

# Regenerator Location Problem and survivable extensions: A hub covering location perspective



Barış Yıldız, Oya Ekin Karaşan \*

*Bilkent University, Department of Industrial Engineering, Bilkent, 06800 Ankara, Turkey*

## ARTICLE INFO

### Article history:

Received 20 January 2014

Received in revised form 7 October 2014

Accepted 8 October 2014

Available online 12 November 2014

### Keywords:

Regenerator location

Hub location

Survivable network design

Branch and cut

## ABSTRACT

In a telecommunications network the reach of an optical signal is the maximum distance it can traverse before its quality degrades. Regenerators are devices to extend the optical reach. The regenerator placement problem seeks to place the minimum number of regenerators in an optical network so as to facilitate the communication of a signal between any node pair. In this study, the Regenerator Location Problem is revisited from the hub location perspective directing our focus to applications arising in transportation settings. Two new dimensions involving the challenges of survivability are introduced to the problem. Under partial survivability, our designs hedge against failures in the regeneration equipment only, whereas under full survivability failures on any of the network nodes are accounted for by the utilization of extra regeneration equipment. All three variations of the problem are studied in a unifying framework involving the introduction of individual flow-based compact formulations as well as cut formulations and the implementation of branch and cut algorithms based on the cut formulations. Extensive computational experiments are conducted in order to evaluate the performance of the proposed solution methodologies and to gain insights from realistic instances.

© 2014 Elsevier Ltd. All rights reserved.

## 1. Introduction

Campbell (1994a) introduces and formulates a variant of the hub covering location problem under an interesting coverage criterion, namely, that demand nodes are considered covered when served via “close” hubs mutually interconnected in a complete fashion of “close” diameter. Although this specific covering criterion did not receive much attention in the succeeding hub location literature, it is of close kinship to the problem considered in this study. Consider the following generic application setting. There are several nodes spread over a wide geographical area. Some commodity needs to be exchanged between any pair of nodes. This commodity travels in the network via pre-built links. Sent from a node, the commodity cannot travel more than a certain distance without going through a replenishment (regeneration) process, which can only be conducted at the nodes by an expensive piece of equipment called a regenerator. Because it is costly to have a regenerator at all nodes, some nodes must be chosen as centers (hubs) that serve others. Then, the problem is finding the minimum number of regenerators (and their locations) to facilitate transportation of the commodity between any two nodes. In telecommunications literature applications, this version of the hub covering problem is known as the Regenerator Location Problem (RLP) (Chen et al., 2009). In this study, we introduce two new dimensions to it:

\* Corresponding author.

- First, we examine the survivability of a network that hedges against a single failure in the regeneration equipment. In this case, we assume that only nodes with regeneration capabilities (hubs) are susceptible to failure or that the high setup and operating costs of regeneration equipment prohibit employing redundancy strategies as freely as in the case of nodes with no regenerators (non-hub nodes). We thus define the RLP with resilience against regenerator failures (RLPRF) as the problem of finding the minimum number of regenerators (and their locations) that can facilitate transportation of the commodity between any two nodes even if an arbitrary regeneration point fails.
- Second, we extend the previous survivability notion to all nodes in the network. We define the RLP with resilience against node failures (RLPNF) as the problem of finding the minimum number of regenerators (and their locations) that can facilitate transportation of the commodity between any two nodes even if any node (hub or not) in the network fails.

The original practical motivation for the RLP comes from the thriving field of optical networks (Yetginer and Karasan, 2003; Chen et al., 2009). With its vast data transfer capacities, an optical network is the only mature solution able to cope with the explosive growth in mobile communication devices (smart phones, tablets, etc.) that has taken the Internet age to a new stage (Agrawal, 2012). In 2011 global mobile data traffic was eight times the size of the whole Internet in 2000, and is expected to increase 18-fold by 2016 (Index, 2012). Such breakneck growth on the demand side has been met by the vast capacity built over the years on the supply side by fiber optic technologies. From 45 Mb/s in 1980, the data transfer capacities of fiber optic cables jumped by a factor of more than 70,000 by 2003, to reach 3.2 Tb/s. In 2010 the world record for capacity over a single fiber optic cable was set at 64 Tb/s (Agrawal, 2012). Despite their immense capacity to transmit digital data, fiber optic networks suffer from transmission impairments that limit transmission ranges. As globalization connects more people, the need to transfer more data over longer distances becomes more pronounced exacerbating the problem of signal degradation. Therefore, when designing an optical network that spans a wide geographical area, facilitating signal regeneration must be considered. Because regeneration costs make up a significant portion of a network's setup and management costs (Yang and Ramamurthy, 2005b), there is great motivation to design an optical network with few regeneration points. Although they are less costly, sparse networks in general and telecommunication networks in particular are vulnerable to damage and equipment failure. Therefore, survivability is also a big concern for optical networks designers (Monma and Shallcross, 1989; Fortz et al., 2000; Kerivin and Mahjoub, 2005).

Though the motivating application settings for the problems under our scope originate in telecommunications, we shall adopt a hub location perspective in our discussion so as to emphasize the inherent transportation nature of these problems. O'Kelly (1986a,b) and O'Kelly (1987) are seminal works on hub location research. Campbell (1994b), and more recently, Alumur and Kara (2008) provide a comprehensive survey of this literature. Campbell (1994a) studies the hub covering problem that is closely related to the RLP, defines three coverage criteria for hubs and provides the first mixed integer programming (MIP) formulations for the problem. Kara and Tansel (2000) prove that the single allocation hub covering problem is NP-hard and provide a linearization for the original quadratic model, which performs better than its previous counterparts. Wagner (2007) provides high-performance preprocessing techniques to reduce the number of the variables and improve the problem formulations. Hamacher and Meyer (2006) delineate facet-defining valid inequalities for the hub covering problem. In all these studies, the underlying hub network is assumed to be complete. Campbell et al. (2005a) relax the fundamental complete hub network assumption and instill a network design perspective to the hub location problems. In a companion study, Campbell et al. (2005b) provide integer programming formulations and optimal solution algorithms. Calik et al. (2009) and Alumur et al. (2009) also address the more general incomplete but connected hub network topologies. We refer the interested reader to Campbell (1993), Racunica and Wynter (2005), Yaman et al. (2007), Alumur et al. (2009), Yaman (2009), Correia et al. (2010), Meng and Wang (2011) for the hub location studies that directly deal with the transportation networks. None of these studies consider system survivability in the case of hub failure or destruction as a significant aspect of the problem. The current study also relaxes the fundamental complete hub network assumption and builds hub networks resilient to hub or node failures.

Kim and O'Kelly (2009) introduce the reliable  $p$ -hub location problems in hub-and-spoke networks. Using the probabilities of successful edge or hub flow transmissions as reliabilities, the reliability of the network performance can be measured. Kim and O'Kelly (2009) formulate and solve two hub location models namely the  $p$ -hub maximum reliability and the  $p$ -hub mandatory dispersion models focusing on maximizing the network performance in terms of reliability based on empirical traffic loss rates among origin destination pairs. Both single and multiple allocation versions of the problem are addressed. Considering hub unavailability and alternative routes in air transportation systems, Zeng et al. (2010) propose different versions of reliable hub location models. Davari et al. (2010) and Zarandi and Davari (2011) design reliable hub networks using fuzzy goal programming. Lei (2013) and Hamidi et al. (2014) utilize a hub interdiction viewpoint and present hub protection and preventive reliable hub location problems, respectively, to the literature. Kim and O'Kelly (2008), An et al. (2011), Kim (2012) and Azizi et al. (2014) bring the survivable network design perspective into the hub-and-spoke networks. With this perspective, backup hubs and alternative routes are designed to provide a continuum of service with a typical objective of minimizing the transportation cost. In our study, the fundamental complete hub network assumption, which is present in the mentioned relevant studies in survivable hub location literature, is relaxed. Moreover, the designs should obey the transport range (optical reach) limitations respecting the edge lengths of an input transportation (optical) network. In particular, for RLP, given an underlying network (typically a sparse one), hubs (regenerators) should be located in such a way that between any origin destination pair, there exists a path visiting perhaps more than two hubs such that each segment of this path is within the transport range. With RLPRF, the hub network design should respect RLP connectivity requirements even if

an arbitrary hub node fails. Finally, with RLPNF, the hub network design should respect RLP connectivity even if any node in the original network fails. In other words, for the survivable versions, connectivity induced by the original network after the particular failure should be considered. On the other hand, our designs as per the application requirements are sought solely from the viewpoint of enabling communication under failures and transportation costs are overlooked.

There is a rich literature on network survivability from a topological design viewpoint. Parallel to the resilience criteria we use in our model, [Grotschel et al. \(1995\)](#) argue that two-connected networks are able to provide a sufficient level of survivability because in most cases, the probability of two nodes failing at the same time is significantly small. This resilience criterion is widely accepted throughout the operations research literature; the reader can refer to [Steiglitz et al. \(1969\)](#), [Monma and Shallcross \(1989\)](#), [Fortz et al. \(2000\)](#) and [Lau et al. \(2009\)](#) for relevant studies and detailed surveys on this topic. Similar to our approach, [Chimani et al. \(2008\)](#) show that supported by a strong MIP formulation, the branch and cut procedure can efficiently solve two-node-connected Steiner network problems of realistic sizes. More recently, [Kerivin and Mahjoub \(2005\)](#) investigate techniques used to solve survivable network design problems where the measure of robustness is modeled in terms of node or link connectivity. In our context we find it more relevant to build resilience against node failures rather than on link (edge) failures because in applications where nodes represent complex processes (regeneration, charging, refueling, amplification, encryption, etc.), nodes can be much more prone to malfunction, expensive to build redundancy into or more likely to be targets of deliberate attacks.

Regarding the RLP literature, [Yetginer and Karasan \(2003\)](#) are the first to introduce the sparse regenerator placement in a static routing environment. The authors take network survivability into account and provide heuristics to place the regenerators, giving one working and one restoration path between any two nodes. However, they establish survivability by edge disjointness, which is not adequate when the context dictates node disjointness as a survivability criterion. The paper works under the simple path assumption which unnecessarily limits the solution space and hence may result in suboptimal solutions. [Konak \(2014\)](#) is a recent work dealing with the same design problem and providing heuristic solution methodologies.

Taking the geographical aspect of the RLP into account, [Chen et al. \(2009\)](#) introduce it to the operations research literature, proving its NP-completeness and showing that it can be modeled as a special Steiner arborescence problem (SAP). Using this fact, they formulate a MIP model that can provide optimal solutions for relatively small instances of the problem. They adapt a branch and cut algorithm tailored for the SAP as a solution methodology, however, because the solution times are prohibitive for large instances of the RLP, they offer three heuristics and test the performances with an extensive experimental study. Similar to this study, we formulate the RLP as a MIP and solve it with a branch and cut procedure. However, [Chen et al. \(2009\)](#) first transform the RLP to a SAP and then apply branch and cut. This transformation necessitates duplicating the nodes in the original input graph and hence prolongs the solution times. This does not occur in our formulation because we apply the branch and cut procedure directly to the original graph. Moreover, with the hub viewpoint, we are able to apply the separation algorithms in the restricted hub subgraph which leads to further efficiency. The maximum leaf spanning tree problem (MLSTP) and the minimum connected dominating set problem (MCDSP) are equivalent problems to the RLP. Using this equivalence, [Sen et al. \(2008\)](#) provide another heuristic for the RLP. [Lucena et al. \(2010\)](#) provide branch and cut solution methodologies for the MLSTP and MCDSP and ultimately for the RLP. In a recent study, [Gendron et al. \(2014\)](#) propose efficient exact solution methodologies built on Benders decomposition and branch and cut techniques. Our experimental results prove that none of these exact solution methodologies can compete with the branch and cut algorithm developed in this study which has the advantage of working only in the hub network due to the hub location perspective we adopt.

We take the geographical aspect of the RLP as the focal point of our study. The interested reader can refer to [Savasini et al. \(2007\)](#), [Ouyang et al. \(2005\)](#), [Yang and Ramamurthy \(2005a\)](#) and [Saradhi and Subramaniam \(2009\)](#) for studies that investigate the RLP with a focus on different aspects such as blocking probabilities in dynamic optical networks with wavelength division multiplexing.

Although the literature on survivable network design and the RLP is extensive, few studies consider the survivability of networks where connectivity is established by using regeneration nodes as prescribed in this paper. In particular, studies such as [do Forte et al. \(2013\)](#) on the two-connected dominating set problem have a much weaker survivable connectivity requirement. We believe our study is an important step in closing this gap. Our contributions are twofold.

1. Incorporating the hub location perspective to the problem, we relate RLP to various transportation settings and introduce the problems RLPRF and RLPNF to the literature.
2. We study all three versions of the RLP under a unifying solution methodology, where for each problem we provide a compact flow formulation and a cut formulation suitable for efficiently utilizing a branch and cut procedure. Indeed, the performances of our exact algorithms challenge the best heuristic algorithms available in the literature.

### 1.1. Applications/limitations in transportation settings

In their insightful analysis of the last twenty-five years of hub location research, [Campbell and O'Kelly \(2012\)](#) draw attention to the differences of transportation and telecom hub networks. These entities though seemingly similar in the abstract level of mathematical models could be quite different in their operations, service measures, costs and constraints due to the inherent differences of the transported objects. The lack of costs in pure service-oriented hub center and hub covering models have found more reasonable applications in telecom than in transportation. Confirming this analysis of [Campbell and O'Kelly \(2012\)](#), RLP literature discussed in this paper as well as a closely related network design problem with relays of

Cabral et al. (2007), Laporte and Pascoal (2011) and Kewcharoenwong and Uster (2014) seem to be confined to the field of telecommunications. In this regard, we shall attempt to describe transportation-related settings and motivations where RLP and its close variants can be of interest.

Regenerators with their potential of extending the transport range via rejuvenation mainly adopt the connectivity feature (in the terminology of Campbell and O'Kelly (2012)) as hubs. Perhaps the closest transportation related application setting to RLP is the relay network design for full truckload transportation. Relay points in transportation networks are physical locations where various activities like the exchange of drivers, trucks and trailers can take place. Drivers can rest or switch transportation means or simply leave the consignment at relay points to be picked up by other drivers. The relay network design problem in freight transportation systems as studied in Ali et al. (2002) is almost identical with RLP. The hub-and-spoke aspect of relay networks has originated with the work of Taha and Taylor (1994) and has been adopted as an alternative dispatching method for TL transportation in order to alleviate the common driver retention problem in several studies (please see Taylor et al. (1999), Ali et al. (2002), Uster and Maheshwari (2013), Uster and Kewcharoenwong (2011), Konak (2012), Vergara and Root (2013) and the references therein). Though relay networks and hub-and-spoke networks (not necessarily complete hub networks) are very similar, there are operational limitations such as circuitry, distances between relay points, load balance at relay points, and number of handlings from origin to destination that are specific to TL transportation in relay networks. Incorporating some of these limitations within an exact solution methodology for relay network designs has been the challenge in works of Uster and Maheshwari (2013), Uster and Kewcharoenwong (2011) and Vergara and Root (2013) who manage to achieve optimal or near-optimal results for moderate sized problem instances. Our RLP designs from a relay network perspective inherently satisfy the spatial requirements among the relay points though overlook other requirements. In applications where the demand is dynamic and dimensions are high, our methodology for RLP might prove useful in the strategic decision of a relay network design. Consider a logistics company planning to receive parking/switching/berthing services from available rest areas scattered throughout the country. Once the best locations for relays are determined serving to needs of all potential pairs of demands, a second stage optimization phase might be called for to tackle the more dynamic operational decisions of routing the specific trucks (point-to-point or through relays) in an effective manner so as to minimize circuitry, imbalance, and number of handlings. Our survivable designs might bring further flexibility to this end during this second stage optimization phase.

