OPERATIONS RESEARCH CENTER

Working Paper

Intuitive Solution-Double Techniques for Worst-case analysis of Some Survivable Network Design Problems.

by

T. Magnanti
A. Balakrishnan
P. Mirchandani

OR 342-00 February 2000

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

Intuitive Solution-Doubling Techniques for Worst-case Analysis of Some Survivable Network Design Problems

Anantaram Balakrishnan
Smeal College of Business Administration
Penn State University
University Park, PA

Thomas L. Magnanti
School of Engineering
Massachusetts Institute of Technology
Cambridge, MA

Prakash Mirchandani
Katz Graduate School of Business
University of Pittsburgh
Pittsburgh, PA

February 2000

Abstract

An intuitive solution-doubling argument establishes well-known results concerning the worst-case performance of spanning tree-based heuristics for the Steiner network problem and the traveling salesman problem. This note shows that the solution-doubling argument and its implications apply to certain more general Low Connectivity Steiner (LCS) problems that are important in the design of survivable telecommunication networks. We use the doubling strategy to establish worst-case upper bounds on the value of tree-based heuristics relative to the optimal value for some variants of the LCS problem, and also provide a lower bound based on solutions to matching problems.

1. Introduction

Motivated by the telecommunications industry's desire to guarantee minimal service disruptions, particularly to institutional customers, while also limiting the total investment in the network needed to provide such guarantees, operations researchers have studied various Survivable Network Design (SND) problems. In its general form, the SND problem seeks a minimum cost network topology containing a prespecified number of edge-disjoint paths, say r_{ij}, between every pair of nodes *i* and *j* in an undirected network $G:(N,E)$ with node set *N* and edge set *E*. Each edge $(i,j) \in E$ has a nonnegative cost c_{ij}, and the objective is to minimize the total cost of the chosen edges. For each node *i*, we define the node connectivity parameter ρ_i as $\max\{r_{ij} : j \in N\}$. We refer to nodes with connectivity parameters ρ_i equal to zero, one, and greater than one as *Steiner*, *regular*, and *critical* nodes, and refer to the regular and critical nodes together as *terminal* nodes. Let *C* and *T* denote the critical and terminal nodes. Steiner nodes are optional intermediate points that the design might use to connect the terminal nodes.

Recent research has focused on developing strong formulations, algorithms, and heuristic worst-case bounds for SND problems. In particular, researchers and practitioners have devoted considerable attention to the special class of *Low Connectivity Steiner (LCS)* problems in which $\rho_i = 0,\ 1,\ \text{or } 2$ for every node *i*. Both the Steiner network problem and the traveling salesman problem are special cases of the LCS problem. We will consider LCS problems with or without *edge duplication*, and with general or triangular edge costs (i.e., costs satisfying the triangle inequality). Edge duplication refers to the option of installing multiple copies of the same edge to provide alternate edge-disjoint paths between critical nodes. We assume that the connectivity requirements r_{ij} are such that both the overall design and the subnetwork spanning the critical nodes are connected in any feasible solution.

This paper is motivated by two primary questions. First, do some of the heuristic worst-case results developed for the Steiner tree and traveling salesman problems extend to LCS problems? For instance, a well-known Steiner tree result states that the minimum cost tree spanning just the terminal nodes is no more than twice as expensive as the optimal Steiner tree (Takahashi and Matsuyama [1980]). Similarly, for traveling salesman problems defined over graphs with triangular edge costs, a spanning tree-based heuristic generates a solution that costs no more than twice the optimal value (Papadimitriou and Steiglitz [1982]). Is there an unified view of these results, and does a similar approach apply more broadly to other LCS problems? Specifically, can we show that the solution to the "terminal" version of the LCS problem, ignoring the Steiner nodes, costs at most twice the optimal value? We confirm this conjecture using a "solution-doubling" argument. Second, a natural heuristic strategy for solving LCS problems is to build a solution in one of two ways: (a) design a two-connected network spanning all the critical nodes and then extend this network to span the remaining regular nodes, or (b) compose a LCS solution by adding the edges of two subnetworks, one that spans all the terminal nodes and one that spans just the critical nodes. What is the worst-case performance of either of these heuristic methods? Again, we use solution-doubling arguments to answer some of these questions. As by-products of structural properties that we examine for these problems, we also establish some worst-case bounds that were previously obtained using linear programming-based arguments. Our bounds compare the heuristic solution value to the optimal integer value, whereas some of the previous bounds are stronger because they compare the heuristic value to the optimal value of the linear programming relaxation. However, the doubling argument is intuitive and is easier to develop.

2. Bounding by Doubling

Using a graphical doubling procedure, we establish bounds on the optimal value of some LCS problems. In particular, using this argument we show that a Tree Completion heuristic has a worst-case bound of two for LCS problems as well as a Ring-on-Steiner tree problem and certain other SND special cases. We begin with a general bounding result.

Notation.

