Available online at www.sciencedirect.com

SCIENCE DIRECT®

Information Processing Letters 91 (2004) 183–190

Information Processing Letters

www.elsevier.com/locate/ipl

Survivable network design: the capacitated minimum spanning network problem

Raja Jothi *, Balaji Raghavachari

Department of Computer Science, University of Texas at Dallas, Richardson, TX 75080, USA

Received 8 December 2003; received in revised form 14 April 2004

Available online 18 May 2004

Communicated by S. Albers

Abstract

We are given an undirected graph $G = (V, E)$ with positive weights on its vertices representing demands, and non-negative costs on its edges. Also given are a capacity constraint k, and root vertex $r \in V$. In this paper, we consider the *capacitated minimum spanning network* (CMSN) problem, which asks for a minimum cost spanning network such that the removal of r and its incident edges breaks the network into a number of components (groups), each of which is 2-edge-connected with a total weight of at most k. We show that the CMSN problem is NP-hard, and present a 4-approximation algorithm for graphs satisfying triangle inequality. We also show how to obtain similar approximation results for a related 2-vertex-connected CMSN problem.

Keywords: Access network design; Approximation algorithms; Combinatorial optimization; 2-connectivity

1. Introduction

Consider a given undirected graph $G = (V, E)$ with non-negative weights on its vertices representing demands, and non-negative costs on its edges. Also given as input are a capacity constraint k, and root vertex $r \in V$. The extensively studied *capacitated minimum spanning tree* (CMST) problem [4,10,17] asks for a minimum cost spanning tree rooted at r in which the sum of the vertex weights in every subtree (local access network) hanging off r is at most k. The problem is feasible only when k is at least as much as the largest vertex weight. The CMST problem is NP-hard [13]. In telecommunication network design, the CMST problem corresponds to the designing of a minimum cost tree network by installing cables along the edges of a network. The cables have a prespecified capacity constraint on the amount of traffic they can transmit, and can be bought at a certain cost per unit length. Each node, but the root node, in the network has traffic associated with it, which must be routed to the root node. The goal is to construct a minimum length tree network so as to facilitate simultaneous routing of traffic from all the nodes to the root node.

The CMST problem has been extensively studied in computer science and operations research for the past

* Corresponding author.
E-mail addresses: raja@utdallas.edu (R. Jothi), rbk@utdallas.edu (B. Raghavachari).

0020-0190/$ – see front matter
doi:10.1016/j.ipl.2004.04.007

40 years [4,10,17]. One generalization of the CMST problem that has received attention recently is the *single-sink buy-at-bulk* problem [14] (also known as the *single-sink edge installation* problem) in which we are given several cable types instead of just one. Each cable has a given capacity and a cost per unit length.

1.1. Problem definition

A CMST can be viewed as a collection of local access tree networks, each with a total demand of at most k, connected to the root node. Most often, local access networks are prone to node/edge failures. To prevent such failures, the notion of survivable networks has been studied. Survivable networks are resilient to node/edge failures. That is, a node/edge failure still allows communication between functioning sites. We call a network to be α-vertex-connected, if the failure of $\alpha - 1$ vertices leaves the remaining network connected (α-edge-connectivity is defined with respect to edge failures analogously). 2-Connectivity is a major feature in today's fast and reliable communication networks, since without 2-connectivity, a single vertex/edge failure could cause intolerable losses. In this paper, we consider the *capacitated minimum spanning network* (CMSN) problem, a variant of the CMST problem, which requires that each of the local access networks be 2-edge-connected. In other words, the CMSN problem requires that the local access networks be resilient under an edge failure. The formal definition of the problem is given below.

> CMSN: Consider an undirected graph $G = (V, E)$, root $r \in V$, and capacity k. Each vertex $v \in V$ is associated with a positive number w_v representing the demand that v wishes to route to r, and each edge has a cost associated with it. The capacitated minimum spanning network problem asks for a minimum cost spanning network such that the removal of r and its incident edges breaks the network into a number of components (groups), each of which is 2-edge-connected with a total weight of at most k.

