satisfies the cut inequalities (2) and non-overlapping inequalities (8). Usually, $(\bar{x}, \bar{z})$ does not satisfy the inequalities (2) and (8). As a result, $(\bar{x}, \bar{z})$ is not feasible for the C-RSA problem. For that, we generate several valid inequalities violated by a solution $(\bar{x}, \bar{z})$ at each iteration of our Branch-and-Cut algorithm. This is known under the name "Separation Problem" which consists in identifying for a given class of valid inequalities the existence of one or more 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. As a result, the final solution is optimal for the linear relaxation of our cut 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 a fractional variable (variable branching rule) or some constraints using the Ryan & Foster branching rule (constraint branching rule). Based on this, we devise a basic Branch-and-Cut algorithm by combining a cutting-plane algorithm based on the separation of the cut inequalities (2) and non-overlapping inequalities (8), and a Branch-and-Bound algorithm.

5.2 Separation Procedures: Complexity and Algorithms

On another hand, to accelerate our Branch-and-Cut algorithm, we already introduced several classes of valid inequalities used to obtain tighter LP bounds. Based on this, and at each iteration in a certain level of our Branch-and-Cut algorithm, one can identify one or more than one violated inequality by the current fractional solution for a given class of valid inequalities. To do so, we study the separation problem of each valid inequality as follows.

Separation of Non-Overlapping Inequalities Consider a fractional solution $(\bar{x}, \bar{z})$, and an edge $e \in E$ and a slot $s \in \mathbb{S}$. The separation problem associated with the inequality (8) consists in identifying all pairs of demands $k, k' \in K$ s.t.

$$\bar{x}_e^k + \bar{x}_e^{k'} + \sum_{s'=s}^{\min(s+w_k-1,\bar{s})} \bar{z}_{s'}^k + \sum_{s"=s}^{\min(s+w_{k'}-1,\bar{s})} \bar{z}_{s"}^{k'} > 3.$$

To do so, we propose an exact algorithm in $\mathcal{O}(|E| * \bar{s} * |K| * \log(|K|))$ which works as follows. We select each pair of demands $k, k' \in K$ with $x_e^k > 0$, $\sum_{s'=s}^{min(s+w_k-1,\bar{s})} z_{s'}^k > 0$, $\bar{x}_e^{k'} > 0$ and $\sum_{s"=s}^{\min(s+w_{k'}-1,\bar{s})} \bar{z}_{s"}^{k'} > 0$. We then add the inequality (8) induced by each selected pair of demands k, k' for the slot s ove edge e, to the current LP if it is violated, i.e.,

$$x_e^k + x_e^{k'} + \sum_{s'=s}^{\min(s+w_k-1,\bar{s})} z_{s'}^k + \sum_{s"=s}^{\min(s+w_{k'}-1,\bar{s})} z_{s"}^{k'} \leq 3.$$

Otherwise, we conclude that such inequality does not exist for the current solution $(\bar{x}, \bar{z})$. On the other hand, given that the inequalities (7) are taken in format of equations when implementing our B&C algorithm (i.e., $\sum_{s=w_k}^{\bar{s}} z_s^k = 1$ for all $k \in K$). Based on this, and taking into account the non-overlapping inequalities (8), we already proposed a new non-overlapping inequality (29) more efficient compared to the ones of (8). To do so, we propose an exact algorithm in $\mathcal{O}(|E| * \bar{s} * |K| * (|K|-1))$ which works as follows. For each demand k and each slot $s \in \{w_k, ..., \bar{s}\}$ over edge e with $x_e^k > 0$, $z_s^k > 0$, we select each demand $k' \in K$ with $\bar{x}_e^{k'} > 0$ and $\sum_{s"=s-w_k+1}^{\min(s+w_{k'}-1,\bar{s})} \bar{z}_{s"}^{k'} > 0$. We then add the following inequality to the current LP if it is violated, i.e.,

$$x_e^k + x_e^{k'} + z_s^k + \sum_{s"=s-w_k+1}^{\min(s+w_{k'}-1,\bar{s})} z_{s"}^{k'} \leq 3.$$

Otherwise, we conclude that there does not exist an inequality from the non-overlapping inequalities (29) violated current solution $(\bar{x}, \bar{z})$. Note that, from an efficiency point of view, the inequalities (29) replace the inequalities (8) in our B&C algorithm.

Separation of Cut Inequalities In this section,frameworksss the separation problem of our cut inequalities (2). Its associated separation problem consists in identifying a cut inequalities (2) that is violated by a given fractional solution $(\bar{x}, \bar{z})$. For each demand $k \in K$, this can be done in polynomial time [22] as shown in the theorem of Ford and Fulkerson by finding a minimum cut separating the origin-node o_k and destination-node d_k. As a result, this can be done exactly [22] and very effectively in $\mathcal{O}(|V \setminus V_0^k|^2 * \sqrt{|E \setminus E_0^k|})$ using an efficient implementation of minimum cut algorithm based on the so-called preflow push-relabel algorithm of Goldberg and Tarjan [24] to compute maximum flow/minimum cut in the proper graph G_k of demand k by assigning a positive weight $\bar{x}_e^k$ for each edge e in the graph G_k. For that, we use a C++ library proposed by the LEMON GRAPH library [38] which calls the algorithm of Goldberg and Tarjan for the minimum cut computation. Based on this, we conclude that the separation of the cut inequalities (2) can be done in $\mathcal{O}(|V|^2 * \sqrt{|E|} * |K|)$ in the worst case.

Separation of Edge-Slot-Assignment Inequalities Consider a fractional solution $(\bar{x}, \bar{z})$, and an edge $e \in E$ and a slot $s \in \mathbb{S}$. The separation problem associated with the inequality (15) consists in identifying a subset of demands $\tilde{K}^* \subset K$ s.t.

$$\sum_{k \in \tilde{K}^*} \bar{x}_e^k + \sum_{s'=s}^{\min(s+w_k-1,\bar{s})} \bar{z}_{s'}^k > |\tilde{K}^*| + 1.$$

To do so, we propose an exact algorithm in $\mathcal{O}(|K| * |E| * \bar{s})$ which works as follows. The main idea is to iteratively add each demand $k \in K$ to $\tilde{K}^*$ iff $x_e^k > 0$ and $\sum_{s'=s}^{min(s+w_k-1,\bar{s})} z_{s'}^k > 0$. We then add the inequality (15) induced by $\tilde{K}^*$ to the current LP if it is violated, i.e.,

$$\sum_{k \in \tilde{K}^*} x_e^k + \sum_{s'=s}^{\min(s+w_k-1,\bar{s})} z_{s'}^k \le |\tilde{K}^*| + 1.$$

Otherwise, we conclude that such inequality does not exist for the current solution $(\bar{x}, \bar{z})$. Moreover, if such violated inequality is identified, it can be easily lifted introducing the inequality (17) induced by $\tilde{K}^*$ and a subset of demands $K_e \setminus \tilde{K}^*$ as follows

$$\sum_{k \in \tilde{K}^*} x_e^k + \sum_{s'=s}^{\min(s+w_k-1,\bar{s})} z_{s'}^k + \sum_{k' \in K_e \setminus \tilde{K}^*} \sum_{s'=s}^{\min(s+w_{k'}-1,\bar{s})} \le |\tilde{K}^*| + 1.$$

Separation of Edge-Slot-Assignment-Clique Inequalities Consider an edge $e \in E$, and a fractional solution $(\bar{x}, \bar{z})$. The separation algorithm for the inequality (28) consists in identifying a maximal clique C^* in the conflict graph $\tilde{G}_S^e$ s.t.

$$\sum_{v_{k,s} \in C^*} \bar{x}_e^k + \bar{z}_s^k > |C| + 1.$$

To do this, we use the greedy algorithm introduced by Nemhauser and Sigismondi in [51] 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 [59]. Based on this, we first assign a positive weight $\bar{z}_s^k * \bar{x}_e^k$ 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 $\bar{z}_{s'}^{k'} > 0$ and $\bar{x}_e^{k'} > 0$. At the end, we add the inequality (28) induced by the clique C^* for edge e to the current LP if it is violated, i.e., we add the following inequality

$$\sum_{v_{k,s} \in C^*} x_e^k + z_s^k \le |C| + 1.$$

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 [51] 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 inequality (28) induced by the clique $C^* \cup N^*$ to the current LP, i.e.,

$$\sum_{v_{k,s} \in C^*} (x^k_e + z^k_s) + \sum_{v_{k',s'} \in N^*} z^{k'}_{s'} \leq 1.$$

Separation of Edge-Interval-Cover Inequalities Let's discuss the separation problem of the inequality (20). Given a fractional solution $(\bar{x}, \bar{z})$, 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_e is identified by generating two slots s_i and s_j randomly 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 (20) is Np-Hard [60] given that it consists in identifying a cover $\tilde{K}^*$ for the interval $I = [s_i, s_j]$ over the edge e, s.t.

$$\sum_{k \in \tilde{K}^*} \bar{x}^k_e + \sum_{s'=s_i+w_k-1}^{s_j} \bar{z}^k_{s'} > 2|\tilde{K}^*| - 1.$$

