Title	A Decomposition Algorithm for the Ring Spur Assignment Problem
Author(s)	Carroll, Paula; McGarraghy, Sean
Publication date	2013-01
Publication information	International Transactions in Operational Research, 20 (1): 119-139
Publisher	Wiley
Item record/more information	http://hdl.handle.net/10197/9283
Publisher's statement	This is the author's version of the following article: Carroll, P., McGarraghy, S. (2012) A decomposition algorithm for the ring spur assignment problem, International Transactions on Operational Research, 20 (1) :119–139 which has been published in final form at http://dx.doi.org/10.1111/j.1475-3995.2012.00867.x
Publisher's version (DOI)	http://dx.doi.org/10.1111/j.1475-3995.2012.00867.x

A Decomposition Algorithm for the Ring Spur Assignment Problem

Paula Carroll and Seán McGarraghy

Centre for Business Analytics, Management Information Systems,

Quinn School of Business, University College Dublin, Belfield, Dublin 4, Ireland

paula.carroll@ucd.ie, sean.mcgarraghy@ucd.ie

This paper describes the Ring Spur Assignment Problem (RSAP), a new problem arising in the design of Next Generation Networks. The RSAP complements the Sonet Ring Assignment Problem (SRAP). We describe the RSAP, positioning it in relation to problems previously addressed in the literature. We decompose the problem into two IP problems and describe a branch-and-cut decomposition heuristic algorithm suitable for solving problem instances in a reasonable time. We present promising computational results.

Keywords: Networks; graphs; applications; combinatorics; Integer Programming formulation; Telecommunications Network Topology Design; Cutting Plane Algorithm

1. Introduction

We address an interesting new problem, the Ring Spur Assignment Problem (RSAP), introduced in (Carroll and McGarraghy, 2009a) and (Carroll and McGarraghy, 2009b). The problem is motivated by a practical situation: a telecommunications network operator seeks to identify an economical fault tolerant Next Generation Network (NGN) topology that can be overlaid on existing physical infrastructure. This problem arose in discussions with an industry partner who wished to identify a survivable backbone topology design in the physical network layer as part of an overall network upgrade plan. The operator wished to achieve this by exploiting existing spare capacity with no (or minimal) further capital investment. At about the same time, the Irish Government proposed a project to build a backbone network on the infrastructure of its local agencies aiming to connect EU funded Metropolitan Area Networks (MANs) which aim to provide broadband access in all areas of the country.

The solution topology we propose would be suitable for both the industry partner and the government project. In both instances, we seek to mine some value from the existing infrastructure and harness the benefits from emerging technology. A ring topology is preferred but if a ring solution is not possible we allow spurs. A sample RSAP topology is shown in Figure 1.
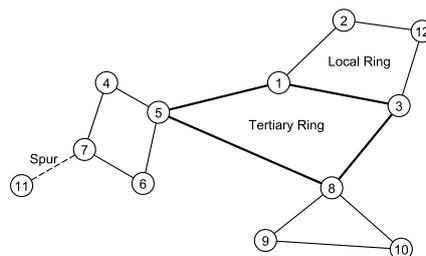

Figure 1: Sample RSAP topology

In this paper we provide a brief summary of some telecommunications background theory and related literature on topology design problems, focusing on ring-based topologies. We describe the RSAP in detail and explain how it relates to problems previously addressed in the literature. We describe how the problem can be decomposed into two IP sub-problems. We use an IP modelling approach which identifies feasible ring structures without having to resort to column generation. We describe our decomposition approach and our branch-and-cut implementation which links the two sub-problems. Finally, we summarise our computational results and give our conclusions.

2.   Background Material

Fibre optic cable allows speeds in the Tb/s range to be used in implementing Next Generation Networks (NGNs). However the higher the speed (bandwidth), the greater the loss if an individual cable or piece of equipment develops a fault. Thus, it has become mandatory for backbone networks to be designed with survivability in mind. Despite the *Dial before you dig* campaigns of many utility companies, it is still a frequent occurrence for cables to be accidently cut as noted in (Grover, 2003). Survivability issues are reviewed in detail in (Grover et al., 2002) and (Kerivin and Mahjoub, 2005).

Synchronous Digital Hierarchy (SDH) is a transmission standard which allows for ease of access to individual channels. SDH, also known as Synchronous Optical NETwork (SONET), provides a fast managed response to failures and so can provide the survivability protection required by modern sparse networks. SDH promotes the use of Self Healing Rings (SHRs) to increase network reliability. In SHRs, described in (Cosares et al., 1995), the node switching equipment is arranged in cycles (rings) connected by transmission links so that traffic affected by node or link failure in one part of the network can be routed on an alternative path. Cosares et al. (1995) note that many computationally challenging problems arise in the design and management of survivable networks.

Wavelength Division Multiplexing (WDM) is used on fibre optic networks to achieve higher bandwidths. Papadimitriou et al. (2001) give an introduction to optical networking issues and indicate that Internet Protocol (IP) over WDM may be the preferred option for NGNs. This protocol can be implemented over the physical SDH layer. The use of WDM introduces additional graph colouring type problems. Wavelength conversion equipment must be installed where traffic is routed across neighbouring rings (Bayvel, 2000). Dense Wave Division Multiplexing (DWDM) offers the potential of even higher bandwidths, (Brackett, 1990).

The Two Connected Bounded Ring problem (2CNBR) and its variants that arise in the design of SDH/SONET and WDM networks are described in (Fortz and Labbé, 2004; Fortz et al., 2000, 2006, 2003). This problem concerns designing a minimum cost network where at least two node-disjoint, or alternatively edge-disjoint, paths exist between every pair of nodes. Each edge of the network belongs to at least one cycle whose length (number of edges) is bounded by a given constant. This bound is imposed to ensure the quality of the telecommunications signal. As an alternative, the ring size could be specified in terms of the ring recovery time (Fumagalli et al., 2003) or ring length in km (Grover et al., 2002). Rings in the resulting 2CNBR topology are not necessarily disjoint but constraints that limit the number of rings that share an edge can be added.

The NP-Hard SDH Ring Assignment Problem (SRAP) is described in (Goldschmidt et al., 2003) as a high level design problem that seeks to identify which SHRs should be built; they choose to minimise network costs by minimising the number of disjoint rings while satisfying customer demand and satisfying a common ring capacity. A special ring, called the federal ring, of the same capacity as the other rings in the network, interconnects the other rings. Another formulation of SRAP as a set partitioning model with additional knapsack constraints is given in (Macambira et al., 2006).

Grover (2003) gives a comprehensive introduction to all of the transmission technologies mentioned above, along with a discussion on survivability issues that affect specific topologies. He describes the technical and engineering network background in detail, and reviews the computational techniques to solve the resulting network design optimisation problems.

We also mention the Ring Star Problem (Labbé et al., 2004). A Ring Star is used to connect terminals to concentrators where not all nodes are required to be 2-connected. A single node is a designated hub node that must be connected to the ring. All other nodes can either be assigned to the ring using concentrators or assigned to a node on the ring. The objective is to minimise the total cost of the ring and star assignments. The authors give an exact algorithm and a polyhedral analysis of the RSP. They also present computational results of a branch-and-cut implementation.

We mention an extension to the RSP, the Capacitated m-Ring-Star Problem (CmRSP) described in (Baldacci et al., 2007) in relation to the design of an optical network in an urban area in an Italian city. The problem consists of finding m node disjoint ring stars that visit a central depot. The number of customers allocated or visited in a ring cannot be greater than some capacity Q. The objective is to minimise the total ring star assignment costs. The authors give two ILP formulations, a two index edge and a two commodity flow formulation and they develop a branch-and-cut algorithm. They assess the effectiveness of the two formulations in a branch-and-cut framework and show the equivalence of the formulations, i.e., the two models represent the same solution. They report that their algorithms can handle instances up to about 100 nodes. Naji-Azimi et al. (2010) give a heuristic method for solving larger CmRSP problem instances, while Hoshino and de Souza (2009) give an exact branch-and-cut-and-price algorithm which they report outperforms the branch-and-cut implementation of Baldacci et al. (2007).

Finally, we note that a polyhedral approach is used to solve many of these combinatorial optimisation problems. The polyhedral approach is described in detail in (Lawler et al., 1985) and (Nemhauser and Wolsey, 1988). The degree constraints of the traditional TSP formulation force each node to have exactly two incident edges while the Subtour Elimination Constraints (SECs) require that any proper subset of nodes be connected by at least two edges.