We also consider a variant of the CMSN problem, which we call the 2-vertex-connected CMSN problem (2VC-CMSN). The 2VC-CMSN problem asks for a minimum cost partitioning of the set of vertices $V \setminus \{r\}$ into groups of weight at most k, with the vertices in each group along with r being 2-vertex-connected. A feasible solution to the 2VC-CMSN problem ensures that the network remains connected even after the removal (failure) of a non-root vertex.

Neither of the above problems have been studied before. But, CMST algorithms designed for graphs satisfying triangle inequality can be altered to produce feasible CMSN solutions. In particular, Altinkemer and Gavish's CMST algorithm [3] can be used to find a feasible solution that is within factor 6 of an optimal solution. The idea is to double the edges of the local access tree networks to construct tours. For the 2VC-CMSN problem, Altinkemer and Gavish's algorithm for the delivery problem [2] will guarantee a feasible solution that is within factor 4 of an optimal solution.

1.2. Our results

We show that the CMSN and 2VC-CMSN problems are NP-hard using a reduction from the minimum cost 2-connected spanning subgraph problem. For the CMSN problem, we present a 4-approximation algorithm for graphs satisfying triangle inequality. For the 2VC-CMSN problem, we show that there is an algorithm with a performance ratio of 3.5. We also show that the 2VC-CMSN problem is polynomial-time solvable if all vertices have unit weights and $k = 2$.

1.3. Related work

Most minimum spanning tree (MST) algorithms can be modified to find a feasible solution to the CMST problem. Several exact algorithms and mathematical formulations are available for the CMST problem [4, 17]. The instance sizes that can be solved by these algorithms to optimality, in reasonable amount of time, is still far from the size of the real-life instances. Numerous heuristics for the CMST problem have been proposed during the past 40 years. Some of the best heuristics, in terms of the quality of solutions obtained, are due to Amberg et al. [4], Ahuja et al. [1], and Sharaiha et al. [15]. Despite the quality of the solutions produced, their worst-case time complexity is high and their running time could be exponential [12,16]. These heuristics start with an initial feasible solution, and improve the initial solution by local re-alignments of nodes during every iteration. The problem with these techniques is that the amount of improvement in each iteration could be so small that the number of iterations could possibly be as large as the maximum possible objective value [16]. The best heuristic, both in terms of the quality of the solutions obtained and the worst-case running time ($O(n^2 \log n)$), is due to Jothi and Raghavachari [9], who recently improved Esau–Williams' heuristic [6]. For a complete survey on heuristics for the CMST problem, we refer the reader to [4,10,17].

The first algorithm for the CMST problem with any approximation guarantee was given by Gavish and Altinkemer [8]. They later presented improved approximation algorithms with ratios $3 - 2/k$ and 4 for the uniform and non-uniform vertex-weighted graphs [3]. In a recent work, Jothi and Raghavachari [10] presented improved approximation algorithms for the CMST problem. Their algorithms guarantee ratios of 2.9 for uniform vertex-weighted graphs in L_p metric plane, and $\gamma + 2$ for non-uniform vertex-weighted graphs, where γ is the inverse *Steiner ratio*.[1] All of the above ratios hold only for graphs whose edges satisfy triangle inequality.

2. Preliminaries

In this paper, we consider graphs whose edges obey triangle inequality. Hence, we assume that there always exists an edge between any two vertices. Even though the input is a vertex-weighted graph, we safely assume that the root vertex r has weight zero as the cost of the CMSN is not dependent upon r's weight. Let $|uv|$ denote the cost of the edge between vertices u and v. Let OPT denote an optimal CMSN and let C_{opt} denote its cost. Let T_v denote the subtree rooted at v in a given tree T rooted at r. Let w_v denote the weight of vertex v and let $w(T_v)$ denote the sum of the weights of vertices in T_v. We call as *spokes*, the edges incident on r in a CMSN. By *level* of a vertex, in a tree T rooted at vertex r, we mean the number of tree edges on its path to r.

