An Integer-Programming-Based Approach to the Close-Enough Traveling Salesman Problem

Behnam Behdani, J. Cole Smith
Department of Industrial and Systems Engineering, University of Florida, 303 Weil Hall, P.O. Box 116595, Gainesville, FL 32611-6595, USA, {behdani@ufl.edu, cole@ise.ufl.edu}

We address a variant of the Traveling Salesman Problem known as the Close-Enough Traveling Salesman Problem (CETSP), where the traveler visits a node if it enters a compact neighborhood set of that node. We formulate a mixed-integer programming model based on a discretization scheme for the problem. Both lower and upper bounds on the optimal CETSP tour length can be derived from the solution of this model, and the quality of the bounds obtained depends on the fidelity of the discretization scheme. Our approach first develops valid inequalities that enhance the bound and solvability of this formulation. We then provide two alternative formulations, one which yields an improved lower bound on the optimal CETSP tour length, and one which greatly improves the solvability of the original formulation by recasting it as a two-stage problem amenable to decomposition. Computational results demonstrate the effectiveness of the proposed methods.

Key words: Close-Enough Traveling Salesman Problem; Geometric Routing Problems; Mixed-Integer Programming; Computational Geometry; Discretization.

---

1. Introduction

Given a collection of nodes and the set of distances between each pair of nodes, the Traveling Salesman Problem (TSP) seeks to find a shortest tour that visits each node exactly once. The broad applicability of the TSP, or slight variations thereof, have resulted in an impressive slate of research on TSP-related problems. Several TSP extensions can be found in the literature, such as the Covering Salesman Problem (Current and Schilling, 1989), Prize Collecting TSP (Balas, 1989, 2004), and the Covering Tour Problem (CTP) (Gendreau et al., 1997). For a comprehensive survey on the history, algorithms, and applications of the TSP, the reader is referred to Applegate et al. (2006).

The Close-Enough Traveling Salesman Problem (CETSP) is a generalization of the TSP in which the salesman does not need to visit the exact location of each customer. Instead, a compact region of the plane containing each node is specified as its *neighborhood set*, and the goal is to find a shortest tour that starts from a specified depot location and intersects all of these neighborhood sets. Intuitively speaking, in the CETSP, each of the salesman's clients is willing to travel to any point inside its particular neighborhood to meet with the salesman. A typical application of the CETSP arises when an airborne vehicle is trying to find a stationary target on a two-dimensional field by scanning a collection of candidate locations. Suppose that detection occurs when the aircraft is no more than r units away from the target. Then the neighborhood set of a target is a disc of radius r at the target's location. The problem of finding the shortest trajectory for the aircraft is equivalent to finding a tour of minimum length that intersects each neighborhood set.

Another application for the CETSP arises in wireless sensor network operations. In these networks, sensor nodes periodically relay data that they have accumulated to a mobile data-collecting device called a *sink*. Energy minimization is an important component of wireless sensor networks. Ciullo et al. (2010) show that by letting the sink come closer to sensors with higher data generation rates, one can significantly reduce the consumed energy for the transmission of data. Therefore, it is reasonable to assign a sensing radius r_i to each sensor node i based on its data generation rate, and require that the sink must visit a point within the r_i-neighborhood of node i to collect its data. Again, the problem of finding a shortest trajectory of the sink to retrieve data from all sensors is an instance of the CETSP (see also Yuan et al. 2007).

Gulczynski et al. (2006) propose several heuristics for a common special case of the CETSP where the neighborhood sets of all nodes are discs of the same radius. The problem arises in situations where radio frequency identification (RFID) tags are used that can remotely provide a mobile data collector with the required data. As an example, many utility companies are now using RFID-based automated meter readers that can read the usage of each customer remotely (Shuttleworth et al., 2008). Mennell (2009) proposes a heuristic algorithm based on *Steiner zones*, which are nonempty intersections of the neighborhood sets. Their approach consists of three phases: (a) identifying a collection of Steiner zones that cover every neighborhood set; (b) representing each Steiner zone with one of its points; and (c) finding a TSP tour over these representative points.

Arkin and Hassin (1994) propose polynomial-time approximation algorithms for several special cases of the problem. In particular, they provide algorithms that yield error bounds for the CETSP in which the neighborhood sets either take the form of parallel unit segments, translates of polygonal regions, or discs. Other approximation algorithms include the work of Mata and Mitchell (1995) and Dumitrescu and Mitchell (2003).

The heuristic algorithm of Mennell (2009) and the polynomial-time approximation algorithms in Arkin and Hassin (1994), Dumitrescu and Mitchell (2003), and Mata and Mitchell (1995) are able to efficiently find a good feasible CETSP tour. One difficulty in evaluating the quality of such feasible solutions is the lack of exact algorithms for the CETSP in the literature. Hence, developing tight lower bounds for the CETSP is of crucial importance. Such lower bounds would enable one to (conservatively) evaluate the quality of a feasible tour obtained by a non-exact algorithm. While there exist several methods of efficiently obtaining such lower bounds for the TSP, developing lower bounds for the CETSP is considered to be a difficult task. Our contribution is targeted toward finding arbitrarily tight lower and upper bounds on the optimal CETSP tour length via mixed-integer programming models.

