A robust variant of the Ring Star Problem

Julien Khamphousone
julien.khamphousone@dauphine.psl.eu
Université Paris Dauphine, PSL
Paris, 75016, France

Fabián Castaño
fabian.castano@frubana.com
Frubana
Bogota, Colombia

André Rossi
andre.rossi@dauphine.psl.eu
Université Paris Dauphine, PSL
Paris, 75016, France

Sonia Toubaline
sonia.toubaline@dauphine.psl.eu
Université Paris Dauphine, PSL
Paris, 75016, France

ABSTRACT

This paper addresses a robust variant of the Ring Star Problem where we assume that at most one hub can fail, among a given subset of nodes of the network. The network should remain functional despite this failure, which means there is an edge connecting the neighbors of any hub that can fail, and every terminal is connected to two different hubs that can fail or a single hub that cannot fail. The objective is to minimize the cost of such a robust Ring Star Network structure. The problem is addressed through an integer linear programming formulation, and a Benders decomposition is proposed as an alternative solution method. Computational experiments are carried out to compare these two approaches, and the results are analyzed.

KEYWORDS

Ring star problem, Robustness, Integer linear programming, Benders decomposition.

1 INTRODUCTION

Several network design, telecommunication, transportation, and facility location problems, among many others, involve designing networks in a tributary or backbone architecture. Different designs of tributary and backbone networks have been proposed, see for instance Klincewicz[8]. In this manuscript, we consider a Ring Star network design, where a complete mixed graph with both, arcs from and to every node, as well as edges between any pair of different nodes and a particular node called the depot are given. The Ring Star Problem (RSP) introduced by Labbé et al.[10] consists in selecting a subset of nodes that includes the depot, named hubs, and link them with a cycle to form the ring. A node that is not a hub is called a terminal. Each terminal is connected to exactly one hub, forming the star topology part. The aim of RSP is to minimize the sum of three costs corresponding to (i) selecting the subsets of hubs, (ii) forming the ring, and (iii) connecting the terminals to the ring. RSP is *NP*-hard since it contains the Traveling Salsman Problem as a special case when the assignment costs are very large compared to the ring costs.

RSP has been widely studied in the literature. Labbé et al.[10] proposed a Mixed Integer Programming model, strengthened with valid inequalities resulting from a polyhedral analysis and solved with a Branch-and-Cut algorithm. Another exact approach that takes advantage of the fact that the depot must be in the ring is introduced by Kedad-Sidhoum and Nguyen[6]. Calvete et al.[1] addressed the problem using a bilevel optimization approach and proposed an evolutionary-based heuristic for solving RSP while Zang et al.[13] recently proposed an ant colony system algorithm. As a consequence of some hazardous events (failures, attacks, etc.), designing a reliable topological network might be essential. Survivable network design problems have largely been studied in the literature as a means to provide robustness to networks [4, 7].

In [11], Labbé *et al.* consider a fully connected star problem where the selected hubs form a clique. The authors investigate the polyhedral properties of the proposed model and develop a custom branch-and-cut algorithm for solving it. Fouilhoux *et al.* [3] study a 2 edge-connected star problem where the backbone structure is a 2 edge-connected subgraph (A graph is k edge-connected, for a non-negative integer k, if there are at least k edge-disjoint paths between any pair of nodes). In [5], Karaşan *et al.* consider 2 edge-connected star problems where each terminal is connected to two selected hubs. Both papers provide integer programming models and valid inequalities for the studied problems, analyze facet-defining inequalities and present exact and heuristic separation algorithms. In these works, the survivability is considered either only for the backbone network [3, 11] or for both tributary and backbone networks [5]. In all cases, the topological structure may not be preserved in the case one hub fails. Indeed, as shown in Figure 1, the backbone structure is disconnected when the hub in the central position fails.

As a consequence of some hazardous events (failures, attacks, etc.), designing a reliable topological network might be essential. Survivable network design problems have largely been studied in the literature as a means to provide robustness to networks [4, 7]. In this work, we introduce a robust version of RSP where the ring star structure should be preserved whenever a single hub fails. The notion of robustness adopted complies with the one introduced in [9].

The rest of this paper is organized as follows. Section 2 defines the robust RSP. In Section 3 an integer linear program (ILP) is proposed to model a robust version of the RSP. In Section 4, a Benders decomposition is presented as an alternative solution method. Finally, computational results are presented and discussed in Section 6.