We can safely assume that all the vertices have integer weights. The assumption is not restrictive as any CMSN problem with rational vertex weights can be converted to an equivalent problem with integer vertex weights. The optimal solution for the scaled problem is identical to that of the original problem [3].

[1] The Steiner ratio is the maximum ratio of the costs of the minimum cost Steiner tree versus the minimum cost spanning tree.

3. The CMSN problem

3.1. Problem complexity

Given an edge weighted graph with the cost of an edge being its weight, the minimum cost 2-edge-connected spanning subgraph (2ECSS) problem asks for a minimum cost subgraph such that there exists at least two edge-disjoint paths between any two nodes [11]. The 2-vertex-connected spanning subgraph (2VCSS) problem is defined analogously with respect to vertex-disjoint paths [11]. In what follows, we show that the CMSN problem is NP-hard using a simple reduction from the 2ECSS problem.

Theorem 3.1. *The capacitated minimum spanning network problem is* NP*-hard.*

Proof. To prove that the CMSN problem is NP-hard, we show that the 2ECSS problem is polynomial-time reducible to the CMSN problem. Let $G = (V, E)$ be an instance of 2ECSS problem. The sum of costs of all the edges in G is given by $\Delta = \sum_{i,j \in V} |ij|$. Let $n = |V|$. Construct an instance $G' = (V', E')$ of CMSN problem, where $V' = V \cup \{r\}$ and $E' = E \cup \{(i, r) \mid i \in V\}$, and set the cost of each edge incident on r to be $\Delta + 1$. Also, set k to be n.

We now show that graph G has a 2-edge-connected spanning subgraph of cost at most C, if and only if graph G' has a CMSN of cost at most $C + \Delta + 1$. Suppose that G has a 2-edge-connected spanning subgraph S of cost at most C. One can easily construct a feasible CMSN in G' using S by just adding an edge from r to one of the other nodes. The cost of the resulting CMSN would be at most $C + \Delta + 1$ by definition. Conversely, suppose that graph G' has a CMSN S^* of cost at most $C + \Delta + 1$. Removing edges incident on r from S^* will result in a feasible 2-edge-connected spanning subgraph for G. Such a solution would be of cost at most C, since we will be removing at least one edge of cost $\Delta + 1$ incident on r. $\square$

Remark. Since 2ECSS problem is NP-hard even for graphs satisfying triangle inequality, we conclude that the CMSN problem on graphs whose edges satisfy triangle inequality is NP-hard as well, as the transformed graph in Theorem 3.2 still obeys triangle inequality.

Theorem 3.2. *The 2-vertex-connected capacitated minimum spanning network problem is NP-hard.*

Proof. The proof is similar to that for Theorem 3.2 except that the reduction is from the Hamiltonian path problem. □

3.2. Lower bounds

For the CMSN problem, we use two different lower bounds on the cost of an optimal solution. The first is the cost of the minimum spanning tree, which is obvious. The second lower bound is $\frac{1}{k}\sum_{v\in V} w_v|rv|$, which is presented as Lemma 3.1.

Lemma 3.1. *Consider a given graph $G=(V,E)$, root $r\in V$, and a capacity constraint k. Let OPT be an optimal CMSN for G, rooted at r, whose cost is C_{opt}. Then,*

$$C_{\mathrm{opt}} \geqslant \frac{1}{k}\sum_{v\in V} w_v|rv|.$$

Proof. Let t be the number of local access networks connected to r in OPT. Let q be one such local access network in OPT that is connected to r. Let S_q be the set of nodes in q. Let C_q be the sum of the cost of the local access network q and the cost of the edge connecting q to r. By triangle inequality,

$$\begin{aligned} C_q &\geqslant \max_{v\in S_q}\bigl\{|rv|\bigr\} \geqslant \frac{\sum_{v\in S_q} w_v|rv|}{\sum_{v\in S_q} w_v} \\ &\geqslant \frac{\sum_{v\in S_q} w_v|rv|}{k} \quad \Bigl(\text{since} \sum_{v\in S_q} w_v \leqslant k\Bigr). \end{aligned}$$