Given a graph G, "triangularizing" this graph corresponds to constructing a complete graph G_Δ with edge costs a_{ij} equal to the length of the shortest path from node i to node j in G. For any set N' of nodes of a graph G, the induced graph $G(N')$ contains all edges of G that have both endpoints in N'. Let $MST(N')$ denote the minimum spanning tree problem defined on the induced subgraph $G_\Delta(N')$ of the triangularized graph G_Δ. Similarly, $TSP(N')$ is the traveling salesman problem defined on $G_\Delta(N')$.

If P is any optimization problem, we let Z_P denote its optimal objective value. If M is any solution method (typically a heuristic procedure) for solving P, we let Z^M denote the objective value of the solution that the method produces. For any subgraph S of a graph G, let $Z(S)$ represent the total cost (sum of edge costs) of the subgraph.

Recall that any Eulerian graph (that is, any connected graph in which each node has an even degree) contains a walk, called an Eulerian walk, that uses each edge exactly once. In our analysis, we will use the following technical result, which involves an Eulerian subgraph EG and a subset of nodes N' that, in general, intersects the nodes of EG.

Proposition 1.

Given a graph G with nonnegative edge costs and a nonempty subset N' of its nodes, for any connected subgraph SG containing the nodes of N' and any Eulerian (possibly empty) subgraph EG of SG,

$$Z_{MST(N')} + Z(EG) \le (1-\frac{1}{|N'|})Z_{TSP(N')} + Z(EG) < 2Z(SG) \qquad \text{if } EG \text{ is nonempty, and} \qquad (1a)$$

$$Z_{MST(N')} \le (1-\frac{1}{|N'|})Z_{TSP(N')} \le 2(1-\frac{1}{|N'|})Z(SG) \qquad \text{if } EG \text{ is empty.} \qquad (1b)$$

Proof:

Let DSG be the Eulerian graph formed by doubling the edges of SG, and let RG be the residual graph obtained by removing one copy of EG from DSG. RG is a connected Eulerian graph. Let $i_1 - i_2 - \cdots - i_K - i_1$ be the node sequence of an Eulerian walk W containing all the edges of RG. For convenience, assume that $i_1 \in N'$. In the triangularized graph G_Δ, form a traveling salesman tour $TOUR$ of the nodes in N' by deleting from the walk W every node $j \notin N'$ and every second and later occurrence of any node $j \in N'$ (except the final occurrence of i_1). When deleting any such node j we replace two adjacent edges (i, j) and (j,k) of W by the edge (i,k) in G_Δ. If $Z(TOUR)$ denotes the cost of this tour in G_Δ, then $Z_{TSP(N')} \le Z(TOUR) \le Z(W)$. By removing the largest cost edge from the optimal solution to $TSP(N')$, we obtain a tree $TREE$, spanning all the nodes in N', whose cost satisfies $Z_{MST(N')} \le Z(TREE) \le (1-1/|N'|)Z_{TSP(N')}$. These arguments imply that $Z_{MST(N')} \le (1-1/|N'|)Z_{TSP(N')} \le (1-1/|N'|)Z(W)$. Combining these inequalities with $Z(W) + Z(EG) = Z(DSG) = 2Z(SG)$, we obtain the desired results. ♦

Note that this proposition remains valid both with and without duplicated edges.

Observations.

(i) For the *Steiner tree* (*ST*) problem over the terminal nodes T, by considering the optimal Steiner tree as the subgraph SG with EG as the null graph, Proposition 1 becomes the familiar Steiner tree result (Takahashi and Matsuyama [1980]):

$$Z_{MST(T)} \le 2(1-\frac{1}{|T|})Z_{ST}. \qquad (2)$$

(ii) If G has triangular edge costs and no Steiner nodes, then with SG as a minimum spanning tree of G and EG as the null graph, Proposition 1 becomes the familiar TSP result (e.g., Papadimitriou and Steiglitz [1982]):

$$Z_{TSP(N)} \le 2Z_{MST(N)}. \qquad (3)$$

(iii) The *Ring-on-Steiner tree* (*RST*) problem is a constrained LCS problem in which all the critical nodes must lie on one simple circuit (that can contain non-critical nodes). Let *SG* be an optimal solution to this problem and *EG* the embedded circuit in *SG* spanning all the critical nodes. Since the cost of the optimal traveling salesman tour in $G_\Delta(C)$ does not exceed the cost of *EG*, Proposition 1 implies the inequality:

$$Z_{TSP(T)} + Z_{TSP(C)} < 2Z_{RST}. \qquad (4)$$

Given any LCS problem instance, we call the version of the problem restricted to just the terminal nodes (i.e., defined over the subgraph $G(T)$ induced by the terminal nodes *T*) as the *Terminal Low Connectivity (TLC)* problem. How much do we sacrifice in solution quality by ignoring the Steiner nodes?

Corollary 2.