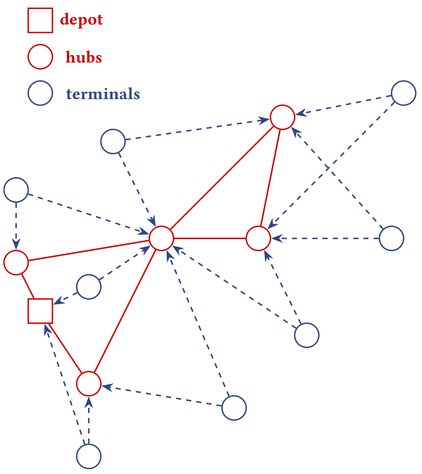

Figure 1: 2 edge-connected star (dual homed) network [5]

2 ROBUST RING STAR PROBLEM DEFINITION

In this section we first recall the RSP definition and introduce the robust version we study in this paper.

2.1 Ring Star Problem

We first recall the Ring Star Problem definition as proposed in Labbé et al.[10]. It consists of selecting a subset of nodes, called hubs, to be linked by a cycle (ring topology) and join all the terminals to the cycle (star topology). We consider a mixed graph $G = (V, E \cup A)$ with $V = \{1, 2, \dots, n\}$ a node set where node 1 is a specific node called the depot, $E = \{ij| \quad (i, j) \in V^2, \quad i < j\}$ an edge set and $A = \{(c, h)| \quad (c, h) \in V^2\}$ an arc set.

- The **Ring** part aims to select a subset $H \subseteq V$ and link up all hubs of H with a cycle using edges of E. The cost of opening a hub $i \in V$ is $o_i \in \mathbb{R}_+$ and the cost of selecting an edge $ij \in E$ between two hubs i and j is $r_{ij} \in \mathbb{R}_+$. The depot node has to be in H.
- The **Star** requires that each terminal in $T = V \setminus H$ must be connected to exactly one hub in H. The cost of selecting arc $(t, h) \in A$ to connect terminal $t \in T$ to a hub $h \in H$ is $s_{th} \in \mathbb{R}_+$.

Finally, the RSP is to design a minimum-cost Ring Star network, composed of the sum of selecting the hubs, linking them to form the ring, and connecting the terminals.

2.2 Robust Ring Star Problem

The Robust Ring Star Problem, referred to as ρ-RSP where ρ stands for *robust*, has an additional input compared to RSP: $\widetilde{V} \subseteq V$ is a possibly empty subset of nodes that can fail if they are selected as hubs. A hub in $\widetilde{V}$ is called *uncertain* because it may fail, whereas a hub in $V \backslash \widetilde{V}$ is called *certain* as it is not supposed to fail [12].

The ρ-RSP is to build a minimal cost subgraph of G that will always contain a ring-star topology even if a hub in $\widetilde{V}$ fails. By contrast with RSP, if a hub belongs to $\widetilde{V}$, an additional edge has to join its two neighbors in the ring. This edge will be used if the hub in question fails. Furthermore, each terminal is linked to the ring by either connecting it to a single hub in $V \backslash \widetilde{V}$ (if there exist one), or by connecting it to two hubs in $\widetilde{V}$. The ρ-RSP is then to design a minimal cost robust ring-star network. Thus, it can be observed that ρ-RSP reduces to RSP when $\widetilde{V}$ is empty. Figure 2 shows an illustration of the ρ-RSP with $\widetilde{V} = V \setminus \{1, 5, 6\}$.

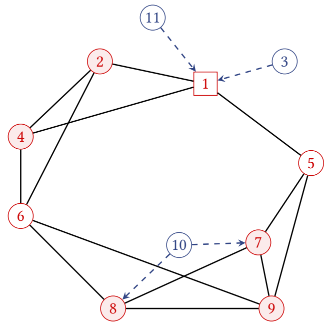

Figure 2: An instance of the ρ-RSP with $\widetilde{V} = V \backslash \{1, 5, 6\}$ and $H = V \backslash \{3, 10, 11\}$.

3 ILP FORMULATION OF ROBUST RSP

The proposed ILP formulation of ρ-RSP is based on the following decision variables: $x_{ij} \in \{0, 1\}, \ \forall (i, j) \in V^2, i < j$ is set to 1 if and only if edge $(i, j) \in E$ belongs to the ring. Note that we may sometimes refer to x_{ij} for $i > j$, in which case the actual computer implementation will simply replace x_{ij} with x_{ji}. Variable $y_{ij} \in \{0, 1\}, \ \forall (i, j) \in V^2$ is set to 1 if and only if terminal i is assigned to hub j and; y_{jj} is set to 1 if j is selected as a hub, and 0 if it is a terminal. Finally, $x'_{ik} \in \{0, 1\}$ is set to 1 if and only if i and k are hubs that have a common neighbor j that can fail in the ring (j is a hub in $\widetilde{V}$). Note that the edges for which x' is one do not need to form a cycle, for e.g. see Figure 4 and Figure 3. In Figure 3, x' and x variables equal to one are displayed. Since there are eight hubs in the ring, there are exactly eight x_{ij} nonzero variables and since there are five uncertain hubs, there are exactly five x'_{ij} nonzero variables. Finally, since the smallest ring has size 3, a robust solution must have a ring of size at least 3+1=4 in ρ-RSP unless all the hubs are in $V \backslash \widetilde{V}$. Variable σ is an integer that enforces that the ring has a size at least 4 whenever there is at least one selected hub in $\widetilde{V}$.