For that, we use a greedy algorithm introduced by Nemhauser and Sigismondi in [51] as follows. We first select a demand $k \in K$ having the largest number of requested slot w_k with $\bar{x}^k_e > 0$ and $\sum_{s'=s_i+w_k-1}^{s_j} \bar{z}^k_{s'} > 0$, and then set $\tilde{K}^*$ to $\tilde{K}^* = \{k\}$. After that, we iteratively add each demand $k' \in K \setminus \tilde{K}^*$ to $\tilde{K}^*$ with $\bar{x}^{k'}_e > 0$ and $\sum_{s'=s_i+w_{k'}-1}^{s_j} \bar{z}^{k'}_{s'} > 0$, 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 (20) induced by the minimal cover $\tilde{K}^*$ for the interval I and edge e if it is violated, i.e., we add the following valid inequality to the current LP

$$\sum_{k \in \tilde{K}^*} x^k_e + \sum_{s'=s_i+w_k-1}^{s_j} z^k_{s'} \leq 2|\tilde{K}^*| - 1.$$

Furthermore, the inequality (20) induced by the minimal cover $\tilde{K}^*$ can be lifted in polynomal time $\mathcal{O}(K_e \setminus \tilde{K})$ by introducing an extended cover inequality (21) as follows

$$\sum_{k \in \tilde{K}^*} x^k_e + \sum_{s'=s_i+w_k-1}^{s_j} z^k_{s'} + \sum_{k' \in \tilde{K}^*_e} \sum_{s'=s_i+w_{k'}-1}^{s_j} z^{k'}_{s'} \leq 2|\tilde{K}^*| - 1,$$

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 (22) is NP-hard [55][59] 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 $I = [s_i, s_j]$ s.t.

$$\sum_{k \in C^*} \bar{x}^k_e + \sum_{s'=s_i+w_k-1}^{s_j} \bar{z}^k_{s'} > |C^*| + 1,$$

for a given fractional solution $(\bar{x}, \bar{z})$ of the current LP.
We start our procedure of separation by constructing a set of intervals of contiguous slots $I = [s_i, s_j] \in I_e$ 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 $\tilde{G}_I^e$. We then use a greedy algorithm introduced by Nemhauser and Sigismondi in [51] to identify a maximal clique in conflict graph $\tilde{G}_I^e$ as follows. We first associate a positive weight for each node v_k in $\tilde{G}_I^e$ equals to $\bar{x}_e^k * \sum_{s'=s_i+w_k-1}^{s_j} \bar{z}_{s'}^k$. We then set $C^* = \{k\}$ s.t. k is a demand in K having the largest number of slots w_k and weight $\bar{x}_e^k * \sum_{s'=s_i+w_k-1}^{s_j} \bar{z}_{s'}^k$. After that, we iteratively add each demand k' having $\bar{x}_e^{k'} > 0$ and $\sum_{s'=s_i+w_{k'}-1}^{s_j} \bar{z}_{s'}^{k'} > 0$ 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 (22) induced by the maximal clique C^* for the interval I and edge e is violated or not. If so, we add the inequality (22) induced by the maximal clique C^* to the current LP, i.e.,

$$\sum_{k \in C^*} x_e^k + \sum_{s'=s_i+w_k-1}^{s_j} z_{s'}^k \leq |C^*| + 1.$$

One can strengthen this additional inequality by adding the inequality (23) induced by the maximal clique C^* and $C_e^* \subset K_e \setminus C^*$, i.e.,

$$\sum_{k \in C^*} x_e^k + \sum_{s'=s_i+w_k-1}^{s_j} z_{s'}^k + \sum_{k' \in C_e^*} \sum_{s'=s_i+w_{k'}-1}^{s_j} z_{s'}^{k'} \leq |C^*| + 1,$$

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 $(\bar{x}, \bar{z})$, and an interval of contiguous slots $I = [si, s_j]$. Our separation algorithm for the inequality (25) consists in identifying a maximal clique C^* in the conflict graph $\tilde{G}_I^E$ s.t.

$$\sum_{k \in C^*} \sum_{s'=s_i+w_k-1}^{s_j} \bar{z}_{s'}^k > 1.$$

As result, its associated separation problem is NP-hard given that computing a maximal clique in a given graph is known to be a NP-hard problem [59]. For that, we also use the greedy algorithm introduced by Nemhauser and Sigismondi in [51] to identify a maximal clique in conflict graph $\tilde{G}_I^E$ 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 given for each slot $s_i \in \mathbb{S}$ and slot s_j with $s_j \in \{s_i + \max_{\substack{k \in K, \\ |E_1^k| \geq 1}} w_k, ..., \min(\bar{s}, s_i + 2 \max_{\substack{k \in K, \\ |E_1^k| \geq 1}} w_k)\}$. We then consider an interval of contiguous slots $I = [s_i, s_j] \in I_E$ and its associated conflict graph $\tilde{G}_I^E$. We associate a positive weight $\sum_{s'=s_i+w_k-1}^{s_j} \bar{z}_{s'}^k$ for each node v_k in $\tilde{G}_I^E$. We select a demand k s.t. k is a demand in K having the largest number of slots w_k and weight $\sum_{s'=s_i+w_k-1}^{s_j} \bar{z}_{s'}^k$, and then set $C^* = \{k\}$. After that, we iteratively add each demand k' having $\sum_{s'=s_i+w_{k'}-1}^{s_j} \bar{z}_{s'}^{k'} > 0$ 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 (25) induced by the maximal clique C^* if it is violated, i.e., by adding the following inequality to the current LP

$$\sum_{k \in C^*} \sum_{s'=s_i+w_k-1}^{s_j} z_{s'}^k \leq 1.$$

Moreover, this additional inequality can be strengthened as follows

$$\sum_{k \in C^*} \sum_{s'=s_i+w_k-1}^{s_j} z_{s'}^k + \sum_{k' \in C_e^*} \sum_{s'=s_i+w_{k'}-1}^{s_j} z_{s'}^{k'} \leq 1,$$

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

- $w_{k'} + w_k \geq |I| + 1$ and $E_1^k \cap E_1^{k'} \neq \emptyset$ for each $k \in C^*$ and $k' \in C_E^*$,
- $w_{k'} + w_{k"} \geq |I| + 1$ and $E_1^{k'} \cap E_1^{k"} \neq \emptyset$ 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 (26), we propose a separation algorithm that consists in identifying an odd-hole H^* in the conflict graph $\tilde{G}_I^E$ for a given Interval I and a fractional solution $(\bar{x}, \bar{z})$ s.t.

$$\sum_{k \in H^*} \sum_{s'=s_i+w_k-1}^{s_j} \bar{z}_{s'}^k > \frac{|H^*|-1}{2}.$$

This can be done in polynomial time as shown by Rebennack et al. in [68] and [69]. 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 5.2. 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 $\frac{1-\sum_{s'=s_i+w_a-1}^{s_j} \bar{z}_{s'}^a - \sum_{s'=s_i+w_b-1}^{s_j} \bar{z}_{s'}^b}{2}$. 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 $\frac{1}{2}$,

$$\sum_{(v_a,v_b) \in E(p_{v_k,v_k'})} \frac{1-\sum_{s'=s_i+w_a-1}^{s_j} \bar{z}_{s'}^a - \sum_{s'=s_i+w_b-1}^{s_j} \bar{z}_{s'}^b}{2} < \frac{1}{2}.$$

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 (26) induced by the odd-hole H^* to the current LP, i.e.,

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

It can be lifted using the greedy algorithm introduced by Nemhauser and Sigismondi in [51] to identify a maximal clique C^* in conflict graph $\tilde{G}_I^E$ 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 first 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 (27) induced by the odd-hole H^* and clique C^* to the current LP, i.e.,

$$\sum_{k \in H^*} \sum_{s'=s_i+w_k-1}^{s_j} z_{s'}^k + \frac{|H^*|-1}{2} \sum_{k' \in C^*} \sum_{s"=s_i+w_{k'}-1}^{s_j} z_{s"}^{k'} \leq \frac{|H^*|-1}{2}.$$

Separation of Slot-Assignment-Clique Inequalities Now, we describe the separation algorithm for the inequality (32). It consists in identifying a maximal clique C^* in the conflict graph $\tilde{G}_S^E$ s.t.

$$\sum_{v_{k,s} \in C^*} \bar{z}_s^k > 1,$$

for a given fractional solution $(\bar{x}, \bar{z})$ of the current LP.
To do so, we use the greedy algorithm introduced by Nemhauser and Sigismondi in [51] 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 [59]. Based on this, we first assign a positive weight $\bar{z}_s^k$ 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 $\bar{z}_{s'}^{k'} > 0$. At the end, we add the inequality (32) induced by the clique C^* to the current LP if it is violated, i.e., we add the following inequality

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

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 [51] 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 (32) induced by the clique $C^* \cup N^*$ to the current LP, i.e.,

$$\sum_{v_{k,s} \in C^*} z_s^k + \sum_{v_{k',s'} \in N^*} z_{s'}^{k'} \leq 1.$$

Separation of Slot-Assignment-Odd-Hole Inequalities The separation algorithm of the inequality (34) can be performed by identifying an odd-hole H^* in the conflict graph $\tilde{G}_S^E$ for a given fractional solution $(\bar{x}, \bar{z})$ s.t.