Another transportation setting where RLP or its close variants can be of practical use is that of finding refueling station locations for alternative-fuel vehicles (AFVs). Vehicle range is the key factor for determining the locations and number of refueling stations for completing a journey, especially a long-distance one. The location of switching/recharging stations for accommodating electric/hybrid vehicles has been the scope of recent research (including but not limited to Kuby and Lim (2005), Wang and Lin (2009), Wang and Wang (2010), MirHassani and Ebrazi (2012), He et al. (2013), Nie and Ghamami (2013), Xi et al. (2013)). It is not hard to see that the refueling station placement problem with its inherent coverage and range limitations is closely related with the RLP. However, the former problem has some dimensions which RLP lacks in its current form. For example in RLP establishing a feasible (regeneration-enabled) route between two nodes is considered sufficient when meeting the connection demand. However, the physical length of the regeneration-enabled-routes might be unacceptable for a typical AFV driver. The current study can still be useful in several ways in this application setting as well. From the practical point of view, providing much more alternative regeneration-enabled-routes, RLPRF or RLPNF solutions can enable driver friendly routes and constitute a reasonable solution for the refueling station placement problem. Once the strategic decision of locations of the refueling stations is made from the viewpoint of enabling the communication of all source destination pairs, individual optimizations can be carried out either to find the minimum cost path (in terms of refueling decisions) or the shortest path for a specific source destination pair. From an algorithmic point of view, RLP solution which can be attained very fast for high dimensions can provide a good lower bound for this specific transportation application. Another limitation of RLP might be the implicit assumption of ample capacity of the regenerators allowing them to handle any incoming flow. This might not be the case for a refueling station that can simply serve a certain number of vehicles simultaneously. Again, a load-balancing optimization can be called for as a second phase optimization in order to make the best use of designs provided by RLP/RLPRF/RLPNF in the first stage and determine the proper capacity allocation to the designated refueling stations jointly with the routing decisions. Therefore, having an efficient algorithm to solve the RLP can be useful to develop more realistic models to study the refueling station location problem.

Optimal deployment of combat logistics units in a military campaign and air-to-air refueling for unmanned aerial vehicles (or refueling for unmanned underwater vehicles) are just a few other examples where RLP-related problems can arise in transportation settings. In these applications, the objective is to facilitate the constant movement of combat units or sophisticated surveillance equipment that needs some kind of replenishment to continue functioning, while keeping the number of logistics support units at a minimum. Due to either tactical or physical restrictions, the set of candidate locations to place such logistic units is usually a finite one which sets a context very similar to the one assumed by RLP. In the presence of an intelligent adversary who actively targets the logistics network, survivability is even more important, and the need for solving the versions RLPRF and RLPNF is more apparent.

A futuristic transportation application for RLP and its variants could be prime air delivery settings utilizing unmanned aerial vehicles like the drones of Amazon (Ozkok, 2014). Drones have limited reach capabilities beyond which they have to be recalibrated/recharged. One can envision a hub-and-spoke network with a covering aspect similar to that of RLP when designing a network including depots, recharging locations of drones and customers.

The location problems tackled in this study employ cover type hub location requirements and ignore costs. In this regard, in their original form, they may be more realistic in public sector and military applications. However, we believe, our effective methodologies developed in this paper can be a first step in solving more realistic and larger hub location instances from transportation. Especially when survivability is a concern, our findings in Section 3 suggest that separating the location and routing decisions might be one viable way to tackle such challenging problems.

## 2. Mathematical models

In this section, we provide some background information on the RLP, establish the NP-completeness results of the two new problems introduced and provide the notation that will be adopted throughout the text. For each of the three problems under the scope of this paper, a compact formulation and a cut formulation are proposed. We shall use hubs and regenerators interchangeably throughout the text.

### 2.1. Notation and background

Throughout the text  $G = (V, E, w)$  will be our input weighted graph with node set  $V = \{1, 2, \dots, n\}$ , edge set  $E, w_e > 0$  the length of edge  $e \in E$ . We shall assume that  $d_{max} > 0$  is the given threshold of regeneration-free communication, i.e., the transport range or the optical reach. In other words, two nodes of distance at most  $d_{max}$  in  $G$  can communicate without any need for regeneration. We assume without loss of generality that  $w_e \leq d_{max}$  for every  $e \in E$  since any edge violating this condition can simply be deleted from  $G$ . Let  $A = \{(i, j) \cup (j, i) : \{i, j\} \in E\}$  be the arc set induced by the edges in  $G$ . We assume that  $w_{ij} = w_{ji} = w_e$  for every  $e = \{i, j\} \in E$ . Let  $d_{ij}$  be the length of the shortest path from  $i$  to  $j$  in  $G$ . After solving an all pairs shortest path problem in  $G$ , we create the unweighted closure graph  $(V, E^c)$ , where the edges correspond to pairs of nodes that can communicate with no need for regeneration, i.e.,  $E^c = \{\{i, j\} : i, j \in V, i \neq j, d_{ij} \leq d_{max}\}$ . Let  $A^c = \{(i, j) \cup (j, i) : \{i, j\} \in E^c\}$  be the set of arcs induced by the edges in the closure graph. A similar construction, under the name of “communication graph”, is introduced to the literature in Chen et al. (2009). Throughout the text the notation  $[G]^c$  will correspond to the operation of “taking the closure” of an input graph  $G = (V, E)$ . In other words,  $[G]^c = (V, E^c)$ .

For two disjoint subsets  $S$  and  $T$  of  $V$ , we denote by  $[S, T]$  the set of edges with one endpoint in  $S$  and the other in  $T$ . For  $S \subseteq V$ , let  $\delta(S) = [S, V \setminus S]$  and  $E(S)$  be the set of edges in  $E$  with both endpoints in  $S$ . We use  $G(S)$  to denote the subgraph induced by  $S$ , i.e.,  $G(S) = (S, E(S))$ . Let  $V_p = V \setminus p, G_p = G(V_p), G^c = [G]^c = (V, E^c)$  and  $G_p^c = [G_p]^c = (V_p, E_p^c)$ . Note that  $E_p^c \subseteq E^c(V_p)$  as the closure edges of  $E^c$  induced by node  $p$  are not present in  $E_p^c$ .

A given regenerator or hub set  $R \subseteq V$  and  $d_{max}$  jointly induce a restricted form of connectivity in  $G$ , namely,  $s, t \in V$  are  $R$ -connected if there exists a walk  $\mathcal{P}$  from  $s$  to  $t$  on which any two consecutive nodes in  $\{s, t\} \cup R$  are at most  $d_{max}$  apart. Such a walk in  $G$  corresponds to a path in  $G^c$  where every internal node is in set  $R$ .

The decision versions of the three problems studied in this paper can be formalized as follows.

Given  $G = (V, E, w), d_{max}$  value, and  $k \leq n$ , does there exist a regenerator set  $R \subseteq V, |R| \leq k$  such that:

- RLP Regenerator Location Problem: Any node pair is  $R$ -connected in  $G$ ?
- RLPRF Regenerator Location Problem under Regenerator Failures: Any node pair is  $R$ -connected in  $G_p$  for every  $p$  in set  $R$ ?
- RLPNF Regenerator Location Problem under Node Failures: Any node pair is  $R$ -connected in  $G_p$  for every  $p$  in set  $V$ ?

With the notion of  $R$ -connectivity non simple paths might be more advantageous. In the following example (Fig. 1) with  $d_{max} = 5$ , the RLP has an optimal solution of one regenerator located at node 4. In particular, the connection between nodes 1

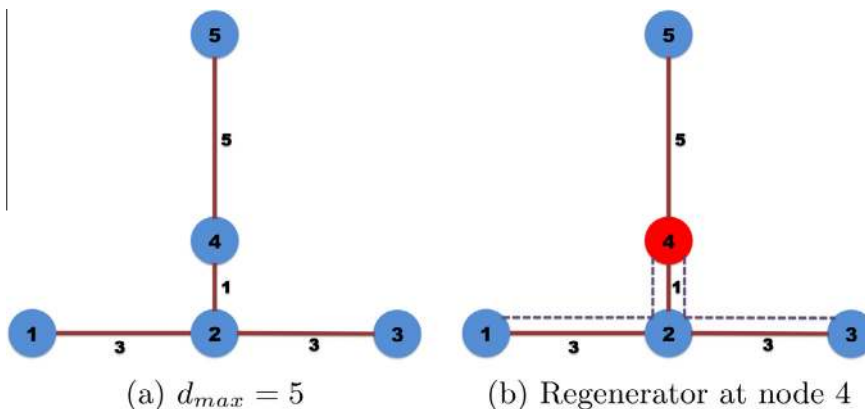


Fig. 1. Non simple paths.

and 3 can be established by using the walk 1-2-4-2-3 and any solution requiring simple path connectivity results in a higher number of regenerators.

RLP is shown to be NP-complete in Chen et al. (2009). This result can also be derived from the NP-completeness proof in Kara and Tansel (2000). The following theorem is an immediate consequence of this result.

**Theorem 1.** *RLPRF and RLPNF are NP-complete.*

**Proof.** Both problems are in NP because checking for the feasibility of a given regenerator subset  $R$  for a specific origin-destination pair, and ultimately for all pairs, can be accomplished in polynomial time.

Let  $G = (V, E, w)$  with  $V = \{1, \dots, n\}$ ,  $d_{\max}$  and  $k$  be an instance of RLP. Let  $V' = \{i_1 \cup i_2 : i \in V\}$  and  $E' = \{\{i_1, j_1\} \cup \{i_2, j_2\} : \{i, j\} \in E\} \cup \{\{i_1, i_2\} : i \in V\}$ . Furthermore, let  $w'$  be the cost vector associated with  $E'$  edges.  $w'_{\{i_1, j_1\}} = w'_{\{i_2, j_2\}} = w_{\{i, j\}}$  for all  $\{i, j\} \in E$  and  $w'_{\{i_1, i_2\}} = 0$  for all  $i \in V$ . Let  $G' = (V', E, w')$ ,  $d_{\max}$  and  $2k$  be an instance of RLPRF(RLPNF). It is easy to note that the RLP has a yes answer if and only if the RLPRF(RLPNF) has a yes answer.  $\square$

We assume that the input graph  $G$  is two-connected. To eliminate trivial cases in our formulations we shall also assume that the input networks necessitate the use of at least three regenerators. This is not a binding assumption because for each of our problem definitions, it is easy to enumerate all possible hub sets of size at most two and check for feasibility in polynomial time.

### 2.2. Regenerator Location Problem: compact formulation

In this subsection, we provide new compact and cut formulations for the RLP. Within the framework of hub covering, given the closure graph  $G^c = (V, E^c)$ , we seek a (hub) set  $R \subseteq V$  of minimum cardinality such that  $G^c(R)$  is connected (not necessarily in a complete fashion) and that for every node  $i \in V \setminus R$ , there exists a node  $j \in R$  such that  $\{i, j\} \in E^c$ , i.e., every demand node is covered by at least one hub, i.e., multi-allocation is allowed in hub location terminology. In this fashion, we guarantee that any two nodes in  $V$  are  $R$ -connected. In other words, solving RLP in  $G^c$  is equivalent to solving the hub covering location problem with the “closeness” criterion of Campbell (1994a) in  $G^c$  with the relaxation of the complete hub network assumption. Note that the design is only pertinent to the choice of hub locations. Once such locations are fixed, the union of closure edges interconnecting these locations, i.e.,  $E^c(R)$  make up the hub network.

In the compact formulation for the RLP, we shall require a tree rooted at a chosen regenerator node visiting all other regenerator nodes using only the edges in  $E^c$  to guarantee the connectivity of the hub network. We define the following decision variables.

$$s_i = \begin{cases} 1, & \text{if node } i \in V \text{ is the chosen root regenerator} \\ 0, & \text{otherwise,} \end{cases}$$

$$r_i = \begin{cases} 1, & \text{if node } i \in V \text{ is a regeneration point} \\ 0, & \text{otherwise,} \end{cases}$$

$x_{ij}$  = amount of flow on arc  $(i, j) \in A^c$  in the designed tree.

We would like the flow variables to satisfy the following balance equations.

$$\sum_{j:(i,j) \in A^c} x_{ij} - \sum_{j:(j,i) \in A^c} x_{ji} = \begin{cases} \sum_j r_j - 1, & \text{if node } i \in V \text{ is the chosen root regenerator} \\ -1, & \text{if node } i \in V \text{ is a non - root regenerator node} \\ 0, & \text{if node } i \in V \text{ is not a regenerator node.} \end{cases}$$

These relations can be induced by the following nonlinear form of the flow balance equations.

$$\sum_{j:(i,j) \in A^c} x_{ij} - \sum_{j:(j,i) \in A^c} x_{ji} = \left( \sum_j r_j - 1 \right) s_i - r_i(1 - s_i) \quad \forall i \in V.$$

The formulation for the RLP is as follows.

$$\min \sum_{i \in V} r_i \tag{1}$$

$$\text{s.t. } \sum_{i \in V} s_i = 1 \tag{2}$$

$$s_i \leq r_i \quad \forall i \in V, \tag{3}$$

$$\sum_{j:(i,j) \in E^c} r_j \geq 1 \quad \forall i \in V, \tag{4}$$

$$\sum_{j:(i,j) \in A^c} x_{ij} - \sum_{j:(j,i) \in A^c} x_{ji} \geq \sum_j r_j - 1 - (n-1)(1-s_i) \quad \forall i \in V, \quad (5)$$

$$\sum_{j:(i,j) \in A^c} x_{ij} - \sum_{j:(j,i) \in A^c} x_{ji} \geq -r_i \quad \forall i \in V, \quad (6)$$

$$\sum_{i:(i,j) \in A^c} x_{ij} \leq (n-1)r_j \quad \forall j \in V, \quad (7)$$

$$r_i, s_i \in \{0, 1\} \quad \forall i \in V, \quad (8)$$

$$x_{ij} \geq 0 \quad \forall (i,j) \in A^c. \quad (9)$$

The objective function (1) is the number of nodes that need to be equipped with the regeneration property. Constraints (2) and (3) force one of the regenerator nodes to be chosen as the root node. Due to constraints (4), any node (demand or hub) will be connected to the hub network in the closure graph. Because we assume the hub network consists of at least three hubs, this constraint is valid for hub nodes as well. Constraint set (5) and (6) provides a linearization for the flow balance equations. It forces the chosen source regenerator to have a net outgoing flow amount equal to the number of remaining regenerator nodes and any other regenerator node to have a net incoming flow value of one unit. The constraints become redundant if node  $i$  is not chosen as a regenerator. The  $x$  variables may take on positive values only in the designed hub network. In this fashion, it is guaranteed that a path will initiate from the root hub to every other hub in the closure graph. If an arc of the closure graph is utilized in any one of these paths, constraints (7) force the head node to become a regenerator point. Constraints (8) and (9) are variable restrictions.