A core task in the polyhedral approach is finding the minimum cut of a graph. Grötschel et al. (1995) note that the Gomory-Hu tree of min cuts can be generated in polynomial time and used to identify violated subtour constraints. We note that some house-keeping is required for this approach as the Gomory-Hu algorithm is intended for use on a directed graph since it in turn uses repeated calls of the Ford-Fulkerson max-flow algorithm. In (Stoer and Wagner, 1997), the authors describe a deterministic non-flow based min cut algorithm based on the principle of maximal adjacency. It takes as input a graph adjacency matrix and returns the min

cut value. It runs through a number of iterations, finding a cut at each iteration, the smallest of which is the min cut of the graph.

Grötschel in Lawler et al. (1985) describes a (perfect) 2-matching as a set of edges such that every vertex is contained in at most (exactly) 2 edges, hence every TSP tour is a perfect 2-matching. Padberg and Rao (1982) give a polynomial time exact algorithm for the separation of 2-matching inequalities.

Our brief summary gives a flavour of just some of the wide variety of approaches, topologies and design parameters used in survivable network design and how emerging DWDM technology is posing new optimisation problems. In particular, we distinguish between approaches that are demand — and hence capacity — driven, yielding solutions in the logical layer and approaches that are topologically focused to provide solutions at the physical layer.

3.   The RSAP

The seven layer Open Systems Interconnection (OSI) model is a standardised framework for data exchange. In this paper we focus on selecting a survivable topology in the physical layer of the OSI model. A node with (exactly) k incident edges is said to be (*exactly*) *k-connected*. We seek to identify a 2-connected (disjoint ring) topology to ensure survivability. If a suitable topology can be identified in the physical layer, the logical network can be implemented and managed at the logical layer.

We wish to use pre-installed capacity in the physical SDH network to minimise costs. Our aim of identifying a highly resilient topology at minimum cost is achieved by assigning locations to rings. Traffic demand for such a project cannot be known with certainty, so we focus on the topological aspects of the problem. Our work can be classified amongst approaches providing physical survivability but differs from other works since we aim to exploit existing physical infrastructure where possible. We focus our attention in this paper on the design aspect. As such, the RSAP is a complement of the SRAP.

We now describe the RSAP in detail. Communities of interest, defined in (Cosares et al., 1995) as geographically close nodes that have high traffic demands between them, are identified and their traffic demands are estimated. If such communities can be clustered on node disjoint rings, no wavelength conversion is required, eliminating the cost of wavelength conversion and/or opto-electronic conversion equipment for intra-ring demand. We call these rings *local rings*. All ring nodes must be exactly 2-connected to two other ring nodes.

Local rings are then connected by a special ring, which we call the *tertiary ring*, often called the federal or backbone ring in the literature. *Tertiary* is a legacy naming convention used by this operator to signify the highest level in the physical infrastructure. The tertiary ring facilitates inter-ring demand; wavelength converters are required where local rings connect to the tertiary ring. Early discussions with the network operator included many design options for the interconnection of the tertiary ring with the local rings. The chosen criteria were that at least one node from each local ring must be on the tertiary ring and the tertiary ring must form a single simple cycle. Other possible options could specify that at least two nodes of each local ring must be on the tertiary ring or that no local ring edges may be used on the tertiary ring.

So far, the problem described is similar to the SRAP problem; we are identifying rings that can carry

the estimated demand. However, no SRAP solution is possible in some real world instances, as the following example shows. For adjacent nodes i and j, let $e_{i,j}$ denote the edge joining them.
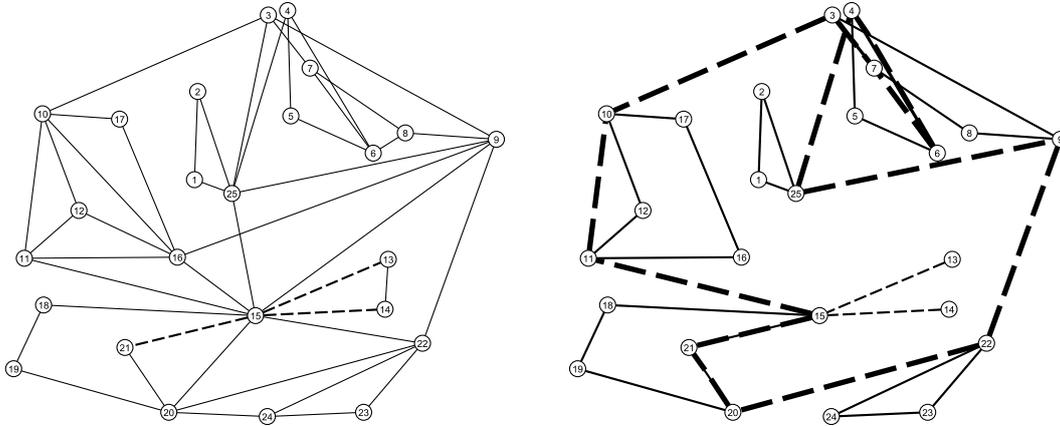

Figure 2: Left: Benchmark problem with no SRAP solution.   Right: RSAP topology

Figure 2 (left) shows *france*, a benchmark problem from (Orlowski et al., 2010). We can see that nodes 13, 14 and 21 are 2-connected and share node 15 as an adjacent node. An SRAP solution could start by connecting edges $e_{13,15}$ and $e_{14,15}$ to attempt to form a ring. Since node 21 is exactly 2-connected and adjacent to node 15, edge $e_{21,15}$ must also be included in a ring; but this violates the requirement that ring nodes be exactly 2-connected, since node 15 is now three-connected; hence, no SRAP solution exists for this problem instance.

As an alternative, where no SRAP solution exists, we allow locations that have insufficient spare capacity or no possible physical route due to limitations of geography, to be connected to SHRs by spurs off the local rings. Spur nodes must be connected to a local ring by a single edge, i.e., we do not allow a chain of edges to connect spur nodes. We call this problem the *Ring Spur Assignment Problem* (RSAP).

An example of the RSAP solution topology for *france* is shown in figure 2 (right). The tertiary ring edges are shown with heavy dashed lines, local rings as thin lines and spur arcs as light dashed lines. For clarity, we omit any edges of the graph that are not part of the solution topology.

An instance of the RSAP is specified by:

- an undirected graph $G = (V, E)$ defined on a set V of nodes, labelled from 1 to n where $n = |V|$, $n \geq 6$ to exclude degenerate cases, and a set of undirected edges E; the underlying set A of oriented arcs contains, for each edge $\{i, j\} \in E$, two arcs (i, j) and (j, i), one in each direction.

- a pair of positional co-ordinates for each node $i \in V$ that allow us to assign positional reference values for the end nodes of each edge $\{i, j\}$. These reference values signify the position of end nodes with respect to each other;

- a non-negative routing cost giving the cost of a link dependent on its length and capacity. Let $c_{ij} \geq 0$ be the cost coefficient of edge $\{i, j\} \in E$ for ring edges. If arc (i, j) is assigned as a spur, then it will be given a cost bc_{ij}, where the penalty factor b is as in §4 below.

5

The RSP topology presented in (Labbé et al., 2004) is a single ring star with no bound on the ring. In the RSAP, there is another level of hierarchy, where bounded local rings are connected to the backbone ring, and spurs then emanate from the local rings. A solution to our problem is a set of disjoint bounded ring stars interconnected by a tertiary ring. In addition, we note that a solution with a single ring and many spurs is not desirable from a reliability point of view so all RSAP solutions consist of at least two local rings interconnected by a tertiary ring. We also note that our results have practical applications in location-allocation and rapid transit network design problems.

4. IP Decomposition Approach

A practical approach to solving large combinatorial problems is to decompose the whole problem into more manageable connected subproblems. While this approach may yield a local rather than the global optimum, the approach provides results for practical problems within a reasonable time frame and is frequently used in network design problems (Cosares et al., 1995; Fumagalli et al., 2003).

We decompose the RSAP into two IP sub-problems; sub-problem 1 (SP1), of finding a minimum cost local ring/spur partition and sub-problem 2 (SP2), of finding a minimum cost tertiary ring to interconnect the local ring/spur partition solution from SP1. We describe IP formulations for both sub-problems.

SONET standards set the recommended maximum number of pieces of node switching equipment (Add-Drop Multiplexers (ADMs)) per ring at sixteen but, in this practical application, local rings are restricted to having no more than eight nodes, which we denote by RB for Ring Bound.