For t local access networks connected to r in OPT,

$$C_{\mathrm{opt}} = \sum_{q=1}^{t} C_q \geqslant \frac{\sum_{v\in V} w_v|rv|}{k}. \qquad \square$$

3.3. Algorithm and analysis

We first construct an MST T, of the given graph $G=(V,E)$ and root T at r. Next, we gather nodes from T in groups such that the sum of vertex weights in each group is between $k/2$ and k. Finally, for each group of vertices, we construct a tour (cycle) spanning all vertices in that group, and connect the resulting tour to r. For easier analysis, we introduce *dummy* vertices with zero weight, in place of real vertices, during the execution of the algorithm, which are removed from the final solution using shortcutting. The formal algorithm is given in Fig. 1.

It can be verified that the algorithm outputs a feasible CMSN for a given k.

Theorem 3.3. *Algorithm* CAPMINSPANET (*Fig.* 1) *is a* 4*-approximation algorithm for the* CMSN *problem.*

Proof. We prove the theorem by showing that, for any given instance, Algorithm CAPMINSPANET outputs a solution that has cost at most 4 times the cost of an optimal CMSN.

Fig. 2 depicts the 3 situations (#1, #2, #3) encountered in the algorithm, where minimum cost tours are constructed. Since the edges of the graph obey triangle inequality, it can be seen that a tour—covering all the vertices in the concerned set—can be constructed by just doubling the necessary MST edges. Remember that the *dummy* vertices, which serve only as placeholders, are introduced to ensure that the underlying MST edges are still available for doubling. The dummy vertices are removed from the final solution using shortcutting.

Except for those tours constructed in the last "for" loop of the algorithm, every tour constructed by the algorithm contains vertices whose weights add up to at least $k/2$. Thus, for every tour the algorithm adds a new spoke, it is guaranteed that the sum of the vertex weights in that tour is at least $k/2$. Notice that every spoke added to T in this manner connects r to the closest vertex in the tour. Let $t_1, t_2, \ldots, t_m$ be the set of tours constructed by Algorithm CAPMINSPANET for which a new spoke was added. Let Γ be the set of vertices in tours $t_1, \ldots, t_m$. Let t_i be one such tour. Let z_i be the vertex in t_i that is connected

Input: MST T, rooted at r, and k.
While there exists a vertex $v \neq r$ such that $w(T_v) \geqslant k$ and level of v is maximum, do
  If $w(T_v) = k$,
    Remove all the edges incident on vertices in T_v.
    Construct a minimum cost tour t visiting all vertices in T_v. // #1
    Add edge rz (spoke) such that $z \in t$ and $|rz| = \min_{u \in t}\{|ru|\}$.
  Else if $\sum_{i \in v\text{'s children}} w(T_{v_i}) < k/2$,
    $w(v)$ must be greater than $k/2$ as $w(T_v) > k$.
    Add edge rv (spoke), and set v as a *dummy* vertex.
  Else
    Initialize: $S = \emptyset$
    Sort v's children, in non-decreasing order, based on the weight of the subtrees rooted at them.
    Let $\{v_1, v_2, \ldots, v_p\}$ be the sorted list of v's children.
    $S = S \cup \{x \mid x \in T_{v_p}\}$.
    While $w(S) < k/2$, do
      Choose an unprocessed child v_i of v.
      $S = S \cup \{x \mid x \in T_{v_i}\}$.
    Remove all the edges incident on vertices in S.
    Construct a minimum cost tour t visiting all vertices in S. // #2
    Add edge rz (spoke) such that $z \in t$ and $|rz| = \min_{u \in t}\{|ru|\}$.
For every child v_j of r, which is not part of any tour
  Remove all the edges in T_{v_j}.
  Construct a minimum cost tour t visiting all vertices in T_{v_j}. // #3
Remove *dummy* vertices from the final solution using shortcutting of the tours.
Install cables along the edges of the network constructed thus far.