$$\sum_{v_{k,s} \in H^*} \bar{z}_s^k > \frac{|H^*| - 1}{2}.$$

This can be done in polynomial time as shown by Rebennack et al. in [68] and [69] 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 also as a bipartite graph by duplicating each node $v_{k,s}$ in $\tilde{G}_S^E$ (i.e., $v_{k,s}$ and $v'_{k,s}$) s.t. 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 $\frac{1-\bar{z}_s^k-\bar{z}_{s'}^{k'}}{2}$. We then compute for each node $v_{k,s}$ in $\tilde{G}_S^E$, the shortest path between $v_{k,s}$ and its copy $v'_{k,s}$ 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 (34) induced by the odd-hole H^*

$$\sum_{v_{k,s} \in H^*} z_s^k \leq \frac{|H^*| - 1}{2},$$

should be added to the current LP. Moreover, one can strengthen the inequality (34) induced by the odd-hole H^* using the greedy algorithm introduced by Nemhauser and Sigismondi in [51] 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 (36) induced by the odd-hole H^* and clique C^*

$$\sum_{v_{k,s} \in H^*} z_s^k + \frac{|H^*| - 1}{2} \sum_{v_{k',s'} \in C^*} z_{s'}^{k'} \leq \frac{|H^*| - 1}{2},$$

Separation of Non-Compatibility-Clique Inequalities Consider now the inequality (39), and a fractional solution $(\bar{x}, \bar{z})$. Its associated separation algorithm consists in identifying a maximal clique C^* in the conflict graph $\tilde{G}_E^K$ s.t.

$$\sum_{v_{k,e} \in C^*} \bar{x}_e^k > 1.$$

The separation problem related to this inequality is NP-hard given that computing a maximal clique in the conflict graph $\tilde{G}_E^K$ is NP-hard problem [59]. For that, we also use the greedy algorithm introduced by Nemhauser and Sigismondi in [51] to identify a maximal clique in conflict graph $\tilde{G}_E^K$ taking into account the fractional solution $(\bar{x}, \bar{z})$ as follows. We first assign a positive weight $\bar{x}_e^k$ to each node $v_{k,e}$ in the conflict graph $\tilde{G}_E^K$. We then select a node $v_{k,e}$ in the conflict graph $\tilde{G}_E^K$ having the largest weight $\bar{x}_e^k$, and set $C^* = \{v_{k,e}\}$. After that, we iteratively add each node $v_{k',e'}$ to the current C^* if it is linked with all the nodes $v_{k,e} \in C^*$ and $\bar{x}_{e'}^{k'} > 0$. At the end, the following inequality (39) induced by the clique C^*

$$\sum_{v_{k,e} \in C^*} x_e^k \leq 1,$$

should be added to the current LP if it is violated. Furthermore, one can strengthen the additional inequality (39) by identifying a maximal clique N^* s.t. each $v_{k',e'} \in N^*$ is linked with all the nodes $v_{k,e} \in C^* \cup (N^* \setminus \{v_{k',e'}\})$ in $\tilde{G}_E^K$. For that, we use also the greedy algorithm introduced by Nemhauser and Sigismondi in [51] to identify the clique N^* as follows. We first set $N^* = \{v_{k',e'}\}$ with $v_{k',e'} \notin C^*$ a node in $\tilde{G}_E^K$ having the largest degree $|\delta(v_{k',e'})|$ in $\tilde{G}_E^K$ and should be also linked with all the nodes $v_{k,e} \in C^*$ in $\tilde{G}_E^K$. We then iteratively add each node $v_{k',e'} \notin C^* \cup N^*$ to the current N^* if it is linked in $\tilde{G}_E^K$ with all the nodes already assigned to C^* and N^*. At the end, we add the inequality (39) induced by the clique $C^* \cup N^*$ to the current LP, i.e.,

$$\sum_{v_{k,e} \in C^*} x_e^k + \sum_{v_{k',e'} \in N^*} x_{e'}^{k'} \leq 1.$$

Separation of Non-Compatibility-Odd-Hole Inequalities The separation algorithm related to the inequality (40) can be done in polynomial time by finding a minimum weighted odd-cycle in the conflict graph $\tilde{G}_E^K$ as shown by Rebennack et al. in [68] and [69]. For that, our aims is to identify an odd-hole H^* in the conflict graph $\tilde{G}_E^K$ s.t.

$$\sum_{v_{k,e} \in H^*} \bar{x}_e^k > \frac{|H^*| - 1}{2},$$

for a given fractional solution $(\bar{x}, \bar{z})$ of the current LP.
We start its procedure of separation by constructing an auxiliary conflict graph $\tilde{G}_E'^K$ by duplicating each node $v_{k,e}$ in $\tilde{G}_E^K$ (i.e., $v_{k,e}$ and $v'_{k,e}$) s.t. each two nodes are linked in $\tilde{G}_E'^K$ if their original nodes are linked in $\tilde{G}_E^K$. We assign to each link $(\tilde{v}_{k,e}, \tilde{v}_{k',e'})$ in $\tilde{G}_E'^K$ a weight $\frac{1-\bar{x}_e^k-\bar{x}_{e'}^{k'}}{2}$. After that, we compute for each node $v_{k,e}$ in $\tilde{G}_E^K$, the shortest path between $v_{k,e}$ and its copy $v'_{k,e}$. We denote this shortest path by $p_{v_{k,e},v'_{k,e}}$. Note that if the total sum of weight over edges belonging to this path is smaller than $\frac{1}{2}$, this means that there exists odd-hole H^* composed by all the original nodes of nodes belong the computed shortest path $p_{v_{k,e},v'_{k,s}}$, i.e., $V(p_{v_{k,e},v'_{k,s}}) \setminus \{v'_{k,s}\}$, s.t. its associated inequality (40) is violated by the current fractional solution $(\bar{x}, \bar{z})$ to the current LP. As a result, we add following inequality (40) induced by the odd-hole H^*

$$\sum_{v_{k,e} \in H^*} x_e^k \leq \frac{|H^*| - 1}{2}.$$

Moreover, the inequality (40) induced by the odd-hole H^* can be lifted using the greedy algorithm introduced by Nemhauser and Sigismondi in [51] by identifying a maximal clique C^* in the conflict graph $\tilde{G}_E^K$ s.t. each node $v_{k',e'} \in C^*$ should have a link with all the nodes $v_{k,e} \in H^*$, and all the nodes $v_{k''',e''} \in C^* \setminus \{v_{k',e'}\}$ in the conflict graph $\tilde{G}_E^K$. To do so, we first assign a node $v_{k',e'} \notin H^*$ to the clique C^* (i.e., $C^* = \{v_{k',e'}\}$) having the largest degree $|\delta(v_{k',e'})|$ in $\tilde{G}_E^K$, and $v_{k',e'}$ should be linked with all the nodes $v_{k,e} \in H^*$ in $\tilde{G}_E^K$. After that, we iteratively add each node $v_{k',e'} \notin H^* \cup C^*$ to the current clique C^* if it is linked in $\tilde{G}_E^K$ with all the nodes already assigned to $H^* \cup C^*$. We then add the inequality (41) induced by the odd-hole H^* and the clique C^*

$$\sum_{v_{k,e} \in H^*} x_e^k + \frac{|H^*|-1}{2} \sum_{v_{k',e'} \in C^*} x_{e'}^{k'} \leq \frac{|H^*|-1}{2}.$$

Separation of Edge-Capacity-Cover Inequalities Let's now study the separation problem of the inequality (43). Given a fractional solution $(\bar{x}, \bar{z})$, and an edge $e \in E$. The separation problem associated with the inequality (43) is Np-Hard [60] given that it consists in identifying a cover $\tilde{K}^*$ the edge e, s.t.

$$\sum_{k \in \tilde{K}^*} \bar{x}_e^k > |\tilde{K}^*| - 1.$$

To do so, we propose a separation algorithm based on a greedy algorithm introduced by Nemhauser and Sigismondi in [51]. We first select a demand $k \in K \setminus K_e$ having largest number of requested slot w_k with $\bar{x}_e^k > 0$, and set $\tilde{K}^*$ to $\tilde{K}^* = \{k\}$. After that, we iteratively add each demand $k' \in K \setminus (K_e \cup \tilde{K}^*)$ to $\tilde{K}^*$ 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 (43) induced by the minimal cover $\tilde{K}^*$ for the edge e to the current LP if it is violated, i.e.,

$$\sum_{k \in \tilde{K}^*} x_e^k \leq |\tilde{K}^*| - 1.$$

Furthermore, the inequality (43) induced by the minimal cover $\tilde{K}^*$ can be lifted by introducing an extended cover inequality (44) as follows

$$\sum_{k \in \tilde{K}^*} x_e^k \leq |\tilde{K}^*| - 1,$$

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

5.3 Primal Heuristic

Here, we propose a primal heuristic to boost the performance of our Branch-and-Cut algorithm. It is based on a hybrid method between a local search algorithm and a greedy-algorithm. Given an optimal fractional solution $(\bar{x}, \bar{z})$ in a certain node of the B&C tree, our primal heuristic consists in constructing an integral "feasible" solution from this fractional solution. To do so, we first construct several paths R_k for each demand $k \in K$ based on the fractional values $\bar{x}_e^k$ using network flow algorithms s.t. each path $p \in R_k$ satisfies the cut inequalities (2). We then use a local search algorithm which consists in generating at each iteration a sequence of demands L (order) numeroted with $L = 1', 2', ..., |K|' - 1, |K|'$. Based on this sequence of demands, our greedy algorithm selects a path p from R_k and a slot s for each demand $k' \in L$ with $\bar{z}_s^{k'} \neq 0$ and $\bar{x}_e^k \neq 0$ for each $e \in E(p)$, 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 $\bar{z}_s^{k'} = 0$ with $s \in \{w_{k'}, ..., \bar{s}\}$ and $\bar{x}_e^k \neq 0$ for each $e \in E(p)$ while respecting the non-overlapping constraint with the set of demands that precede the demand k' in the list L. Algorithm 5.3 summarizes the different steps of our greed-algorithm for a given sequence of demands.