### 2.3. Regenerator Location Problem: cut formulation

To facilitate the use of a branch and cut algorithm in the solution methodology, we provide an equivalent cut formulation for the RLP. The following decision variables will be used.

$$x_e = \begin{cases} 1, & \text{if edge } e \in E^c \text{ is present in the (designed) regenerator subgraph} \\ 0, & \text{otherwise,} \end{cases}$$

$$r_i = \begin{cases} 1, & \text{if node } i \in V \text{ is a regeneration point} \\ 0, & \text{otherwise.} \end{cases}$$

Note that there is no design with respect to the  $x$  variables; the choice of the hub nodes defines these variables uniquely. In other words, we have the following nonlinear relationship.

$$x_{ij} = r_i r_j \quad \forall \{i,j\} \in E^c. \quad (10)$$

Similar to the compact formulation, the cut formulation will force the hub network to be connected and every demand node to be covered by some hub in this subgraph.

$$\min \sum_{i \in V} r_i \quad (11)$$

$$\text{s.t.} \quad \sum_{j:\{i,j\} \in E^c} r_j \geq 1 \quad \forall i \in V, \quad (12)$$

$$x_{ij} \leq r_i \quad \forall \{i,j\} \in E^c, \quad (13)$$

$$x_{ij} \leq r_j \quad \forall \{i,j\} \in E^c, \quad (14)$$

$$x_{ij} \geq r_i + r_j - 1 \quad \forall \{i,j\} \in E^c, \quad (15)$$

$$x(\delta(S)) \geq r_i + r_j - 1 \quad \forall S \subset V, i \in S, j \notin S, \quad (16)$$

$$x_e \in \{0, 1\} \quad \forall e \in E^c, \quad (17)$$

$$r_i \in \{0, 1\} \quad \forall i \in V. \quad (18)$$

The number of regenerator nodes is minimized through (11). Every node  $i$  should be able to reach the regenerator subgraph via a closure edge (valid due to our assumption of having more than one regenerator in the resulting design), and this is forced with constraints (12). Constraints (13)–(15) linearize the relationship (10). Constraints (16) are the cut constraints. If  $S$  and  $V \setminus S$  induce a partition of  $V$ , each having a regenerator node, then there should be at least one edge crossing this partition in the regenerator subgraph for connectivity. Finally, constraints (17) and (18) force the required variable restrictions. It is enough to force the integrality of the  $r$  variables only since they imply the integrality of the  $x$  variables.

We would like to remark that both the compact and the cut formulations provided above are novel in the RLP as well as the hub location literature. In particular, our single-commodity flow formulation is different from its multi-commodity counterpart in Alumur et al. (2009) since the hub network is induced by the closure edges and there is no need to keep the connectivity information explicitly. Similarly, the cut formulation is quite different from its counterpart in (Chen et al., 2009) where a rooted Steiner tree is sought on a transformed graph rather than a rooted tree on the closure graph. Indeed, we believe that the differences in the performances of the algorithms are merely due to the differences in the dimensions of the underlying input graphs.

2.4. Regenerator Location Problem under regenerator failures: compact formulation

Let  $R \subseteq V$  be the set of nodes selected for the regeneration property. Under the RLPRF, we need to guarantee that the removal of any node in  $R$  will not destroy connectivity. Unfortunately, the closure graph utilization for this problem is not as direct as it is for the RLP. Namely, simply forcing the resulting hub network design to be two-connected and every demand node to be covered by at least two hubs will not be sufficient, due to the information loss that will occur in the transfer from the original graph to the closure graph. The following example is included to clarify this issue. For the original graph depicted in Fig. 2a, consider the closure graph induced by  $d_{max} = 5$  and nodes 2, 3 and 6 as hubs. The resulting hub network is two-connected and every demand node is connected to at least two hub nodes as seen in Fig. 2b. However, in  $G_3^c$ , demand node 4 is disconnected from the hub network and hence  $\{2, 3, 6\}$  is not a feasible solution for the RLPRF. Indeed, the optimal design shown in Fig. 2c has 4 hub nodes.

We first present a compact formulation for the RLPRF. Let  $s^1$  and  $s^2$  be two distinct hub nodes, to be chosen from  $V$  in the resulting design. For any node  $p \in V$  chosen as a regenerator node in the design, our model will force  $s^1$  to reach every  $t \in V_p \setminus s^1$  using edges in  $E_p^c$ . In other words, the model will build a rooted tree at  $s^1$  using only edges in  $E_p^c$ , visiting all the remaining regenerator nodes in  $V_p$ . Similarly, since node  $s^1$  is a regeneration point, its removal should not destroy connectivity. This is forced by rooting a tree at  $s^2$  to every regenerator node in  $V_{s^1} \setminus s^2$  using only  $E_{s^1}^c$  edges. In this fashion, we shall guarantee that even if a regenerator node is deleted from the graph, the closure graph induced by the remaining edges is connected, and hence there is resilience against failures. Let

$$s_i^1 = \begin{cases} 1, & \text{if node } i \in V \text{ is the chosen root regenerator} \\ 0, & \text{otherwise,} \end{cases}$$

$$s_i^2 = \begin{cases} 1, & \text{if node } i \in V \text{ is the chosen root regenerator when node } s^1 \text{ is deleted} \\ 0, & \text{otherwise,} \end{cases}$$

$$r_i = \begin{cases} 1, & \text{if node } i \in V \text{ is a regeneration point} \\ 0, & \text{otherwise,} \end{cases}$$

$$x_{ij}^p = \text{amount of flow on arc } (i,j) \in A_p^c \text{ in the designed tree rooted at } s^1 \text{ or } s^2 \text{ (when } p = s^1).$$

In particular, we would like the flow variables to satisfy the following balance equations for all  $p \in V, i \in V_p$ .

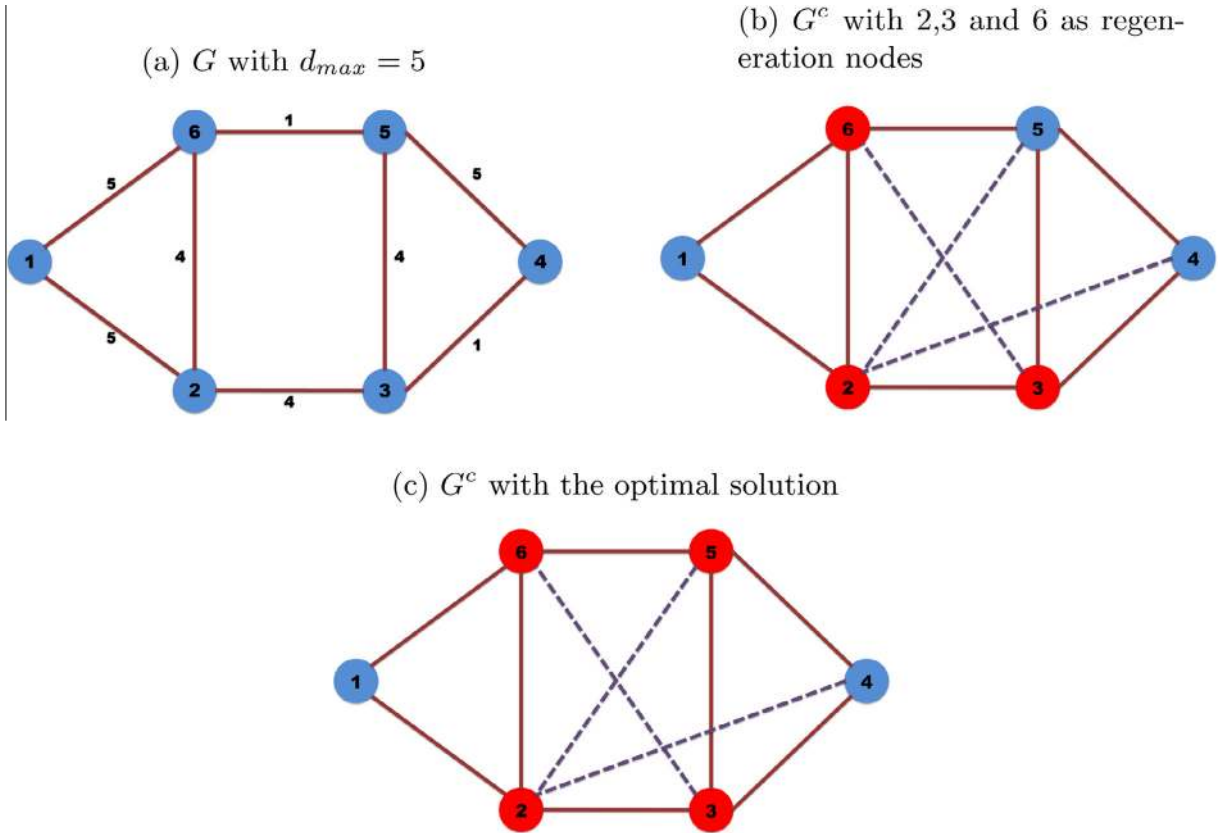


Fig. 2. Working with  $G_p^c$  rather than  $G^c$ .



$$\sum_{j:(i,j) \in A_p^c} x_{ij}^p - \sum_{j:(j,i) \in A_p^c} x_{ji}^p = \begin{cases} \sum_j r_j - 2, & \text{if } s_i^1 = 1 \text{ and } r_p = 1, \\ \sum_j r_j - 2, & \text{if } s_i^2 = 1 \text{ and } s_p^1 = 1, \\ -1, & \text{if } s_i^1 = 0, s_i^2 = 0, r_i = 1, \text{ and } r_p = 1, \\ 0, & \text{otherwise.} \end{cases}$$

Our compact formulation for the RLPRF after proper linearization of the above relations is as follows.

$$\min \sum_{i \in V} r_i \quad (19)$$

$$\text{s.t. } \sum_{i \in V} s_i^1 = 1 \quad (20)$$

$$\sum_{i \in V} s_i^2 = 1 \quad (21)$$

$$s_i^1 + s_i^2 \leq r_i \quad \forall i \in V, \quad (22)$$

$$\sum_{j:(i,j) \in E^c} r_j \geq 2 \quad \forall i \in V, \quad (23)$$

$$\sum_{j:(i,j) \in E_p^c} r_j \geq r_p \quad \forall p \in V, i \in V_p, \quad (24)$$

$$\sum_{j:(i,j) \in A_p^c} x_{ij}^p - \sum_{j:(j,i) \in A_p^c} x_{ji}^p \geq \sum_j r_j - 2 - M(1 - s_i^1) - M(1 - r_p) \quad \forall p \in V, i \in V_p, \quad (25)$$

$$\sum_{j:(i,j) \in A_p^c} x_{ij}^p - \sum_{j:(j,i) \in A_p^c} x_{ji}^p \geq \sum_j r_j - 2 - M(1 - s_i^2) - M(1 - s_p^1) \quad \forall p \in V, i \in V_p, \quad (26)$$

$$\sum_{j:(i,j) \in A_p^c} x_{ij}^p - \sum_{j:(j,i) \in A_p^c} x_{ji}^p \geq -r_i - M(1 - r_p) \quad \forall p \in V, i \in V_p, \quad (27)$$

$$\sum_{i:(i,j) \in A_p^c} x_{ij}^p \leq M r_j \quad \forall p \in V, j \in V_p, \quad (28)$$

$$r_i, s_i^1, s_i^2 \in \{0, 1\} \quad \forall i \in V, \quad (29)$$

$$x_{ij}^p \geq 0 \quad \forall p \in V, (i, j) \in A_p^c. \quad (30)$$

The objective function (19) minimizes the cardinality of the regenerator set. Constraints (20)–(22) ensure that two distinct regenerator points will be chosen as roots  $s^1$  and  $s^2$ . Constraints (23) and (24) are the cover constraints for resilience against regenerator failures. In particular, every node (demand or hub) will be reachable from two different hubs in the closure graph  $G^c$ , and when a regenerator node  $p$  is deleted, every node (demand or hub) will be reachable from another hub node in the resulting closure graph  $G_p^c$ . Adapted flow balance constraints (25)–(27) guarantee that when an arbitrary node  $p$  (different from  $s_1$ ) selected as a regeneration point is deleted from  $G$ , there will be a directed path from regenerator  $s_1$  to every other regenerator node in the closure graph  $G_p^c$ . Similarly, when the deleted node is the chosen root node  $s^1$ , there will be another tree rooted at  $s^2$  reaching every other regenerator node in the resulting closure graph. A node visited by any of these rooted tree arcs should be a regenerator node, which is forced via constraints (28).  $M = n - 1$  is a large enough number to ensure that constraints (25)–(28) are proper linearizations. Finally, constraints (29) and (30) are the required variable restrictions.

### 2.5. Regenerator Location Problem under regenerator failures: cut formulation

For the resulting design to be survivable against regenerator failures, the removal of any regenerator node should not spoil connectivity. This in turn will imply two-connectivity of the resulting design. The following cut formulation is based on these facts, and we use the same decision variables as those in the cut formulation of the RLP.

$$\min \sum_{i \in V} r_i \quad (31)$$

$$\text{s.t. } \sum_{j:(i,j) \in E^c} r_j \geq 2 \quad \forall i \in V, \quad (32)$$

$$\sum_{j:(i,j) \in E_p^c} r_j \geq r_p \quad \forall p \in V, i \in V_p, \quad (33)$$

$$x_{ij} \leq r_i \quad \forall \{i, j\} \in E^c, \quad (34)$$

$$x_{ij} \leq r_j \quad \forall \{i, j\} \in E^c, \quad (35)$$

$$x_{ij} \geq r_i + r_j - 1 \quad \forall \{i, j\} \in E^c, \tag{36}$$

$$\sum_{\{k,l\} \in E_p^c, k \in S, l \notin S} x_{kl} \geq r_i + r_j + r_p - 2 \quad \forall p \in V, S \subset V_p, i \in S, j \notin S, \tag{37}$$

$$x_e \in \{0, 1\} \quad \forall e \in E^c, \tag{38}$$

$$r_i \in \{0, 1\} \quad \forall i \in V. \tag{39}$$

As usual, the objective given in (31) is to minimize the number of regenerator nodes. Assuming at least three regenerator nodes in the design, any node should reach two regenerator nodes in the closure graph  $G^c$  through constraints (32). Once a regenerator node  $p$  is deleted from the graph, any node should be able to reach a regeneration point in the resulting closure subgraph  $G_p^c$  for resilient connectivity. This is forced by constraints (33). Since the  $x$  variables are only defined in the regenerator subgraph of the design, using constraints (34)–(36), we linearize relationship (10). Constraints (37) are the adapted form of cut inequalities. Once a regenerator node  $p$  is deleted from  $G$ , if there exists a partition of the remaining subgraph such that both sets in the partition include regenerator nodes, then a closure edge in  $G_p^c$  should cross the cut in the design for connectivity. Finally, (38) and (39) are the variable restrictions and (38) can be relaxed due to (34)–(36).