For any graph with triangular costs or with edge duplication,

$$Z_{LCS} \leq Z_{TLC} \leq Z_{TSP(T)} \leq 2Z_{LCS}. \qquad (5)$$

Proof:

Select *SG* as the optimal LCS solution and *EG* as the null graph in Proposition 1, and note that since the optimal solution to *TSP(T)* is feasible for the TLC problem (when costs are triangular), $Z_{TLC} \leq Z_{TSP(T)}$. Hence, the inequalities in (5) hold for LCS problems with triangular costs. If edge duplication is permitted, we can assume triangular costs without loss of generality. ♦

Corollary 2 implies that any polynomial-time heuristic with a worst-case bound of α for the TLC problem is a polynomial-time heuristic with a worst-case bound of at most 2α for the LCS problem defined on a graph with triangular edge costs.

In some situations, it is possible to solve the TLC problem optimally. For example, consider the *Dual Path tree* problem: given an undirected graph with triangular edge costs and two critical nodes 1 and 2, find the minimum cost connected subgraph that spans all the nodes and contains two edge-disjoint paths between nodes 1 and 2. The *Dual Path Steiner tree* problem contains additional Steiner nodes that the solution can optionally use to reduce total cost. Balakrishnan, Magnanti, and Mirchandani [1998] describe a polynomial, matroid intersection-based algorithm for solving the Dual Path tree problem. Corollary 2 implies that the optimal solution obtained using this algorithm costs no more than twice the optimal value of the Dual Path Steiner tree problem.

2.1 EG-Tree problems

Given a graph G with terminal nodes $T \subseteq N$ and a collection $\mathbf{C} = \{S_1, S_2, \ldots, S_L\}$ of subgraphs of G, the *subgraph extension problem* seeks the minimum cost design that spans all the terminal nodes (via optional Steiner nodes) and contains a subgraph S_l belonging to $\mathbf{C}$. The LCS problem is a special case of this subgraph extension problem in which the collection $\mathbf{C}$ consists of all subgraphs that provide the required two connectivity among the critical nodes. We examine the performance of the following Tree Completion heuristic for solving this problem.

Tree Completion heuristic:

Step 1: Find an approximate or optimal (i.e., minimum cost) subgraph OG from the collection $\mathbf{C}$ of subgraphs of G.

Step 2: Contract OG into a single node, 0, and delete all but the lowest-cost edge from any parallel edges that this contraction creates. Triangularize the edge costs in this reduced graph to obtain graph G^*. Let T^* denote all the nodes in T that do not belong to OG. Find the minimum spanning tree *TREE* in the subgraph of G^* induced by $T^* \cup \{0\}$.

Step 3: The Tree Completion heuristic solution is the union of the edges in OG and the edges of the shortest paths in G connecting nodes i and j, for all edges (i, j) in *TREE*.

We will focus on collections $\mathbf{C}$ containing only Eulerian subgraphs. To emphasize this special case, we will denote the collection as $\mathbf{C}_{\mathrm{EG}}$, and refer to the corresponding subgraph extension problems as *EG-tree problems*. One example is the *k-path Steiner tree* problem, an extension of the Dual Path Steiner tree problem that requires k edge disjoint paths connecting the designated critical nodes 1 and 2. We assume, without loss of generality (by adding, if necessary, an artificial zero-cost path from node 1 to node 2), that k is even. For this problem, the class of Eulerian graphs $\mathbf{C}_{\mathrm{EG}}$ is the set of all k edge-disjoint paths in G connecting nodes 1 and 2. Finding the optimal solution in this collection $\mathbf{C}_{\mathrm{EG}}$ (in Step 1) is easy, with or without edge duplication. With edge duplication, the optimal solution consists of k copies of the shortest 1-to-2 path. Without edge duplication, the problem reduces to a minimum cost network flow problem with unit edge capacities, and a demand and supply of k units at nodes 1 and 2, respectively. The *Ring-on-Steiner tree* problem that we previously discussed is another example of EG-tree problems, with $\mathbf{C}_{\mathrm{EG}}$ equal to the set of simple circuits that visit all the critical nodes (via optional regular or Steiner nodes).

The next result uses Proposition 1 to bound the worst-case performance of the Tree Completion heuristic for EG-tree problems as a function of the worst-case performance of the method used to solve Step 1 in the Tree Completion heuristic.

Proposition 3.

For EG-tree problems with nonnegative costs, if we find an α-approximate solution to the problem in Step 1, then the cost of the solution produced by the Tree Completion heuristic is no more than $\beta = \min\{2\alpha, \alpha + 2(1 - \frac{1}{|T|})\}$ times the optimal EG-tree cost.