Fig. 1. Algorithm CapMinSpaNet.

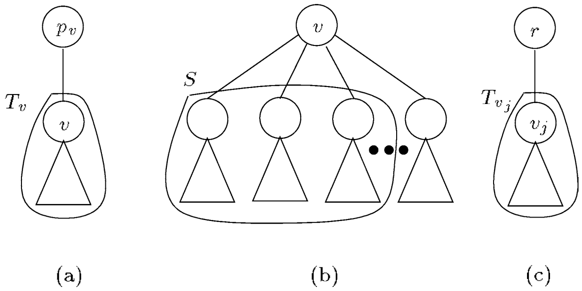

Fig. 2. (a) Doubling the edges in T_v and shortcutting results in a tour. (b) Doubling the edges incident on the vertices in S and shortcutting results in a tour. (c) Doubling the edges in T_{v_j} and shortcutting results in a tour.

to r. Among the vertices in t_i, since z_i is the closest to r,

$$|rz_i| \leqslant \frac{\sum_{v \in t_i} w_v |rv|}{k/2}.$$

Thus, for m tours for which the new spokes were added,

$$\begin{aligned} C_{\text{spokes}} &= \sum_{i=1}^{m} |rz_i| \leqslant \frac{\sum_{v \in \Gamma} w_v |rv|}{k/2} \\ &\leqslant 2 \times \frac{\sum_{v \in V} w_v |rv|}{k} \quad (\text{since } \Gamma \subset V) \\ &\leqslant 2 \times C_{\text{opt}} \quad (\text{by Lemma 3.1}). \end{aligned}$$

Since the cost of the local access networks is at most twice the cost of the MST, and the cost of the new spokes that were added is at most twice the optimal CMSN, we conclude that Algorithm CapMinSpaNet outputs a solution of cost at most 4 times the cost of an optimal CMSN. □

4. The 2VC-CMSN problem

For a given graph $G = (V, E)$, root $r \in V$, and a capacity k, the 2VC-CMSN problem asks for a minimum cost partitioning of the set of vertices $V \backslash \{r\}$ into groups of weight at most k, with the vertices in each group along with r being 2-vertex-connected.

Notice that a feasible solution to the 2VC-CMSN problem ensures that the network remains connected even after the removal (failure) of a non-root vertex. For the 2VC-CMSN problem, we show that Altinkemer and Gavish's algorithm for the delivery problem [2] finds a feasible solution that is within a factor 3.5 of an optimal solution.

4.1. Delivery problem

Given a set of customers (nodes), each having a positive demand, a depot (root node) s, and vehicles of capacity q, the delivery problem asks for routes for the vehicles such that the vehicles depart from s, serve a set of customers following their designated routes and return to s. The objective is to minimize the total length (cost) of the routes without violating the capacity limit q and visiting each customer exactly once. Altinkemer and Gavish [2] presented a traveling salesman (TSP) tour partitioning algorithm for this problem, which finds a feasible solution of cost at most $2 + \rho_{\text{TSP}}$ times the cost of an optimal solution, where ρ_{TSP} is the best achievable approximation ratio for the TSP problem. Their idea is to construct a TSP tour visiting all the customers, optimally partition the tour into paths with the sum of customer demands in each path being at most k, and connect the two ends of each path to s. Using the fact that an optimal TSP tour is a lower bound on an optimal solution for the delivery problem, they were able to show that the cost of the solution produced by such an algorithm is at most $2 + \rho_{\text{TSP}}$ times than that of an optimal solution (showing that the cost to connect the paths to s is at most 2 times the cost of an optimal solution).

Notice that the solution returned by the delivery algorithm for capacity $q = k$ with depot $s = r$ is a feasible solution for the 2VC-CMSN problem. Since an optimal TSP tour is not a lower bound for the 2VC-CMSN problem, their approximation ratio of $2 + \rho_{\text{TSP}}$ does not apply to the 2VC-CMSN problem. Rather, one can interpret the cost of the TSP tour, which they partition, as 2 times the cost of an MST. Since the MST cost is a lower bound for the 2VC-CMSN problem, Altinkemer and Gavish's analysis will guarantee a ratio of 4 for the 2VC-CMSN problem.