2.6. Regenerator Location Problem under node failures: compact formulation

For the RLPNF, we first present a compact formulation: Let  $s^1$  and  $s^2$  be two distinct nodes in  $V$ . For any node  $p \in V_{s^1}$ , independent of whether chosen as a regenerator point or not, we shall guarantee that  $s^1$  can reach every other node in  $V$  using edges in  $E_p^c$ . In this fashion, we shall ensure that even if node  $p$  is deleted from the graph, the closure graph induced by the edges in  $E_p^c$  is connected. We shall ensure the same is true when  $s^1$  is deleted from  $G$  by forcing  $s^2$  to reach every other node in  $V_{s^2} \setminus s^1$  through the edges induced by  $E_{s^1}^c$ .

The compact formulation for the RLPNF below parallels that of the RLPRF except that there is no conditioning on the node removed from the graph to be a regeneration point.  $M = n - 1$  is a large enough value for this formulation to work properly.

$$\min \sum_{i \in V} r_i \tag{40}$$

$$\text{s.t. } \sum_{i \in V} s_i^1 = 1 \tag{41}$$

$$\sum_{i \in V} s_i^2 = 1 \tag{42}$$

$$s_i^1 + s_i^2 \leq r_i \quad \forall i \in V, \tag{43}$$

$$\sum_{j: \{i,j\} \in E^c} r_j \geq 2 \quad \forall i \in V, \tag{44}$$

$$\sum_{j: \{i,j\} \in E_p^c} r_j \geq 1 \quad \forall p \in V, i \in V_p, \tag{45}$$

$$\sum_{j: \{i,j\} \in A_p^c} x_{ij}^p - \sum_{j: \{j,i\} \in A_p^c} x_{ji}^p \geq \sum_j r_j - 1 - r_p - M(1 - s_i^1) \quad \forall p \in V, i \in V_p, \tag{46}$$

$$\sum_{j: \{i,j\} \in A_p^c} x_{ij}^p - \sum_{j: \{j,i\} \in A_p^c} x_{ji}^p \geq \sum_j r_j - 2 - M(1 - s_i^2) - M(1 - s_p^1) \quad \forall p \in V, i \in V_p, \tag{47}$$

$$\sum_{j: \{i,j\} \in A_p^c} x_{ij}^p - \sum_{j: \{j,i\} \in A_p^c} x_{ji}^p \geq -r_i \quad \forall p \in V, i \in V_p, \tag{48}$$

$$\sum_{i: \{i,j\} \in A_p^c} x_{ij}^p \leq M r_j \quad \forall p \in V, j \in V_p, \tag{49}$$

$$r_i, s_i^1, s_i^2 \in \{0, 1\} \quad \forall i \in V, \tag{50}$$

$$x_{ij}^p \geq 0 \quad \forall p \in V, (i, j) \in A_p^c. \tag{51}$$

2.7. Regenerator Location Problem under node failures: cut formulation

For the resulting design to be two-connected, the removal of any node should not spoil connectivity. The following cut formulation is based on this fact: For any node  $p$ ,  $G_p^c$  should be connected.

$$\min \sum_{i \in V} r_i \tag{52}$$

$$\text{s.t. } \sum_{j: \{i,j\} \in E^c} r_j \geq 2 \quad \forall i \in V, \tag{53}$$

$$\sum_{j:(i,j) \in E_p^c} r_j \geq 1 \quad \forall p \in V, i \in V_p, \quad (54)$$

$$x_{ij} \leq r_i \quad \forall \{i,j\} \in E^c, \quad (55)$$

$$x_{ij} \leq r_j \quad \forall \{i,j\} \in E^c, \quad (56)$$

$$x_{ij} \geq r_i + r_j - 1 \quad \forall \{i,j\} \in E^c, \quad (57)$$

$$\sum_{\{k,l\} \in E_p^c: k \in S, l \notin S} x_{kl} \geq r_i + r_j - 1 \quad S \subset V, i \in S, j \notin S, p \in V \setminus \{i,j\}, \quad (58)$$

$$x_e \in \{0, 1\} \quad \forall e \in E^c, \quad (59)$$

$$r_i \in \{0, 1\} \quad \forall i \in V. \quad (60)$$

Similar to the compact formulation, the cut formulation of the RLPNF requirements closely follows that of the RLPRF; hence the explanations are not repeated here.

### 3. Path recovery

In all of the envisioned applications of our problems, it is imperative to know the underlying transportation paths between every source destination pair. For the first two problems recovery of such paths can be done efficiently once the design is attained, but, this task turns out to be more challenging for our third problem.

Consider a specific source destination pair  $s$  and  $t$ . Let  $R_*$  be a solution in the resulting design and let  $H = (R_* \cup \{s, t\}, E_H)$  be such that  $E_H$  is a subset of  $E^c(R_* \cup \{s, t\})$ , including only the closure edges  $\{i, j\}$  for which there exists a path from  $i$  to  $j$  in  $G$  of length at most  $d_{\max}$ , such that none of the intermediate nodes on this path belongs to  $R_* \cup \{s, t\}$ . In other words, the closure edges in  $E^c(R_* \cup \{s, t\})$  induced by using nodes in  $R_* \cup \{s, t\}$  as intermediate nodes are omitted in  $E_H$ . Let  $A_H$  be the arc set induced by the edges in  $E_H$ . Note that  $H$  is connected for the RLP and two-connected for the RLPRF and RLPNF.

Finding a walk from  $s$  to  $t$  for  $R_*$ -connectivity is easy. Let  $\mathcal{P}$  be a directed  $s - t$  path in  $H$ . The required walk can be constructed simply by concatenating the original paths in  $G$  from  $i$  to  $j$  for every induced arc  $(i, j) \in \mathcal{P}$ .

To satisfy the connectivity requirements of the RLPRF, for every source destination pair  $s$  and  $t$  we need to construct two  $s - t$  walks in  $G$  that are  $R_*$ -connected and that do not share a node from  $R_*$ . This can still be done in polynomial time by simply constructing two node disjoint paths from  $s$  to  $t$  in  $H$  and simply expanding each closure arc appearing in a path by one of its corresponding paths in the original graph and concatenating these paths.

The situation is somewhat different when trying to find two walks from  $s$  to  $t$  in  $G$  which are  $R_*$ -connected and node disjoint, i.e., satisfying the connectivity requirements of the RLPNF. A special case of this problem when  $|R_*| = 0$  is equivalent to finding two node disjoint paths between  $s$  and  $t$  such that the longer one is the smallest, which is an NP-complete problem (see Garey and Johnson, 1979).

The following model will construct two paths from  $s$  to  $t$  in  $H$ , indicated by variables  $P1$  and  $P2$ , respectively. For each closure arc induced by these variables, variables  $p1$  and  $p2$  will indicate the expanded path segments of length at most  $d_{\max}$  in the original graph. The union of these path segments will be forced to be node disjoint and will comprise the two required  $s - t$  walks.

$$p1^{ij} = \begin{cases} 1, & \text{if closure arc } (i, j) \in A_H \text{ is used on the first closure path from } s \text{ to } t \\ 0, & \text{otherwise,} \end{cases}$$

$$p2^{ij} = \begin{cases} 1, & \text{if closure arc } (i, j) \in A_H \text{ is used on the second closure path from } s \text{ to } t \\ 0, & \text{otherwise,} \end{cases}$$

$$p1_{kl}^{ij} = \begin{cases} 1, & \text{if arc } (k, l) \in A \text{ is used on the first path's closure segment from } i \text{ to } j \\ 0, & \text{otherwise,} \end{cases}$$

$$p2_{kl}^{ij} = \begin{cases} 1, & \text{if arc } (k, l) \in A \text{ is used on the second path's closure segment from } i \text{ to } j \\ 0, & \text{otherwise,} \end{cases}$$

$$m1_l = \begin{cases} 1, & \text{if node } l \in V \text{ is used as an intermediate node on the first walk from } s \text{ to } t \\ 0, & \text{otherwise,} \end{cases}$$

$$m2_l = \begin{cases} 1, & \text{if node } l \in V \text{ is used as an intermediate node on the second walk from } s \text{ to } t \\ 0, & \text{otherwise.} \end{cases}$$

The flow based formulation is as follows.

$$\min \sum_{(i,j) \in A_H, (k,l) \in A} c_{kl}(p1_{kl}^{ij} + p2_{kl}^{ij}) \tag{61}$$

$$\text{s.t. } \sum_{j:(i,j) \in A_H} P1^{ij} - \sum_{j:(j,i) \in A_H} P1^{ji} = \begin{cases} 1, & \text{for } i = s \\ -1, & \text{for } i = t \\ 0 & \text{otherwise} \end{cases} \quad \forall i \in R_* \cup \{s, t\}, \tag{62}$$

$$\sum_{j:(i,j) \in A_H} P2^{ij} - \sum_{j:(j,i) \in A_H} P2^{ji} = \begin{cases} 1, & \text{for } i = s \\ -1, & \text{for } i = t \\ 0 & \text{otherwise} \end{cases} \quad \forall i \in R_* \cup \{s, t\}, \tag{63}$$

$$\sum_{l:(k,l) \in A} p1_{kl}^{ij} - \sum_{l:(l,k) \in A} p1_{lk}^{ij} = \begin{cases} P1^{ij}, & \text{for } k = i \\ -P1^{ij}, & \text{for } k = j \\ 0 & \text{otherwise} \end{cases} \quad \forall k \in V, (i,j) \in A_H, \tag{64}$$

$$\sum_{l:(k,l) \in A} p2_{kl}^{ij} - \sum_{l:(l,k) \in A} p2_{lk}^{ij} = \begin{cases} P2^{ij}, & \text{for } k = i \\ -P2^{ij}, & \text{for } k = j \\ 0 & \text{otherwise} \end{cases} \quad \forall k \in V, (i,j) \in A_H, \tag{65}$$

$$\sum_{(k,l) \in A} c_{kl} p1_{kl}^{ij} \leq d_{\max} P1^{ij} \quad \forall (i,j) \in A_H, \tag{66}$$

$$\sum_{(k,l) \in A} c_{kl} p2_{kl}^{ij} \leq d_{\max} P2^{ij} \quad \forall (i,j) \in A_H, \tag{67}$$

$$\sum_{k:(k,l) \in A, (i,j) \in A_H} p1_{kl}^{ij} \leq M \ln1_l \quad \forall l \in V \setminus t, \tag{68}$$

$$\sum_{k:(k,l) \in A, (i,j) \in A_H} p2_{kl}^{ij} \leq M \ln2_l \quad \forall l \in V \setminus t, \tag{69}$$

$$\ln1_l + \ln2_l \leq 1 \quad \forall l \in V \setminus t, \tag{70}$$

$$P1^{ij}, P2^{ij} \in \{0, 1\} \quad \forall (i,j) \in A_H, \tag{71}$$

$$p1_{kl}^{ij}, p2_{kl}^{ij} \in \{0, 1\} \quad \forall (i,j) \in A_H, (k,l) \in A, \tag{72}$$

$$\ln1_l, \ln2_l \in \{0, 1\} \quad \forall l \in V \setminus t, \tag{73}$$

The objective function (61) minimizes the total length of the two  $s - t$  walks to be constructed. The given objective function is simply chosen for completeness' sake. Flow balance constraints (62) and (63) construct two paths from  $s$  to  $t$  in  $H$ . If closure arc  $(i, j)$  is used in either of these closure paths, it should decompose into a path segment from  $i$  to  $j$  using the arcs in the original input network. This situation is satisfied with flow balance constraints (64) and (65). Moreover, because each of these constructed path segments is regenerator free, their lengths should be bounded by the  $d_{\max}$  value, which is forced via constraints (66) and (67). Any node used as an intermediate node in a constructed walk will be indexed via constraints (68) and (69) and node disjointness will be enforced through Constraints (70).  $M = |R_*| + 1$  is satisfactorily large enough since in the worst case each one of the path segments might utilize the same node. Finally, constraints (71)–(73) enforce the proper variable restrictions.

#### 4. Separation algorithms

As our cut formulations in Section 2 contain an exponential number of constraints, we propose branch and cut algorithms for the three problems. The cut inequalities of all the formulations can be separated exactly in polynomial time by solving a series of minimum cut problems similar to the ones employed in the literature such as Labbé et al. (2004), Fouilhoux et al. (2012) and Karasaan et al. (2014). Even though our cut inequalities can be separated exactly by solving minimum cut (maximum flow) problems as typically done in the literature, since the design of the hub network is induced by the design of the hub nodes, we can employ very effective heuristic separation procedures that are particular to our problems. In the following subsections, the three separation algorithms employed in the branch and cut procedures will be detailed.

#### 4.1. Separation algorithm for the RLP

Let  $(r^*, x^*)$  be a given solution of the LP relaxation to model (11)–(18) including a subset of the cut inequalities. We define the following sets.  $R^* = \{i \in V : r_i^* > 0\}$ ,  $E^* = \{\{i, j\} \in E^c | x_{ij}^* > 0\}$  and the undirected capacitated graph  $G^* = (R^*, E^*, c)$  with the capacity of edge  $e$ , say  $c_e$  set to  $x_e^*$ , i.e.,  $G^*$  is the support graph induced by the fractional solution at hand. A cut inequality (16) is defined by a node set  $S \subset V$  and fixed nodes  $i \in S$  and  $j \notin S$ . For fixed  $i, j \in R^*$ , the most violated cut inequality can be found by solving a minimum cut problem on  $G^*$  separating  $i$  and  $j$ . If the minimum cut capacity is less than  $r_i^* + r_j^* - 1$ , then there is a violated cut inequality. Thus inequalities (16) can be separated exactly by solving  $O(n^2)$  minimum cut problems. Due to constraints (13)–(15), a violation for (16) can only occur for a pair  $i$  and  $j$  for which  $\{i, j\} \notin E^c$ .

To speed up the separation of cut inequalities, we employ a three phase procedure. In the first two phases violations are heuristically generated and only in the last phase we resort to the more costly exact separation.

#### 4.2. Connectivity separation (CS)

In the first phase we use a heuristic that can achieve separation in  $O(|E^c(R^*)|)$  time complexity. If the support graph  $G^*$  is not connected then violated cut inequalities (16) can be easily derived. More precisely, let  $S$  be any connected component in this graph. The capacity of the cut  $(S, V \setminus S)$  in the closure graph is zero, i.e.,  $x^*(\delta_{G^c}(S)) = 0$  which implies that for any pair  $i \in S, j \in R^* \setminus S$  for which  $r_i^* + r_j^* > 1$ , the cut inequality induced by  $S, i$ , and  $j$  is violated. Hence, a simple breadth first search in  $G^*$  might lead to violations. This search can further be simplified. In particular, due to constraints (13)–(15), if there exists a violated inequality induced by  $S, i$ , and  $j$  because  $G^*$  is not connected, then the same triplet will induce a violation if simply the connectivity of  $G^c(R^*)$  is checked. Ultimately, in this phase of our separation algorithm, we may ignore the  $x$  variables and work simply with the closure graph which provides great savings in CPU times since the numerical challenges of working with very small positive values are eliminated. Each connected component of  $G^c(R^*)$  may lead to violated cut inequalities. In our application, we chose a strong component  $S$  for which  $|\delta_{G^c}(S)|$  is smallest to generate cuts.