Some of the most successful exact algorithms for solving the symmetric TSP combine the use of cutting planes and efficient heuristics in a branch-and-cut scheme. Examples of such branch-and-cut algorithms are the work of Padberg and Rinaldi (1991) and the well known Concorde TSP solver. The approach that we propose in this paper requires solving integer programming problems to obtain a series of upper and lower bounds that converge to an optimal CETSP tour.

The rest of this paper is organized as follows. Section 2 establishes foundational concepts related to optimal CETSP tours and the discretization scheme used in this paper. Section 3 provides a mathematical programming formulation that yields a lower bound on the optimal CETSP tour length. Based on the anticipated difficulties of solving this model, we propose an alternative model and associated two-stage optimization algorithm in Section 4. We present computational results that demonstrate the efficiency of our approach in Section 5, and conclude the paper in Section 6.

2. Preliminaries

We begin by stating the formal definition of the CETSP and describing the notation used in this paper in Section 2.1, and then present our core discretization scheme in Section 2.2.

2.1 Definitions and Notation

Let M be a set of points in a two-dimensional plane along with a depot point p_0. For each point $m \in M$, let S_m be a compact set that contains m. The CETSP seeks to find a shortest tour (with respect to Euclidean distance) that starts from p_0, intersects every set $S_1, \ldots, S_{|M|}$ (in any order), and terminates at p_0.

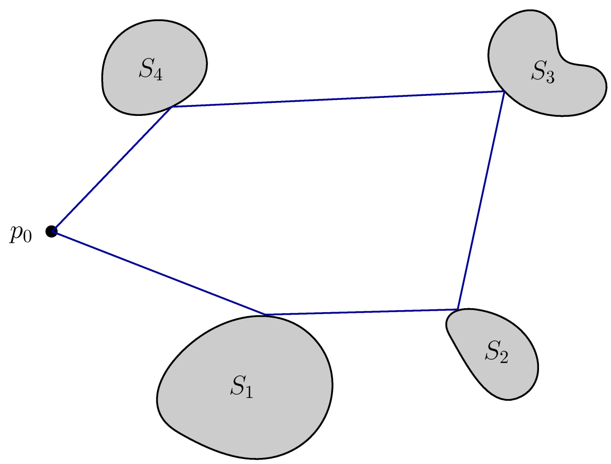

Figure 1: A feasible CETSP tour with $|M| = 4$

Figure 1 illustrates a CETSP tour that intersects four compact sets. Note that the CETSP is clearly NP-hard as it reduces to the TSP when each set S_m consists of a single point. Without loss of generality, we assume that $p_0 \notin S_m$, $\forall m \in M$, or else we can simply ignore all $m \in M$ for which $p_0 \in S_m$.

We will refer to the elements of M as the *target locations* and their associated compact sets as the *neighborhood sets*. These neighborhood sets are often discs, e.g., for the air monitoring and meter reading applications mentioned previously. However, the approach that we present in this paper for solving the CETSP in general does not depend on the shape of the neighborhood sets.

The following proposition is useful in the development of our algorithm.

Proposition 1. All optimal solutions to the CETSP consist of a finite set of connected line segments (p_0p_1), (p_1p_2), ..., $(p_{k-1}p_k)$, (p_kp_0), where $k \leq |M|$. Moreover, for each point p_i, $i = 1, \ldots, k$, there exists at least one $m \in M$ such that p_i is on the boundary of S_m.

Proof. Let T be any feasible CETSP tour. We can select a finite number of points $p_1, \ldots, p_k$ on T such that $k \leq |M|$ and $\left(\cup_{j=1}^{k} p_j\right) \cap S_m \neq \emptyset$, $\forall m \in M$. Note that the line segment between any pair of these points is the unique minimizer of distance between them. Therefore, the class of solutions composed of line segments connecting these points dominates all other classes of solutions, and so an optimal CETSP tour must consist of some $k \leq |M|$ line segments.

Hence, we now consider an optimal tour consisting of line segments $(p_0 p_1)$, $(p_1 p_2)$, $\ldots$, $(p_{k-1} p_k)$, $(p_k p_0)$, where $1 \leq k \leq |M|$. For each $i = 1, \ldots, k$, define V_i as the set of target neighborhoods visited by p_i, but not by $p_1, \ldots, p_{i-1}$, i.e.,

$$V_i = \{m \in M : p_i \in S_m \text{ and } m \notin V_j \text{ for } j < i\}. \tag{1}$$

We assume that each V_i is nonempty; otherwise, if $V_i = \emptyset$ then we can skip the corresponding point p_i in the tour (moving from p_{i-1} to p_{i+1}, where $p_{k+1} = p_0$), which results in a tour whose length is no more than the original $(k+1)$-link tour. Let

$$R_i = \cap_{m \in V_i} S_m, \qquad \forall i = 1, \ldots, k. \tag{2}$$

Each R_i is a closed set since all the sets S_m are so. We claim that there exists an optimal tour in which p_i is on the boundary of R_i. By contradiction, suppose that p_i belongs to the interior of R_i and the two line segments $(p_{i-1} p_i)$ and $(p_i p_{i+1})$ intersect the boundary of R_i at points p_i' and p_i'', respectively (Figure 2). An alternative tour that traverses from p_i' directly to p_i'' and bypasses p_i is no longer than T, but still intersects R_i. Therefore, there must exist an optimal solution in which each point p_i, $i = 1, \ldots, k$, lies on the boundary of at least one set S_m for some $m \in M$. This completes the proof. $\square$