Algorithm 1 Greedy-Algorithm for the Branch-and-Cut Algorithm

Data: A set of edges E, a spectrum $\mathbb{S}$, a multi-set K of demands, an optimal solution (x^*, z^*) of the current LP, a set R_k of precomputed feasible paths for each demand $k \in K$ based on the values ${x^*}_e^k$, 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&C, and a sequence of demands $L = 1', 2', ..., |K|' - 1, |K|'$

Result: integral solution

```
Set E_k = ∅, and S_k = ∅ for each demand k ∈ K   for each demand k' ∈ L do
    Set SERVED = FALSE  Order the set of paths in R_k' in increasing order according to the total length
     of the paths p ∈ R_k', and let R'_k' denote the set of ordered paths in R_k'  for each path p ∈ R'_k' do
        for each slot s ∈ {w_k', ..., s̄} do
            if SERVED = FALSE then
                if z_s^k' ∈ FIX_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}, and SERVED = TRUE
                    end
                end
                else
                    if z_s^k' ∉ FIX_0 and 0 < z*_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}, and SERVED = TRUE
                        end
                    end
                end
            end
        end
    end
    if SERVED = FALSE then
        for each path p ∈ R'_k' do
            for each slot s ∈ {w_k', ..., s̄} do
                if SERVED = FALSE then
                    if z_s^k' ∉ FIX_0 and z*_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}, and SERVED = TRUE
                        end
                    end
                end
            end
        end
    end
end
Let 𝒮 = ({E_k for all k ∈ K}, {S_k for all k ∈ K}) be the final solution obtained by our greedy-algorithm.
 It is feasible for the C-RSA iff E_k ≠ ∅ and S_k ≠ ∅ for each demand k ∈ K  return integral solution 𝒮
 for current node in the tree of our B&C algorithm
```

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 (i.e., $\sum_{k \in K} \sum_{e \in E_k} l_e$). 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 5.3 summarizes the different steps of our local search algorithm which calls our greedy-algorithm 5.3 at each iteration.

Algorithm 2 Primal Heuristic Based on a Hybrid Algorithm Between a Local Search Algorithm and Greedy-Algorithm for the B&C Algorithm.

Data: A set of edges E, a spectrum $\mathbb{S}$, a multi-set K of demands, and a maximum number of iterations $iter$, the maximal size of neighborhood n
Result: integral solution
Let (x^*, z^*) 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&C Let FIX_1 be the fixed variables to 1 in the current node in the tree of B&C Construct several paths R_k for each demand $k \in K$ based on the fractional values ${x^*}_e^k$ using network flow algorithms s.t. each path $p \in R_k$ satisfies the cut inequalities (2) Set $val^* = INF$, and best solution $\mathcal{S}^* = \emptyset$ Consider a sequence of demands $L = 1', 2', ..., |K|' - 1, |K|'$ Call the greedy-algorithm 5.3 based on the sequence L Let $\mathcal{S}$ be the final solution obtained by our greedy-algorithm 5.3 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 5.3 based on the sequence L_j Let $\mathcal{S}_j$ be the final solution obtained by our greedy-algorithm 5.3 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&C algorithm

6 Computational Study

6.1 Implementation's Feature

We have used C++ Programming Language to implement our B&C algorithm under Linux using three frameworks, CPLEX 12.9 [14], Gurobi 9.0 [28], and "Solving Constraint Integer Programs" (SCIP 7.0) [76] framework using CPLEX 12.9 as LP solver. It has been tested on LIMOS high-performance servers with a memory size limited to 64 Gb while benefiting from parallelism by activating 8 threads using Gurobi or SCIP (which is not possible using Cplex), and with a CPU time limited to 5 hours (18000 s).

6.2 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 [52] 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 2. Note that we tested 4 instances for each triplet $(G, K, \bar{s})$ with $|K| \in \{10, 20, 30, 40, 50\}$, and $\bar{s}$ up to 180 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 2. Characteristics of different topologies used for our experiments.

6.3 Computational Results

We first studied the impact of each family of valid inequalities introduced before on the effectiveness of our B&C algorithm using Cplex, Gurobi, and SCIP considering 4 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 violated inequalities added (Ineq_Add). 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 topologies (German, Nsfnet, and Spain). For each instance, we used Cplex with benefiting of its automatic cut generation (denoted by B&C_Cplex in the different tables), Cplex using our valid inequalities and disabling its proper cut generation (denoted by B&C_Cplex_Additional_Ineq), Gurobi with benefiting of its automatic cut generation (denoted by B&C_Gurobi), Gurobi using our valid inequalities and disabling the Gurobi proper cut generation (denoted by B&C_Gurobi_Additional_Ineq), SCIP with benefiting of its automatic cut generation (denoted by B&C_SCIP), SCIP using our valid inequalities and disabling the SCIP proper cut generation (denoted by B&C_SCIP_Additional_Ineq). Note that the gap values given in red, represent the instances solved to optimality.
The results show that the cover-based inequalities (43) and (20) are efficient than the clique-based inequalities (32), (28) and (22). Our B&C algorithm is very efficient using SCIP and Gurobi when adding the cover-based inequalities (43) and (20). We notice that adding these families of valid inequalities allows solving to optimality some instances that are not solved to optimality using B&C_Cplex, B&C_Gurobi, and B&C_SCIP. Furthermore, they allow reducing the average gap, average number of nodes, and the average CPU time. On the other hand, we observed that our valid inequalities do not work well when using Cplex. This is due to deactivating the inequalities of

the proper Cplex cut generation, and Cplex does not work well without its proper cut generation even if our valid inequalities are shown to be efficient using Gurobi and Cplex for the instances tested. The results show also that several inequalities of the cover-based inequalities (43) and (20), and clique-based inequalities (32), (28) and (22), they are generated along our B&C algorithm. However, the number of clique-based inequalities (32) generated is very less compared with other inequalities. Based on these results, we conclude that our valid inequalities are very useful to obtain tighter LP bounds using Gurobi and SCIP. 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 s.t. their separation is performed along with the B&C algorithm (using Cplex, Gurobi, and SCIP) in the following order

1. edge-capacity-cover inequalities (43),
2. edge-interval-cover inequalities (20),
3. edge-slot-assignment-clique inequalities (28),
4. edge-interval-clique inequalities (22),
5. slot-assignment-clique inequalities (32).

After that, we provide a comparative study between Cplex, Gurobi, and SCIP using the B&C (without additional valid inequalities) algorithm. To do so, we evaluate the impact of the valid inequalities used within our B&C algorithm. For this, we present some computational results using several instances with a number of demand ranges in $\{10, 20, 30, 40, 50\}$ and $\bar{s}$ up to 180 slots. We use two types of topologies: real, and realistic ones from SND-LIB already described in Table 2. Our first series of computational results presented in Table 3, concerns the results obtained for the B&C algorithm using real topologies. On the other hand, in the second series of computational results are shown in Table 4, we present the results found for the B&C algorithm using realistic topologies.

The results show that adding several families of valid inequalities is very efficient. They improve the effectiveness of our B&C algorithm compared with the last approach described in the last subsequent when adding just one family of valid inequalities within our B&C algorithm. We first notice that introducing valid inequalities allows solving several instances to optimality that are not solved to optimality using B&C_Cplex, B&C_Gurobi, and B&C_SCIP. Furthermore, they enable reducing the average number of nodes in the B&C 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 in which adding valid inequalities does not improve the results of the B&C algorithm. We further observe that using our valid inequalities within Gurobi (i.e., B&C_Gurobi_Ineq) is shown to be very efficient for the small-sized instances compared with Cplex and SCIP (see for example the Tables 3 and 4). However, and looking at the instances that are solved to optimality introducing our valid inequalities using Gurobi and SCIP, we notice that we have less number of nodes and time CPU using SCIP compared with Gurobi (see for example the Tables 3 and 4). Furthermore, there exist some instances in which introducing our valid inequalities using SCIP works much better than Gurobi s.t. B&C_SCIP_Ineq can solve several instances to optimality that are not solved using B&C_Gurobi_Ineq. Based on these results, we conclude that using our valid inequalities allows obtaining tighter LP bound.