#### 4.3. Global minimum cut separation (GMCS)

In this phase, our algorithm searches for violations using a heuristic based on the Stoer-Wagner global minimum cut algorithm (SW) (Stoer and Wagner, 1997). SW finds the global minimum cut (a cut with the minimum capacity among all the partitions of the node set  $V$ ) and its running time is equivalent to that of a single minimum cut problem. Let  $(S, T)$  be a global minimum cut in  $G^*$ . If the cut capacity  $SWC = cap(S, T) = x^*(\delta_{G^c}(S))$  is not smaller than  $\max\{r_i^* + r_j^* - 1 : i, j \in R^*, i \neq j\}$ , there cannot be a violated cut and the separation procedure is terminated. Else,  $(S, i, j)$  for every  $i \in S, j \in T$  for which  $r_i^* + r_j^* - 1 > cap(S, T)$  defines a violated cut inequality. Among all potential pairs, we pick ones for which the violation is largest.

#### 4.4. Minimum cut separation (MCS)

The last stage of the algorithm is an exact separation method. In the second phase, SW finds a global minimum cut without taking  $r^*$  values into account. Perhaps another cut with larger capacity separating two nodes with higher  $r^*$  values could lead to a violation. Note that for this to happen, the two nodes should belong to the same partition in the global minimum cut  $(S, T)$ . Therefore, in this phase a series of minimum cut problems are solved separating pairs of nodes  $i$  and  $j$  that are both in  $S$  or both in  $T$ . The Ford and Fulkerson maximum flow algorithm (Ahuja et al., 1993) is implemented for solving these minimum cut problems. Only the pairs  $i$  and  $j$  for which  $r_i^* + r_j^* - 1 > SWC$  need to be considered.

The details of these phases are depicted in Algorithm 1. As our computational results will attest to, the high performance of this three phase approach makes the branch and cut application for the RLP a very successful one. Most of the time, instead of directly solving up to  $O(n^2)$  minimum cut problems, we can generate enough violated inequalities by a single graph search. One nice property of the problem is that when separating integral LP solutions, i.e., when  $r^*$  and hence  $x^*$  is integral, the first two separation algorithms become also exact. In our computational studies we observed that most of the time enough violated inequalities are found in the first phase and there are very few instances for which both the heuristics fail to produce a cut. Even in those rare cases the information derived during the second phase is very useful in reducing the number of minimum cut problems to be solved during the exact separation phase.

#### 4.5. Separation algorithm for the RLPRF

Let  $(r^*, x^*)$  be a given solution of the LP relaxation to model (31)–(39) including a subset of the cut inequalities (37) inducing sets  $R^*$  and  $E^*$  and graph  $G^*$  as defined in Section 4.1. For every  $p$  in  $R^*$ , we additionally define the set  $E_p^* = \{\{i, j\} \in E_p^c | x_{ij}^* > 0\}$  and the undirected capacitated graph  $G_p^* = (R^* \setminus p, E_p^*, c)$  with the capacity of edge  $e$ , say  $c_e$  set to  $x_e^*$ . For a fixed node  $p \in V$ , a cut inequality (37) is defined by a node set  $S \subset V$  and fixed nodes  $i \in S$  and  $j \notin S$ . For fixed  $p \in R^*$ , the most violated cut inequality can be found by solving a minimum cut problem on  $G_p^*$  separating  $i$  and  $j$ . If the min-

imum cut capacity is less than  $r_i^* + r_j^* + r_p^* - 2$ , then there is a violated cut inequality. Thus inequalities (37) can be separated exactly by solving  $O(n^3)$  minimum cut problems.

**Algorithm 1.** RLP separation

---

```

Input:  $G^c, x^*, r^*, G^*$ 
Output: Cuts
1 begin
2   Cuts= $\emptyset$ 
3   Apply CS on  $G^c$  to get connected components  $S_1, \dots, S_t$ 
4   if  $t > 1$  then
5     Let  $q = \operatorname{argmin}\{|\delta_{G^c}(S_i)| : i = 1, \dots, t\}$ 
6     foreach  $i, j \in R^* : i \in S_q, j \notin S_q, r_i^* + r_j^* > 1$  do
7       | Cuts=Cuts  $\cup (S_q, i, j)$ 
8     end
9   end
10  Apply GMCS on  $G^*$  to get the global minimum cut  $(S, T)$ 
11  Let  $SWC = x^*(\delta_{G^c}(S))$ 
12  if  $SWC < \max\{r_i^* + r_j^* - 1 : i, j \in R^*, i \neq j\}$  then
13    Let  $s^* = \operatorname{argmax}\{r_i^* : i \in S\}$ 
14    Let  $t^* = \operatorname{argmax}\{r_i^* : i \in T\}$ 
15    if  $SWC < r_{s^*}^* + r_{t^*}^* - 1$  then
16      foreach  $t \in T : r_{s^*}^* + r_t^* - 1 > SWC$  do
17        | Cuts=Cuts  $\cup (S, s^*, t)$ 
18      end
19      foreach  $s \in S : r_s^* + r_{t^*}^* - 1 > SWC$  do
20        | Cuts=Cuts  $\cup (S, s, t^*)$ 
21      end
22    end
23    foreach  $i, j \in S : r_i^* + r_j^* > SWC + 1$  do
24      | Apply MCS in  $G^*$  to get minimum cut  $(U, V)$  separating  $i$  and  $j$ 
25      | if  $\operatorname{cap}(U, V) < r_i^* + r_j^* - 1$  then
26        | Cuts=Cuts  $\cup (U, i, j)$ 
27      | end
28    end
29    foreach  $i, j \in T : r_i^* + r_j^* > SWC + 1$  do
30      | Apply MCS in  $G^*$  to get minimum cut  $(U, V)$  separating  $i$  and  $j$ 
31      | if  $\operatorname{cap}(U, V) < r_i^* + r_j^* - 1$  then
32        | Cuts=Cuts  $\cup (U, i, j)$ 
33      | end
34    end
35  end
36 end

```

---

Our separation algorithm mimics that for the RLP with an additional heuristic separation phase which became possible due to the need for survivability.

#### 4.6. Two-connectivity separation (TCS)

A necessary condition to survive from regenerator failures is to have a two-connected hub network, i.e.,  $G^c(R^*)$  should be two-connected. In other words, the following constraint set must be satisfied.

$$\sum_{\{k,l\} \in E^c: k \in S, l \notin S} x_{kl} \geq 2(r_i + r_j - 1) \quad S \subset V, i \in S, j \notin S$$

Note that the above inequalities will be implied by their stronger form of cut inequalities (37) appearing in the model. However, if  $G^c(R^*)$  is not connected then violated inequalities to the above constraints can be easily generated. In particular, for any connected component  $S$  of  $G^c(R^*)$ , the triplet  $(S, i, j)$  where  $r_i + r_j - 1 > 0$  induces a violation for the above constraint set. Similar to the RLP separation, in our application we chose strong components with minimum number of outgoing closure edges to generate cuts.

For every  $p \in R^*$ , the following three phases similar to those of RLP separation are applied to the individual closure graphs.

#### 4.7. Connectivity separation ( $CS_p$ )

If  $G_p^c$  is not connected, then for every component  $S$ ,  $i \in S$ ,  $j \notin S$  such that  $r_i^* + r_j^* + r_p^* - 2 > 0$  a cut inequality of type (37) is violated. We pick the strong component with minimum  $|\delta_{G_p^c}(S)|$  value to generate all such possible violations.

#### 4.8. Global minimum cut separation ( $GMCS_p$ )

Let  $(S, T)$  be the global minimum cut in  $G_p^*$  with capacity SWC. If  $SWC \geq \max\{r_i^* + r_j^* + r_p^* - 2 : i, j \in R^* \setminus p, i \neq j\}$ , there cannot be any violated cuts. Otherwise, every  $i \in S$ ,  $j \in T$  for which  $r_i^* + r_j^* + r_p^* - 2 > 0$  defines a violated cut inequality. For fixed  $p \in R^*$ , as done in RLP Separation,  $i$  and  $j$  are chosen such that the violation is the greatest.

#### 4.9. Minimum cut separation ( $MCS_p$ )

If the previous heuristic phases fail to generate enough cuts, this exact phase of the separation algorithm is executed. If  $(S, T)$  is the global minimum cut with capacity SWC, minimum cuts separating pairs of nodes both in  $S$  or both in  $T$  can lead to violations. In particular, only pairs  $i, j \in S(T)$  for which  $r_i^* + r_j^* > SWC + 2 - r_p^*$  might generate violations.

#### 4.10. Separation algorithm for the RLPNF

Apart from choosing the fixed node  $p$  in  $V$  rather than in  $R^*$ , the phases of this separation procedure for inequalities (58) are identical to those of the RLPRF.

The pseudo codes for the RLPRF and RLPNF separation procedures are not presented due to their close resemblance to Algorithm 1.

#### 4.11. Branch and cut implementation details

In order to reduce the high symmetry present in the feasible region for an RLP instance, we augment our model (11)–(18) with some optimality cuts commonly used in the location literature (Francis et al., 1992). In particular let  $N(i) = \{j : \{i, j\} \in E^c\}$  define the neighborhood of each node  $i \in V$  in the closure graph  $G^c$ . Then, we have:

**Lemma 1.** *If  $N(i) \setminus \{j\} \subseteq N(j) \setminus \{i\}$  for  $i \neq j$ , then there exists an optimal solution for which  $r_i = 0$ .*

**Proof.** In order for  $i$  to be present in every optimal solution, it should have a neighbor that is only covered through a hub at node  $i$ . The above condition excludes this situation.  $\square$

In order to strengthen the initial LP relaxations, we also added the following valid inequality forcing the hub network to be including at least as many edges as a tree hub network.

$$\sum_{(i,j) \in E^c} x_{ij} \geq \sum_{i \in V} r_i - 1$$

The branch and cut procedure is guided by three parameters, namely, Cutlimit, Totalcutlimit, and ESnodelimit. At every node of the branch and cut tree with node number at most ESnodelimit, we look for violations through CS, GMCS, and MCS in that order. We stop the separation algorithm as soon as Cutlimit number of cuts are generated. If the current branch and cut tree node size is larger than ESnodelimit, we stop with exact separation and look for violations (at most Cutlimit in number) through CS and GMCS only. In order to keep the cut augmented LPs at reasonable sizes, if the number of user cuts generated in this fashion is more than Totalcutlimit, we change the Cutlimit to 1. There is no limitation on the number of CPLEX generated cuts at each node.

We use the three parameters of the RLP algorithm with the same meanings during the branch and cut implementations for the RLPRF and the RLPNF as well. Given a solution  $(x^*, r^*)$ , violations are first searched through TCS in  $G^c$ . Then, for each  $p \in R^*$ , more violations are looked for by applying  $CS_p$  in  $G_p^c$ , and  $GMCS_p$  and  $MCS_p$  in  $G_p^*$ . The next phase of the separation algorithm is executed only if the number of user generated cuts so far is below Cutlimit. The specific values of the search parameters are Cutlimit = 10, Totalcutlimit = 5000, and ESnodelimit = 1.

We implemented our branch and cut algorithms using Java under Linux and ILOG CPLEX 12.4. CPLEX default settings are used throughout the branch and cut implementations. To integrate our algorithms with CPLEX we utilized the *UserCutCallback* and *LazyConstraintCallback* classes provided by the ILOG CPLEX 12.4. In each branch and bound node:

- If the LP solution is not integral, CPLEX pauses and calls the former class which runs the proposed separation algorithms to find violated inequalities. If any are found, they are added to the model and CPLEX re-solves the modified model with the added constraints. This process repeats until either an integral solution is found or separation algorithms fail to detect a violated inequality in which case branching occurs. In variable selection we set the priority levels of each binary variable  $r_i, i \in V$  to the total number of edges incident to the node  $i$  and CPLEX chooses the variables with the highest priority level first. This is done by employing the *setPriority* method available in the ILOG CPLEX 12.4.
- In case of an integral solution, CPLEX pauses and calls *LazyConstraintCallback* class to run CS algorithm (which can perform exact separation in this case) to check whether the incumbent solution is feasible. If connectivity separation detects a violated inequality it is added to the model and CPLEX continues with the modified model. Otherwise, the incumbent solution is treated as a feasible integral solution and the branch and bound node is pruned by integrality.

## 5. Computational results

Extensive computational experiments are conducted to test the performances of the proposed models and algorithms and to derive insights from the instances closely representing the real world problems. All experiments are done on a 32 × 2.6 GHz AMD Opteron 6282 SE processor with 50 GB RAM.

### 5.1. Performances of the models and algorithms

There are only a few exact algorithms in the literature that can solve moderate size RLP instances. [Chen et al. \(2009\)](#) and [Lucena et al. \(2010\)](#) present the most efficient of those algorithms. To check the performances of our algorithms against this benchmark we emulated the random graph generation procedures commonly adopted in these two studies.

#### 5.1.1. Branch and cut algorithms

For a fixed  $d_{\max}$  value, graphs are randomly generated based on four parameters. Parameter  $n$  is the number of nodes in the instance and  $p \in [0, 1]$  is the ratio of edges included in the input graph  $G$  among all potential edges. Edge lengths are chosen from a uniform distribution  $U[a * d_{\max}, b * d_{\max}]$  where  $(0 < a \leq b \leq 1)$ . To ensure the feasibility for the RLPRF and the RLPNF, an arbitrary cycle spanning all the nodes is included in the set of chosen edges. In addition to the cycle edges,  $\max(0, \lfloor \frac{p(n-1)n}{2} - n \rfloor)$  more edges are added. To keep the instances consistent with the instances in [Chen et al. \(2009\)](#), we fix  $a = 0.2$  and  $b = 1$ , and  $d_{\max} = 100$ . We enlarge the experimentation set of [Chen et al. \(2009\)](#) to values of  $n \in \{100, 150, 200, 250, 300\}$  and  $p \in \{0.01, 0.02, 0.03, 0.04, 0.05, 0.06, 0.07, 0.08, 0.09, 0.10, 0.20\}$  and generate 5 random instances for each pair of  $(n, p)$ . [Table 1](#) aggregates the results of 825 instances. Under the multicolumn heading “Graph Specifications”, for each  $(n, p)$  pair, we provide the minimum, maximum, and mean degrees in the 5 randomly generated input graphs. Under column “cl. d”, acronym for closure density, we depict the average percentage of edges present in the closure graphs of these 5 instances. Under headings “NR” and “CPU”, and for each problem, the average number of regenerators in the optimal solutions as well as the average times in seconds to reach optimality are provided, respectively.