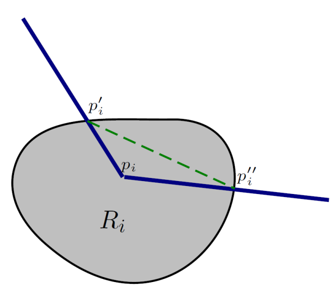

Figure 2: Illustration of boundary point optimality

Proposition 1 implies that any optimal solution to the CETSP can be characterized by a discrete set of points in the plane, where each point belongs to the boundary of at least one neighborhood set. In the rest of the paper, we will call these the *turn points* of the corresponding tour. Proposition 2 further characterizes those sections of the boundary sets that may contain a turn point of an optimal tour. (The proof of this proposition is contained in Section A-1 of the Online Supplement.)

Proposition 2. Suppose that every neighborhood set S_m is a disc centered at $m \in M$. Let T be an optimal CETSP tour that is characterized by a set of turn points $\{p_0, \ldots, p_k\}$. Then, $p_i \in conv(M \cup \{p_0\})$, for $i = 1, \ldots, k$, where $conv(M \cup \{p_0\})$ denotes the convex hull of the corresponding points.

2.2 CETSP Partitioning Schemes

Since an optimal CETSP tour can be represented by a finite number of turn points, a natural way of obtaining a feasible tour is to approximate the solution space by a discrete set of points. However, such an approach results only in an upper bound on the optimal CETSP tour length. Our approach for obtaining lower bounds on the optimal CETSP tour length in this paper is based on partitioning the continuous solution space into smaller sets and identifying those partitions that possibly contain a turn point of an optimal CETSP tour.

Definition 1. A set $C = \{C_0, \ldots, C_n\}$ is called a CETSP-partitioning of the two dimensional plane if:

1. $C_0 = \{p_0\}$.
2. C_i is a nonempty compact set, for all $1 \leq i \leq n$.
3. $conv(C_i \cap C_j)$ has an empty interior, for all $1 \leq i, j \leq n$.
4. $(\cup_{i=1}^{n} C_i) \supseteq (\cup_{i \in M} B_m)$, where B_m is the boundary of S_m.

When every neighborhood set S_m is a disc that is centered at its corresponding target location, B_m in the above definition can be restricted to include those points on the boundary of S_m that lie inside $conv(M \cup \{p_0\})$. □

We refer to the elements of a CETSP-partitioning C as *cells*. For any two cells C_i and C_j, l_{ij} is defined as the shortest line segment length that connects the boundaries of C_i and C_j. See Section A-2 of the Online Supplement for details on computing l_{ij}-values.

In this paper, we consider two ways of partitioning the plane: grid-based and arc-based. In a grid-based partitioning, each grid cell C_i is a rectangle that intersects at least one neighborhood set (see Figure 3). The grid cells are chosen so that they collectively contain all the neighborhood sets and the depot. Moreover, we will restrict ourselves to grid-based partitionings in which there exists no neighborhood set that is entirely contained in one of the rectangles. We define a set P of *grid points* as the collection of p_0 and any other point that is a vertex of at least one grid cell. Throughout the paper, we will represent grid cell i by a tuple $(a_i, b_i, w_{1i}, w_{2i})$, where (a_i, b_i) denotes the lower-left vertex, and w_{1i} and w_{2i} are the two side lengths of the rectangle, so that $(a_i + w_{1i}, b_i + w_{2i})$ denotes the upper-right vertex.

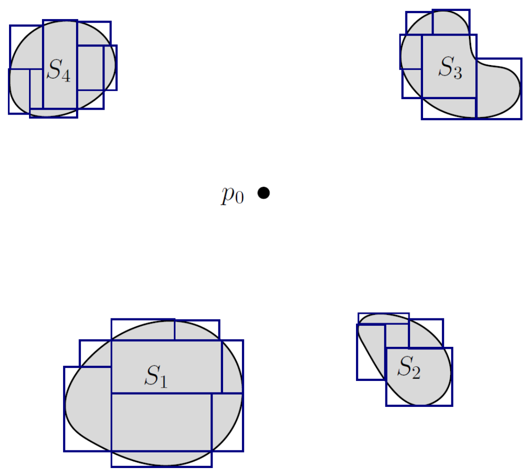

Figure 3: Illustration of grid-based partitioning

As opposed to the grid-based scheme, arc-based partitioning is primarily intended for circular neighborhood sets. Figure 4 illustrates an example of an arc-based partitioning for a CETSP instance. Here, a cell is defined as an arc (specified by its two endpoints) on the boundary of a set S_m that lies on or inside the convex hull of the target locations and the depot. Note that by Proposition 2, every turn point of an optimal CETSP tour belongs to at least one arc in this partitioning. We will specify an arc with $(a_i, b_i, r_i, \alpha_{1i}, \alpha_{2i})$ where (a_i, b_i) and r_i are respectively the center and radius of the corresponding disc, and α_{1i} and α_{2i} respectively denote the start and finish angles of the corresponding arc. The *central angle* of an arc is defined as $\alpha_i = \alpha_{2i} - \alpha_{1i}$. Throughout the paper, we will assume that $\alpha_i < \pi$ for all CETSP-partitioning arcs. Similar to the grid-based case, set P is defined as the set of endpoints of all arcs unioned with p_0.