Proof:

The total cost *Z(TREE)* of the edges that the Tree Completion heuristic adds to *OG* in Step 2 is no more than $Z_{MST(T)}$. Let *SG** be an optimal solution to the EG-tree problem. By definition, this solution contains a subgraph *S* from the given collection $\mathbf{C}_{\mathrm{EG}}$; therefore, $Z(S) \le Z_{EG-tree}$. Since the embedded heuristic in Step 1 selects a subgraph *OG* that costs no more than α times the minimum cost subgraph in $\mathbf{C}_{\mathrm{EG}}$, $Z(OG) \le \alpha Z(S)$. Since the cost Z^{TC} of the tree completion heuristic solution equals $Z(OG) + Z(TREE)$, Proposition 1 with $EG = S$ and $SG = SG^*$ implies that $Z^{TC} \le Z_{MST(T)} + Z(OG) \le 2\alpha Z_{EG-tree}$. Moreover, with $EG = \phi$ and $SG = SG^*$, Proposition 1 implies that $Z_{MST(T)} \le 2(1 - 1/|T|) Z_{EG-tree}$. Hence, the result. ♦

Proposition 3 implies that for the *k*-path Steiner tree problem, the Tree Completion heuristic is a polynomial-time algorithm with a worst-case performance of at most two. Note that this analysis does not require triangular costs, and if the given class of Eulerian graphs does not duplicate any edges, then neither does the heuristic solution.

Worst-case example.

The Dual Path Steiner tree example in Figure 1 shows that the worst-case bound of 2 in Proposition 3 is best possible. In Figure 1(a), nodes 1 and 2 are the critical nodes; all the other nodes are regular nodes. Four paths connect nodes 1 and 2, each of length *q*. Two of these paths have one intermediate node, while the other two paths each have *q*–1 intermediate nodes. Figure 1(b) shows the Tree Completion heuristic solution. The first step chooses the two paths having a single intermediate node. Step 2 greedily connects the remaining regular nodes to this dual path. This solution costs $2q + 2(q-1) = 4q - 2$. The optimal solution (Figure 1(c)) costs $2q$. As *q* becomes large, the ratio of these costs approaches two.

Insert FIGURE 1 about here

2.2 A structural property for LCS problems

The doubling argument also permits us to establish a structural property for the broader class of LCS problems without requiring Eulerian subgraphs. Consider the "critical-connectivity" special case of the LCS problem in which every node is either a critical node, with connectivity requirement of two, or a Steiner node. We refer to this problem as the *Critical Connectivity Steiner (CCS)* problem, and denote its optimal value as Z_{CCS}. If we can establish a bound on the optimal value of the LCS or the CCS problem, can we establish a bound for the other problem?

We first consider a general result. Let *CLASS* be any family of subgraphs of graph *G*, and for any given subset of nodes $N' \subseteq N$, let *CLASS*(N') denote the members of *CLASS* that span (at least) the nodes N'. For instance, in one context, *CLASS* might represent subgraphs used to construct heuristic CCS solutions; in another context, *CLASS* might represent solutions to the Steiner network problem. We denote the value of the minimum cost subgraph in $CLASS(N')$ as $Z_{CLASS(N')}$.

Proposition 4.

Let $\beta(\cdot)$ be a nondecreasing, nonnegative real-valued function of the positive integers. For any graph *G* with nonnegative costs, if we permit edge duplication, the following two properties are equivalent.

(a) $Z_{CLASS(C)} \leq \beta(|C|)Z_{CCS}$ for the CCS problem defined over any set of critical nodes $C \subseteq N$.

(b) $Z_{CLASS(T)} + Z_{CLASS(C)} \leq 2\beta(|T|)Z_{LCS}$ for the LCS problem defined over any sets of terminal nodes $T \subseteq N$ and critical nodes $C \subseteq T$.

Proof:

If (b) is true, then since the CCS problem is a special case of the LCS problem with $T = C$, substituting *C* for *T* in (b) gives (a), i.e., (b) implies (a).

To establish the converse, choose any optimal solution *OS* to the LCS problem. Between every pair of critical nodes, the solution contains two edge-disjoint paths. Let Γ be the union of the edges in these paths. Consider a doubled solution containing two copies of each edge in *OS*. From this solution, extract one copy of Γ, and let *RG* denote the residual graph. Γ contains two paths joining the critical nodes, and *RG* contains two paths joining every pair of terminal nodes.

Let CCS and $\overline{CCS}$ denote the Critical Connectivity Steiner problems with T and C as the critical nodes. Property (a) implies that, $Z_{CLASS(T)} \le \beta(|T|)Z_{CCS} \le \beta(|T|)Z(RG)$, and property (a) and the monotonicity of $\beta(\cdot)$ implies that $Z_{CLASS(C)} \le \beta(|C|)Z_{\overline{CCS}} \le \beta(|T|)Z(\Gamma)$. But, by construction, $2Z_{LCS} = Z(RG) + Z(\Gamma)$ which, together with the previous inequalities, implies (b). ♦

Since the graph defined by the edges Γ need not be Eulerian, this result uses the doubling argument in a slightly different way than we have used it previously. The following three results are special cases of Proposition 4.

Corollary 5.

$Z_{MST(T)} + Z_{MST(C)} \le 2(1 - \frac{1}{|T|})Z_{LCS}$ for LCS problems with edge duplication.