Results presented in [Table 1](#) show that in general RLPRF and RLPNF take more time to solve the same instance of the problem than RLP. The behaviors of RLPRF and RLPNF are similar with slightly higher solution times for the RLPNF. Very sparse as well as very dense networks are favored for both of these problems in terms of CPU times. This is in contrast to the performance for the RLP where the most challenging instances correspond to very sparse input networks. In particular, we ran into out of memory problems for the pairs  $n = 250, p = 0.01$  and  $n = 300, p = 0.01$ . This is in part due to the weak LP bounds of the RLP cut formulation. As networks get dense, the RLP solution times also tend to decrease dramatically. This observation is consistent with the findings in [Chen et al. \(2009\)](#) and [Lucena et al., 2010](#). In fact, both of these papers could report exact solutions for either very small dimensions (at most 100 nodes) or very dense networks ( $p$  values at least 0.1). The solution times reported in the current study are comparable with the heuristic times provided in the mentioned two studies. Our RLP branch and cut implementation could provide results for moderately dense networks in reasonable times for dimensions up to 500 nodes. However, since our scope has all three problems and since the RLPRF and RLPNF solution times did not scale to such dimensions, we do not provide the corresponding statistics here.

The price of survivability manifests itself by increasing the regenerator number 1.4–117% with an average of 74.5% in going from RLP to RLPRF solutions and 0–100% with an average of 8.4% in going from RLPRF to RLPNF solutions, respectively. This cost is typically at its minimum for very sparse networks and at its maximum for very dense ones. In this setting of network generation, networks with larger number of nodes are more connected as can be observed through increasing closure densities with increasing  $n$  values. This in turn results in a decrease in the average number of regenerators necessary for each problem type. Though the absolute number of regenerators decreases with increasing dimension, the relative price of survivability seems to stay close throughout the  $n$  variations.



**Table 1**  
Cut formulation results.

$n$	$p$	Graph Specifications				NR			CPU		
		min	max	mean	cl. d.	RLP	RLPRF	RLPNF	RLP	RLPRF	RLPNF
100	0.01	2.00	2.00	2.00	2.65	78.20	80.20	80.20	1.00	0.00	0.00
	0.02	2.00	2.00	2.00	2.65	78.20	80.20	80.20	1.00	0.00	0.00
	0.03	2.00	6.00	2.98	5.20	30.20	55.80	58.20	195.60	0.60	0.00
	0.04	2.00	7.60	3.98	8.94	18.40	37.00	39.00	1.60	1.20	0.20
	0.05	2.00	9.60	4.96	13.46	12.80	25.40	27.60	1.20	3.00	0.00
	0.06	2.00	11.80	5.96	18.30	9.20	18.40	20.00	1.40	3.20	1.20
	0.07	2.60	13.40	6.94	24.23	6.80	12.40	13.80	2.60	6.60	1.80
	0.08	3.00	14.80	7.92	29.94	5.80	10.80	11.80	2.00	7.80	3.20
	0.09	3.40	16.20	8.92	35.36	4.80	8.20	8.80	4.40	10.40	3.60
	0.1	4.20	17.40	9.90	42.59	4.00	7.20	7.20	3.00	8.80	5.40
	0.2	11.00	28.20	19.80	91.82	1.20	2.20	3.00	0.00	0.60	7.40
	150	0.01	2.00	2.00	2.00	1.75	119.40	121.40	121.40	1.80	1.00
0.02		2.00	6.60	2.99	3.76	45.40	83.20	86.00	72.70	1.00	0.20
0.03		2.00	9.60	4.48	7.56	21.60	45.00	48.40	3.60	6.80	1.00
0.04		2.20	11.80	5.97	12.66	14.00	26.60	29.20	3.20	19.20	3.40
0.05		2.40	14.80	7.45	18.43	9.40	16.60	18.20	5.60	38.00	8.40
0.06		3.00	16.60	8.95	26.63	6.60	12.20	13.00	5.20	63.20	11.60
0.07		4.40	18.80	10.44	33.98	5.00	8.80	9.80	4.00	42.60	13.20
0.08		5.40	20.00	11.92	43.52	4.20	7.20	7.60	5.80	58.40	21.80
0.09		6.00	22.00	13.41	52.41	3.60	5.80	6.20	8.80	40.80	28.20
0.1		7.20	24.60	14.91	60.56	3.00	5.00	5.00	5.80	36.60	52.00
0.2		19.00	43.20	29.80	98.70	1.00	2.00	2.40	1.80	2.20	11.20
200		0.01	2.00	2.00	2.00	1.33	157.60	159.80	159.80	3.00	1.00
	0.02	2.00	9.40	3.99	4.42	36.20	71.20	75.40	142.00	14.60	1.40
	0.03	2.00	12.40	5.98	9.30	17.60	34.60	37.40	6.40	234.60	11.40
	0.04	2.60	16.40	7.97	16.11	10.80	19.80	21.80	8.60	122.00	17.00
	0.05	3.80	18.60	9.96	25.29	7.20	13.20	14.40	9.40	157.20	50.20
	0.06	5.20	21.40	11.95	34.88	5.20	8.60	9.40	13.00	171.60	74.40
	0.07	6.60	23.60	13.93	45.39	4.00	7.00	7.20	11.80	174.00	88.40
	0.08	7.60	26.60	15.92	56.11	3.20	5.40	5.60	16.60	122.40	157.80
	0.09	9.00	29.20	17.91	65.83	2.60	4.60	4.60	10.80	95.60	185.60
	0.1	10.80	31.80	19.90	74.48	2.20	3.80	3.80	11.60	76.20	203.80
	0.2	26.00	55.00	39.80	99.91	1.00	2.00	2.00	6.80	7.20	7.00
	250	0.01	2.00	5.20	2.50	1.58	mem.	171.20	174.60	mem.	1.00
0.02		2.00	10.60	4.98	5.55	30.00	60.40	65.40	35.20	62.60	7.20
0.03		2.80	15.40	7.47	11.81	15.20	27.20	30.00	59.20	689.00	85.60
0.04		3.80	18.60	9.97	20.42	8.80	16.20	17.40	22.60	710.60	360.40
0.05		5.00	22.40	12.46	31.89	6.20	11.20	11.80	20.20	323.60	126.20
0.06		6.20	25.60	14.94	43.07	4.40	7.40	7.60	27.40	304.60	363.20
0.07		7.60	29.60	17.43	55.85	3.20	5.80	5.80	25.80	328.80	252.00
0.08		10.00	35.00	19.92	66.81	2.80	4.40	4.60	27.00	176.60	586.20
0.09		11.00	36.80	22.41	76.27	2.00	3.80	3.80	21.40	195.00	402.40
0.1		13.20	39.00	24.90	85.21	2.00	3.00	3.00	22.20	78.60	388.20
0.2		33.00	68.20	49.80	99.98	1.00	2.00	2.00	12.20	13.80	14.20
300		0.01	2.00	6.80	2.99	1.75	mem.	166.80	172.60	mem.	6.80
	0.02	2.00	13.00	5.99	6.40	26.40	51.80	57.60	85.20	1948.80	347.80
	0.03	3.00	18.20	8.97	14.17	13.40	23.80	26.20	46.60	2355.60	3448.00
	0.04	4.00	22.20	11.97	24.99	7.80	13.40	14.80	78.20	2178.60	2975.00
	0.05	6.20	27.00	14.95	38.00	5.20	8.40	9.40	37.20	766.00	1928.20
	0.06	8.00	31.40	17.95	51.91	4.00	6.00	6.60	61.80	672.40	2336.60
	0.07	9.60	34.80	20.93	65.63	3.00	4.20	4.80	42.80	307.60	3579.80
	0.08	11.80	39.20	23.92	77.77	2.00	3.40	3.80	34.60	261.20	833.60
	0.09	13.40	41.60	26.91	86.39	2.00	3.00	3.00	167.20	54.60	55.20
	0.1	15.60	44.80	29.90	92.25	1.20	2.60	3.00	30.40	43.60	64.20
	0.2	40.80	80.60	59.80	100.00	0.40	0.80	1.60	20.20	28.80	30.20

### 5.1.2. Flow formulations

In our testbed instances to compare the flow and cut formulations, in the random graph generation scheme detailed above, we chose  $p$  as  $\delta/(n-1)$  in order to generate graphs with average degree of  $\delta$ . We ranged  $n \in \{40, 60, 80, 100\}$  and  $\delta \in \{3, 4, 5\}$  and randomly generated 5 instances for each parameter pair. Table 2 depicts the results of 360 runs. As done in the previous table, we report the average closure graph densities of the 5 instances. The average optimal solution values are presented under multicolumn NR for each of the three problems. The average CPU times in seconds for respective flow and cut formulations of the three problems are also tabulated. Due to the sparse nature of the generated graphs, the

**Table 2**  
Flow vs cut formulations.

n	$\delta$	cl. d.	NR			CPU					
			RLP	RLPRF	RLPNF	RLP		RLPRF		RLPNF	
						Flow	Cut	Flow	Cut	Flow	Cut
40	3	13.23	12.00	23.40	23.80	0.80	0.00	4.80	0.00	4.00	0.00
	4	20.38	7.60	14.60	15.80	0.00	0.00	3.00	0.00	2.40	0.00
	5	30.41	4.80	10.00	11.00	0.00	0.00	3.80	0.20	12.00	0.00
60	3	8.82	18.40	33.80	35.20	70.00	0.80	19.40	0.00	9.40	0.00
	4	13.99	11.60	23.40	24.80	0.20	0.40	11.80	0.00	17.20	0.00
	5	20.63	7.60	15.40	17.00	0.00	0.00	28.80	0.40	30.40	0.00
80	3	6.49	23.80	45.00	46.60	418.20	1.40	50.20	0.00	69.20	0.00
	4	10.98	14.60	29.80	31.40	1.00	0.80	224.40	0.40	65.80	0.00
	5	17.01	10.00	19.80	22.20	0.20	0.60	74.80	1.20	180.40	0.20
100	3	5.20	30.20	55.80	58.20	7400.60	195.60	431.20	0.60	232.00	0.00
	4	8.94	18.40	37.00	39.00	9.40	1.60	756.00	1.20	1379.40	0.20
	5	13.46	12.80	25.40	27.60	0.20	1.20	152.20	3.00	613.00	0.00

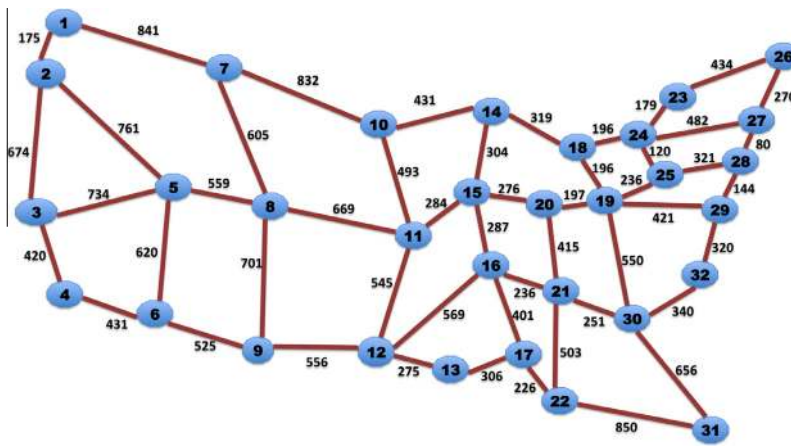


Fig. 3. 32-Node network.

performance of RLP is the weakest for some instances. In general, the cut formulations are quite fast. However, for  $\delta = 4$  and  $\delta = 5$ , the performance of RLP flow formulations is also quite good. Even with  $n = 100$  optimum solutions for the RLP flow formulations are obtained in a few seconds.

Unlike the results of Table 1, the number of regenerators increases with increasing dimension due to the decrease in the closure graph densities resulting from fixing the average degrees. The price of survivability tends to increase slightly with the increasing density and shows a nearly steady behavior with increasing dimension.

### 5.2. Spatial insights

For most of the real world applications of the RLP, the length of an edge between two nodes is closely related to the Euclidean distance between those nodes. Therefore, we believe that investigating problem instances with edge costs computed as the Euclidean distances can be beneficial to better understand the nature of the problem and gain insights which would be helpful in dealing with the real world problems. To this end, in the following sections we study networks of this nature.

#### 5.2.1. 32 Node network

In this set of problems we study a relatively small size transport network based on US haul optical network as introduced in Yetginer and Karasan (2003) and depicted in Fig. 3. In Fig. 4, solutions (NR numbers) of the three problems for varying transport range ( $d_{max}$ ) values are illustrated. As expected, the larger the transport range the less the number of regenerators needed. Fig. 4 confirms this intuition and also shows that the optimal solutions become less sensitive to  $d_{max}$  as it gets higher. This conclusion is expected and valid for all three problem types RLP, RLPRF and RLPNF.

Confirming the conclusions of Section 5.1, the high price of survivability is also apparent in Fig. 4. The average number of regenerators required for RLP, RLPRF, and RLPNF are 1.68, 3.46 and 3.84, respectively. We see that the number of

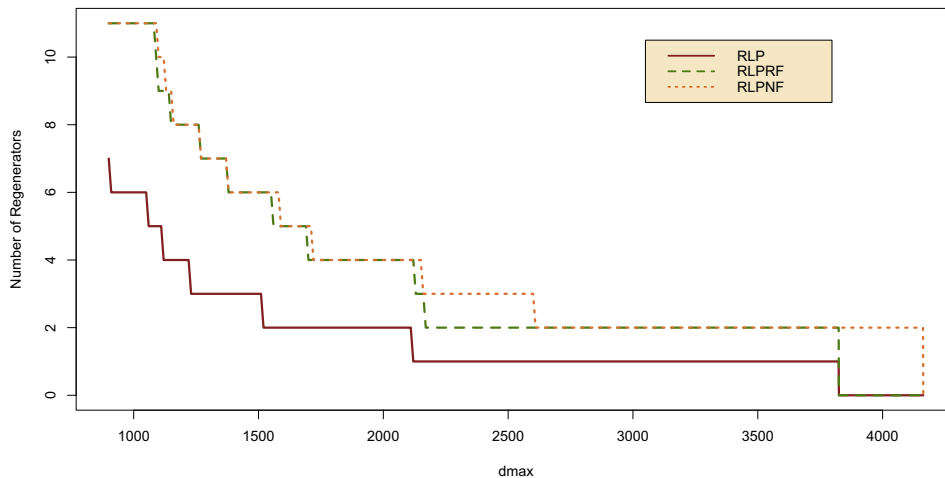


Fig. 4. 32-Node network: number of regenerators versus  $d_{\max}$ .

Table 3

Computational results for the 32-node network example.