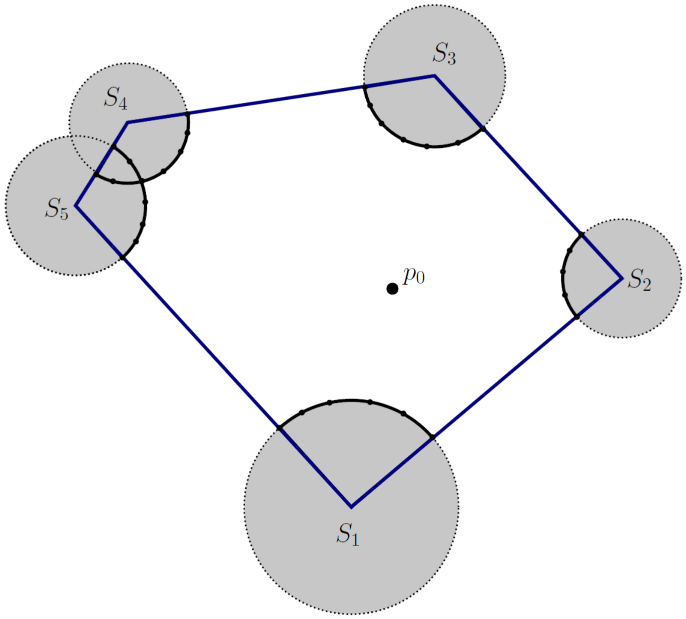

Figure 4: Illustration of arc-based partitioning

3. Lower Bounding Model

We first formulate a mixed-integer program (MIP) for obtaining lower bounds to the optimal CETSP tour length in Section 3.1, along with a closed-form expression for an accompanying upper bound. We then describe two cutting-plane strategies to aid in the solution of this model in Section 3.2, and comment on the complexity of their separation routines.

3.1 MIP Formulation and Bounds

Let $C = \{C_0, \ldots, C_n\}$ be a CETSP-partitioning of the plane with a pairwise distance matrix $L = \{l_{ij}\}$. For each $m \in M$ define the set of cells intersecting S_m as $N(m) = \{1 \leq i \leq n : C_i \cap S_m \neq \emptyset\}$. Note by Definition 1 we have $N(m) \neq \emptyset$ for all $m \in M$. Consider the following MIP.

$$\text{Min} \quad \sum_{i=0}^{n} \sum_{j=0}^{n} l_{ij} x_{ij} \tag{3a}$$

$$\text{s.t.} \quad \sum_{j=0}^{n} x_{ij} = \sum_{j=0}^{n} x_{ji}, \quad \forall i = 0, \ldots, n \tag{3b}$$

$$y_i = \sum_{j=0}^{n} x_{ij}, \quad \forall i = 0, \ldots, n \tag{3c}$$

$$\sum_{i \in N(m)} y_i \geq 1, \quad \forall m \in M \tag{3d}$$

$$\sum_{i \in \mathcal{S}} \sum_{j \notin \mathcal{S}} x_{ij} \geq y_v, \quad \forall \mathcal{S} \subset \{1, \ldots, n\} : 2 \leq |\mathcal{S}| \leq |C| - 2 \text{ and } v \in \mathcal{S} \tag{3e}$$

$$x_{ij} \in \{0, 1\}, \quad \forall i = 0, \ldots, n, \ j = 0, \ldots, n \tag{3f}$$

$$0 \leq y_i \leq 1, \ \forall i = 1, \ldots, n; \quad y_0 = 1. \tag{3g}$$

An optimal solution to (3) is in fact a shortest TSP tour with respect to distance matrix L, over a subset of C that visits at least one element in $N(m)$ for each $m \in M$. Decision variable x_{ij} indicates whether the corresponding tour moves from C_i to C_j, while auxiliary binary variable y_i indicates whether C_i is visited on the tour. The objective function (3a) minimizes the total distance traveled in the tour. Constraints (3b) ensure that for each node i, the number of incoming tour arcs equals the number of outgoing tour arcs. Constraints (3c) define y-variables in terms of x-variables. (In fact, the formulation can be given without the y-variables; they are included only for convenience in presentation.) Constraints (3d) ensure that for each $m \in M$, at least one element of C is visited that covers m. Constraints (3e) are subtour elimination constraints (see Gendreau et al., 1998). Finally, (3f) and (3g) state logical restrictions and bounds on the variables.

Problem (3) is a special case of the CTP (Gendreau et al., 1997). The CTP is defined on a graph $G = (V \cup W, E)$ where V is the set of vertices that can be visited and W is the set of targets that must be covered by vertices in V. For every $w \in W$, there exists a nonempty subset $V^w \subseteq V$ of vertices that cover w. There is also a subset $\overline{V} \subseteq V$ containing the vertices that must be visited by any feasible tour. The goal in the CTP is to find a shortest tour on a subset of V that visits all vertices of $\overline{V}$ as well as at least one vertex of V^w, for every $w \in W$. Therefore, problem (3) is a special case of the CTP on a complete directed graph with $V = C \cup \{p_0\}$, $W = M$, $\overline{V} = \{p_0\}$, and $V^m = N(m)$.