For this corollary, *CLASS(N')* is the set of all spanning trees on the induced graph *G(N')* (since edge duplication is permitted, we assume without loss of generality that G is triangular). The function $\beta(|N'|) = (1 - 1/|N'|)$ is nonincreasing in $|N'|$. Furthermore, since deleting the most expensive edge in the optimal CCS solution (with C as the critical nodes) gives a spanning tree, we have $Z_{MST(C)} \le (1 - 1/|C|)Z_{CCS}$. Consequently, the result follows from Proposition 4.

The corollary implies that if we construct a heuristic solution to the LCS problem by adding the edges of $MST(T)$ to the edges of *MST(C)*, then this solution costs no more than twice the optimal LCS value. Note that both Proposition 4 and Corollary 5 apply even if the graph G does not contain any Steiner nodes.

The bound in Corollary 5 is tight. Consider a ring graph with n nodes, shown in Figure 2, on which every alternate node is a critical node. The cost of each edge on the circumference is 1, while each chordal edge connecting pairs of critical nodes has a cost of 2. When n is even, $Z_{MST(T)} + Z_{MST(C)} = (n-1)*1 + (n/2 - 1)*2 = 2n - 3$. Since $Z_{LCS} = n$, the right hand side of Corollary 5 equals $2n - 2$. Therefore, the lefthand side and the righthand side of Corollary 5 are asymptotically equal as n approaches infinity.

Insert FIGURE 2 about here

Balakrishnan et al. [2000] strengthen the bound in Corollary 5 (and its generalization to SND and CCS problems in which critical nodes have connectivity parameter $\rho \ge 2$), and show that the bound applies relative to the optimal value of the linear programming relaxation of a new class of connectivity-dividing cutset formulations (instead of the optimal integer programming value Z_{LCS} in Corollary 5).

Corollary 6.

$$Z_{TSP(T)} + Z_{TSP(C)} \leq \frac{8}{3} Z_{LCS} \text{ for LCS problems with edge duplication.}$$

In this context, *CLASS(N')* is the set of all Hamiltonian circuits through the nodes *N'*. Monma, Munson, and Pulleyblank [1990] showed that, for graphs with triangular costs, $Z_{TSP(C)} \leq \frac{4}{3} Z_{CCS}$, i.e., $\beta(\cdot) = \frac{4}{3}$. So, the result follows from Proposition 4. In Corollary 2 we showed that $Z_{TSP(T)} \leq 2Z_{LCS}$ if costs satisfy the triangle inequality (and hence when edge duplication is permitted). Corollary 6 provides a sharper lower bound than Corollary 2 whenever $Z_{TSP(C)} \geq \frac{1}{3} Z_{TSP(T)}$.

The example shown in Figure 3, a modified version of the example in Monma et al. [1990], demonstrates that the bound in Corollary 6 is tight. The shaded and clear circles in the network represent respectively critical and regular nodes. Each of the three parallel paths contains $2q+1$ nodes, and all the edges shown have unit cost. The LCS solution consists of all the edges on the three parallel paths, and so costs $6q$. The $TSP(T)$ solution, shown in Figure 3(b), has cost $8q$, while the $TSP(C)$ solution is the same as the $TSP(T)$ but with all regular nodes short-circuited. Each of these solutions cost $8q$ and, therefore, for this example the bound of Corollary 6 is tight.

Insert FIGURE 3 about here

Corollary 7.

If C and T have even cardinalities,

$$Z_{Match(T)} + Z_{Match(C)} \leq Z_{LCS} \text{ for LCS problems with edge duplication.}$$

In this context, *CLASS(N')* is the set of all matchings over *N'*, and $\beta(\cdot)$ equals $\frac{1}{2}$. Goemans and Bertsimas [1993] develop a parsimonious property and use Edmonds' [1965] perfect matching polytope result to establish the bound $Z_{Match(C)} \leq \frac{1}{2} Z_{CCS}$. Therefore, Corollary 7 follows from Proposition 4.

The bound in Corollary 7 is tight. To verify this observation, consider a problem instance defined over a graph G with four nodes, connected by the edges of the square and one diagonal edge, as shown in Figure 4a. The nodes incident to the diagonal edge (say the top left and the bottom right nodes) are the critical nodes. The diagonal edge costs two, while all other edges cost one. As Figures 4(b)–4(d) show, $Z_{Match(C)} + Z_{Match(T)} = 2 + 2 = Z_{LCS}$.

Insert FIGURE 4 about here

3. Conclusions

In this paper, we have developed a result for LCS problems that is analogous to a well-known Steiner tree result: if we solve an LCS problem without the Steiner nodes, the resulting solution costs at most twice the optimal value of the original LCS problem. The solution doubling argument used to prove this result applies to other related problems as well. For example, it permits us to use any heuristic with a worst-case bound of α for optimization problems defined over Eulerian graphs problems to develop a Tree Completion heuristic with a worst-case bound of 2α for subgraph extension versions of these problems. A similar doubling argument establishes relationships among the optimal objective value of certain LCS problems and MST solution values, TSP solution values, and costs of optimal matchings over regular and critical nodes.