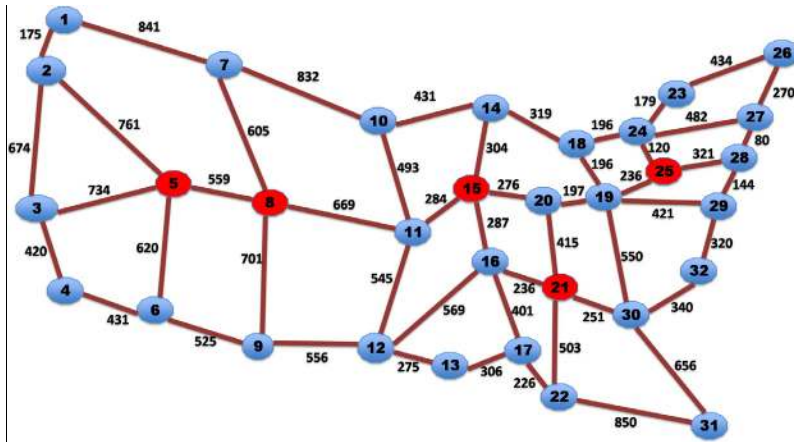
$d_{\max}$	RLP		RLPRF		RLPNF	
	NR	Regeneration nodes	NR	Regeneration nodes	NR	Regeneration nodes
1090	5	5-8-15-21-25	10	3-5-6-8-10-12-15-21-25-32	11	1-3-6-7-9-10-12-18-21-22-29
1150	4	5-8-15-32	8	1-5-7-9-10-16-18-32	9	2-5-7-9-10-12-22-25-32
1560	2	5-15	5	4-5-12-15-19	6	1-6-8-12-19-21
1700	2	5-15	4	5-6-15-16	5	1-6-8-16-18
2130	1	10	3	1-11-14	4	1-9-11-32
2170	1	10	2	8-11	3	7-11-14
3825	0	–	0	–	2	8-11

regenerators can increase significantly when the network is required to be resilient against regenerator failures or resilient against node failures. On the average, the number of regenerators increases by 107% with regenerator resilience and by 125% with node resilience. These high percentage changes are in part due to having a small number of regenerators as solutions for all three problem types in this network instance. This price is lower for more strict transport range values. However, the price of adding node survivability over regenerator survivability is on the average quite low at 8%. Our more extensive experiments presented in the next subsection also confirm all of these observations albeit with slightly smaller percentages due to the high number of regenerators in the solutions.

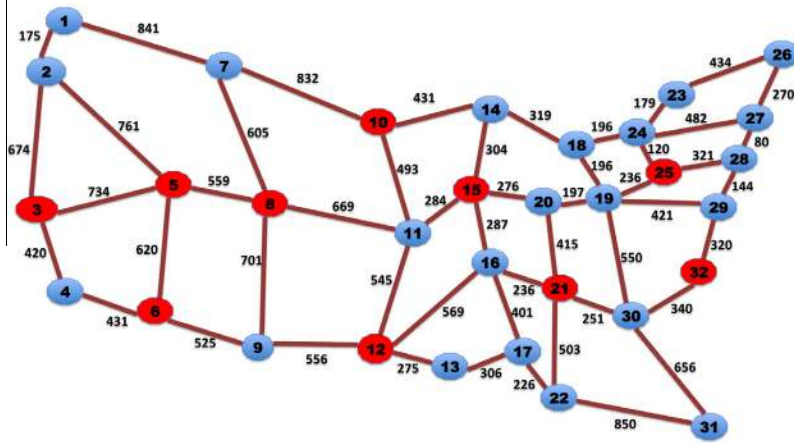
In Table 3 we list the regenerator locations for the transport range values at which RLPRF and RLPNF have different solutions. Revealing the different natures of the RLP problems, Fig. 5 illustrates the optimal solutions for the instance  $d_{\max} = 1090$ . Looking at these figures we can observe that RLPRF basically adds new regenerators to the RLP solution to establish a cycle among the regenerator nodes to guarantee that each node can reach at least two regenerators. In the RLPNF, not only one more regenerator is required but also the placement of those regenerators is different than that of RLPRF. Looking at node 31 we can see what makes RLPRF and RLPNF to have different solutions in this example. Far away peripheral nodes like 1 and 31 increase the NR values for the RLPNF solution. These nodes have just two neighbors and at least one of them is quite far. In this situation establishing node failure resilience forces two regenerators to be placed rather close to these nodes, which restricts the solution space significantly hence resulting in higher number of regenerators in total. Therefore for the problem instances where node degrees are small and edge lengths are close to  $d_{\max}$ , it is more likely for RLPRF and RLPNF to have different solutions.

It is important to understand if any spatial property of a node makes it desirable as a hub. In this respect, the number of times a node appears in the alternative optimal solutions reveals crucial information. For four selected  $d_{\max}$  values, Table 4 lists for each node its degree in the corresponding closure graph ( $\Delta(G^c)$ ) and the percentage of optimal solutions having this particular node as a hub node for the three problem types (denoted as 1, 2 and 3 to save space). The number of alternative optimal solutions is reported at the bottom of the table for each instance. Fig. 4 can be referred to for optimal values.

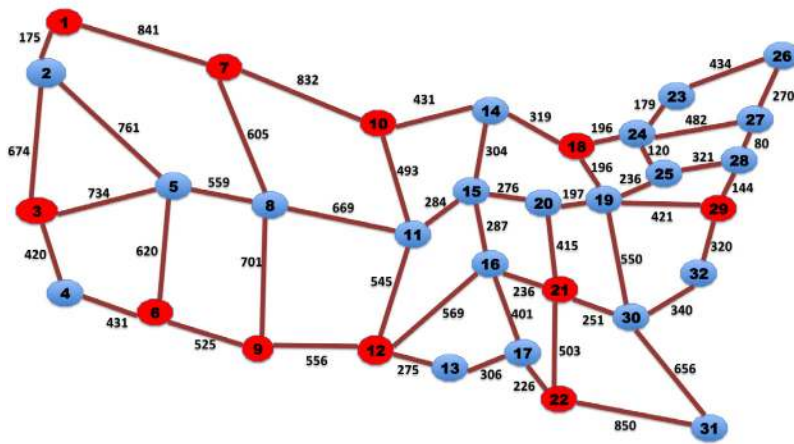
Inspecting the table, we observe that peripheral nodes like 26 and 31 are consistently ruled out in all the solutions. The RLP designs are quite different from those of the survivable ones. In the RLP, the aim is to build a connected hub network covering all nodes. For this reason, the nodes on the central band  $\{3, 5, 8, 11, 30\}$ , which have relatively high closure graph degrees when compared with their neighbors, are more likely to be present in the optimal solution. This picture is more clear



(a) RLP Solution



(b) RLPRF Solution



(c) RLPNF Solution

Fig. 5. Hub locations for  $d_{\max} = 1090$ .

for smaller  $d_{\max}$  values. When  $d_{\max}$  gets larger, node 3 is excluded and in a typical solution of value 2 for  $d_{\max} = 1800$  node 8 pairs up with a central-east node of high closure graph degree.

On the other hand, when resilience is an issue and  $d_{\max}$  values are small, the nodes on the periphery, preferably those with relatively high closure graph degrees among their neighbors, begin to appear in the solutions to build a two-connected hub

**Table 4**  
Analysis of alternative solutions.

Node	$d_{\max} = 900$			$d_{\max} = 1200$			$d_{\max} = 1500$			$d_{\max} = 1800$						
	$\Delta(G^c)$	1	2	3	$\Delta(G^c)$	1	2	3	$\Delta(G^c)$	1	2	3	$\Delta(G^c)$	1	2	3
1	3	0	100	100	4	0	13	13	6	10	2	0	8	0	0	0
2	3	0	0	0	5	0	27	25	7	10	16	2	7	0	0	0
3	5	100	100	100	5	0	0	0	7	14	39	49	8	0	0	0
4	2	0	0	0	5	0	27	25	6	5	5	0	9	0	0	0
5	4	100	0	0	8	100	87	88	9	38	49	49	14	24	20	0
6	4	0	100	100	6	6	47	50	8	10	21	6	13	12	100	100
7	3	0	100	100	5	0	100	100	8	5	44	51	14	0	0	0
8	4	100	0	0	7	18	0	0	17	57	49	49	23	65	80	100
9	4	0	100	100	9	76	67	63	14	29	80	94	18	0	0	0
10	5	3	100	100	13	0	100	100	20	0	13	9	26	0	0	0
11	10	100	0	0	18	12	0	0	24	24	15	9	27	18	0	0
12	8	0	50	50	13	12	33	38	17	0	5	0	23	0	0	0
13	6	0	50	50	9	0	0	0	16	0	21	11	20	0	0	0
14	11	5	0	0	18	6	0	0	23	0	59	83	25	6	20	0
15	14	16	0	0	21	12	0	0	24	26	24	0	26	12	40	0
16	12	14	0	0	19	53	40	0	23	0	10	21	26	18	100	100
17	7	0	0	0	13	12	40	75	17	0	8	13	24	0	0	0
18	14	22	100	100	17	6	20	0	21	2	11	0	24	6	20	50
19	15	22	0	0	19	12	20	0	23	17	19	36	24	6	20	50
20	14	22	0	0	20	12	20	0	23	17	23	0	25	12	0	0
21	13	19	0	0	20	0	7	0	24	26	39	38	24	6	0	0
22	7	5	100	100	12	0	13	25	18	0	9	17	22	0	0	0
23	10	5	0	0	15	0	20	0	17	0	0	0	20	0	0	0
24	11	8	0	0	17	6	0	0	18	0	0	0	23	6	0	0
25	14	19	25	0	17	6	20	100	20	2	11	26	23	6	0	0
26	8	0	0	0	12	0	0	0	14	0	0	0	17	0	0	0
27	11	14	25	50	14	0	0	0	17	0	0	0	20	0	0	0
28	11	14	25	50	14	0	0	0	18	2	4	9	20	0	0	0
29	12	14	25	0	16	6	0	0	19	2	6	13	21	0	0	0
30	14	95	100	100	21	35	47	88	22	2	11	0	24	6	0	0
31	2	0	0	0	6	0	0	0	14	0	0	0	20	0	0	0
32	9	5	0	0	16	12	53	13	20	2	8	17	22	0	0	0
#Alt		37	8	4		17	75	8		42	158	47		17	5	2

network. For the most strict transport range, RLPRF and RLPNF optimal solutions replace nodes like 5 and 8 in a solution of RLP with pairs of nodes  $\{6, 7\}$  and  $\{9, 10\}$ , respectively. But if  $d_{\max}$  gets larger, the nodes in the center like 8 and 16 become advantageous in achieving two-connectivity. Even though for all the chosen  $d_{\max}$  values, the RLPRF and RLPNF provide the same optimal values, the listed percentages interestingly point out differences in locations for medium  $d_{\max}$  values.

We would like to emphasize that the previous findings linking spatial properties to optimal hub locations are only pertinent to the hub covering aspect of our problems where demands as well as costs are not influential. One would expect that in a cost based hub model, the number of alternative solutions in Table 4 would diminish and hub locations of high demand and geographical advantages would appear as more robust optimal hub locations.

### 5.2.2. Euclidean networks

To study Euclidean graphs further, we studied various randomly generated problem instances of different size and structure. In fact Chen et al. (2009) have test instances for Euclidean networks. In that setting, to establish feasibility, an arbitrary spanning tree is constructed by drawing the lengths of the edges in the tree from a uniform distribution. In other words,  $n - 1$  edges do not have Euclidean edge lengths. Moreover, with the parameters used in that study the closure graphs get too dense as the average degree of a node in the closure graph can be as large as 50. Although it is computationally easier to generate and solve the RLP on such instances, we believe that they may not be close approximations of the real world. In our case ensuring feasibility for the RLPRF and the RLPNF would add even more artificialities. So we chose to generate Euclidean networks in a slightly different way.

We uniformly locate  $n$  nodes on a  $80 \times 80$  grid. The parameter  $c_{\max} \leq d_{\max}$  controls the connectivity of the graph, namely, edge  $\{i, j\}$  is included if and only if the Euclidean distance between nodes  $i$  and  $j$  is less than  $c_{\max}$ . If the graph generated in this fashion is not two-connected, we disregard it and repeat the procedure with a different random number seed. It is clear that for smaller values of  $c_{\max}$  it gets harder to construct the graph and the resulting graph tends to concentrate on some smaller portion of the  $80 \times 80$  grid. Thus  $c_{\max}$  cannot be chosen arbitrarily small.

Table 5 lists the solution averages of 10 instances for differing  $n$ ,  $c_{\max}$  and  $d_{\max}$  values. As expected, going from left to right in the table, the number of regenerators decreases due to the increased connectivity of the graphs with high  $c_{\max}$  values. This trend is persistent in all the three problems. However, it is interesting to see that the decrease is not more pronounced in the RLPRF and the RLPNF. Intuitively, one would expect that, in the case of survivability, the number of regenerators would be

**Table 5**  
Computational results for the Euclidean networks.

$n$	$c_{\max}$	RLP			RLPRF			RLPNF			
		12	19	25	12	19	25	12	19	25	
80	$d_{\max}$										
	25	10.2	8.0	7.5	15.4	12.8	12.0	15.9	13.0	12.3	
	30	7.7	5.8	5.4	11.9	9.8	9.0	12.2	10.1	9.3	
100	35	5.8	4.1	4.0	9.4	8.0	7.9	9.6	8.2	8	
	25	9.6	7.9	7.3	14.9	12.6	11.9	16.1	13.0	12.2	
	30	6.9	5.7	5.4	11.3	9.6	9.2	12	9.8	9.4	
120	35	5.5	4.1	4.0	8.9	7.9	7.8	9.1	8.0	7.9	
	25	9.1	7.6	7.2	14.3	12.2	11.6	14.8	12.7	12.1	
	30	6.6	5.3	5.1	10.8	8.9	8.4	10.9	9.1	8.6	
150	35	5.1	4.1	4.0	8.5	7.9	7.7	8.5	7.9	7.7	
	25	8.2	7.3	6.9	13.3	11.7	11.5	13.7	12.0	11.9	
	30	6.0	5.1	5.0	9.8	8.6	8.1	10	8.6	8.3	
	35	4.3	4.0	4.0	8	7.8	7.8	8	7.8	7.8	

**Table 6**  
Euclidean network statistics.

$n$	$c_{\max} = 12$			TEL	$c_{\max} = 19$			TEL	$c_{\max} = 25$			TEL
	Node degrees				Node degrees				Node degrees			
	min	max	mean		min	max	mean		min	max	mean	
80	2.0	9.0	4.3	1515.9	3.1	19.7	10.8	5527.2	4.8	30.7	17.9	11784.2
100	2.1	10.9	5.0	2257.8	3.7	23.6	13.4	8680.6	5.9	38.2	23.0	18540.7
120	2.0	13.3	6.5	3266.9	4.4	28.4	16.7	12690.1	7.7	44.9	27.2	26541.7
150	2.3	17.3	8.8	5328.9	6.2	35.8	21.2	20051.7	10.0	57.0	34.7	41791.6

more sensitive to the changes in the connectivity of the graph than in the case of RLP. However, at least for these instances, results do not confirm this expectation. For a fixed  $n$  value, we observe a sharper decrease in the optimal values with increasing  $d_{\max}$ . This time we see that the solutions of the RLP are much more sensitive than the RLPRF and the RLPNF to the increasing transportation range. As a result, the relative price of survivability increases when  $d_{\max}$  increases.