We will use the set of indices $\widetilde{J} = \{(i, j, k) \in V^3 : j \in \widetilde{V}, i \neq j, j \neq k, i < k\}$ and $V^{\neq} = \{(i, j) \in V^2 : i \neq j\}$ in the ILP formulation of ρ-RSP:

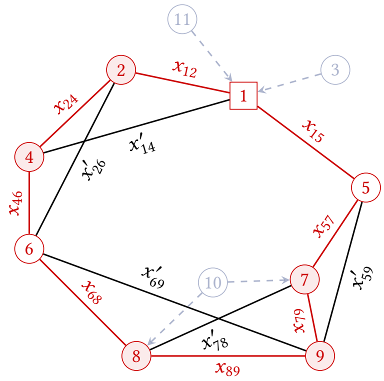

Figure 3: Same instance as in Figure 2 with x and x' variables equal to one displayed and non-hubs with lesser opacity

$$\text{Minimize} \sum_{i \in V} \sum_{\substack{j \in V \\ i<j}} r_{ij}(x_{ij} + x'_{ij}) + \sum_{i \in V} o_i y_{ii} + \sum_{i \in V} \sum_{j \in V \setminus \{i\}} s_{ij} y_{ij}$$

$$\sum_{\substack{j \in V \\ i<j}} x_{ij} + \sum_{\substack{j \in V \\ i>j}} x_{ji} = 2y_{ii} \qquad \forall i \in V \qquad (1)$$

$$|S| - \frac{1}{|V \setminus S|} \sum_{i \in V \setminus S} y_{ii} \geq \sum_{i \in S} \sum_{\substack{j \in S \\ i<j}} x_{ij} \qquad \forall S \subset V : |S| \leq \tfrac{1}{2}|V| \qquad (2)$$

$$\sum_{\substack{j \in V \setminus \widetilde{V} \\ i \neq j}} 2y_{ij} + \sum_{\substack{j \in \widetilde{V} \\ i \neq j}} y_{ij} = 2(1 - y_{ii}) \qquad \forall i \in V \qquad (3)$$

$$\sum_{i \in V} \sum_{\substack{j \in V \\ i<j}} x_{ij} \geq 3 + \sigma \qquad (4)$$

$$\sigma \geq y_{ii} \qquad \forall i \in \widetilde{V} \qquad (5)$$

$$x_{ij} + x_{jk} \leq 1 + x'_{ik} \qquad \forall (i,j,k) \in \widetilde{J} \qquad (6)$$

$$y_{ij} \leq y_{jj} \qquad \forall (i,j) \in V^{\neq} \qquad (7)$$

$$y_{11} = 1 \qquad (8)$$

$$\sigma \in \mathbb{N}$$

$$y_{ij} \in \{0,1\} \qquad \forall (i,j) \in V^2$$

$$x_{ij} \in \{0,1\} \qquad \forall (i,j) \in V^2, i<j$$

$$x'_{ij} \in \{0,1\} \qquad \forall (i,j) \in V^2, i<j$$

Constraints (1) correspond to connectivity constraints for the ring while (2) to subtour elimination constraints. Constraints (3) enforce that each terminal is connected to exactly two distinct hubs if these hubs are in $\widetilde{V}$, or to a single hub if it is in $V \setminus \widetilde{V}$. Constraints (4) and (5) state that the ring has size at least three if all hubs are in $V \setminus \widetilde{V}$, or four if at least one hub can fail. Constraints (6) enforce that for each hub $j \in \widetilde{V}$ having hubs i and k as neighbors in the ring, there is an edge that joins i and k, that can serve when j fails. There are $\frac{1}{2}\widetilde{n}(n-1)(n-2) = O(\widetilde{n}n^2)$ such inequalities where $n = |V|$ and $\widetilde{n} = |\widetilde{V}|$. Constraints (7) ensure that a terminal can only be linked to a hub. Finally, (8) force the depot to be part of the ring.

This model can be solved using a branch-and-cut approach, where subtour elimination constraints are added on-the-fly as Lazy Constraints.

4 BENDERS DECOMPOSITION OF ROBUST RSP

We can observe that whenever the ring is known (*i.e.*, the y_{ii} variables, the x_{ij} variables, and σ are fixed), determining all the other variables is an easy problem. Indeed, if the ring has 5 or more hubs, we can simply set x'_{ik} to one if and only if $j \in \widetilde{V}$, and $y_{jj} = x_{ij} = x_{jk} = 1$, and zero otherwise, and for each terminal i, we connect it either to the two nearest hubs in $\widetilde{V}$ or to the closest hub in $V \setminus \widetilde{V}$, the selected option being the one that incurs the minimum cost. Hence, we can devise a Benders decomposition whose master problem decides on the y_{ii}, x_{ij} and σ variables, whereas the subproblem is to set the remaining variables.