References

BALAKRISHNAN, A., T. L. MAGNANTI, and P. MIRCHANDANI. 2000. Connectivity-Splitting Models for Survivable Network Design. Working Paper 00-3, Department of Management Science & Information Systems, Pennsylvania State University.

BALAKRISHNAN, A., T. L. MAGNANTI, and P. MIRCHANDANI. 1998. Designing Hierarchical Survivable Networks. *Operations Research*, Vol. 46, No. 1, pp. 116-136.

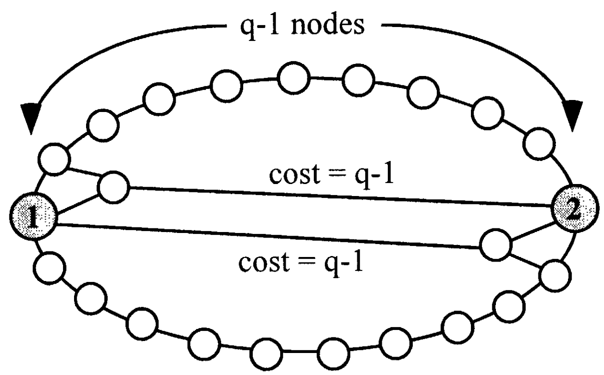
EDMONDS, J. 1979. Matroid Intersection. *Annals of Discrete Mathematics*, **4**, 39-49.

GOEMANS, M. X., and D. J. BERTSIMAS. 1993. Survivable Networks, Linear Programming Relaxations and the Parsimonious Property. *Math. Prog.*, **60**, 145-166.

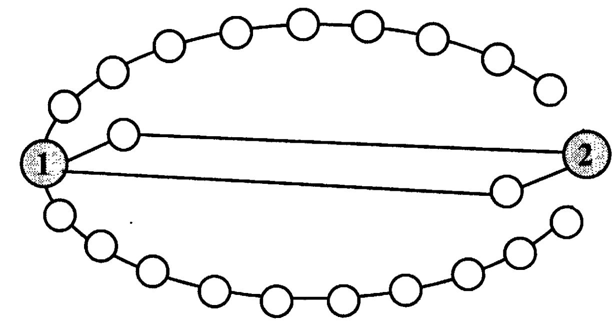
MONMA, C. L., B. S. MUNSON, and W. R. PULLEYBLANK. 1990. Minimum-weight Two-connected Spanning Networks. *Math. Prog.,* **46**, 153-171.

PAPADIMITRIOU, C. H., and K. STEIGLITZ. 1982. *Combinatorial Optimization: Algorithms and Complexity*. Prentice-Hall, New Jersey.

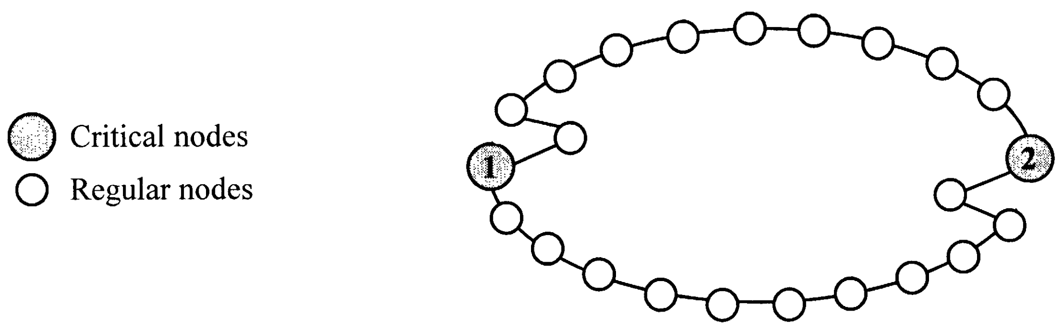
TAKAHASHI, H., and A. MATSUYAMA. 1980. An Approximation Solution for the Steiner Problem in Graphs. *Mathematica Japonica*, **24**, 573-577.



(a) Dual Path Steiner tree example
All edges except diagonal edges have unit cost.