Table 6 provides further statistics for the Euclidean graphs under consideration when  $d_{\max} = 25$ . In addition to the minimum, maximum, and mean degree averages of the 10 instances for specific  $n$  and  $c_{\max}$  pairs, the total edge length (TEL) averages are also reported. Since increased connectivity results in fewer hubs, a natural question is whether we can reduce the total hub setup cost by building new connections between nodes. Table 6 shows that this cannot be done in a naive way. Starting from a graph with  $c_{\max} = 12$  with 150 nodes and augmenting this graph by connecting all the nodes that are closer than  $d_{\max}$  requires total edge length to grow from 5329 to 41,792 (around 700% increase), while such an effort results in around 15% fewer regenerators on average without survivability and around 13% fewer when survivability is considered. However, not all those new connections are essential. Thus, it is an interesting optimization problem in its own right to choose the critical set of edges to add to an existing network such that the total construction cost is minimum while the network provides the regeneration capability of a complete network.

In the case of change in the number of nodes in the network, two opposite effects are in place. The higher number of nodes potentially increases the optimal value while increased connectivity pushes it down. Note that in this set of graphs the underlying grid has a fixed size and more nodes mean more connectivity. This is also apparent in Table 6 which shows that the average node degree almost doubles when we have 150 nodes instead of 80. This is the same for all  $c_{\max}$  values. Between these competing factors results show that the effect of connectivity is stronger than that of the number of nodes.

## 6. Conclusions

This paper revisits the Regenerator Location Problem with the hub location viewpoint and introduces two new variations of it involving grades of survivability. For the three problems under our scope, flow formulations, cut formulations, and branch and cut implementations are proposed in a unifying framework. The hub location perspective enabled the formulations to be mainly defined on the hub network thus resulting in higher efficiency. The RLP literature is computationally extended and it has been shown that optimal designs under hub and node failure resilience are tractable even for very high network dimensions.

Extensive computational experiments are conducted to understand what makes some nodes desirable as hubs. It has been observed that survivability has a great price and drastically changes the resulting designs. On the other hand,

interestingly, designs hedging against all node failures rather than just hub failures are not much more costly. Though the designs might have similar numbers of regenerators, their spatial properties seem to differ.

The problems as defined in this paper are intrinsic parts of several transportation problems and the methodologies developed can be used as building blocks to devise efficient solution algorithms for more specific and complex applications in these settings.

## Acknowledgments

We are thankful to two anonymous referees for their constructive feedbacks, in particular for suggestions that shaped up the discussion in SubSection 1.1

This work was supported by the Scientific and Technological Research Council of Turkey (TUBITAK), Grant No: BIDEB-2211-A.

## References

- Agrawal, G., 2012. Fiber-Optic Communication Systems. In: Wiley Series in Microwave and Optical Engineering. Wiley.
- Ahuja, R., Magnanti, T., Orlin, J., 1993. Network Flows: Theory, Algorithms and Applications. Prentice Hall.
- Ali, T., Radhakrishnan, S., Pulat, S., Gaddipati, N., 2002. Relay network design in freight transportation systems. *Transportation Research Part E* 38, 405–422.
- Alumur, S., Kara, B., 2008. Network hub location problems: The state of the art. *European Journal of Operational Research* 190, 1–21.
- Alumur, S.A., Kara, B.Y., Karasan, O.E., 2009. The design of single allocation incomplete hub networks. *Transportation Research Part B* 43, 936–951.
- An, Y., Zhang, Y., Zeng, B., 2011. The reliable hub-and-spoke design problem: Models and algorithms, optimization-online.org.
- Azizi, N., Chauhan, S., Salhi, S., Vidyarthi, N., 2014. The impact of hub failure in hub-and-spoke networks: Mathematical formulations and solution techniques. *Computers & Operations Research*. <http://dx.doi.org/10.1016/j.cor.2014.05.012> (to appear in print).
- Cabral, E., Erkut, E., Laporte, G., Patterson, R., 2007. The network design problem with relays. *European Journal of Operational Research* 180, 834–844.
- Calik, H., Alumur, S., Kara, B., Karasan, O., 2009. A tabu-based heuristic for the hub covering problem over incomplete hub networks. *Computers & Operations Research* 36, 3088–3096.
- Campbell, J., 1994a. Integer programming formulations of discrete hub location problems. *European Journal of Operational Research* 72, 387–405.
- Campbell, J., 1994b. A survey of network hub location. *Studies in Locational Analysis* 6, 31–49.
- Campbell, J., Ernst, A., Krishnamoorthy, M., 2005a. Hub arc location problems: Part i – Introduction and results. *Management Science* 51, 1540–1555.
- Campbell, J., Ernst, A., Krishnamoorthy, M., 2005b. Hub arc location problems: Part ii – Formulations and optimal algorithms. *Management Science* 51, 1556–1571.
- Campbell, J., O’Kelly, M., 2012. Twenty-five years of hub location research. *Transportation Science* 46, 153–169.
- Campbell, J.F., 1993. Continuous and discrete demand hub location problems. *Transportation Research Part B* 27, 473–482.
- Chen, S., Ljubić, I., Raghavan, S., 2009. The regenerator location problem. *Networks* 55, 205–220.
- Chimani, M., Kandyba, M., Ljubić, I., Mutzel, P., 2008. Strong formulations for 2-node-connected steiner network problems. *Combinatorial Optimization and Applications*, 190–200.
- Correia, I., Nickel, S., da Gama, F.S., 2010. Single-assignment hub location problems with multiple capacity levels. *Transportation Research Part B* 44, 1047–1066.
- Davari, S., Zarandi, M., Turksen, I., 2010. The fuzzy reliable hub location problem. In: *Fuzzy Information Processing Society (NAFIPS), 2010 Annual Meeting of the North American*, pp. 1–6.
- do Forte, V.L., Lucena, A., Maculan, N., 2013. Formulations for the minimum 2-connected dominating set problem. *Electronic Notes in Discrete Mathematics* 41, 415–422.
- Fortz, B., Labbé, M., Maffioli, F., 2000. Solving the two-connected network with bounded meshes problem. *Operations Research* 48, 866–877.
- Fouilhoux, P., Karasan, O., Mahjoub, A., Ozkok, O., Yaman, H., 2012. Survivability in hierarchical telecommunications networks. *Networks* 59, 37–58.
- Francis, R., McGinnis, J.L., White, J., 1992. Facility Layout and Location: An Analytical Approach. Prentice Hall.
- Garey, M., Johnson, D., 1979. Computers and Intractability: A Guide to the Theory of NP-completeness. W.H. Freeman & Co.
- Gendron, B., Lucena, A., da Cunha, A.S., Simonetti, L., 2014. Benders decomposition, branch-and-cut and hybrid algorithms for the minimum connected dominating set problem. *Inform. Journal on Computing*. <http://dx.doi.org/10.1287/ijoc.2013.0589> (to appear in print).
- Grotschel, M., Monma, C., Stoer, M., 1995. Design of survivable networks. *Handbooks in Operations Research and Management Science* 7, 617–672.
- Hamacher, H., Meyer, T., 2006. Hub cover and hub center problems. Department of Mathematics, University of Kaiserslautern, Germany.
- Hamidi, M., Gholamian, M., Shahanaghi, K., 2014. Developing prevention reliability in hub location models. *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability*.
- He, F., Wu, D., Yin, Y., Guan, Y., 2013. Optimal deployment of public charging stations for plug-in hybrid electric vehicles. *Transportation Research Part B* 47, 87–101.
- Index, C., 2012. Global mobile data traffic forecast update, 2011–2016, white paper.
- Kara, B., Tansel, B., 2000. On the single-assignment p-hub center problem: Models and linearizations. *European Journal of Operational Research* 125, 648–655.
- Karasan, O., Mahjoub, A., O. Ozkok, H., 2014. Survivability in hierarchical telecommunications networks under dual homing. *Inform. Journal on Computing* 26, 1–15.
- Kerivin, H., Mahjoub, A., 2005. Design of survivable networks: A survey. *Networks* 46, 1–21.
- Kewcharoenwong, P., Uster, H., 2014. Benders decomposition algorithms for the fixed-charge relay network design in telecommunications. *Telecommunication Systems* 56, 441–453.
- Kim, H., 2012. P-hub protection models for survivable hub network design. *Journal of Geographical Systems* 14, 437–461.
- Kim, H., O’Kelly, M., 2009. Reliable p-hub location problems in telecommunication networks. *Geographical Analysis* 41, 283–306.
- Kim, H., O’Kelly, M.E., 2008. Reliable p-hub location problems and protection models for hub network design. Ph. D. thesis, Ohio State University.
- Konak, A., 2012. Network design problem with relays: A genetic algorithm with a path-based crossover and a set covering formulation. *European Journal of Operational Research* 218, 829–837.
- Konak, A., 2014. Two-edge disjoint survivable network design problem with relays: a hybrid genetic algorithm and lagrangian heuristic approach. *Engineering Optimization* 46, 130–145.
- Kuby, M., Lim, S., 2005. The flow-refueling location problem for alternative-fuel vehicles. *Socio-Economic Planning Sciences* 39, 125–145.
- Labbé, M., Laporte, G., Martin, I., Gonzalez, J., 2004. The ring star problem: polyhedral analysis and exact algorithm. *Networks* 43, 177–189.
- Laporte, G., Pascoal, M., 2011. Minimum cost path problems with relays. *Computers & Operations Research* 38, 165–173.
- Lau, L., Naor, J.S., Salavatipour, M., Singh, M., 2009. Survivable network design with degree or order constraints. *SIAM Journal on Computing* 39, 1062–1087.
- Lei, T., 2013. Identifying critical facilities in hub-and-spoke networks: A hub interdiction median problem. *Geographical Analysis* 45, 105–122.

- Lucena, A., Maculan, N., Simonetti, L., 2010. Reformulations and solution algorithms for the maximum leaf spanning tree problem. *Computational Management Science* 7, 289–311.
- Meng, Q., Wang, X., 2011. Intermodal hub-and-spoke network design: Incorporating multiple stakeholders and multi-type containers. *Transportation Research Part B* 45, 724–742.
- MirHassani, S., Ebrazi, R., 2012. A flexible reformulation of the refueling station location problem. *Transportation Science* 47, 617–628.
- Monma, C., Shallcross, D., 1989. Methods for designing communications networks with certain two-connected survivability constraints. *Operations Research* 37, 531–541.
- Nie, Y.M., Ghamami, M., 2013. A corridor-centric approach to planning electric vehicle charging infrastructure. *Transportation Research Part B* 57, 172–190.
- O’Kelly, M., 1986a. Activity levels at hub facilities in interacting networks. *Geographical Analysis* 18, 343–356.
- O’Kelly, M., 1986b. The location of interacting hub facilities. *Transportation Science* 20, 92–106.
- O’Kelly, M., 1987. A quadratic integer program for the location of interacting hub facilities. *European Journal of Operational Research* 32, 393–404.
- Ouyang, Y., Zeng, Q., Wei, W., 2005. Dynamic lightpath provisioning with signal quality guarantees in survivable translucent optical networks. *Optics Express* 13, 10457–10468.
- Ozkok, O., 2014. Private communication. *Operations Research Scientist*, Amazon.
- Racunica, I., Wynter, L., 2005. Optimal location of intermodal freight hubs. *Transportation Research Part B* 39, 453–477.
- Saradhi, C., Subramaniam, S., 2009. Physical layer impairment aware routing (pliar) in wdm optical networks: issues and challenges. *IEEE Communications Surveys & Tutorials* 11, 109–130.
- Savasini, M., Monti, P., Tacca, M., Fumagalli, A., Waldman, H., 2007. Regenerator placement with guaranteed connectivity in optical networks. *Optical Network Design and Modeling LNCS* 4534, 438–447.
- Sen, A., Murthy, S., Bandyopadhyay, S., 2008. On sparse placement of regenerator nodes in translucent optical network. In: *Global Telecommunications Conference, 2008. IEEE GLOBECOM*. IEEE, pp. 1–6.
- Steiglitz, K., Weiner, P., Kleitman, D., 1969. The design of minimum-cost survivable networks. *IEEE Transactions on Circuit Theory* 16, 455–460.
- Stoer, M., Wagner, F., 1997. A simple min-cut algorithm. *Journal of the ACM* 44, 585–591.
- Taha, T., Taylor, G., 1994. An integrated modeling framework for evaluating hub-and-spoke networks in truckload trucking. *Logistics and Transportation Review* 30, 141–166.
- Taylor, G., Meinert, T., Killian, R., Whicker, G., 1999. Development and analysis of alternative dispatching methods in truckload trucking. *Transportation Research Part E* 35, 191–205.
- Uster, H., Kewcharoenwong, P., 2011. Strategic design and analysis of a relay network in truckload transportation. *Transportation Science* 45, 505–523.
- Uster, H., Maheshwari, N., 2013. Strategic network design for multi-zone truckload shipments. *IIE Transactions* 39, 177–189.
- Vergara, H.A., Root, S., 2013. Mixed fleet dispatching in truckload relay network design optimization. *Transportation Research Part E* 54, 32–49.
- Wagner, B., 2007. Model formulations for hub covering problems. *Journal of the Operational Research Society* 59, 932–938.
- Wang, Y.W., Lin, C.C., 2009. Locating road-vehicle refueling stations. *Transportation Research Part E* 45, 821–829.
- Wang, Y.W., Wang, C.R., 2010. Locating passenger vehicle refueling stations. *Transportation Research Part E* 46, 791–801.
- Xi, X., Sioshansi, R., Marano, V., 2013. Simulation-optimization model for location of a public electric vehicle charging infrastructure. *Transportation Research Part D* 22, 60–69.
- Yaman, H., 2009. The hierarchical hub median problem with single assignment. *Transportation Research Part B* 43, 643–658.
- Yaman, H., Kara, B.Y., Tansel, B., 2007. The latest arrival hub location problem for cargo delivery systems with stopovers. *Transportation Research Part B* 41, 906–919.
- Yang, X., Ramamurthy, B., 2005a. Dynamic routing in translucent wdm optical networks: the intradomain case. *Journal of Lightwave Technology* 23, 955–971.
- Yang, X., Ramamurthy, B., 2005b. Sparse regeneration in translucent wavelength-routed optical networks: Architecture, network design and wavelength routing. *Photonic Network Communications* 10, 39–53.
- Yetginer, E., Karasan, E., 2003. Regenerator placement and traffic engineering with restoration in gmpls networks. *Photonic Network Communications* 6, 139–149.
- Zarandi, M., Davari, S., 2011. Design of a reliable hub-and-spoke network using an interactive fuzzy goal programming. In: *2011 IEEE International Conference on Fuzzy Systems*. IEEE, 2955–2959.
- Zeng, B., An, Y., Zhang, Y., Kim, H., 2010. A reliable hub-spoke model in transportation systems. In: *Proceedings of the 4th International Symposium on Transportation Network Reliability*, Minneapolis, Minnesota, USA, 22–23 Jul 2010.