Since local rings can have a minimum of three nodes and a maximum of eight, it would be impractical to consider all possible feasible rings. We do not explicitly identify a subset of rings to be considered, the approach taken by some authors e.g., Fortz et al. (2003) or resort to column generation to identify potential rings, an approach used by e.g., Thomadsen and Stidsen (2005). Instead, our formulation ensures that feasible node disjoint local rings are constructed, by ensuring that all local ring nodes are exactly 2-connected and that rings contain no more than eight nodes.

Goldschmidt et al. (2003) propose an upper bound of n on the number of rings to be used. Since any valid ring must contain at least three nodes, we note that $\lfloor n/3 \rfloor$ is an upper bound. We do not predetermine the number of rings but give the set of ring indices $R := \{1, \ldots, \lfloor n/3 \rfloor\}$.

SP1 can be formulated as a binary integer programming problem as follows: Let x_{ijr} be a binary variable equal to 1 if and only if edge $\{i, j\}$ appears on ring r, and equal to 0 otherwise; i.e., both i and j are assigned to the same ring $r \in R$.

For each arc $(i, j) \in A$, let y_{ij} be a binary variable equal to 1 if and only if vertex i is assigned to vertex j as a spur; we set $y_{ii} = 1$ for any vertex i that is on a ring. Let z_{ij} be a binary variable equal to 1 if and only if edge $\{i, j\}$ appears on the tertiary ring.

The set of nodes adjacent to node i is denoted by $\mathrm{adj}(i)$. The *cut* of $S \subset V$ is denoted by $\delta(S) := \{\{i, j\} \in E : i \in S, j \notin S\}$, i.e., the set of edges having only one endpoint in S. We say a cut is *odd* if $|\delta(S)|$ is odd, *even* otherwise. The set of nodes on the ring of index r is denoted by $N(r)$ and the set of edges that form the

ring of index r is denoted by $E(r)$. For simplicity, we refer to ring r, meaning the ring of index r. The support graph G_r associated with ring r is $G_r := (N(r), E(r))$ i.e., those edges for which $x_{ijr} > 0$. The support graph G_T associated with the tertiary ring is $G_T := (N(T), E(T))$ i.e., those edges for which $z_{ij} > 0$.

As previously noted, we wish to foster high resilience by having locations assigned to rings where possible. By assigning a sufficiently high weight, b_{ij}, to links that are spurs, we achieve this objective. A similar approach is used in (Labbé et al., 2004).

First, consider the case shown in Figure 3. We prefer the use of e_{ij} and e_{jk} as ring edges (the leftmost arrangement) over the spur assignment of j to i or j to k. Then, for every clique of nodes $\{i, j, k\} \in V$,
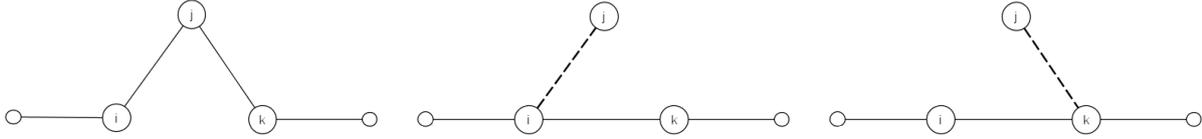

Figure 3: Ring versus Spur assignment, Case 1

$$c_{ij} + c_{jk} < b_{ji} + c_{ik} \tag{a}$$

and
$$c_{ij} + c_{jk} < b_{jk} + c_{ik}, \tag{b}$$

so
$$c_{ij} + c_{jk} - c_{ik} < b_{ji} \tag{c}$$

and
$$c_{ij} + c_{jk} - c_{ik} < b_{jk}. \tag{d}$$

Thus, we set

$$b_{ji} := c_{ij} + c_{jk} - c_{ik} + \varepsilon \tag{e}$$

where ε is some positive value, chosen to be small.

Now, consider the case shown in Figure 4. Again, we prefer the ring solution on the left of Figure 4 over the spur solution on the right. For every pair of adjacent nodes j, l which could be assigned as either ring nodes or spur nodes (assigned to i, k respectively), choose b_{ji} and b_{lk} so that
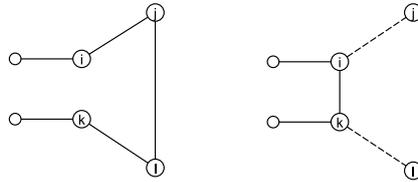

Figure 4: Ring versus Spur assignment, Case 2

$$c_{ij} + c_{jl} + c_{lk} < b_{ji} + b_{lk} + c_{ik}. \tag{f}$$

Thus, for every arc (j, i) it suffices to choose

$$b_{ji} = c_{ij} + c_{jl} + c_{lk} - c_{ik} \tag{g}$$

7

since b_{lk} will be positive. Therefore, to ensure b_{ji} is sufficiently high to guarantee a ring solution where one is possible, we set b_{ji} to the larger of (e) and (g). For simplicity, we let $b = \max_{ij \in A}\{b'_{ij}\}$ where $b'_{ij} := b_{ij}/c_{ij}$, and set the linear coefficient of each arc $(i,j) \in A$, to be bc_{ij} in our objective function, i.e., the cost of using a spur arc is the network cost of that edge, c_{ij}, multiplied by the penalty weighting value of b for the network.

We now address the topology considerations of the local ring spur partitions and specify the formulation for SP1 as follows:

$$SP1 \quad \min \sum_{\{i,j\} \in E} \sum_{r \in R} c_{ij} x_{ijr} + \sum_{(i,j) \in A} bc_{ij} y_{ij} \tag{1}$$

subject to Local Ring Topological constraints:

$$\sum_{j \in \text{adj}(i)} y_{ij} + y_{ii} = 1 \qquad \forall\, i \in V \qquad \text{Assignment} \tag{2}$$

$$\sum_{j \in \text{adj}(i)} \sum_{r \in R} x_{ijr} = 2y_{ii} \qquad \forall\, i \in V \qquad \text{Node Connectivity} \tag{3}$$

$$\sum_{l \in \text{adj}(j), l \neq i} x_{ljr} \geq x_{ijr} \qquad \forall\, \{i,j\} \in E, r \in R \qquad \text{Edge Connect (Head)} \tag{4}$$

$$\sum_{k \in \text{adj}(i), k \neq j} x_{ikr} \geq x_{ijr} \qquad \forall\, \{i,j\} \in E, r \in R \qquad \text{Edge Connect (Tail)} \tag{5}$$

$$\sum_{\{i,j\} \in E} x_{ijr} \leq RB \qquad \forall\, r \in R \qquad \text{Ring Bound} \tag{6}$$

$$\sum_{r \in R} x_{ijr} + y_{ji} \leq y_{ii} \qquad \forall\, i \in V, j \in \text{adj}(i) \qquad \text{Spur Assign} \tag{7}$$

$$x_{ijr} \in \{0,1\} \,\forall\, \{i,j\} \in E, r \in R \qquad y_{ij} \in \{0,1\} \,\forall\, (i,j) \in A,\, y_{ii} \in \{0,1\} \,\forall\, i \in V \qquad \text{Binaries} \tag{8}$$

We wish to ensure that each local ring consists of a single simple cycle of the edges assigned to that local ring. A *subtour* on local ring r is defined as a simple cycle on a proper subset of the edges assigned to ring r. The traditional TSP formulation enforces connectivity with degree constraints. However, use of degree constraints alone does not prohibit subtours. Our formulation above also allows for subtours; local rings are connected but may be split into two or more cycles. In the next section we describe how these subtours can be eliminated.

Note also that if no nodes are assigned as spurs, the resulting topology is a set of disjoint local rings similar to an SRAP topology albeit without having yet verified that the topology capacity supports the estimated demands. The assignment constraints (2) ensure that each site is assigned, either as a ring node (assigned to itself) or as a spur (assigned to another site). Connectivity constraints (3) are used to ensure that any node assigned as a ring node with $y_{ii} = 1$ is exactly 2-connected, i.e., exactly two ring edges x_{ijr} are incident to node i.

Using only constraints (2) and (3) allows an assignment $x_{ijr_1} = 1 = x_{ikr_2}$, i.e., Node Connectivity is satisfied but the local rings are not correctly constructed so Edge Connect (Head) constraints (4) and Edge Connect (Tail) constraints (5) are added to force the cut of any active ring edge to be at least one on the same ring. We can consider these to be dis-aggregated connectivity constraints.