The following proposition establishes a relationship between the optimal objective function value of problem (3) and the optimal CETSP tour length.

Proposition 3. Suppose that the optimal CETSP tour length is l^*, and consider an optimal solution $(\mathbf{x}^*, \mathbf{y}^*)$ to (3) with objective function value z_{LB1}. Define $I = \{i : y_i^* = 1\}$, which indicates the set of *turn cells* associated with this optimal solution, and let $N_i = |\{m \in M :$

$i \in N(m)\}|$ be the number of neighborhood sets intersected by C_i. Then

$$z_{LB1} \leq l^* \leq z_{LB1} + \sum_{i \in I} h_i, \tag{4}$$

where

$$h_i = \begin{cases} 2N_i r_i \sin\left(\frac{\alpha_i}{2}\right) & \text{if } N_i \leq 2, \\ 2r_i \left(2\sin\left(\frac{\alpha_i}{4}\right) + (N_i - 1)\sin\left(\frac{\alpha_i}{2(N_i-1)}\right)\right) & \text{if } N_i \geq 3, \end{cases} \tag{5}$$

for an arc-based CETSP-partitioning, and

$$h_i = \begin{cases} 2\sqrt{{w_{1i}}^2 + {w_{2i}}^2} & \text{if } N_i = 1, \\ w_{1i} + w_{2i} + \sqrt{{w_{1i}}^2 + {w_{2i}}^2} & \text{if } N_i = 2, \\ 2(w_{1i} + w_{2i}) & \text{if } N_i = 3, \\ 2(w_{1i} + w_{2i}) + \sqrt{{w_{1i}}^2 + {w_{2i}}^2} & \text{if } N_i \geq 4, \end{cases} \tag{6}$$

for a grid-based CETSP-partitioning.

Proof. Suppose that an optimal CETSP tour T is characterized by an ordered set of turn points $(p_0, \ldots, p_k)$. Let $C'_t = C_{i_t} \in C$ be an element of the CETSP-partitioning that contains p_t, for $t = 1, \ldots, k$ and define $C'_{k+1} = C'_0 = C_0$ and $i_{k+1} = i_0 = 0$. Because cells might contain multiple turn points, some elements of $C' = \{C'_1, \ldots, C'_k\}$ may be identical. Suppose that

- $y^R_{i_t} = 1$, for $t = 0, \ldots, k$,
- $x^R_{0,i_1} = 1$, and
- $x^R_{i_t,i_s} = 1$, for $t = 1, \ldots, k$ and $s > t$, if and only if $x^R_{i_u,i_t} = 1$ for some $u < t$ and s is the first index strictly greater than t such that $C'_s \notin \{C'_1, \ldots, C'_t\}$.

By construction, $(\mathbf{x}^R, \mathbf{y}^R)$ defines a tour over a subset $\{C_{i_0}, \ldots, C_{i_k}\} \subseteq C$ that covers all the targets in M. Hence, $(\mathbf{x}^R, \mathbf{y}^R)$ is feasible to (3). Let $z^R = \sum_{(i,j) \in A^R} l_{i,j}$ denote the objective function value of $(\mathbf{x}^R, \mathbf{y}^R)$, where $A^R = \{(i,j) : 0 \leq i, j \leq n, x^R_{i,j} = 1\}$. Now suppose $x^R_{i_t,i_s} = 1$. Because l_{i_t,i_s} is defined as the minimum distance between C_{i_t} and C_{i_s} and $p_t \in C_{i_t}$ and $p_s \in C_{i_s}$, we have

$$l_{i_t,i_s} \leq \sum_{j=t}^{s-1} |p_j p_{j+1}|, \tag{7}$$

where $|p_i p_{i+1}|$ is the length of line segment between p_i and p_{i+1}, with $p_{k+1} = p_0$. Aggregating (7) over all elements of A^R, we obtain $z^R \leq l^*$. Therefore, $z_{LB1} \leq z^R \leq l^*$.

To prove the validity of the upper bound in (4), we form a feasible CETSP tour $\hat{T}$ whose length is bounded by $z_{LB1} + \sum_{i \in I} h_i$. Recall that an optimal solution $(\mathbf{x}^*, \mathbf{y}^*)$ to (3) consists of a collection of (possibly disconnected) links between turn cells in I (see Figure 5 for example). Suppose $x^*_{ji} = x^*_{ik} = 1$. That is, the corresponding optimal solution to (3) contains an incoming link (with length l_{ji}) from C_j to a point p^i_1 in C_i. Similarly, there is a point p^i_2 in C_i from which there exists a link to C_k. (Note that p^i_2 may not be the same point as p^i_1.) One can construct $\hat{T}$ from this optimal solution to (3) by connecting p^i_1 to p^i_2 for each $i \in I$ via a shortest trajectory that visits all N_i targets that are covered by C_i.