4.2. Improved approximation analysis

Notice that an optimal 2VCSS cost is a lower bound for the 2VC-CMSN problem as 2VC-CMSN problem is a generalization of the 2VCSS problem (2VCSS problem is the 2VC-CMSN problem with $k = \infty$). Suppose that Altinkemer and Gavish's algorithm for the delivery problem uses Christofides' TSP algorithm [5] to construct the initial tour. In this approach, we start with an MST T of the graph. A minimum weight perfect matching of the odd-degree nodes of T in an arbitrary graph is then computed and added to T. The resulting Eulerian graph can then be converted into a TSP tour using shortcutting. Frederickson and JáJá [7] showed that the matching found by Christofides' algorithm, (on its way to finding an approximate TSP tour) is no more than half the cost of an optimal 2VCSS. As MST cost is a lower bound for the 2VCSS problem, the cost of the Christofides' TSP tour is at most 1.5 times the cost of an optimal 2VCSS [7]. Since an optimal 2VCSS is a lower bound for the 2VC-CMSN problem, Altinkemer and Gavish's algorithm actually guarantees a ratio of 3.5 for the 2VC-CMSN problem.

Theorem 4.1. *Altinkemer and Gavish's algorithm for the delivery problem is a* 3.5*-approximation for the* 2VC-CMSN *problem.*

4.3. Unit vertex weights with $k = 2$

The special case when all vertices have unit weights and $k = 2$ is polynomial time solvable using min-cost matching. Let $G = (V, E)$ be the given graph. Let $|uv|_G$ be the cost of the edge connecting vertices u and v in G. Let r be the root vertex in G. Construct a graph $G' = (V_1 \cup V_2, E_1 \cup E_2 \cup E_3)$ as follows:

- Set $V_1 = \{v' \mid v \in V\}$.
- Introduce an edge, into E_1, of cost $|xy|_G + |rx|_G + |ry|_G$ between vertices x' and y' in V_1, if $x' \neq y'$.
- Set $V_2 = \{v'' \mid v \in V\}$.
- Introduce an edge, into E_2, of cost zero between vertices x'' and y'' in V_2, if $x'' \neq y''$.
- for each $v \in V$, introduce an edge of cost $|rv|_G$, into E_3, between vertices v' and v'' in G' (v' and v'' correspond to the same vertex $v \in V$).

Lemma 4.1. *The cost of a min-cost perfect matching of the vertices in G' is equal to the cost of an optimal solution for the* 2VC-CMSN *problem on G.*

Proof. We use the notations used above to complete the proof. It suffices to prove the following:

(1) Given a 2VC-CMSN of cost ω in G, there exists a perfect matching of cost ω in G'.
If a local access network in the given 2VC-CMSN contains 2 vertices x and y, then match the corresponding x and y, belonging to V_1, in G'. If a local access network in the given 2VC-CMSN contains just one vertex x, then match $x' \in V_1$ and $x'' \in V_2$. By doing this, all vertices belonging to V_1 in G' will be matched while there might be some vertices in V_2 that are not matched. Since all vertices in V_1 are matched, an even number of vertices in V_2 will be left unmatched, and they can be paired-up arbitrarily. Since the cost of matching $x \in V_2$ and $y \in V_2$ is zero, the cost of the perfect matching would be the same as the cost of the 2VC-CMSN.