Instances			B&C_Cplex			B&C_Cplex_Additional_Cons				B&C_Gurobi			B&C_Gurobi_Additional_Cons				B&C_SCIP			B&C_SCIP_Additional_Cons			
Topology	**K**	**S**	**Nbr_Nd**	**Gap**	**T_CPU**	**Nbr_Nd**	**Gap**	**T_CPU**	**Ncons_Add**	**Nbr_Nd**	**Gap**	**T_CPU**	**Nbr_Nd**	**Gap**	**T_CPU**	**Ncons_Add**	**Nbr_Nd**	**Gap**	**T_CPU**	**Nbr_Nd**	**Gap**	**T_CPU**	**Ncons_Add**
Conus75	10	40	1	0,00	17,27	88345,25	17,39	17169,59	17075,25	7619,25	0,00	1188,84	3960,25	0,00	878,05	1222	1	0,00	5,15	1	0,00	7,11	437
Conus75	20	40	1	0,00	44,17	33681,75	33,53	17709,16	56502	3491	0,00	1177,57	6021	0,00	2682,32	4653,75	1	0,00	12,20	1	0,00	20,26	2063
Conus75	30	40	46973,50	0,68	17440,65	7422,50	48,69	17987,09	126023	36751,75	0,56	18000	30535,75	0,19	14429,07	79730	92019	0,69	17628,22	81349,50	0,44	17670,60	83977,75
Conus75	40	40	21223,75	0,85	13057,54	3029,25	56,90	17992,33	78597,25	21442,25	0,53	13689,01	15722	0,56	14886,90	122031,50	47580,75	0,71	13352,35	39957,50	0,66	13413,09	96430
Conus75	50	80	1	0,00	187,98	3066	51,10	17989,84	44001,25	37,50	0,00	575,52	2827,50	0,00	6576,50	22585,50	1	0,00	31,29	1	0,00	167,69	13082,75
Coronet100	10	40	1	0,00	521,22	2441,75	63,74	17997,31	960,50	11159	0,00	9705,61	3886,50	187,78	18000	1188	1	0,00	36,58	1	0,00	1554,46	967,25
Coronet100	20	40	1	0,00	346,57	1284,75	47,80	18000	1584,75	2548,25	0,00	3653,55	2367	12,76	15249,39	3041,75	1	0,00	25,48	1	0,00	585,13	1945
Coronet100	30	40	15557	0,72	13321,77	9433,75	48,41	17812,14	73828,75	14412,50	0,81	13765,25	15333,25	0,23	14793,11	123638,50	40418,75	3,34	13323,08	43584,75	1,48	13334,17	85708,75
Coronet100	40	40	23015	0,84	17958,15	1228	50,40	14622,69	95882,25	29702	0,82	18000	9319,25	3,78	15224,61	134684,75	50527,50	1,17	17793,53	32372	1,71	13984,13	127457,50
Coronet100	50	80	226	0,00	462,99	3888	50,43	17991,81	52119	453,50	0,00	883,99	5281,50	22,98	11950,06	33617,75	4915	0,00	2809,11	2078,50	0,00	1436,55	18772
German	10	15	7686	0,00	165,87	9235,75	0,00	115,13	1585,25	6096,50	0,00	263,38	3709,50	0,00	200,24	2061,75	2789,50	0,00	26,66	380,75	0,00	4,33	616
German	20	45	18632,75	0,27	4454,34	61074	6,57	16447,88	92132	60521,25	0,27	4583,90	3872	0,00	644,94	10044,75	177236	0,27	3769,93	197,50	0,00	9,72	2872,50
German	30	45	7120	0,19	4471,62	18171	27,39	17972,26	93045,75	14882,50	0,20	4558,14	3279,25	0,00	967,80	27432,50	78292,50	0,95	4020,46	111610,50	0,58	3876,79	61595,75
German	40	45	66539,75	3,88	11926,56	2758,75	53,57	17989,40	86243,50	41226,25	3,56	18000	12842,75	2,15	14349,59	204155,50	283331	4,52	16287,71	220937	4,77	16671,41	271131
German	50	55	54422,25	4,69	8356,88	986,75	61,10	17995,21	69203	26208	6,01	18000	8498,25	7,01	18000	297978,50	213247,75	5,85	16706,36	181223	7,52	16879,17	240595,25
Nsfnet	10	15	1	0,00	0,44	64330,25	0,00	811,29	3717	883,25	0,00	60,74	6358	0,00	216,64	4077,75	1	0,00	0,27	1	0,00	0,17	93,25
Nsfnet	20	20	47956,75	0,70	5596,36	68010	12,88	13616,67	35484,25	84354,25	0,70	7008,50	20945,50	0,00	2384,70	34409,75	249014,75	1,14	8153,62	250663,25	0,66	6296,84	253515,25
Nsfnet	30	30	8490,25	2,47	8988,43	9515,75	38,52	17980,98	65155,25	21817,25	2,47	9172,79	41426,75	0,67	9629,51	118950,75	271523,25	2,47	7613,18	168300,25	3,79	8063,73	446796,75
Nsfnet	40	35	15382,50	1,34	8980,37	2458,75	53,36	17593,17	64946,75	34928	1,34	9175,36	28030,25	0,28	9327,03	162393,25	250109	1,34	7609,34	124003,75	1,34	8341,22	222712,50
Nsfnet	50	50	14929,25	4,33	17974,16	1225,25	58,03	16983,09	52238,75	46333,50	4,33	18000	6505,25	0,91	12468,67	279539,75	329760	4,46	16418,03	123115	7,98	17328,54	268086,50
Spain	10	15	1	0,00	1,77	322877	11,55	12904,73	9381,33	982,67	0,00	169,42	1196,67	0,00	176,07	940	1	0,00	0,88	1	0,00	0,83	301,33
Spain	20	20	85526	0,14	497,69	118123,25	22,73	17702,56	18928,50	66218,25	0,15	6434,54	1987,50	0,00	376,63	3603,50	140426,25	0,15	5298,98	67	0,00	9,64	1913
Spain	30	25	302,25	0,00	28,51	64967,25	26,52	17119,40	35464,50	3387	0,00	343,41	1622,25	0,00	541,76	4867,75	1114	0,00	71,98	1	0,00	5,31	3760,50
Spain	40	30	83910,50	4,99	10794,05	21328,50	36,50	17966,76	81144	26715	1,39	13572,18	37239,75	0,07	9771,34	90666,25	140206	2,56	12641,22	107698,50	0,29	8524,02	31380,50
Spain	50	35	36547	0,50	9956,99	14875,75	48,78	17978,25	62618,25	33518	0,34	13692,45	17801,25	0,48	10780,32	83047,75	109464,50	1,09	12855,13	82476,75	0,13	9533,33	69090

Table 3. Table of comparison between B&C : Cplex Vs Gurobi Vs SCIP using real topologies.