Rings are restricted to having no more than RB nodes by the Ring Bound Constraints (6). The Spur Assignment Constraints (7) ensure that any node i is on a ring if any node j is assigned to i as a spur and

is strengthened by noting that node j cannot be assigned as a spur to node i and adjacent to it on any ring. Finally, constraints (8) ensure that the decision variables are binary integers.

For each integer solution found to SP1, we solve an instance of SP2, a Generalised Travelling Salesman Problem, to find a tertiary ring that interconnects the local rings.

SP2 can be solved using the following model:

$$\text{SP2} \quad \min \quad \sum_{\{i,j\} \in E} c_{ij} z_{ij} \tag{9}$$

subject to:

$$\sum_{k \in \text{adj}(i), k \neq j} z_{ik} \geq z_{ij} \qquad \forall \, \{i,j\} \in E \tag{10}$$

$$\sum_{l \in \text{adj}(j), l \neq i} z_{lj} \geq z_{ij} \qquad \forall \, \{i,j\} \in E \tag{11}$$

$$\sum_{j \in \text{adj}(i)} z_{ij} \leq 2 \qquad \forall \, i \in V \tag{12}$$

$$z_{ij} + y_{ij} + y_{ji} \leq 1 \qquad \forall \text{ spur arc } (i,j) \text{ of SP1} \tag{13}$$

$$\sum_{i \in N(r)} \sum_{j \in \text{adj}(i), j \notin N(r)} z_{ij} \geq 2 \qquad \forall \text{ ring } r \text{ of SP1} \tag{14}$$

along with the Binary Integer constraints:

$$z_{ij} \in \{0,1\}, \quad \forall \, \{i,j\} \in E. \tag{15}$$

The head of each tertiary ring edge is forced to be connected to another tertiary edge by constraints (10). Similarly for the tail with constraints (11). We limit the use of each node to at most two incident edges with (12). The next two constraints are explicitly written for the particular SP1 solution. Constraints (13) ensure that no spur node forms part of the tertiary ring as this would be unreliable. Constraints (14) force at least two tertiary edges to be incident with each local ring identified in SP1. In other words, the cut of the tertiary ring across each local ring is at least two. Finally, constraints (15) are the binary integer constraints.

Let $V_T \subset V$ be the subset of nodes of the SP1 local rings that connect the local rings to the tertiary ring by constraints (14). In the case of the tertiary ring, we wish to form a single simple cycle on all nodes of V_T, and to avoid having the tertiary ring split into multiple cycles. A subtour on the tertiary ring is defined as a simple cycle on a proper subset of V_T. As described, our formulation enforces connectivity of the nodes of V_T but allows subtours on the tertiary ring. In the sections below we describe how tertiary ring subtours can be eliminated.

4.1 Additional Valid Inequalities

Our decomposition algorithm uses a branch-and-cut approach to SP1. Subtour elimination constraints are valid and necessary for ring subsets, i.e., each ring must form a single simple cycle. However, in the case of the RSAP we do not know which local rings will form the optimal SP1 solution, so we need modified versions of the known SECs. In this section we address how the running time can be improved by tightening the lower LP bound of

SP1 with valid inequalities added at the root node and how subtours on the SP1 local rings and SP2 tertiary rings are eliminated during the branch-and-cut search.

We use the following approach; for each local ring $r \in R$, create the support graph G_r associated with ring r and calculate the min cut of G_r. Details of how we calculate the min cut are given in §4.2 below. If the min cut is zero, we have detected subtours on G_r, i.e., on ring r.

Let us assume we detect subtours r_a and r_b on a specific ring r', as shown in the example in Figure 5. We partition the nodes of r' into subsets $N(r_a)$ and $N(r_b)$, and the edges of r' into subsets $E(r_a)$ and $E(r_b)$. Correspondingly, we denote the subtour partitions of $G_{r'}$ by G_{r_a} and G_{r_b} i.e., $N(r') = N(r_a) \cup N(r_b)$ and $E(r') = E(r_a) \cup E(r_b)$. Now, let $\{k, l\}$ be an edge in $E(r_b)$. If this edge is in r' then, clearly, not all edges in $E(r_a)$ can be in r' since this would cause the formation of a subtour in r'. From this observation, the following valid inequality can be derived:

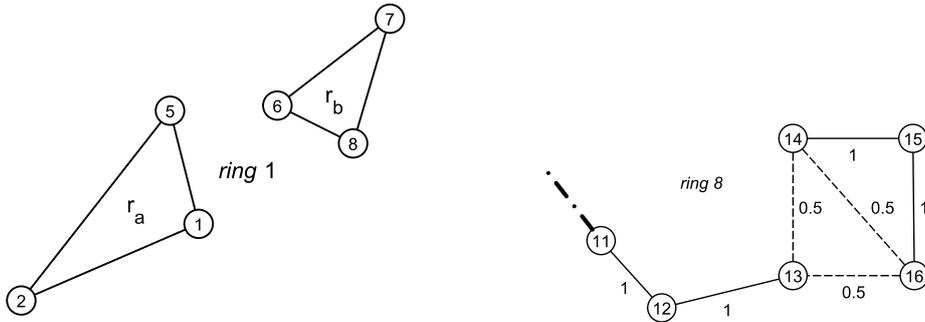$$\sum_{\{i,j\} \in E(r_a)} x_{ijr'} \leq |r_a| - x_{klr'} \tag{16}$$

Figure 5: Left: Subtour example, Right: 2-matching example

Back to Figure 5, let ring 1 and the edge $\{6, 7\}$, respectively, play the roles of ring r' and of edge $\{k, l\}$ in Eq. (16). The corresponding constraint reads: $x_{1,2,1} + x_{1,5,1} + x_{2,5,1} \leq 3 - x_{6,7,1}$. This says that a cycle on ring 1 of the edges $\in E(r_a)$ can only be formed if the edge $\{k, l\} \in E(r_b)$ on ring 1 is set to zero. If the edge $\{k, l\}$ is not set to zero on ring 1, then not all the edges of $E(r_a)$ can be on ring 1.

Likewise, we can write the constraint in respect of r_b: $x_{6,7,1} + x_{6,8,1} + x_{7,8,1} \leq 3 - x_{1,2,1}$.

We note that this inequality is valid for all ring indices, not just the ring where the subtour is detected. In our examples, the inequalities are valid for all $r \in R$, not just on $r = 1$.

The right hand side of the traditional SEC is $|r_a| - 1$. Our modified SEC is not as strong as the traditional form since in a fractional solution, the value of the edge decision variable $x_{klr'}$ may be less than one.

The 2-matching constraints force the number of edges across a cut to be even. The traditional 0-1 formulation of the TSP uses $x_e \in \{0, 1\}$ for all $e \in E$ and for $E' \subseteq E$, defines $x(E') := \sum_{e \in E'} x_e$. In the traditional formulation, 2-matching constraints are given by:

$$\sum_{E(H)} x_e + \sum_{E'} x_e \leq |H| + \left\lfloor \frac{|E'|}{2} \right\rfloor$$

where $H \subset V$ is called the handle and $E' \subset E$ is an odd set of k disjoint edges, $k \geq 3$, with exactly one end in H, known as teeth.

Nemhauser and Wolsey (1988) note that the 2-matching constraints are dominated by the subtour inequalities when the number of teeth is 1. The 2-matching inequalities can be derived by taking linear combinations with weights of 0.5 on the degree constraints of the handle nodes, weights of 0.5 for teeth edge trivial upper bound inequalities and rounding down the RHS. Except for the rounding of the RHS, the 2-matching inequality with one tooth can be obtained by similar linear combinations. Moreover, even after rounding down the RHS, this leads to an inequality dominated by a traditional subtour elimination constraint. Note that the solution on the right of Figure 5 violates the traditional SECs but not our SECs modified for the RSAP.

The 2-matching constraints are valid for the local rings of SP1, but again, modified versions are needed. $T \subset E$ is an odd set of disjoint edges, known as teeth, each with exactly one end in H. In our case, the use of the dis-aggregated connectivity constraints allow a violated 2-matching on a single tooth. For the RSAP, $|T| \geq 1$, i.e., we may have a violated 2-matching inequality on one tooth while 2-matching constraints on a single tooth are redundant in the traditional TSP formulation.