The master problem is:

$$\text{Minimize} \sum_{i \in V} \sum_{\substack{j \in V \\ i<j}} r_{ij} x_{ij} + \sum_{i \in V} o_i y_{ii} + \lambda$$

Subject to (1), (2), (4), (5), (8), and

$$\sigma \in \mathbb{N}$$

$$y_{ii} \in \{0,1\} \qquad \forall i \in V$$

$$x_{ij} \in \{0,1\} \qquad \forall (i,j) \in V^2, i<j$$

$$\lambda \geq 0$$

The numerical value of the x_{ij} and y_{jj} variables after solving the master problem are stored as $\hat{x}_{ij}$ and $\hat{y}_{jj}$, then passed to the subproblem, whose primal can be stated as:

$$\text{Minimize } \lambda = \sum_{i \in V} \sum_{\substack{j \in V \\ i<j}} r_{ij} x'_{ij} + \sum_{i \in V} \sum_{j \in V \setminus \{i\}} s_{ij} y_{ij}$$

$$\sum_{\substack{j \in V \setminus \widetilde{V} \\ i \neq j}} 2y_{ij} + \sum_{\substack{j \in \widetilde{V} \\ i \neq j}} y_{ij} = 2(1 - \hat{y}_{ii}) \qquad \forall i \in V \qquad (9)$$

$$x'_{ik} \geq \hat{x}_{ij} + \hat{x}_{jk} - 1 \qquad \forall (i,j,k) \in \widetilde{J} \qquad (10)$$

$$y_{ij} \leq \hat{y}_{jj} \qquad \forall (i,j) \in V^{\neq}, \qquad (11)$$

$$y_{ij} \in \{0,1\} \qquad \forall (i,j) \in V^{\neq} \qquad (12)$$

$$x'_{ij} \in \{0,1\} \qquad \forall (i,j) \in V^2, i<j$$

Where (9), (10), and (11) are derived from (3), (6), and (7) respectively.

This subproblem is easy to solve. Indeed, for all $(i,j,k) \in \widetilde{J}$ such that $\hat{x}_{ij} + \hat{x}_{jk} = 2$ we should set $x'_{ik} = 1$ if $\hat{y}_{jj} = 1$. If i is a terminal meaning $\hat{y}_{ii} = 0$, we compute m_i and m'_i as the closest hubs in $\widetilde{V}$.

- $m_i = \underset{j \in \widetilde{V} : \hat{y}_{jj}=1}{\arg\min} \; s_{ij}$ and
- $m'_i = \underset{j \in \widetilde{V} \setminus \{m_i\} : \hat{y}_{jj}=1}{\arg\min} \; s_{ij}$.

If no two such hubs exist, m_i and m'_i are set to zero. We also compute $m^\star_i$, as the closest hub in $V \setminus \widetilde{V}$: $m^\star_i = \underset{j \in V \setminus \widetilde{V} : \hat{y}_{jj}=1}{\arg\min} \; s_{ij}$. If no such hub exists, $m^\star_i$ is set to zero. Assuming that $s_{i,0}$ is infinite, if

$s_{i,m_i} + s_{i,m'_i} < s_{i,m^\star_i}$, then terminal i is connected to its two closest uncertain neighbors in the ring, otherwise i is connected to the closest certain hub in the ring. It can be observed that $s_{i,m_i} + s_{i,m'_i}$ and $s_{i,m^\star_i}$ cannot be simultaneously set to infinity, as the ring has at least three hubs.

The subproblem is originally an integer linear program, but it can be stated as a linear program by adding the following constraint: $y_{ij} \leq \sum_{k \in \widetilde{V} \setminus \{i,j\} : \hat{y}_{kk}=1} y_{ik}$ for all $i \in V$ such that $\hat{y}_{ii} = 0$, and for all $j \in \widetilde{V} \setminus \{i\}$ such that $s_{ij} = s_{im_i}$. If no such j exists, the constraint is not enforced. This constraint, labeled as (14) in the sequel, states that if terminal i is connected to an uncertain hub, then it should be connected to at least another one. Since (11) dominates (12), and because the objective function "pushes" the x' variables downward, integrality constraints can be dropped, and it can be shown that the linear relaxation of the subproblem (with the new constraints above) has an integral optimal solution. Let $\Omega = \{(i,j) \in V \times \widetilde{V} \setminus \{i\} : \hat{y}_{ii} = 0, s_{ij} = s_{im_i}\}$, the linear relaxation of the subproblem is:

$$\text{Minimize } \lambda = \sum_{i \in V} \sum_{\substack{j \in V \\ i<j}} r_{ij} x'_{ij} + \sum_{i \in V} \sum_{j \in V \setminus \{i\}} s_{ij} y_{ij}$$

$$\sum_{\substack{j \in V \setminus \widetilde{V} \\ i \neq j}} 2y_{ij} + \sum_{\substack{j \in \widetilde{V} \\ i \neq j}} y_{ij} = 2(1 - \hat{y}_{ii}) \qquad \forall i \in V \tag{13}$$

$$-y_{ij} + \sum_{k \in \widetilde{V} \setminus \{i,j\}} y_{ik} \geq 0 \qquad \forall (i,j) \in \Omega \tag{14}$$

$$x'_{ik} \geq \hat{x}_{ij} + \hat{x}_{jk} - 1 \qquad \forall (i,j,k) \in \widetilde{J} \tag{15}$$

$$-y_{ij} \geq -\hat{y}_{jj} \qquad \forall (i,j) \in V^{\neq}, \tag{16}$$

$$x'_{ij} \geq 0 \qquad \forall (i,j) \in V^2, i < j$$

$$y_{ij} \geq 0 \qquad \forall (i,j) \in V^{\neq}$$

5 DUAL FORMULATION OF BENDERS SUBPROBLEM RELAXATION

In this section, we state the dual of the linear relaxation of the subproblem, and the optimality cut that is used in the Benders decomposition. Then, we introduce a hybrid solution approach to solve the subproblem's dual. Rather than solving the master problem and the subproblem alternatively, the Benders decomposition is implemented as follows: whenever a feasible integer solution to the master problem is found, the subproblem is solved and an optimality cut is added to the master problem (if necessary). If the current integer solution to the master problem contains a subtour, we add a Benders feasibility cut, *i.e.*, a subtour elimination constraint (2).

5.1 Subproblem's dual and optimality cut

The decision variables of the subproblem's dual are associated with the subproblem's primal constraints as follows:

- Constraints (13) in the primal are associated with α_i $\forall i \in V$
- Constraints (14) in the primal are associated with δ_{ij} $\forall (i,j) \in \Omega$
- Constraints (15) in the primal are associated with β_{ijk} $\forall (i,j,k) \in \widetilde{J}$
- Constraints (16) in the primal are associated with γ_{ij} $\forall (i,j) \in V^{\neq}$

The dual of the relaxation of the subproblem is:

$$\text{Maximize } \lambda = \sum_{i \in V} 2(1 - \hat{y}_{ii})\alpha_i + \sum_{(i,j,k) \in \widetilde{J}} (\hat{x}_{ij} + \hat{x}_{jk} - 1)\beta_{ijk} - \sum_{(i,j) \in V^{\neq}} \hat{y}_{jj}\gamma_{ij}$$

$$2\alpha_i - \gamma_{ij} \leq s_{ij}, \qquad \forall i \in V \quad \forall j \in V \setminus (\widetilde{V} \cup \{i\}) \tag{17}$$

$$\alpha_i - \gamma_{ij} \leq s_{ij}, \qquad \forall i \in V, \hat{y}_{ii} = 1 \quad \forall j \in \widetilde{V} \setminus \{i\} \tag{18}$$

$$\alpha_i - \gamma_{ij} - \delta_{ij} + \sum_{\substack{k \in \widetilde{V} \setminus \{i,j\} \\ s_{ik} = s_{im_i}}} \delta_{ik} \leq s_{ij}, \qquad \forall (i,j) \in \Omega \tag{19}$$

$$\alpha_i - \gamma_{ij} + \sum_{\substack{k \in \widetilde{V} \setminus \{i,j\} \\ s_{ik} = s_{im_i}}} \delta_{ik} \leq s_{ij}, \qquad \forall i \in V : \hat{y}_{ii} = 0 \quad \forall j \in \widetilde{V} \setminus \{i\} : s_{ij} > s_{im_i} \tag{20}$$

$$\sum_{j \in \widetilde{V} : j \neq i, j \neq k} \beta_{ijk} \leq r_{ik}, \qquad \forall (i,k) \in V^2, i < k \tag{21}$$

$$\alpha_i \in \mathbb{R}, \qquad \forall i \in V$$

$$\gamma_{ij} \geq 0, \qquad \forall (i,j) \in V^{\neq}$$

$$\delta_{ij} \geq 0, \qquad \forall (i,j) \in \Omega$$

$$\beta_{ijk} \geq 0, \qquad \forall (i,j,k) \in \widetilde{J}$$

The optimality cut to be added to the master problem of the Benders decomposition is then:

$$\lambda \geq \sum_{i \in V} 2(1 - y_{ii})\alpha_i + \sum_{(i,j,k) \in \widetilde{J}} (x_{ij} + x_{jk} - 1)\beta_{ijk} - \sum_{(i,j) \in V^{\neq}} y_{jj}\gamma_{ij}$$