(2) Given a matching of cost ω in G', there exists a 2VC-CMSN of cost ω in G.
Let $n = |V|$. Thus, $|V'| = 2n$. Let (x, y) denote a matching between vertices x and y. If $x, y \in V_1$, connect the corresponding two vertices in G with r to form a cycle. If $x \in V_1$ and $y \in V_2$, connect the corresponding vertex (both x and y will correspond to the same vertex in V) to r. □

Theorem 4.2. *The* 2VC-CMSN *problem with unit vertex weights and $k = 2$ is polynomial time solvable.*

5. Open questions

Unlike the CMST problem, for which the approximation ratio for uniform vertex-weighted graphs is smaller than that for non-uniform vertex-weighted graphs, our algorithm guarantees an approximation ratio of 4 for all graphs. Are there better approximations for the CMSN problem in uniform vertex-weighted graphs? Also, are there better approximations for geometric graphs? The lower bounds that we used are weak as one can easily construct an instance whose optimal CMSN cost is at least $\Omega(n/k)$ times the cost of an MST. Similarly one can construct an instance whose optimal CMSN cost is at least $\Omega(k)$ times the cost of the spoke lower bound (Lemma 3.1). We believe that finding better lower bounds could help find better algorithms. One other interesting open problem would be to approximate CMSNs for general graphs (whose edges may not satisfy triangle inequality). We do not know at this time how to obtain a non-trivial approximation ratio for general graphs.

References

[1] R.K. Ahuja, J.B. Orlin, D. Sharma, A composite neighborhood search algorithm for the capacitated minimum spanning tree problem, Oper. Res. Lett. 31 (2003) 185–194.

[2] K. Altinkemer, B. Gavish, Heuristics for unequal weight delivery problems with a fixed error guarantee, Oper. Res. Lett. 6 (1987) 149–158.

[3] K. Altinkemer, B. Gavish, Heuristics with constant error guarantees for the design of tree networks, Management Sci. 34 (1988) 331–341.

[4] A. Amberg, W. Domschke, S. Voß, Capacitated minimum spanning trees: Algorithms using intelligent search, J. Comb. Optim. Theory and Practice 1 (1996) 9–39.

[5] N. Christofides, Worst case analysis of a new heuristic for the traveling salesman problem, Tech. report GSIA, Carnegie-Mellon Univ., 1976.

[6] L.R. Esau, K.C. Williams, On teleprocessing system design, IBM Sys. J. 5 (1966) 142–147.

[7] G.N. Frederickson, J. JáJá, On the relationship between the biconnectivity augmentation and traveling salesman problems, Theoret. Comput. Sci. 19 (2) (1982) 189–201.

[8] B. Gavish, K. Altinkemer, Parallel savings heuristics for the topological design of local access tree networks, in: Proc. IEEE INFOCOM, 1986, pp. 130–139.

[9] R. Jothi, B. Raghavachari, Design of local access networks, in: Proc. of 15th IASTED Internat. Conf. on Parallel and Distributed Computing and Systems (PDCS), 2003, pp. 883–888.

[10] R. Jothi, B. Raghavachari, Approximation algorithms for the capacitated minimum spanning tree problem and its variants in network design, in: Proc. 31st International Colloquium on Automata, Languages and Programming (ICALP), 2004, in press.

[11] S. Khuller, U. Vishkin, Biconnectivity approximations and graph carvings, J. ACM 41 (1994) 214–235.

[12] J. Orlin, Personal communication, MIT, 2003.

[13] C.H. Papadimitriou, The complexity of the capacitated tree problem, Networks 8 (1978) 217–230.

[14] F.S. Salman, J. Cheriyan, R. Ravi, S. Subramanian, Buy-at-bulk network design: Approximating the single-sink edge installation problem, in: Proc. of Symp. on Discrete Algorithms (SODA), 1997, pp. 619–628.

[15] Y.M. Sharaiha, M. Gendreau, G. Laporte, I.H. Osman, A tabu search algorithm for the capacitated shortest spanning tree problem, Networks 29 (1997) 161–171.
[16] D. Sharma, Personal communication, University of Michigan, Ann Arbor, 2003.
[17] S. Voß, Capacitated minimum spanning trees, in: C.A. Floudas, P.M. Pardalos (Eds.), Encyclopedia of Optimization, vol. 1, Kluwer, Boston, 2001, pp. 225–235.