We use the following modified form of the 2-matching constraints. In (17), r' is a ring where a violated 2-matching constraint is detected.

$$\sum_{E(H)} x_{ijr'} + \sum_{T} x_{ijr'} \leq \sum_{i \in H} y_{ii} + \left\lfloor \frac{|T|}{2} \right\rfloor \qquad (17)$$

On the right of Figure 5 we show an example of a violated 2-matching with one tooth on $r = 8$, ring 8. In this example $H = \{13, 14, 15, 16\}$ and $T = e_{12,13}$, $y_{ii} = 1$ for all $i \in H$. Note that the solution given in Figure 5 not only satisfies all the inequalities in the system, Eqs. 2 to 7, but also does not violate the subtour elimination constraint on vertices 13–16.

Since the ring edges of the SP1 LP solution may be fractional, local rings of the LP solution might not be disjoint and we observe a *ghosting* effect as in Figure 6. We define a ghost ring r_g as a simple cycle on the fractional ring edges of the SP1 LP solution with more than the ring bound number of edges, i.e., $|E(r_g)| > RB$, the ring bound.
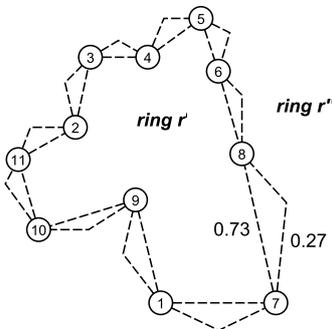

Figure 6: Ghost rings

Figure 6 shows an example, where two local rings, r' and r'' of size 11 coincide, $y_{ii} = 1$ for all 11 nodes with

$x_{ijr'} \approx 0.73$ for all edges on ring r', $x_{ijr''} \approx 0.27$ for all edges on ring r''. In this example, $E(r_g) = E(r') \cup E(r'')$, note that $|r'| = |r''| = 11 > RB$. Both rings r' and r'' satisfy the ring bound constraint (6) but there are more than the allowed number of sites on the ghost ring r_g. Effectively, the set of edges $E(r_g)$ form an infeasible ring, since it exceeds the ring bound. Our objective is to achieve bounded rings but long cycles in the fractional solution can emerge in our formulation.

Such *ghosts* can be eliminated by observing that at most $|E(r_g)| - 2$ of the edges on the ghost ring r_g can be used in an optimal solution to form a feasible ring. Standard SECs break a subtour by forcing at least one edge of the subtour to be dropped. In the case of bounded rings, dropping one edge would leave a path, in our example of 10 edges. These 10 edges forming a path cannot form a bounded ring.

We observe that $\lceil |r_g|/RB \rceil$ is a lower bound on the number of rings that would be required to cover the nodes of a ghost ring. Thus, we obtain a stronger constraint by forcing at least $\lceil |r_g|/RB \rceil$ edges of r_g to be dropped, as follows:

$$\sum_{\{i,j\} \in E(r_g), r \in R} x_{ijr} \leq |r_g| - \left\lceil \frac{|r_g|}{RB} \right\rceil \tag{18}$$

We call these constraints Ghost Elimination Constraints (GECs) and note that they are another refinement of SECs.

Finally, we address subtours on the SP2 tertiary ring. Recall that SP2 is a generalised TSP problem for each SP1 local ring/spur partition. We ensure that at least one ring node from each SP1 local ring connects the local ring to the tertiary ring with Eqs. (14). Figure 7 shows an example of subtours detected on the tertiary ring connecting the local rings of an SP1 solution. In this example SP1 consists of four local rings. Let $E' \subset E$ be the set of edges of a subtour of the tertiary ring. For such a set E' we add a tertiary ring SEC similar to the traditional TSP SECs, (19).

$$\sum_{\{i,j\} \in E'} z_{ij} \leq |E'| - 1 \tag{19}$$
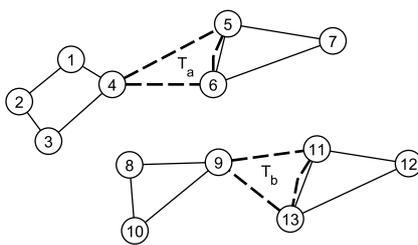
Figure 7: Tertiary ring subtour interconnecting SP1 solution

We identify two tertiary ring SECs in our example in Figure 7: $z_{4,5} + z_{4,6} + z_{5,6} \leq 2$ and $z_{9,11} + z_{9,13} + z_{11,13} \leq 2$.

4.2 The Algorithm

In this section we describe the main components of our branch-and-cut decomposition algorithm which links the two sub-problems SP1 and SP2. The pseudo code of our algorithm is given as Algorithm 1, comments are

shown in curly braces. Our branch-and cut approach to SP1 is as follows: we solve the relaxed SP1 LP. Then at the root node we separate and add SECs Eq. (16), 2-Matching Eq. (17), and GECs Eq. (18) in that order. If the resulting SP1 solution is still fractional, we run a global branch-and-cut search on the amended SP1 problem separating SECs Eq. (16) and 2-Matching Eq. (17) at each node. We solve an instance of SP2 at any integer node in the SP1 branch-and-cut tree.

In our implementation, we create one instance of SP2 since the objective function and constraints, (10), (11) and (12) are the same for all tertiary ring sub-problems. For each integer local ring spur partition solution to SP1 we modify SP2 by adding constraints of type (13) and (14). We solve the SP2 IP adding SECs Eq. (19) as necessary.

Algorithm 1 Decomposition Algorithm

Initialise data, calculate b

create SP1 and SP2 (less Eq.s (13) and (14))

Solve LP (SP1: Local Ring Spur Partition)

while fractional cuts and iterations < MaxIter **do**

    separate and add cuts { modified Local SECs, 2-Matchings, GECs}

end while

if solution is fractional **then**

    call branch-and-cut { modified local ring SECs and 2-Matchings}

    for each integer node of SP1 branch-and-cut **do**

        add constraints of type (13) and (14) to SP2

        solve SP2 IP adding tertiary SECs (19) as necessary

        reset SP2 {remove constraints (13), (14) and (19)}

        save best solution

    end for

else

    add constraints of type (13) to SP2 and (14) to SP2

    solve SP2 IP adding tertiary ring SECs (19) as necessary

    save solution

end if

We save the first complete integer (local and tertiary) solution, if one is found, as the best so far. If any subsequent integer solution is of lower cost than the best so far, it becomes the incumbent best solution. With a decomposition approach it is possible that a globally optimal solution, consisting of a relatively higher local ring cost interconnected by a lower cost tertiary ring, could be missed. By investigating the higher costing SP1 integer solutions, our heuristic attempts to compensate for the weakness of the decomposition approach.

Next we describe our cut separation procedures. Since we only need to identify violated subtours on any ring rather than generate all min-cuts in the solution, we use a modified version of the min cut algorithm of (Stoer and Wagner, 1997). We modified the Stoer Wagner algorithm to run on the support graph G_T for the

tertiary ring (and G_r for each local ring). We track the cuts at each iteration using data structures to store the number of edges in a cut, the cut value and the resulting partitions. We interrupt the min cut algorithm if a zero value cut is detected. Otherwise, the modified Stoer Wagner algorithm runs to completion and in both cases we have access to the stored cuts. A subtour is detected when the min cut of G_T (G_r) is zero.

In the case of subtours of the SP2 tertiary ring we add tertiary ring SECs Eq. (19). It may not be possible to find a tertiary ring to interconnect this SP1 integer solution, in which case we move on the next SP1 integer solution or terminate if there are no further integer SP1 solutions to consider. Having processed the current integer SP1 solution, we then remove those SP1-related constraints of type (13), (14) and (19) from the SP2 model.

In the case of the local rings of SP1, for each ring at each node of the branch-and-cut search, we use the modified check Stoer Wagner algorithm to identify a min cut of zero and check for a violated SEC Eq. (16).

We use the following heuristic to separate 2-matching inequalities on the SP1 local rings: having modified Stoer Wagner's min cut algorithm we have access to any odd cuts detected on G_r during the SEC separation. We note that unless the odd cut detected by our modified Stoer Wagner algorithm is the min cut of the graph, it is not necessarily the minimal odd cut but any odd cut found on the fractional ring edges is a violation of some sort and is worthy of investigation. We check if any odd cut gives a violated 2-matching Eq. (17) to be added to SP1.