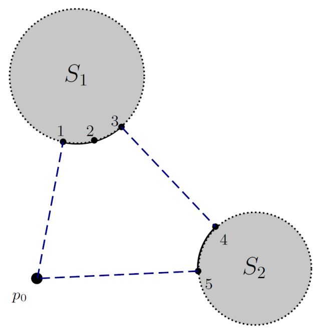

Figure 5: An instance of the lower bound problem (3) and its optimal solution

We first consider an arc-based CETSP-partitioning where each C_i is an arc of radius r_i and central angle α_i. The connecting trajectory in this case starts from p^i_1, visits (at most) $N_i - 1$ *middle points* on the arc to cover all the other targets that are covered by C_i, and then moves to p^i_2. Because this trajectory interconnects a collection of points on the underlying arc, it consists of several chords. The chord length of an arc with radius r and central angle α equals $2r \sin(\alpha/2)$. When $N_i \leq 2$, a shortest connecting trajectory consists of at most N_i chords and therefore (5) holds. For $N_i \geq 3$, we first compute an upper bound on the shortest total length of $N_i - 1$ chords that interconnect p^i_1 and the remaining middle points. This bound is given by the optimal objective function value of the following problem.

$$\text{Max} \sum_{k=1}^{N_i-1} 2r_i \sin\left(\frac{\beta_k}{2}\right) \tag{8}$$

$$\text{s.t.} \sum_{k=1}^{N_i-1} \beta_k \leq \alpha_i \tag{9}$$

$$\beta_k \geq 0, \quad \forall k = 1, \ldots, N_i - 1. \tag{10}$$

Since $\alpha_i < \pi$, the objective function is concave and so the unique optimal solution to the above problem is $\beta_1 = \cdots = \beta_{N_i-1} = \alpha_i/(N_i - 1)$, and the maximum length of these $N_i - 1$ chords is given by $2r_i(N_i-1)\sin(\alpha_i/2(N_i-1))$. However, we possibly need additional chords to complete this connecting trajectory. When p_1^i is an endpoint of the underlying arc, the connecting section starts from p_1^i, visits all the middle points, and then traverses to p_2^i. An upper bound on the length of the trajectory in this case is given by $2r_i \sin(\alpha_i/2) + 2r_i(N_i - 1)\sin(\alpha_i/2(N_i-1))$. Now suppose p_1^i is not an endpoint of the underlying arc, and let P_1 be the endpoint that is closer to p_1^i (compared to p_2^i) and P_2 be the other endpoint. One possible way to connect p_1^i and p_2^i is to start from p_1^i, visit all the middle points on the way to P_1, come back to p_1^i, visit the remaining middle points, and finally move to p_2^i. In this case, two chords ($P_1p_1^i$, and $P_2p_2^i$) are added to the previous $N_i - 1$ chords. Let γ_1 and γ_2 be the central angles corresponding to the two new chords. Because $\gamma_1 + \gamma_2 \leq \alpha_i$, the total length of these two new chords cannot exceed $4r_i \sin(\alpha_i/4)$ (which is greater than $2r_i \sin(\alpha_i/2)$ for all values of α_i). Hence, an upper bound on the length of the shortest connecting trajectory is given by $4r_i \sin(\alpha_i/4) + 2r_i(N_i-1)\sin(\alpha_i/2(N_i-1))$.

Now consider a grid-based CETSP-partitioning. The worst-case scenario for each value of N_i occurs when p_1^i exists at a vertex of the grid cell. When $N_i = 1$, the length of a minimal connecting segment is clearly no more than $2\sqrt{{w_{1i}}^2 + {w_{2i}}^2}$, which happens when $p_1^i = p_2^i$ but a diagonal of the grid cell needs to be traversed once to cover a target, and once to return to p_2^i. When $N_i = 2$, the connecting trajectory consists of three connected line segments. It can be verified that the maximum length of the minimal trajectory in the underlying rectangle is given by $w_{1i} + w_{2i} + \sqrt{{w_{1i}}^2 + {w_{2i}}^2}$, which happens when $p_1^i = p_2^i$ and two of the other grid vertices need to be visited to cover the corresponding targets. When $N_i = 3$, the maximum length happens when $p_1^i = p_2^i$ and the other three grid vertices of the rectangle need to be visited to cover all the targets. (The details showing that the situations above are indeed the worst cases when $N_i = 2$ or 3 are intuitive and are omitted for brevity.) Finally, when $N_i \geq 4$, one can ensure that all N_i targets are covered by starting from p_1^i, traversing the perimeter of the rectangle and then moving to p_2^i. (No interior point of the grid cell needs to be visited by our assumption that no S_m is contained within the interior of any cell.) The length of the connecting trajectory in this case is bounded by $2(w_{1i} + w_{2i}) + \sqrt{{w_{1i}}^2 + {w_{2i}}^2}$.