(b) Tree Completion heuristic solution



(c) Optimal solution

Figure 1: Worst-case example for Dual Path Steiner tree problem

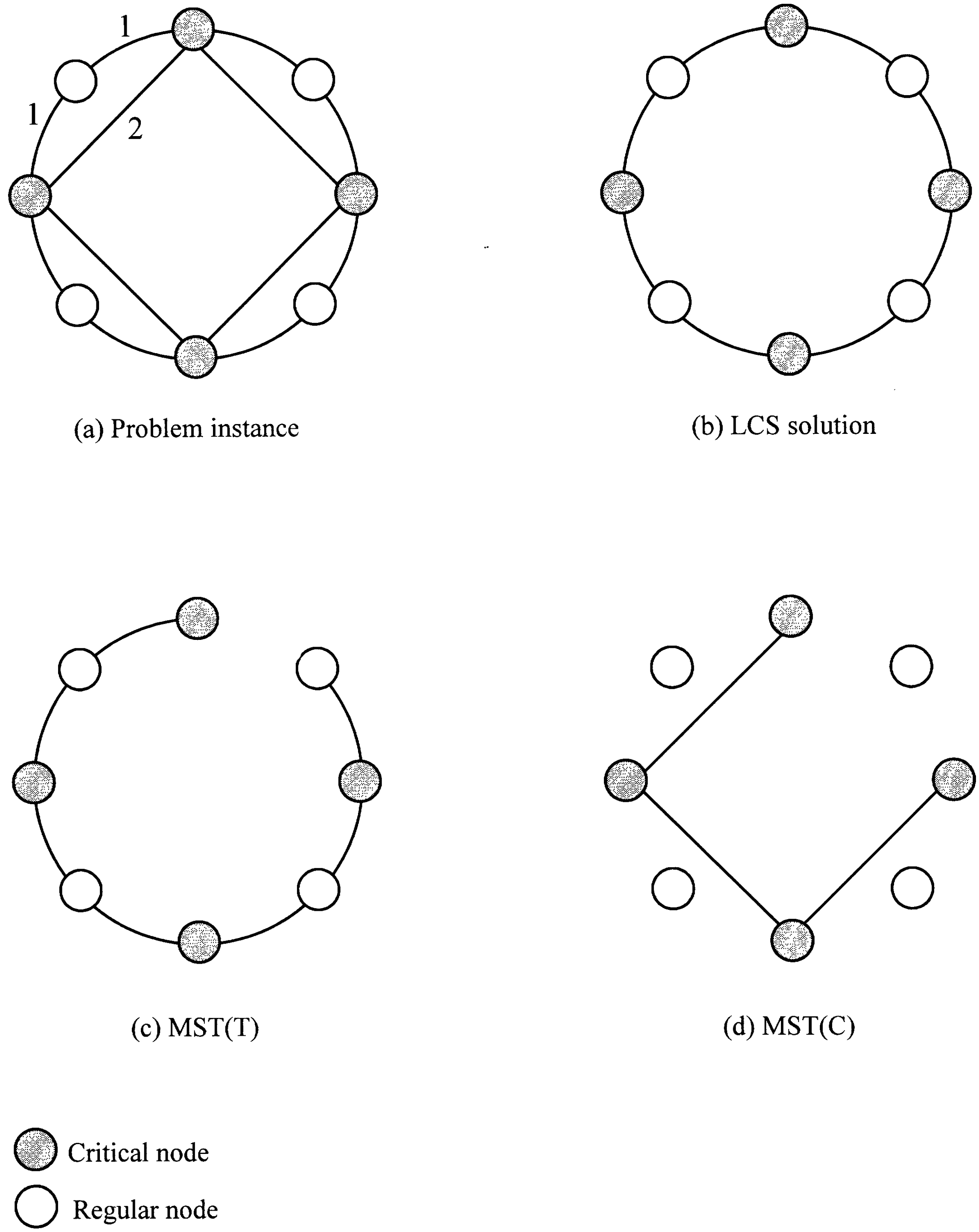
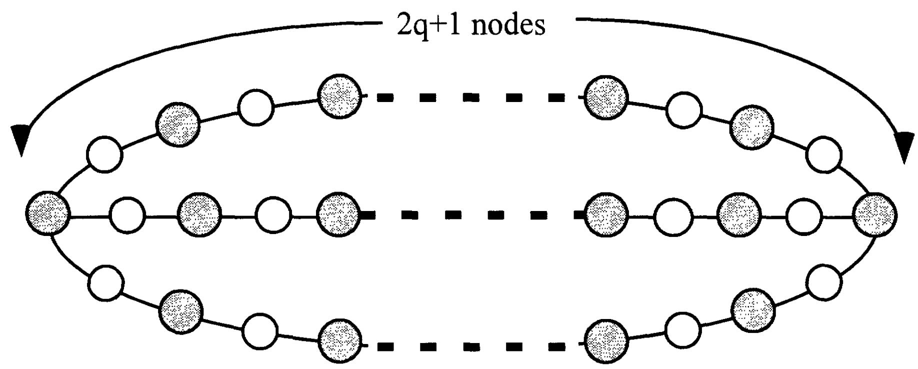
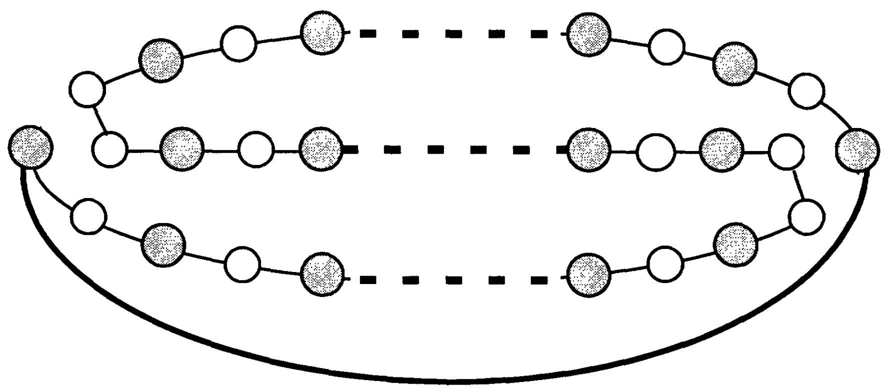