Since there are potentially exponential numbers of both feasible and infeasible rings on a fractional SP1 solution, we use a simple heuristic to detect and add GECs given by Eq. (18); other approaches are also possible. Recall that a ghost ring is a cycle in the fractional solution which exceeds the ring bound number of edges. For each ring index r in the fractional SP1 LP solution, we create the support graph G_r, select any 2-connected node as the start node and attempt to trace a path that returns to the start node to form a simple cycle along the edges of G_r.

Our cycle construction heuristic uses a greedy approach and simply looks at the next node on the path. Our rationale is that one way to to trace a potential ghost cycle is to attempt to steer around the contour of the edges of G_r. At any intermediate node of degree greater than two, we select the next node as the unvisited node that makes the biggest clockwise angle on our current path, by looking in the appropriate quadrant as described below. This choice for navigation was motivated by empirical evidence in the fractional LP solutions. We continue tracing the path until there are no more adjacent unvisited nodes. If we have returned to the start node, we have traced a cycle which may exceed the ring bound.

To select the outermost node in the GEC separation procedure we make use of the positional data of the nodes. Similarly to the idea of using quadtrees in computer graphics, but using simple quadrants, we assign positional reference values for each end node of each edge which signify the position of end nodes with respect to each other. Each node i is assigned as being north west Q_1, north east Q_2, south east Q_3 or south west Q_4 of any adjacent node j.

In Figure 8 we show an example where node i has been selected as the start node, it is two connected and its first adjacent node is node j. So the first edge of the ghost path is e_{ij}. Node j is adjacent to four nodes; the ghost path so far has come from the direction of node i so we wish to select the next node on the path as

the one that makes the biggest angle with the incoming path edge in a clockwise direction. In our example, we wish to leave by edge e_{jk}. Node i is in Q_4 wrt node j so we select as the next node any node in Q_3, in our example, node k.
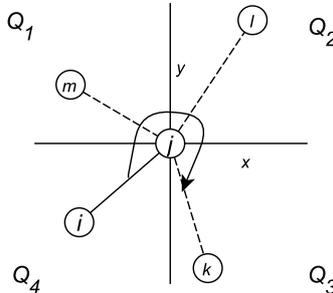

Figure 8: Selecting next node in GEC separation

In this example, if the current lead node has no adjacent node in Q_3, we select the next best as any node in Q_2 and failing that, any node in Q_1. If there are more than two adjacent nodes in a quadrant, we take the first we find rather than the incurring the cost of calculating the angles between the candidate edges. In summary, we check the quadrant of the incoming path against the quadrant of possible outgoing edges and select the outermost as the next edge to add to the path.

If the path returns to the start node or meets a node already visited, we stop the GEC separation heuristic. If the heuristic has stopped because we have returned to the start node, we check whether the sum of the ring edges on the candidate ghost cycle give a violated GEC and if so, add the GEC to the $SP1$ problem definition. We may only have detected a cycle on the fractional ring solution with less nodes than the ring bound so this tolerance check is necessary to avoid the addition of an invalid constraint. If we failed to form a cycle or the cycle thus formed uses less than the ring bound number of edges, no GEC has been detected on this fractional ring and we proceed to check the next ring with a fractional solution.

5. Results

We present promising computational results. A complete set of our results is downloadable from `http://mis.ucd.ie/Members/pcarroll`. Small problems can be solved in a reasonable amount of time and we achieve the desired effect of favouring ring topologies where they exist. We set a time limit of three hours for SP1 for larger problems (of more than 40 nodes), interrupt the branch-and-cut search if it has not completed, and report the best integer solution found by that time.

All code was written in ANSI C, using Xpress-MP suite 7.2 with Xpress-BCL version 4.4.0 Builder Component library routines and Xpress-Optimizer 22.01 and run on a 32 bit Toshiba Satellite Pro with Intel Dual Core Pentium 1.86GHz processors and 2 G of RAM under Windows Vista. The release of the Xpress 7.2 suite (May 2011) includes a facility to exploit parallel processing in the branch-and-bound tree. This is achieved

by synchronising the BCL and optimiser problems at the start and end of each callback, access to the BCL problem is locked to the particular thread in between these two function calls. We allowed two threads in our implementation to exploit the dual core hardware.

Xpress-MP uses floating point arithmetic to encode real numbers. We settled on a tolerance value of 10^{-6}. In our separation routines, if the value of the left hand side (LHS) of an inequality differs from the right hand side (RHS) value by less than this tolerance level, we deem that the constraint is satisfied and we have failed to detect a violated inequality. Using a lower tolerance level proved inefficient in testing as it allowed too many cuts to be added without improving the objective function value.

The test data used was SNDlib (Orlowski et al., 2010), since it provides many real world problem instances with both a network model and positional co-ordinates for each node. The problem instances have associated costs for capacities that can be installed on the edges of the graph. Some instances have pre-installed capacities, in other instances no capacity is pre-installed so we install one unit of the lowest capacity available at the costs specified to allow us to test our algorithm. Two problems, *janos-us-ca* and *zib54* were integer infeasible for SP1, and so were omitted from further testing. These two networks have a small number of nodes of very high degree making them unsuitable for the RSAP topology.

We got the best results from the cutting plane algorithm by allowing a maximum of 50 iterations at the root node where modified SECs Eq. (16), 2-Matching Eq. (17) and GECs Eq. (18) are separated in that order. Modified SECs Eq. (16) and 2-Matching Eq. (17) are then separated at each node of the SP1 branch-and-cut tree. Empirical testing of the GECs in the branch-and-cut showed that in some cases they improved the run time but in many cases they caused a deterioration in run time. Recall that we use a heuristic to separate the GECs. It is possible that a better heuristic would justify their inclusion in the branch-and-cut but in our algorithm, we restrict the use of the GECs to the root node.

Computational results from SP1 (the local ring spur partition problem) are shown in Table 1. Columns from left to right show the problem name and size (the number of nodes n, the number of edges m), the value of penalty weighting b, the SP1 LP objective function value, the tightened Cutting Plane Objective function value and the Integer solution value (on termination or interruption). We then report the time in seconds for the addition of cuts at the root node (including the initial LP relaxation), the time for SP1 IP followed by the number of cuts by type (Modified SECs, 2-Matching and GECs). The comment *INT INF* indicates that the problem is integer infeasible for SP1.

We see that in some cases the LP relaxation yields an integer solution, denoted with $*$ in Table 1. In some cases, such as *dfn-bwin* and *atlanta*, the integer solution has subtours and local ring SECs are added at the root node. As noted previously, *janos-us-ca* and *zib54* were integer infeasible for SP1. We see that SP1 for small problems is solved with a relatively short running time but that for most of the larger problem instances the hardware memory limitation was reached before the maximum run time allowed. In the case of *germany*, the algorithm terminated after just under 1.5 hours. We see for these problems that a large number of modified local ring SECs and 2-matchings are added during the branch-and-cut search.

Results for SP2 are shown in Table 2. SP2 identifies a tertiary ring to interconnect the integer local ring spur solutions of SP1. Columns from left to right show the problem name, the number of tertiary ring SECs

prob	n, m	b	LP(SP1)	CP(SP1)	IP(SP1)	CP(s)	IP(s)	Cuts (SEC; 2M; GEC)
dfn-bwin	10,45	3	*81,333	81,333	81,333	0.579	0.61	3; 0; 0
pdh	11,34	4	1,047,629	1,055,454	1,059,366	0.497	0.49	0; 0; 2
di-yuan	11,42	16	*343,200	-	343,200	0.153	-	0; 0; 0
dfn-gwin	11,47	6	*12,520	-	12,520	0.18	-	0; 0; 0
polska	12,18	3	*2,631	-	2,631	0.164	-	0; 0; 0
atlanta	15,22	17	*33,582,500	33,582,500	33,582,500	0.511	0.28	2; 0; 0
newyork	16,49	7	1,257,600	1,267,733	1,333,600	0.699	18.46	63; 76; 2
ta1	24,51	7	7,931,099	8,329,128	8,329,128	1.366	2.73	2; 0; 4
france	25,45	10	14,000	15,800	15,800	0.685	1.66	2; 1; 0
janos-us	26,42	4	11,799	12,134	13,289	2.397	60.07	433; 88; 4
norway	27,51	6	423,065	427,972	504,880	1.503	707.94	2,186; 2,245; 3
sun	27,51	15	423	425	604	1.532	2,407.98	9,418; 12,224; 0
nobel-eu	28,41	3	164,160	164,447	170,890	1.422	25.25	259; 2; 4
cost266	37,57	13	6,914,610	7,013,655	8,633,700	1.354	294.63	1,317; 49; 3
giul39	39,86	6	662	671	708	2.06	**4,920.12	7,904; 3,645; 3
janos-us-ca	39,60	10	-	-	INT INF	-	-	-
pioro40	40,89	9	5,707	5,724	6,248	3.444	**3,120.56	9,902; 3,881; 3
germany	50,88	7	301,480	303,711	426,950	4.651	**5,340.72	26,867; 2,162; 6
zib54	54,82	12	-	-	INT INF	-	-	-
ta2	65,108	20	23,903,860	29,687,391	64,362,782	8.554	**9,781.31	12,978; 3,476; 7

Table 1: Sub-problem 1 results, $*$ indicates an integer result, $**$ denotes out of memory.

added and the number of SP1 integer solutions considered. Next we show, for the best complete solution found, the local ring integer objective function value and the cost of its interconnecting tertiary ring (SP2 objective function value). We next show the best total solution cost (which is the sum of the local ring spur costs and the tertiary ring costs). Then we show the lower bound for the tertiary ring followed by the problem lower bound and lastly, the gap between our best solution an the problem lower bound. We explain below how the lower bound is calculated.