□

Remark 1. Note that the upper bounds in (4) are obtained by performing a worst-case analysis. In practice, one can design a postprocessing subroutine that uses the solution of (3) as well as the problem's geometry to build a feasible CETSP tour, which would typically yield a smaller upper bound than the one given in Proposition 3. A heuristic algorithm (such as those in Jozefowiez et al. 2007 and Mennell 2009) can also be used for this purpose. Alternatively, one can obtain an upper bound on the optimal CETSP tour length by solving (3) over a set P of grid points, where $p_0 \in P$ and $P \cap S_m \neq \emptyset$, $\forall m \in M$, with respect to the Euclidean distance between the corresponding points. $\square$

Proposition 3 implies that the set of links in an optimal solution to (3) can be used to form a feasible CETSP tour whose length can become arbitrarily close to the optimal CETSP tour length. To that end, one can refine the underlying partitioning in such a way that the grid cell sizes belonging to I are small enough to yield an acceptably small optimality gap. One possible strategy is to solve (3) and obtain I, subdivide all cells in I into smaller cells, and solve the revised instance of (3) in a repeated fashion.

3.2 Cutting-Plane Generation

Subtour Elimination Constraints. Because there are an exponential number of subtour elimination constraints (3e), we initially relax these constraints and add those that are violated (with respect to a pre-determined violation threshold) at each node of the branch-and-bound tree. Here, we discuss the corresponding separation procedures.

To find a subtour elimination constraint (3e) that is violated by a solution $(\overline{\mathbf{x}}, \overline{\mathbf{y}})$, we use a well known separation procedure (see Fischetti et al., 1997; Valle et al., 2009) as follows. Define a complete directed graph with the node set $\{0, \dots, n\}$ in which the capacity of an arc (i, j) equals $\overline{x}_{ij}$. For each $v \in \{1, \dots, n\}$ with $\overline{y}_v > 0$, we find the maximum flow from 0 to v, which will generate a minimum cut $(\overline{S}, S)$ where $0 \in \overline{S}$ and $v \in S$. If the capacity of this cut is less than $\overline{y}_v$, that is, if $\sum_{i \in \overline{S}} \sum_{j \in S} \overline{x}_{ij} < \overline{y}_v$, then S and v define a violated inequality (3e). When this procedure does not find a violated subtour inequality, $(\overline{\mathbf{x}}, \overline{\mathbf{y}})$ satisfies all the subtour elimination constraints (3e).

Model Tightening Inequalities. Recall that the objective function coefficients in (3) are pairwise minimum distances of the corresponding cells. In general, the triangle inequality does not hold for these cost coefficients, which may result in a weak lower bound. To illustrate this point, note that there exists an optimal CETSP tour in which no two turn points cover an identical set of targets (or otherwise, bypassing either of them results in a CETSP tour that is no longer than the original one). However, it is possible that two cells on the tour cover the same set of targets in a unique optimal solution to (3). Consider the CETSP instance in Figure 5, where C_1 is the arc from point 1 to 2, and C_2 is the arc from point 2 to 3. Here, $x_{12} = 1$ at optimality. This optimal solution contains both C_1 and C_2, which cover the same target, and allows the tour to go from 1 to 3 at a distance of zero (noting that $l_{12} = 0$ because point 2 belongs to C_1 and C_2). The following proposition, which is a generalization of similar inequalities for the CTP (Gendreau et al., 1997), can be added to (3) to strengthen the obtained bound.

Proposition 4. Consider two nonempty subsets C^1 and C^2 of C such that $C^1 \cap C^2 = \emptyset$. Define

$$V(C^p) = \cup_{C_i \in C^p} \{m \in M : i \in N(m)\}, \tag{11}$$

for $p = 1, 2$. If $V(C^1) \subseteq V(C^2)$, then the following inequality is valid:

$$\sum_{i \in C^1 \cup C^2} y_i \leq \min\{|C^1| + |C^2| - 1, |V(C^2)|\}, \tag{12}$$

in the sense that solving model (3) augmented with (12) still yields a valid lower bound on the optimal CETSP tour length.

Proof. To cover all the target locations in $V(C^2)$, a shortest CETSP tour needs at most $|V(C^2)|$ turn points. Moreover, an optimal CETSP tour will not contain a turn point in every cell of $C^1 \cup C^2$ because in that case, a shorter feasible tour can be constructed by excluding any of the turn points in C^1. As a result, $(\mathbf{x}^R, \mathbf{y}^R)$ in the proof of Proposition 4 satisfies (12), which completes the proof. □

Remark 2. In Proposition 4, (12) can be strengthened in the following form when $|C^1| \geq 2$, $|C^2| \geq 2$, and $V(C^1) = V(C^2)$:

$$\sum_{i \in C^1 \cup C^2} y_i \leq \min\{|C^1| + |C^2| - 2, |V(C^2)|\}. \tag{13}$$

The reduction in the first term is due to the fact that if an optimal CETSP tour contains a turn point in every cell in C^1 (or C^2), then no cells in C^2 (or C^1) are visited by this optimal tour or otherwise, omitting those turn points in C^2 (or C^1) results in a shorter tour. Since

$|C^1|$ and $|C^2|$ are both at least two, $(\mathbf{x}^R, \mathbf{y}^R)$ in the proof of Proposition 4 satisfies (13) and so the inequality obtains a valid bound in this case. Else, there exists at least one cell in both C^1 and C^2 that does not contain any turn point of an optimal CETSP tour, and (13) holds in this case as well. $\square$

Given a (possibly fractional) vector $\bar{\mathbf{y}}$, a CETSP-partitioning C, a set M of targets, and the covering set $V_i = V(\{C_i\})$ for each $C_i \in C$, the separation problem of (12) seeks to find nonempty disjoint subsets C^1 and C^2 of C such that $V(C^1) \subseteq V(C^2)$ and $\sum_{i \in C^1 \cup C^2} \bar{y}_i > |C^1| + |C^2| - 1$. Let this decision problem be denoted by SP1. Similarly, let SP2 be the decision problem associated with separating (13). The proof for the following complexity result is provided in Section A-3 of the Online Supplement.