Instances			B&C_Cplex			B&C_Cplex_Additional_Cons				B&C_Gurobi			B&C_Gurobi_Additional_Cons				B&C_SCIP			B&C_SCIP_Additional_Cons			
Topology	**K**	**S**	**Nbr_Nd**	**Gap**	**T_CPU**	**Nbr_Nd**	**Gap**	**T_CPU**	**Ncons_Add**	**Nbr_Nd**	**Gap**	**T_CPU**	**Nbr_Nd**	**Gap**	**T_CPU**	**Ncons_Add**	**Nbr_Nd**	**Gap**	**T_CPU**	**Nbr_Nd**	**Gap**	**T_CPU**	**Ncons_Added**
Brain161	10	40	1	0,00	128,17	65187	38,90	17906,44	1347,75	6609,75	0,00	2435,26	21933	0,00	5597,98	634	4,50	0,00	77,13	1	0,00	81,21	1265,50
Brain161	20	40	1	0,00	448,11	30441,25	50,24	17950,09	1529	3589	0,00	3567,93	15472,75	0,00	7267,81	5086	59,25	0,00	379,38	58,50	0,00	508,72	11489,50
Brain161	30	40	5985,75	0,00	4954,43	21226	48,28	17962,03	3073,25	6060,50	0,00	4572,01	13578,50	2,13	11953,56	14272,75	8195,50	0,00	4779,94	559,75	0,05	917,95	12808
Brain161	40	40	7552	0,35	17986,12	14700	53,48	17973,50	20758,25	16636,25	2,88	16188,06	9346,75	7,21	18000	40068,75	17594	0,12	17929,61	6713,75	0,09	9962,17	26459
Brain161	50	40	513	5,20	17984,46	9493,75	54,85	17979,82	28305,25	10079,25	11,86	18000	7738,75	19,25	18000	50580,50	11410,25	3,17	17950,63	12637,75	1,98	17933,65	44029,75
Europe	10	25	90,25	0,00	5,74	63968,50	23,79	4652,48	93975,50	2568	0,00	360,20	1585	0,00	202,42	2507,75	233,25	0,00	11,17	17	0,00	2,01	913,75
Europe	20	60	22878,75	0,63	1569,14	14455	29,11	7094,67	68330,50	19847,25	0,65	4607,67	14995,50	0,32	595,34	32526,25	97062,50	0,65	4128,08	158	0,00	15,35	6714,25
Europe	30	80	85820,50	0,33	9021,71	4790,25	51,49	14236,41	82750	41270,25	0,63	9310,92	4694,75	0,40	6766,36	59859,25	235327,50	2,14	12690,84	120373	0,00	8544,71	77234,25
Europe	40	80	20903,75	1,14	10111,12	2798,25	53,13	17007,62	60690,75	19628,75	1,16	9352,87	6934,50	3,71	12068,74	90175	79644,50	0,98	8758,60	28205	0,65	4723,85	68977
Europe	50	180	1	0,00	82,90	1686,75	56,66	18000	21406,50	38,50	0,00	234,26	1916,50	0,00	8512,06	29754,50	1	0,00	11,40	1	0,00	259,59	11175,50
France	10	50	465	0,00	4503,78	29315,25	6,16	12206,64	70369,25	3149,50	0,00	364,45	4044,50	0,00	563,63	5220	2864,50	0,00	60,79	804,25	0,00	19,02	1212,25
France	20	60	4056	2,68	2085,90	1566,25	45,29	15492,02	109995	83688	2,60	13556,11	14293,75	0,58	5330,56	35095,50	326144	2,60	11870,51	100035,50	1,28	4046,23	24567,25
France	30	80	847	8,89	1504,94	371,25	57,84	17994,65	80452,25	29890,75	6,52	18000	9530,75	5,93	18000	186659	164306,50	6,65	17051,61	31323,25	6,73	13398,87	201121,50
France	40	100	118,25	9,60	1370,90	470,75	59,26	17995,76	54179,25	17456,75	8,39	18000	4408,50	11,31	18000	226833,50	123222	8,46	17265,88	35301,75	7,19	13532,41	222222,50
France	50	120	1	6,63	1210,43	303,50	58,54	17995,94	39008	9275,50	7,64	14166,25	3618	5,73	18000	209611,50	77080,25	4,39	13045,83	61481	5,92	13279,70	144678,50
German50	10	35	1	0,00	8,70	93433,25	24,05	17875,73	8894,75	1341,50	0,00	416,90	6041,75	0,00	735,68	1015,50	1	0,00	2,79	1	0,00	3,34	510,25
German50	20	40	1	0,00	20,93	43132	33,02	17934,72	14547,75	831,50	0,00	443,66	1440,25	0,00	648,34	2384,75	1	0,00	6,22	2	0,00	9,30	1181,50
German50	30	50	1	0,00	38,72	19691,25	44,32	17970,65	14375	274,50	0,00	358,03	1736,25	0,00	1324,33	4891,50	1	0,00	10,07	1	0,00	18,78	3025,25
German50	40	50	1	0,00	49,58	12771,25	48,18	17979,42	29231,75	318,75	0,00	417,79	2068,25	0,00	1990,94	8568,50	9	0,00	17,82	1,75	0,00	32,42	4760,25
German50	50	50	1	0,00	69,78	1934,50	52,70	17982,03	22306,25	24,50	0,00	470,86	2538,50	0,00	3329,87	12210	1	0,00	18,47	1	0,00	45,93	7618,25
Giul39	10	40	1	0,00	9,90	107155,25	21,93	17880,40	3354,50	3418,25	0,00	548,49	1582	0,00	386,72	630,25	1	0,00	2,02	1	0,00	2,06	313,50
Giul39	20	40	1	0,00	19,36	40367,25	32,20	17939,88	18911	497,25	0,00	234,52	1592	0,00	681,27	2564,25	1	0,00	3,71	1	0,00	4,78	1383
Giul39	30	40	1	0,00	29,94	25160,75	40,11	17963,32	33988	2912,75	0,00	463,92	1556,25	0,00	932,83	4964,50	1	0,00	5,96	1	0,00	8,15	2874,25
Giul39	40	40	1	0,00	43,34	15101,50	47,24	17976,69	40031	8,75	0,00	215,85	2183	0,00	1859,79	8615,50	1	0,00	7,95	1	0,00	13,38	4419
Giul39	50	40	16228	0,03	4521,14	9624,75	52,52	17982,61	49665,25	8519,50	0,03	4733,14	2710,25	0,00	2892,89	14895,50	26257	0,03	4410,46	25022,50	0,03	4412,18	20748,75
India35	10	40	109	0,00	12,76	80989,25	26,67	13367,49	35225	2190,25	0,00	715,86	1664,50	0,00	384,63	1780	369,25	0,00	19,81	1	0,00	2,34	685
India35	20	40	1	0,00	16,03	35957,25	33,12	12951,98	28203,75	1120	0,00	697,08	1634	0,00	611,17	4078,75	1	0,00	4,87	1	0,00	6,06	2049,25
India35	30	40	35274,25	0,71	7606,17	14228,75	42,96	13901,97	68667,50	30607,75	0,50	9947,09	1954	0,00	1165,86	10755,75	75905,75	0,47	8692,79	122,25	0,03	42,66	9514,25
India35	40	40	23065,25	3,24	13734,91	9669,50	inf	15359,15	83495,75	28316,75	1,27	14289,51	9527,75	0,00	7476,46	51568,50	115339	3,13	17572,47	93869	1,33	13256,99	45686,75
India35	50	80	7285,50	0,02	3456,75	4618,75	53,70	17992,28	27734,25	8838,25	0,01	4834,34	1941,25	0,00	3843,02	18504,25	27183	0,01	4396,19	32894,50	0,01	4448,39	30332,50
Pioro40	10	40	1	0,00	8,93	39,50	15,18	17864,99	1772,25	1191,50	0,00	259,61	1687,50	0,00	299,88	592,25	96475,50	0,00	1,72	1	0,00	2,32	219
Pioro40	20	40	1	0,00	22,28	85	28,97	17938,33	22683,25	4,75	0,00	165,09	1713,75	0,00	774,76	2576,75	41798	0,00	4,17	1	0,00	6,15	1366,75
Pioro40	30	40	1	0,00	38,17	128,50	50,51	17966,83	73468,75	1,50	0,00	238,47	1924,25	0,00	1273,35	5807	19802,25	0,00	6,52	1	0,00	13,02	3646,75
Pioro40	40	40	27713,25	0,11	8991,06	169,50	61,24	17981,44	85659,75	13291,75	0,11	9225,83	2888,50	0,00	2983,67	20450	9505	0,11	8776,61	683,50	0,00	173,82	21258,50
Pioro40	50	80	12143,50	0,02	4525,44	210,25	63,15	17991,72	13879	8277,75	0,02	5069,33	3594,75	0,20	7063,78	81682,50	6231,50	0,02	4440,92	17655,50	0,04	4485,61	18355,25
Ta65	10	40	1	0,00	19,24	77275,50	24,82	17894,81	2558	3514,25	0,00	572,53	3161,25	0,00	661,27	966,25	1	0,00	5,30	1	0,00	5,87	424,25
Ta65	20	40	1	0,00	50,95	32840,75	42,34	17949,83	30168,75	1288,25	0,00	796,14	2151	0,00	1108,83	3817	1	0,00	10,86	1	0,00	15,53	2280,75
Ta65	30	40	12942,25	0,02	4542,52	21154	52,94	17966,89	20274	15060,75	0,02	4818,70	2049,50	0,00	1806,49	7285,75	28333	0,02	4419,21	637	0,00	203,31	14585,25
Ta65	40	40	30250,75	0,29	13488,48	15521,75	54,52	17977,59	42733,25	33276,25	0,29	13672,92	6129,75	0,19	6748,90	52035	58523,25	0,42	13312,67	688	0,35	280,12	17037,25
Ta65	50	40	19240,50	0,35	17971,39	7159	65,09	17989,66	78757,75	13752,50	0,32	18000	13383,75	0,40	15828,17	121729,75	51167,75	0,39	17821,38	36616,25	0,31	12188,14	107263,50
Zib54	10	40	1	0,00	11,10	129403	13,57	17845,78	3168,75	1643,75	0,00	566,59	2192,50	0,00	468,23	723,75	1	0,00	2,84	1	0,00	3,62	614
Zib54	20	40	1	0,00	25,76	44022	27,40	17940,21	50448,75	754,25	0,00	342,07	3268,50	0,00	1300,90	3280,50	1	0,00	5,62	1	0,00	9,55	3530,75
Zib54	30	40	13,25	0,00	46,19	12691,25	52,81	17978,04	87701	18,50	0,00	341,26	3015,75	0,00	2028,68	13420,25	29	0,00	14,10	12	0,00	25,36	6877,25
Zib54	40	40	14227,50	0,26	4240,50	3432,25	59,21	17991,94	101917,25	2819,50	0,26	4723,54	8086,50	0,29	5928,94	58583	21728	0,26	4412,57	137,50	0,00	80,49	14435,50
Zib54	50	40	40960,50	3,02	13168,84	2274,25	65,25	17993,19	111151,25	13701	1,31	18000	14480	4,50	18000	188913,75	68023,75	3,52	17715,20	63962	2,36	17708,49	180708,25

Table 4. Table of comparison between B&C : Cplex Vs Gurobi Vs SCIP using realistic topologies.