For all of the problems for which SP1 integer solutions are found, except one, a tertiary ring and complete solution can be found. The exception is *nobel-eu* with $b = 4$ where only one integer SP1 solution was found during the branch-and-cut search with five local rings and two spurs. The network does not have a set of edges to allow the local rings to be interconnected by a tertiary ring. This leaves 17 of the original test cases where complete solutions were found.

We see that, on smaller problems, the number of integer solutions to SP1 is often low and generating and solving SP2 takes a small amount of time. For example, only one integer local ring SP1 solution is considered for the 24 node *ta1*. We add 14 tertiary ring SECs to find an interconnecting SP2 tertiary ring. Figure 9 shows an example of the RSAP solution for the *newyork* problem instance. We see that the algorithm considers four integer SP1 solutions and yields a ring solution.
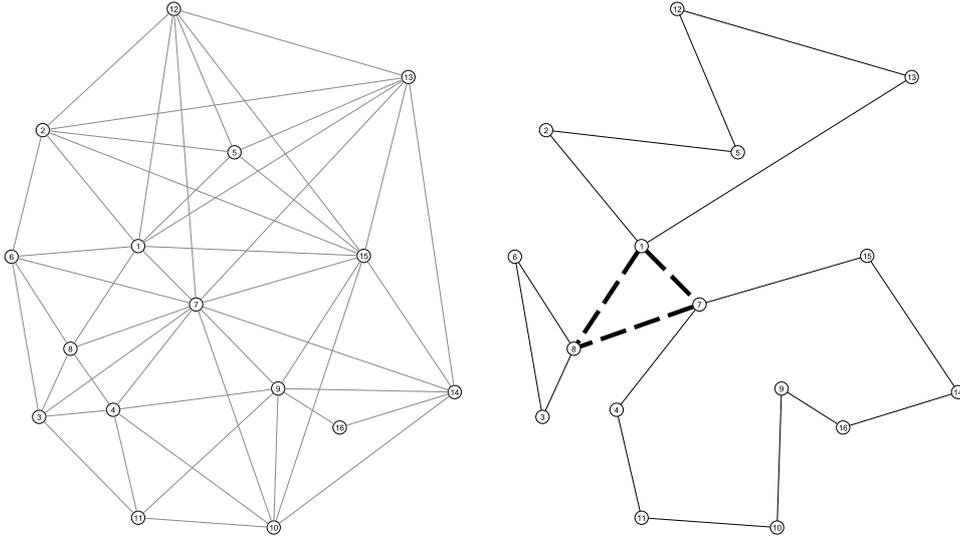
Figure 9: Left: *newyork* instance, Right: Best RSAP solution for *newyork*

prob	T-SEC	Num Int Sols	SP1 best sol	SP2 best sol	best sol	LB_{SP2}	Problem LB	Gap %
dfn-bwin	0	3	81,333	24,549	105,882	24,322	105,655	0.21
pdh	0	1	1,059,366	370,204	1,429,570	260,081	1,315,535	7.98
di-yuan	2	1	343,200	69,100	412,300	68,200	411,400	0.22
dfn-gwin	2	1	12,520	3,204	15,724	2,812	15,332	2.49
polska	2	1	2,631	856	3,487	694	3,325	4.65
atlanta	0	1	33,582,500	21,870,000	55,452,500	5,120,000	38,702,500	30.21
newyork	22	4	1,333,600	194,000	1,527,600	133,200	1,400,933	8.29
ta1	14	1	8,329,128	3,081,040	11,410,169	486,480	8,815,608	22.74
france	8	1	15,800	5,000	20,800	600	16,400	21.15
janos-us	14	5	13,289	4,091	17,380	594	12,728	26.77
norway	22	5	511,950	84,120	596,070	20,890	448,862	24.70
sun	14	7	604	180	783	21	446	43.09
nobel-eu	0	1	170,890	INT INF	-	-	-	-
cost266	40	4	8,633,700	3,628,890	12,262,590	378,180	7,391,835	39.72
giul39	146	6	745	221	966	33	704	27.12
pioro40	228	5	6,269	2,794	9,063	271	5,995	33.86
germany	48	7	437,980	120,610	558,590	11,160	314,871	43.63
ta2	24	2	64,362,782	9,420,022	73,782,804	176,902	29,864,293	59.52

Table 2: Computational Results for the RSAP

We see also that, for larger problems, a higher number of integer SP1 solutions are considered. For example, five local ring solutions are considered for the 40 node *pioro40* problem. We note that the branch-and-cut search is interrupted before the three hour time limit had elapsed as we reach the hardware memory limit. We report

the best feasible solution within the time and hardware limits. In this case, the best solution had an SP1 local ring cost of 6,269 and SP2 tertiary ring cost of 2,794, giving a total solution cost of 9,063. The lowest cost SP1 local ring solution found by the branch-and-bound search by contrast was lower at 6,248 but needed a tertiary ring costing 3,047 giving a higher total cost of 9,104. This is an example of where our branch-and-cut heuristic compensates for the weakness of the decomposition approach. Figure 10 shows an example of an RSAP solution topology for the 50 node *germany* problem. We note in this case that the algorithm yields a ring spur solution.
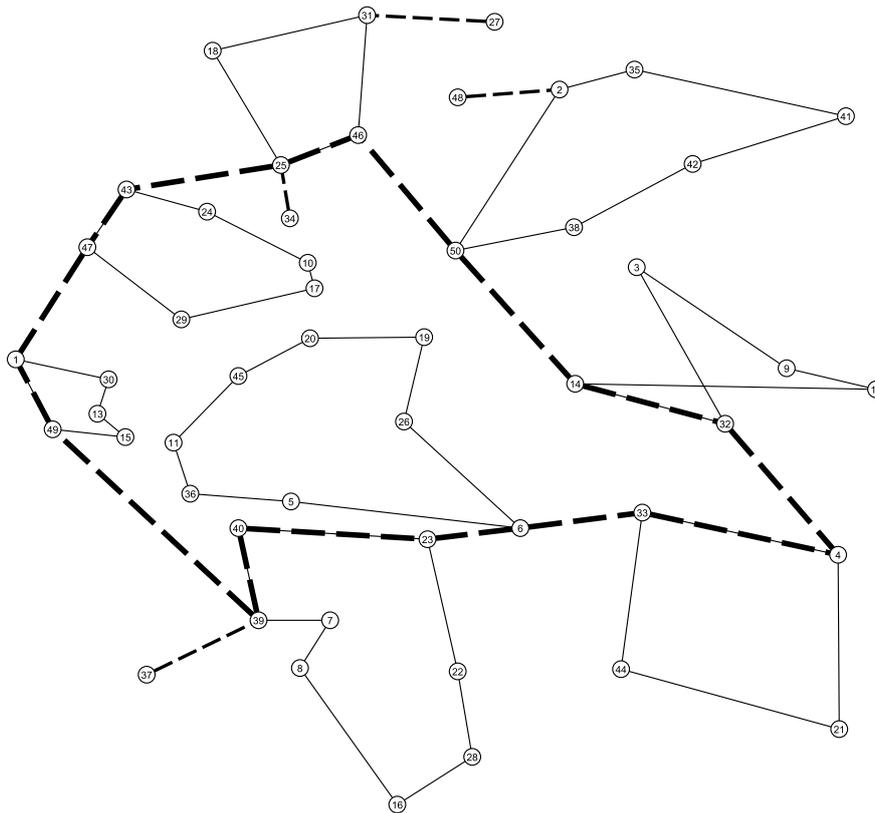

Figure 10: RSAP solution for 50 node *germany* problem

Finally, we seek a measure of the quality of our solutions. We calculate a lower bound on the solution cost for each problem as the sum of the lower bound on the local ring spur solution, LB_{SP1} plus the lower bound on the tertiary ring solution, LB_{SP2}. We set LB_{SP1} to be the Cutting Plane objective value. The minimum number of local rings in an RSAP solution is two, and these must be connected by a tertiary ring of size at least three. Therefore, we set the lower bound LB_{SP2} for the tertiary ring to be the cost of a min cost simple cycle of size at least three for each problem instance. This is a weak lower bound, as in general the RSAP solutions have more than two local rings.