Even if the subproblem can be stated as a linear program, it has a cubic number of β_{ijk} variables, which makes it long to solve. In order to solve it faster, we take advantage of the fact that these variables are independent of the other variables. The next section presents a quadratic time algorithm that finds a partial optimal solution to the subproblem's dual, in the sense that it sets the β_{ijk} variables. Then, the other decision variables of the subproblem's dual are set by solving the subproblem's dual where β_{ijk} variables and constraint (21) are removed. This "light" version of the subproblem dual has $O(|V|^2)$ variables and constraints, and is solved much faster.

5.2 An algorithm for solving partially the subproblem's dual

Algorithm 1 solves partially the dual of Benders subproblem's relaxation. It takes advantage of the fact that at most $|\widetilde{V}|$ of the β_{ijk} will be non-zeros and compute them in a quadratic running time. Knowing $\hat{x}_{ij}$ and $\hat{y}_{jj}$, we initially let β'_j be the adjacency list of node $j \in \widetilde{V}$ such that $\hat{y}_{jj} = 1$: $\beta'_j[1]$ and $\beta'_j[2]$ are its two neighbors in the ring. Next we handle the special case of a four-hub ring, in which an edge joining two nonadjacent hubs may wrongly be counted twice. As an example, in Figure 4, the dashed edge between depot 1 and hub 3 must not be counted twice in the objective function as this edge preserves the ring structure when hub 2 or 4 fails. This is achieved in constant time, lines 13 to 19 in Algorithm 1: if there exist $j_1 \neq j_2$ in $\widetilde{V}$ such that $\hat{y}_{j_1j_1} = \hat{y}_{j_2j_2} = 1$ and $(\beta'_{j_1}[1], \beta'_{j_1}[2]) = (\beta'_{j_2}[1], \beta'_{j_2}[2])$, then we set β'_{j_2} to the empty set. Note that for all $(i, j, k) \in \widetilde{J}$, $\beta_{ijk} \neq 0$ implies that β'_j is nonempty (the converse does not hold when some ring costs are zero).

Algorithm 1: Building the robust edges in the dual of the subproblem of ρ-RSP

1 **Input:** $(\hat{y}_{ii})_{i \in V}$, $(\hat{x}_{ij})_{(i,j) \in V^2 \mid i<j}$ booleans
2 **Output:** $(\beta'_j)_{j \in \widetilde{V}}$
3 **foreach** $j \in \widetilde{V}$ **do**
4     $\beta'_j \leftarrow [\,]$
5 **foreach** $i \in V$ **do**
6     $\alpha_i \leftarrow 0$
7   **foreach** $j \in V : j \neq i$ **do**
8   **if** $j > i$ **and** $\hat{x}_{ij} = 1$ **and** $i \in \widetilde{V}$ **then**
9   Append j to β'_i
10   **if** $j > i$ **and** $\hat{x}_{ij} = 1$ **and** $j \in \widetilde{V}$ **then**
11   Append i to β'_j

/* Particular case of a 4-hub ring */
12 $H \leftarrow \{i \in V : \hat{y}_{ii} = 1\}$
13 **if** $|H| = 4$ **then**
14     $\widetilde{H} \leftarrow H \cap \widetilde{V}$
15   **foreach** $(j_1, j_2) \in \widetilde{H} \times \widetilde{H} : j_1 < j_2$ **do**
16   **if** $(\beta'_{j_1}[1], \beta'_{j_1}[2]) = (\beta'_{j_2}[1], \beta'_{j_2}[2])$ **then**
17             $\widetilde{H} \leftarrow \widetilde{H} \setminus \{j_2\}$
18             $\beta'_{j_2} \leftarrow \emptyset$
19 **return** $(\beta'_j)_{j \in \widetilde{V}}$

When Algorithm 1 terminates, β_{ijk} is set to r_{ik} for all $(i, j, k) \in \widetilde{J}$, and the optimality cut can be written as:

$$\lambda \geq \sum_{i \in V : \hat{y}_{ii}=0} 2(1 - y_{ii})\alpha_i - \sum_{(i,j) \in V^{\neq} : \hat{y}_{ii}=0} \gamma_{ij} y_{jj} + \sum_{j \in \widetilde{V} : \beta'_j \neq \emptyset} (x_{\beta'_j[1]j} + x_{j\beta'_j[2]} - 1) r_{\beta'_j[1]\beta'_j[2]}$$

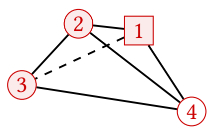

Figure 4: ρ-RSP instance with $\widetilde{V} = \{2, 3, 4\}$ with a dashed edge used when hub 2 or 4 fails.