7 Conclusion

In this paper, we focused on a complex variant of the Routing and Spectrum Assignment (RSA) problem, called the Constrained-Routing and Spectrum Assignment (C-RSA). We first proposed a new integer linear programming formulation based on the so-called cut formulation for the C-RSA. We further identified several families of valid inequalities to obtain tighter LP bounds. Moreover, we presented a separation algorithm for each valid inequality. Based on these results, we devised a Branch-and-Cut (B&C) algorithm to solve the problem. The valid inequalities are shown to be efficient and allow improving the effectiveness of our B&C algorithm. Our next step is to study the impact of the following branching strategies on the effectiveness of the B&C algorithm.

7.1 Cut Formulation Variables Branching Strategy

Here, we use the classical branching schemes. We select a variable from the variables z_s^k or x_e^k induced by a demand $k \in K$ and slot $s \in \{w_k, ..., \bar{s}\}$ or edge e having the largest value ${z^*}_s^k$ or ${x^*}_e^k$ with $0 < {z^*}_s^k < 1$ or $0 < {x^*}_e^k < 1$. Then, if a pair of demand $k \in K$ and edge e is selected, our branching algorithm generates two nodes (also called childs) by using or not the edge e to route the demand k, i.e., $x_e^k = 0$ or $x_e^k = 1$ which induces two new sub-problems. Otherwise, if a pair of demand k and slot s is selected, our branching algorithm generates two nodes by selecting or not the slot s as last-slot for the demand k, i.e., $z_s^k = 0$ or $z_s^k = 1$.

7.2 Demand-Edge-Usage Variables Branching Strategy

In this branching strategy, the variables x_e^k are priorities. It consists in branching on a variable $0 \leq x_e^k \leq 1$ for a demand k and edge e. To do so, we select a demand $k \in K$ and a edge $e \in E \setminus (E_0^k \cup E_1^k)$ having the largest value of ${x^*}_e^k$ with $0 < {x^*}_e^k < 1$. Then, we generate two nodes by imposing the usage of edge e to route the demand k or no, i.e., we create two sub-problem with $x_e^k = 0$ or $x_e^k = 1$. However, if such pair of demand k and edge e does not identified in a certain level of our algorithm, we select a variable z_s^k induced by a demand $k \in K$ and slot $s \in \{w_k, ..., \bar{s}\}$ having the largest value ${z^*}_s^k$ with $0 < {z^*}_s^k < 1$, and then generate two nodes by imposing that $z_s^k = 0$ or $z_s^k = 1$.

7.3 Demand-Slot-Assignment Variables Branching Strategy

Let us present now another branching scheme in which the variables z_s^k are priorities. It consists in branching on a variable $0 \leq z_s^k \leq 1$ for a demand k and slot s. To do so, we select a demand $k \in K$ and a slot $s \in \{w_k, ..., \bar{s}\}$ having the largest value of ${z^*}_s^k$ with $0 < {z^*}_s^k < 1$. Then, we generate two nodes by selecting or not the slot s as last-slot for the demand k, i.e., $z_s^k = 0$ or $z_s^k = 1$. However, if such pair of demand k and slot s does not identified in a certain level of our algorithm, we select a variable x_e^k induced by a demand $k \in K$ and edge $e \in E \setminus (E_0^k \cup E_1^k)$ having the largest value ${x^*}_e^k$ with $0 < {x^*}_e^k < 1$, and then generate two nodes by imposing the usage of edge e to route the demand k or no, i.e., we create two sub-problem with $x_e^k = 0$ or $x_e^k = 1$.

7.4 Demands-Dependency-Edge-Usage Constraints Branching Strategy

Here, we create dependency constraints between demands s.t. it consists in selecting two demands k, k' for an edge e having the largest value of ${x^*}_e^k + {x^*}_e^{k'}$ with $0 < {x^*}_e^k < 1$ and $0 < {x^*}_e^{k'} < 1$. Based on this, we create 4 branches by deciding if the demands k, k' pass together through the edge e or no, i.e., by adding some constraints as follows

- branch 1 by adding the constraint $x_e^k + x_e^{k'} = 0$,
- branch 2 by adding the constraints $x_e^k = 0$ and $x_e^{k'} = 1$,
- branch 3 by adding the constraints $x_e^k = 1$ and $x_e^{k'} = 0$,
- branch 4 by adding the constraints $x_e^k = 1$ and $x_e^{k'} = 1$.

However, if such pair of demands k, k' and edge e does not exist, we select a variable from the variables z^k_s or x^k_e induced by a demand $k \in K$ and slot $s \in \{w_k, ..., \bar{s}\}$ or edge e having the largest value z^{*k}_s or x^{*k}_e with $0 < z^{*k}_s < 1$ or $0 < x^{*k}_e < 1$. Then, if a pair of demand $k \in K$ and edge e is selected, we generate two nodes by deciding if the demand k uses the edge e or not, i.e., $x^k_e = 0$ or $x^k_e = 1$. Otherwise, if a pair of demand k and slot s is selected, we then generate two nodes by selecting or not the slot s as last-slot for the demand k, i.e., $z^k_s = 0$ or $z^k_s = 1$.

7.5 Demands-Dependency-Slot-Assignment Constraints Branching Strategy

Similar to what we just did in the last paragraph, we create dependency constraints between demands s.t. we select two demands k, k' for a slot $s \in \mathbb{S}$ having the largest value of $\sum_{s'=s}^{\min(s+w_k-1,\bar{s})} z^{*k}_{s'} + \sum_{s'=s}^{\min(s+w_{k'}-1,\bar{s})} z^{*k'}_{s'}$ with $0 < \sum_{s'=s}^{\min(s+w_k-1,\bar{s})} z^{*k}_{s'} < 1$ and $0 < \sum_{s'=s}^{\min(s+w_{k'}-1,\bar{s})} z^{*k'}_{s'} < 1$. Based on this, we create 4 branches by deciding if the slot s is assigned to the demands k, k' or no, i.e., we create

- branch 1 by adding the constraint $\sum_{s'=s}^{\min(s+w_k-1,\bar{s})} z^{*k}_{s'} + \sum_{s'=s}^{\min(s+w_{k'}-1,\bar{s})} z^{*k'}_{s'} = 0$,
- branch 2 by adding the constraints $\sum_{s'=s}^{\min(s+w_k-1,\bar{s})} z^{*k}_{s'} = 0$ and $\sum_{s'=s}^{\min(s+w_{k'}-1,\bar{s})} z^{*k'}_{s'} = 1$,
- branch 3 by adding the constraints $\sum_{s'=s}^{\min(s+w_k-1,\bar{s})} z^{*k}_{s'} = 1$ and $\sum_{s'=s}^{\min(s+w_{k'}-1,\bar{s})} z^{*k'}_{s'} = 0$,
- branch 4 by adding the constraints $\sum_{s'=s}^{\min(s+w_k-1,\bar{s})} z^{*k}_{s'} = 1$ and $\sum_{s'=s}^{\min(s+w_{k'}-1,\bar{s})} z^{*k'}_{s'} = 1$.

However, if such pair of demands k, k' and edge e are not found, we select a variable from the variables of our cut formulation z^k_s or x^k_e induced by a demand $k \in K$ and slot $s \in \{w_k, ..., \bar{s}\}$ or edge e having the largest value z^{*k}_s or x^{*k}_e with $0 < z^{*k}_s < 1$ or $0 < x^{*k}_e < 1$. Then, if a pair of demand $k \in K$ and edge e is selected, we generate two nodes by deciding if the demand k uses the edge e or not, i.e., $x^k_e = 0$ or $x^k_e = 1$. Otherwise, if a pair of demand k and slot s is selected, we then generate two nodes by selecting or not the slot s as last-slot for the demand k, i.e., $z^k_s = 0$ or $z^k_s = 1$.
On the other hand, we will study the impact of adding the following symmetry-breaking inequalities on the effectiveness of the B&C algorithm

Proposition 32. *Consider a demand k in K, a slot $s \in \{1, ..., \bar{s} - 1\}$. Let s' be a slot in $\{s, ..., \bar{s}\}$*

$$\sum_{s"=s'}^{\min(s'+w_k-1,\bar{s})} z^k_{s"} - \sum_{k' \in K} \sum_{s"=s}^{\min(s+w_{k'}-1,\bar{s})} z^{k'}_{s"} \leq 0. \tag{45}$$

This ensures that the slot s' can be assigned to the demand k iff the slot s (which precedes the slot s') is already assigned to at least one demand k' in K. A similar idea was proposed by Mendez-Diaz and Zabala in [48] to break the symmetry for the vertex coloring problem. Note that the inequalities (45) are not valid for the polytope $P(G, K, \mathbb{S})$ given that they cut off some feasible regions in our polytope $P(G, K, \mathbb{S})$. In any case, we ensure that there exists at least one optimal solution from our original problem that remains feasible and belongs to the convex hull of non-symmetric solutions of the C-RSA problem.

References

1. Amar, D. : Performance assessment and modeling of flexible optical networks. Theses, Institut National des Télécommunications 2016.
2. Balas, E. : Facets of the knapsack polytope. In: Journal of Mathematical Programming 1975, pp. 146-164.