We solve a modified version of SP2 to calculate LB_{SP2} for each problem instance. We use the same objective function with constraints (10), (11), (12) and (15) and an additional constraint that ensures the sum of tertiary ring edges is at least three.

Each of LB_{SP1}, LB_{SP2} is a lower bound for the appropriate sub-problem in the absence of any other

constraints. Thus, even though we have decomposed the problem, the sum of the proposed lower bounds on the sub-problems is a valid lower bound for the complete problem. We see that, for small problems such as *polska*, the gap is narrow at 4.65% but, for the larger problems, the gap is wide, e.g., the worst gap is for the 65 node *ta2* problem at 59.52%. However, this RSAP has ten local rings demonstrating the weakness of a lower bound of a minimum cost tertiary ring with at least three edges.

We note also that the tertiary rings of our solutions can consist of a large number of edges, since no ring bound was imposed on the tertiary ring. The 50 node *germany* solution shown in Figure 10, has a tertiary ring containing 15 nodes (edges) which exceeds our Ring Bound but satisfies the SONET ring bound recommendation. In the case of the 37 node *cost36* and 65 node *ta2* problems, the tertiary rings exceed the SONET ring bound. These solutions may still be acceptable if alternative ring bounds in terms of time or length can be satisfied. Alternatively, a 3-level topology would be necessary.

6. Conclusions

We have presented a summary of the RSAP problem and positioned it in relation to problems previously described in the literature. The main contribution of our paper is an IP based decomposition branch-and-cut algorithm which we have implemented with promising computational results. We have described valid inequalities for multi-ring problems. We are currently investigating alternative formulations for the complete IP model that may be solvable within a reasonable time frame without having to decompose the problem. We are also working on improving the lower bound to give a better estimation of the quality of our results.

Finally, we note the resulting tertiary ring for large networks may exceed the SONET ring bound standards. We note that the result of our work is a 2-level hierarchal network where the local ring spurs of the stage 1 problems constitute the lower layer and the tertiary ring constitutes the higher layer. A further modification to our work would allow us to create a 3-level topology. We could interconnect the stage 1 local rings by intermediate rings which in turn would be interconnected by the tertiary ring. This corresponds directly to the traditional hierarchal topology of telecommunications networks which consist of primary, secondary and tertiary layers in a tree-like topology. The local ring/spur partitions correspond to the primary layer but now have some ring protection, likewise our intermediate rings would correspond to the secondary layer while the tertiary ring corresponds to the tertiary layer.

Acknowledgements

References

Baldacci, R., Dell'Amico, M., González, L. S. 2007. The capacitated m-ring-star problem. *Operations Research* 55 1147–1162.

Bayvel, P. 2000. Future high-capacity optical telecommunication networks. *Philosophical Transactions - A - Mathematical Physical and Engineering Sciences* 358 303–329.

Brackett, C. 1990. Dense wavelength division multiplexing networks: principles and applications. *Selected Areas in Communications, IEEE Journal on* 8 948–964. doi:10.1109/49.57798.

Carroll, P., McGarraghy, S. 2009a. An algorithm for the ring spur assignment problem. Papova, N., hÉigeartaigh, M. O., (Eds.), *Proceedings of the International Eugene Lawler PhD Summer School 2009 held at WIT, Ireland, June 6–10, 2009*. Scientific Computing, WIT, 180–197.

Carroll, P., McGarraghy, S. 2009b. Investigation of the ring spur assignment problem. Bigi, G., Frangioni, A., Scutellà, M., (Eds.), *Proceedings of INOC 2009*. Pisa, Italy, MB1–3. URL `http://www.di.unipi.it/optimize/Events/proceedings/M/B/1/MB1-3.pdf`.

Cosares, S., Deutsch, D., Saniee, I., Wasem, O. 1995. Sonet toolkit: A decision support system for designing robust and cost-effective fiber-optic networks. *Interfaces* 25 20–40.

Fortz, B., Labbé, M. 2004. Two-connected networks with rings of bounded cardinality. *Computational Optimization and Applications* 27 123–148.

Fortz, B., Labbé, M., Maffioli, F. 2000. Solving the two-connected network with bounded meshes problem. *Operations Research* 48 866–877.

Fortz, B., Mahjoub, A., McCormick, S., Pesneau, P. 2006. Two-edge connected subgraphs with bounded rings: Polyhedral results and Branch-and-Cut. *Mathematical Programming* 105 85–111.

Fortz, B., Soriano, P., Wynants, C. 2003. A tabu search algorithm for self-healing ring network design. *European Journal of Operational Research* 151 280–295.

Fumagalli, A., Cerutti, I., Tacca, M. 2003. Optimal design of survivable mesh networks based on line switched wdm self-healing rings. *IEEE/ACM Transactions on Networking* 11 501–512.

Goldschmidt, O., Laugier, A., Olinick, E. 2003. SONET/SDH ring assignment with capacity constraints. *Discrete Applied Mathematics* 129 99–128.

Grötschel, M., Monma, C., Stoer, M. 1995. Polyhedral and computational investigations for designing communication networks with high survivability requirements. *Operations Research* 43 1012–1024.

Grover, W. 2003. *Mesh Based Sruvivable Networks, Options and Strategies for Optical, MPLS, Sonet and ATM Networking*. Prentice Hall.

Grover, W., Doucette, J., Clouqueur, M., Leung, D., Stamatelakis, D. 2002. New options and insights for survivable transport networks. *Communications Magazine, IEEE* 40 34–41.

Hoshino, E., de Souza, C. 2009. A branch-and-cut-and-price approach for the capacitated m-ring-star problem. *Electronic Notes in Discrete Mathematics* 35 103–108.

Kerivin, H., Mahjoub, A. 2005. Design of survivable networks. *Networks* 46 1–21.

Labbé, M., Laporte, G., Martin, I., Salazar-Gonzalez, J. 2004. The Ring Star Problem: Polyhedral analysis and exact algorithm. *Networks* 43 177–189.

Lawler, E., Lenstra, J., Kan, A. R., Shmoys, D., (Eds.). 1985. *The Travelling Salesman Problem: A Guided Tour of Combinatorial Optimization*. Wiley, Chichester.

Macambira, E., Maculan, N., de Souza, C. 2006. A column generation approach for SONET ring assignment. *Networks* 47 157–171.

Naji-Azimi, Z., Salari, M., Toth, P. 2010. A heuristic procedure for the capacitated m-ring-star problem. *European Journal of Operational Research* 207 1227–1234.

Nemhauser, G., Wolsey, L., (Eds.). 1988. *Integer and Combinatorial Optimization*. Wiley, Chichester.

Orlowski, S., Pióro, M., Tomaszewski, A., Wessäly, R. 2010. SNDlib 1.0–Survivable Network Design Library. *Networks* 55 276–286. doi:10.1002/net.20371. URL `http://sndlib.zib.de`.

Padberg, M., Rao, M. 1982. Odd minimum cut-sets and b-matchings. *Mathematics of Operations Research* 7 67–80.

Papadimitriou, G., Obaidat, M., Pomportsis, A. 2001. Advances in Optical Networking. *Int. J. Commun. Sys.* 15 101–113.

Stoer, M., Wagner, F. 1997. A simple min-cut algorithm. *Journal of the ACM* 44 585–591.

Thomadsen, T., Stidsen, T. 2005. Hierarchical ring network design using branch-and-price. *Telecommunication Systems* 29 61–76.