6 COMPUTATIONAL RESULTS

The ILP formulations given in Sections 3 and 4 are addressed using a branch-and-cut approach. They have been implemented with Julia v1.5.2 and Gurobi v0.9.4 on a 16 GB RAM machine and an Intel(R) Core(TM) i7-10610U processor running at 1.80GHz.

Two types of instances have been used. First, we generated random instances with $n \in \{100, 200\}$ nodes. The nodes' coordinates are randomly drawn in $[1, n]$ for both abscissa and ordinates. The parameter $\alpha \in \{3, 5, 7, 9\}$ allows to compute the ring costs $r_{ij} = \lceil (10 - \alpha)\ell_{ij} \rceil$, for all (i, j) in E where ℓ_{ij} is the euclidean distance between $i \in V$ and $j \in V$. Opening costs o_i, for all $i \in V$ are either randomly distributed over O_i (where O_i is a random variable following the discrete distribution over the set $\mathbb{N} \cap [0.5n; 1.5n]$) or are equal to 1. Star assignment costs s_{ij} for all $(i, j) \in A$ are either randomly distributed over $\mathcal{S}_{ij}$ $\left(\text{where } \mathcal{S}_{ij} \text{ is a random variable such that } \mathcal{S}_{ij} \sim \frac{1}{\mathcal{U}([n; 3n/2])}\right)$, or defined by $s_{ij} = \lceil \alpha \ell_{ij} \rceil$, for all $(i, j) \in A$. For all random instances, we launched 5 runs and computed the average for all of them. Second, we used `eil 51` adapted from TSPLIB as in Labbé 2004 [10]. For this TSPLIB instance, we set $r_{ij} = \lceil (10 - \alpha)\ell_{ij} \rceil$, $o_i = n$. In all cases, $\widetilde{V} = V \setminus \{1\}$, and instance names are of the form `rand` $n - \alpha$ and `eil 51`$-\alpha$ in Table 1.

The formulation of ρ-RSP given in Section 3 is referred to as ILP, the one of Section 4 is referred to as Benders. Table 1 shows a comparison of ILP and Benders, with a time limit of 600 seconds per instance.

The output of our numerical results are as follows: **CPU** is the CPU Time in seconds for both methods. (TL) is indicated when the time limit is reached for at least one instance; **CPU SP** is the CPU time in seconds for Benders subproblem. Note that the master problem execution time is CPU − CPU SP; **Gap** represents the relative optimality gap of both methods computed as $\frac{|obj\_bound - obj\_val|}{|obj\_val|}$ where *obj_bound* and *obj_val* are the ILP objective bound and incumbent objective value; **n_subtour** is the number of subtour elimination constraints, *i.e.*, feasibility cuts for Benders, and Lazy Constraints for ILP; **n_cut** gives the number of optimality cuts in the Benders decomposition; and $\boldsymbol{r^*}$ corresponds to the percentage of hubs over total number of nodes in the best solution found.

We can see from Table 1 that for all random instances we generated, the Benders decomposition approach outperforms the ILP model. Those random instances are designed so that star costs are very low compared to opening and ring costs. For such instances, star costs are approximately n times smaller than the opening costs and the ring costs if opening costs are equal to 1, and n^2 smaller than the opening costs if the opening costs are randomly distributed over O_i. In these instances, we observe that we have 4 hubs in the ring in all optimal solutions. For instances of size $n = 100$, Benders is between 1.5 to 2 times faster than ILP, and for $n = 200$, the speedup lies between 4 and 5. For `eil 51`, Benders decomposition is slower. This might be partially explained because star costs are of the same order of magnitude as opening and ring costs. For such instances, the master problem does not take into account subtour elimination constraints and the contribution of star costs. Hence, a large number of feasibility and optimality cuts are required to achieve lower bounds that are competitive with the ones provided

Table 1: Comparison of ILP and Benders

Instance n–α	Instance type		ILP				Benders Decomposition					
$n = \|V\|$	o_i	s_{ij}	CPU	Gap	r^*	n_subtour	CPU	Gap	r^*	n_subtour	CPU SP	n_cut
rand 100-3	O_i	S_{ij}	38.01	0%	0.04	4.0	16.04	0%	0.04	15.2	12.01	85.8
rand 100-5	O_i	S_{ij}	30.97	0%	0.04	46.2	13.08	0%	0.04	20	9.39	66.8
rand 100-3	1	S_{ij}	30.01	0%	0.04	36.6	19.79	0%	0.04	38.4	14.78	102.2
rand 100-5	1	S_{ij}	28.8	0%	0.04	21.6	19.78	0%	0.04	53.8	15.01	109.8
rand 200-5	1	S_{ij}	349.74	0%	0.02	59.0	71.3	0%	0.02	54	53.84	76.2
rand 200-7	1	S_{ij}	324.28	0%	0.02	81.2	76.03	0%	0.02	68.4	51.59	69.6
eil 51-3	n	$\lceil 3l_{ij} \rceil$	600.00 (TL)	26%	0.92	283.0	600.00 (TL)	47%	0.8	2492.0	58.25	754.0
eil 51-5	n	$\lceil 5l_{ij} \rceil$	600.00 (TL)	29%	0.08	566.0	600.00 (TL)	36%	0.18	3442.0	62.17	1438.0
eil 51-7	n	$\lceil 7l_{ij} \rceil$	600.00 (TL)	7%	0.08	118.0	600.00 (TL)	5%	0.08	1746.0	100.89	2332.0
eil 51-9	n	$\lceil 9l_{ij} \rceil$	11.04	0%	0.08	0.0	15.41	0%	0.08	31.0	11.17	312.0