Figure 2: Worst-case example for Corollary 5



(a) Base graph for worst-case example
All edges shown have unit cost; LCS problem is
defined over triangularized version of base graph.



(b) Optimal TSP solution over terminal nodes

Critical nodes
Regular nodes

Figure 3: Worst-case example for Corollary 6

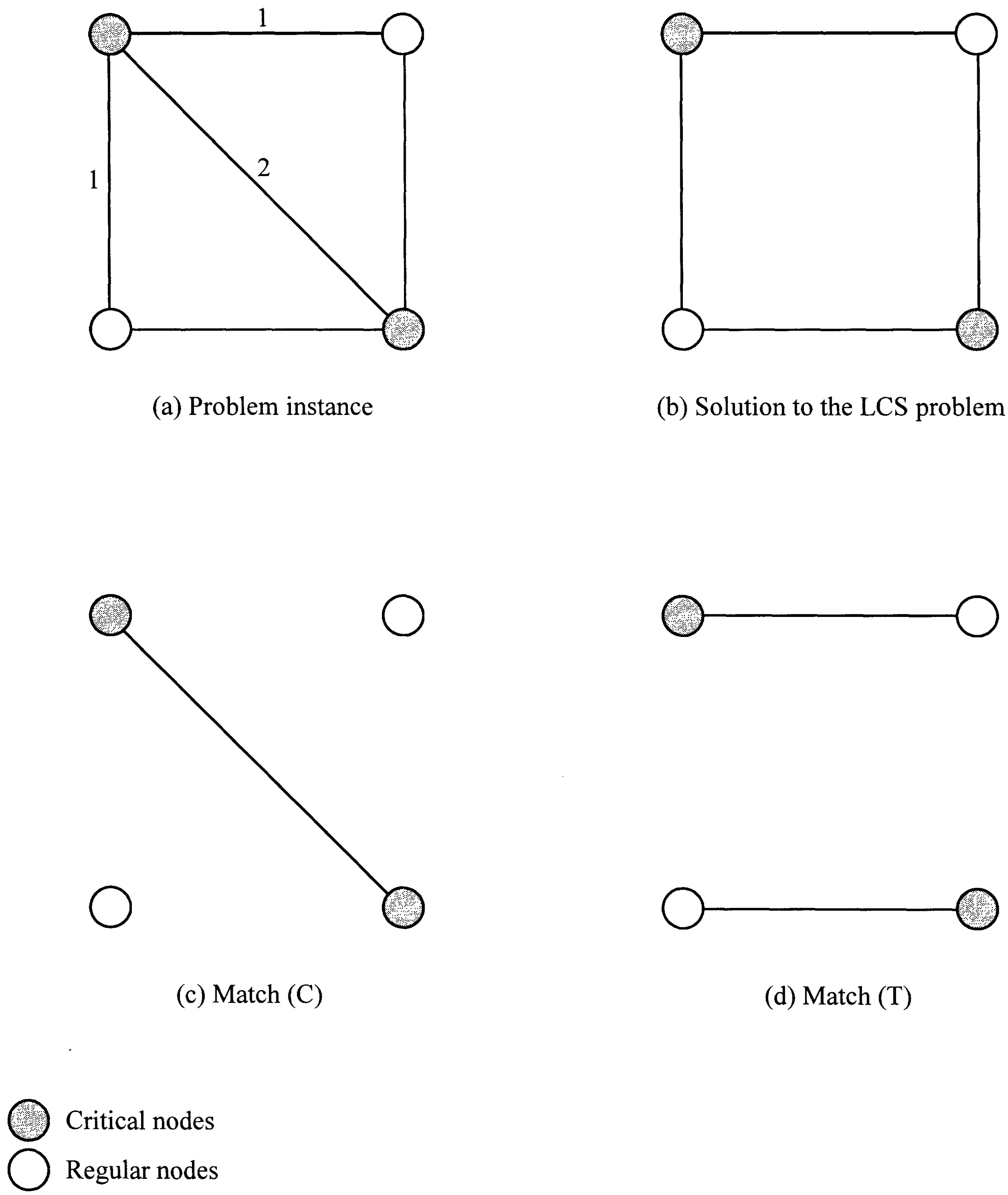


Figure 4: Worst-case example for Corollary 7