3. 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.
4. 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.
5. 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.
6. Chatterjee, B.C., and Ba, S., and Oki, E. : Routing and Spectrum Allocation in Elastic Optical Networks: A Tutorial. In: IEEE Communications Surveys Tutorials 2015, pp. 1776-1800.
7. Chatterjee, B.C., and Ba, S., and Oki, E. : Fragmentation Problems and Management Approaches in Elastic Optical Networks: A Survey. In: IEEE Communications Surveys Tutorials 2018, pp. 183-210.
8. 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.
9. 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.
10. 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.
11. 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.
12. Christodoulopoulos, K., Tomkos, I., and Varvarigos, E.A. : Elastic Bandwidth Allocation in Flexible OFDM-Based Optical Networks. In: Lightwave Technology 2011, pp. 1354-1366.
13. Chudnovsky, M., and Scott, A., and Seymour, P., and Spirkl, S. : Detecting an Odd Hole. In: Journal of the ACM 2020, pp. 1-15.
14. Cplex, I.I., 2020. V12. 9: User's Manual for CPLEX. International Business Machines Corporation, 46(53), pp. 157.
15. 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.
16. 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.
17. 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.
18. Dumitrescu, I., and Boland, N.: Algorithms for the weight constrained shortest path problem. In: International Transactions in Operational Research, pp. 15-29.
19. 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.
20. Eppstein, D. : Finding the k shortest paths. In: 35th Annual Symposium on Foundations of Computer Science, pp. 154-165.
21. Fayez, M., and Katib, I., and George, N.R., and Gharib, T.F., and Khaleed H., and Faheem, H.M. : Recursive algorithm for selecting optimum routing tables to solve offline routing and spectrum assignment problem. In: Ain Shams Engineering Journal 2020, pp. 273-280.
22. Ford, L. R., and Fulkerson, D. R. : Maximal flow through a network. In: Canadian Journal of Mathematics 8, pp. 399404, 1956.
23. 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.
24. Goldberg, A.V., and Tarjan, R.E. : A New Approach to the Maximum Flow Problem. In: Proceedings of the Eighteenth Annual Association for Computing Machinery Symposium on Theory of Computing 1986, pp. 136-146.
25. 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.
26. Grötschel, M., Lovász, L., and Schrijver, A. : Geometric Algorithms and Combinatorial Optimization. In: Springer 1988.
27. Gu, R., Yang, Z., and Ji, Y.: Machine Learning for Intelligent Optical Networks: A Comprehensive Survey. In : Journal CoRR 2020, pp. 1-42.

28. Gurobi Optimization, LLC.: Gurobi Optimizer Reference Manual. In: https://www.gurobi.com, 2021.
29. 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.
30. Hadhbi, Y., Kerivin, H., and Wagler, A. : Routage et Affectation Spectrale Optimaux dans des Réseaux Optiques Élastiques FlexGrid. In: Journées Polyédres et Optimisation Combinatoire (JPOC-Metz) 2019, pp. 1-4.
31. 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.
32. 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.
33. He, S., Qiu, Y., and Xu, J. : Invalid-Resource-Aware Spectrum Assignment for Advanced-Reservation Traffic in Elastic Optical Network. In: Sensors 2020.
34. Jaumard, B., and Daryalal, M. : Scalable elastic optical path networking models. In: 18th International Conference Transparent Optical Networks (ICTON) 2016, pp. 1-4.
35. 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.
36. Jinno, M., Takara, H., Kozicki, B., Tsukishima, Y., Yoshimatsu, T., Kobayashi, T., Miyamoto, Y., Yonenaga, K., Takada, A., Ishida, O., and Matsuoka, S. : Demonstration of novel spectrum-efficient elastic optical path network with per-channel variable capacity of 40 Gb/s to over 400 Gb/s. In: 34th European Conference on Optical Communication 2008.
37. Joksch, H.C. : The shortest route problem with constraints. In: Journal of Mathematical Analysis and Applications, pp. 191-197.
38. https://lemon.cs.elte.hu/trac/lemon.
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. 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.
45. Margot, F. : Symmetry in integer linear programming. In: 50 Years of Integer Programming 1958-2008, Springer, 2010, pp. 647-686.
46. Margot, F. : Pruning by isomorphism in branch-and-cut. In: Mathematical Programming 2002, pp. 71-90.
47. Margot, F. : Exploiting orbits in symmetric ilp. In: Mathematical Programming 2003, pp. 3-21.
48. Méndez-Díaz, I. and Zabala, P. : A Branch-and-Cut algorithm for graph coloring. In: Discrete Applied Mathematics Journal 2006, pp. 826-847.
49. 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.
50. Nemhauser, G.L., and Wolsey, L.A. : Integer and Combinatorial Optimization. In: John Wiley 1988.
51. 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.
52. 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.
53. Ostrowski, J., Anjos, M. F., and Vannelli, A. : Symmetry in scheduling problems. In: Citeseer 2010.
54. Ostrowski, J., Linderoth, J., Rossi, F., and Smriglio, S.: Orbital branching. In: Mathematical Programming 2011, pp. 147-178.

55. Padberg, M.W. : On the facial structure of set packing polyhedra. In: Journal of Mathematical Programming 1973, pp. 199-215.
56. Patel, B., and Ji, H., and Nayak, S., and Ding, T., and Pan, Y. and Aibin, M. : On Efficient Candidate Path Selection for Dynamic Routing in Elastic Optical Networks. In: 11th IEEE Annual Ubiquitous Computing 2020, pp. 889-894.
57. Kaibel, V., and Pfetsch, M. E.: Packing and partitioning orbitopes. In: Mathematical Programming 2008, pp. 1-36.
58. Kaibel, V., Peinhardt, M., and Pfetsch, M. E. : Orbitopal fixing. In: Discrete Optimization 2011, pp. 595-610.
59. 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.
60. Klabjan, D., Nemhauser, G.L., and Tovey, C. : The complexity of cover inequality separation. In: Journal of Operations Research Letters 1998, pp. 35-40.
61. 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.
62. Klinkowski, M., and Walkowiak, K. : Routing and Spectrum Assignment in Spectrum Sliced Elastic Optical Path Network. In: IEEE Communications Letters 2011, pp. 884-886.
63. 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.
64. 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.
65. Panigrahi, D. : Gomory-Hu Trees. In:Encyclopedia of Algorithms 2014, Springer Berlin Heidelberg, pp. 1-4.
66. Ramaswami, R. : Optical Networks: A Practical Perspective, 3rd Edition. In: Morgan Kaufmann Publishers Inc. 2009.
67. Ramaswami, R., Sivarajan, K., and Sasaki, G. : Multiwavelength lightwave networks for computer communication. In: IEEE Communications Magazine 1993, pp. 78-88.
68. Rebennack, S. : Stable Set Problem: Branch & Cut Algorithms. In: Encyclopedia of Optimization Book 2009.
69. 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.
70. 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.
71. 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.
72. 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.
73. Salameh, B.H., Qawasmeh, R., and Al-Ajlouni, A.F. : Routing With Intelligent Spectrum Assignment in Full-Duplex Cognitive Networks Under Varying Channel Conditions. In: Journal of IEEE Communications Letters 2020, pp. 872-876.
74. 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.
75. 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.
76. 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.
77. Selvakumar, S., and Manivannan, S.S. : The Recent Contributions of Routing and Spectrum Assignment Algorithms in Elastic Optical Network (EON). In: International Journal of Innovative Technology and Exploring Engineering (IJITEE) 2020, pp. 1-11.

78. 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.
79. Schrijver, A. : Combinatorial Optimization - Polyhedra and Efficiency. In: Springer-Verlag 2003.
80. Schrijver, A. : Theory of Linear and Integer Programming. In: John Wiley & Sons, Chichester 1986.
81. 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.
82. The Network Cisco's Technology News Site: Cisco Predicts More IP Traffic in the Next Five Years Than in the History of the Internet. In: https://newsroom.cisco.com.
83. Trotter, L.E. : A class of facet producing graphs for vertex packing polyhedra. In: Journal of Discrete Mathematics 1975. pp. 373-388.
84. 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.
85. Walkowiak, K., and Aibin, M. : Elastic optical networks - a new approach for effective provisioning of cloud computing and content-oriented services. In: Przeglad Telekomunikacyjny + Wiadomosci Telekomunikacyjne 2015.
86. Wan, X., and Hua, N., and Zheng, X.: Dynamic Routing and Spectrum Assignment in Spectrum-Flexible Transparent Optical Networks. In: Journal of Optical Communications and Networking 2012, pp. 603-613.
87. Xuan, H., Wang, Y., Xu, Z., Hao, S., and Wang, X. : New bi-level programming model for routing and spectrum assignment in elastic optical network. In: Opt Quant Electron 49-2017, pp. 1-16.
88. 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.
89. Zhou, Y., and Sun, Q., and Lin, S. : Link State Aware Dynamic Routing and Spectrum Allocation Strategy in Elastic Optical Networks. In: IEEE Access 2020, pp. 45071-45083.
90. Zhu, Q., and Yu, X., and Zhao, Y., and Zhang, J.: Layered Graph based Routing and Spectrum Assignment for Multicast in Fixed/Flex-grid Optical Networks. In: Journal of Asia Communications and Photonics Conference/International Conference on Information Photonics and Optical Communications 2020 (ACP/IPOC), pp. 1-3.
91. 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.