by the solver while addressing the monolithic ILP model. This issue may be mitigated by reformulating the problem objective function to let the master problem use more knowledge about star costs. This may improve the lower bounds of the linear relaxation of the master problem.

7 CONCLUSION

Future works may be focused on accelerating the proposed Benders decomposition. A first option is to replace the hybrid method for solving the subproblem's dual by a single quadratic time algorithm to separate Benders optimality cuts faster. This would also open the way for the generation of stronger optimality cuts. Indeed, there may exist many different optimal solutions to the subproblem's dual, that may lead to different optimality cuts. It would be beneficial to return the solution that yields the strongest cut, and this aspect is currently not under control when we resort to a linear programming solver for separating Benders optimality cuts. In addition, a heuristic may be devised to address the master problem as in Costa et al. [2]. This would be useful especially for large instances, that might be beyond the range of exact approaches. Finally, lower bounds could also be proposed in an attempt to address this problem with a branch-and-bound algorithm, as this class of solution approaches does not seem have been explored to tackle ring star problem variants.

REFERENCES

[1] Herminia I Calvete, Carmen Galé, and José A Iranzo. 2013. An efficient evolutionary algorithm for the ring star problem. *European Journal of Operational Research* 231, 1 (2013), 22–33.

[2] Alysson M Costa, Jean-François Cordeau, Bernard Gendron, and Gilbert Laporte. 2012. Accelerating benders decomposition with heuristic master problem solutions. *Pesquisa Operacional* 32 (2012), 03–20.

[3] Pierre Fouilhoux, O. Ekin Karaşan, A. Ridha Mahjoub, Onur Özkök, and Hande Yaman. 2012. Survivability in hierarchical telecommunications networks. *Networks* 59, 1 (2012), 37–58.

[4] Martin Grötschel, Clyde L. Monma, and Mechthild Stoer. 1995. Design of survivable networks. In *Handbooks in operations research and management Science*, C.L. Monma M.O. Ball, T.L. Magnanti and G.L. Nemhauser (Eds.). Vol. 7. Elsevier, Amsterdam, 617–671.

[5] O. Ekin Karaşan, A. Ridha Mahjoub, Onur Özkök, and Hande Yaman. 2014. Survivability in hierarchical telecommunications networks under dual homing. *INFORMS Journal on Computing* 26, 1 (2014), 1–15.

[6] Safia Kedad-Sidhoum and Viet Hung Nguyen. 2010. An exact algorithm for solving the ring star problem. *Optimization* 59, 1 (2010), 125–140.

[7] Hervé Kerivin and A. Ridha Mahjoub. 2005. Design of survivable networks: A review. *Networks* 46 (2005), 1–21.

[8] John G. Klincewicz. 1998. Hub location in backbone/tributary network design: a review. *Location Science* 6, 1-4 (1998), 307–335.

[9] Panos Kouvelis and Gang Yu. 1997. *Robust discrete optimization and its applications.* Vol. 14. Springer Science & Business Media, Springer, Boston, MA.

[10] Martine Labbé, Gilbert Laporte, Inmaculada Rodríguez Martín, and Juan José Salazar González. 2004. The ring star problem: Polyhedral analysis and exact algorithm. *Networks: An International Journal* 43, 3 (2004), 177–189.

[11] Martine Labbé, Hande Yaman, and Éric Gourdin. 2005. A branch and cut algorithm for hub location problems with single assignment. *Mathematical Programming* 102 (2005), 371–405.

[12] André Rossi, Evgeny Gurevsky, Olga Battaïa, and Alexandre Dolgui. 2016. Maximizing the robustness for simple assembly lines with fixed cycle time and limited number of workstations. *Discrete Applied Mathematics* 208 (2016), 123–136.

[13] Xiaoning Zang, Li Jiang, Bin Ding, and Xiang Fang. 2020. A hybrid ant colony system algorithm for solving the ring star problem. *Applied Intelligence* 51 (2020), 3789–3800.