Theorem 1. Problems SP1 and SP2 are strongly NP-complete.

Given the worst-case complexity of separation routines for these inequalities, we add these inequalities to the formulations only when $|C^1| = |C^2| = 1$.

4. Alternative Model Formulations and Algorithms

In this section, we present an alternative model in Section 4.1, which captures a tighter bound on minimum distances traveled in a tour, at the expense of creating additional binary decision variables. We explore subtour elimination constraints for this model in Section 4.2. We then formulate a third lower-bounding model in Section 4.3, and demonstrate that this model is solvable by a Benders decomposition strategy.

4.1 Expanded Formulation

Another way of reducing the conservativeness of formulation (3) is to redefine the decision variables so that they capture some of the travel distances inside the cells, and hence, contribute to a tighter lower bound. To that end, we need to add sequence-related binary variables to the problem. Let $s_{ijk} = x_{ij}x_{jk}$ be a binary variable that equals one if and only if the solution to the lower bound problem consecutively visits C_i, C_j, and C_k. Moreover, let e_{ijk} ($\geq l_{ij} + l_{jk}$) denote the length of a shortest path that goes from a point in C_i to some point in C_j, and then from the same point in C_j to a point in C_k. (Section A-2 of the Online Supplement discusses details pertaining to the calculation of these e_{ijk}-values.) Figure 6 illustrates the definition of e_{ijk} for an arc-based partitioning scheme. Consider the following integer programming problem.

$$
\begin{aligned}
\text{Min} \quad & \frac{1}{2}\sum_{i=0}^{n}\sum_{j=0}^{n}\sum_{k=0}^{n} e_{ijk}s_{ijk} && \text{(14a)}\\
\text{s.t.} \quad & \sum_{l=0}^{n} s_{lij} = \sum_{k=0}^{n} s_{ijk}, \quad \forall i = 0,\dots,n,\ j = 0,\dots,n && \text{(14b)}\\
& y_j = \sum_{i=0}^{n}\sum_{k=0}^{n} s_{ijk}, \quad \forall j = 0,\dots,n && \text{(14c)}\\
& \sum_{i\in N(m)} y_i \geq 1, \quad \forall m \in M && \text{(14d)}\\
& \sum_{i\in S}\sum_{j\notin S}\sum_{k=0}^{n} s_{ijk} \geq y_v, \quad \forall \mathcal{S} \subset \{1,\dots,n\} : 3 \leq |\mathcal{S}| \leq |C| - 3 \text{ and } v \in \mathcal{S} && \text{(14e)}\\
& s_{ijk} \in \{0,1\}, \quad \forall i = 0,\dots,n,\ j = 0,\dots,n,\ k = 0,\dots,n && \text{(14f)}\\
& 0 \leq y_i \leq 1,\ \forall i = 1,\dots,n; \quad y_0 = 1. && \text{(14g)}
\end{aligned}
$$

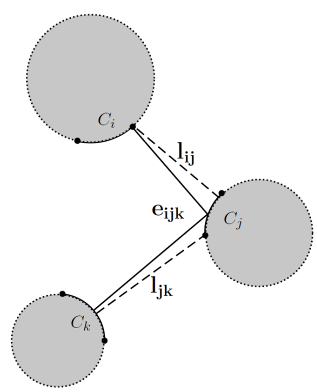

Figure 6: Illustration of the objective function coefficients e_{ijk} in (14)

Proposition 5. Let l^* be the optimal CETSP tour length and suppose z_{LB1} and z_{LB2} are the optimal objective function values of problems (3) and (14), respectively. Then,

$$z_{LB1} \leq z_{LB2} \leq l^*. \tag{15}$$

Proof. Suppose again that an optimal CETSP tour T visits (in order) the set of turn points $(p_0,\dots,p_k)$. Similar to the proof of Proposition 3, let $C'_t = C_{i_t} \in C$ be an element of the CETSP-partitioning that contains p_t, for $t = 1,\dots,k$ and define $C'_{k+1} = C'_0 = C_0$

and $i_{k+1} = i_0 = 0$. Consider $(\mathbf{x}^R, \mathbf{y}^R)$ as defined in the proof of Proposition 3 and let $s^R_{ijk} = x^R_{ij} x^R_{jk}$, for all $0 \leq i, j, k \leq n$. Note that $(\mathbf{s}^R, \mathbf{y}^R)$ defines a feasible solution to (14). Let z^R_{LB1} and z^R_{LB2} denote the objective function value of $(\mathbf{x}^R, \mathbf{y}^R)$ and $(\mathbf{s}^R, \mathbf{y}^R)$, respectively. By definition, if $s^R_{i_t, i_u, i_v} = 1$, we have

$$l_{i_t,i_u} + l_{i_u,i_v} \leq e_{i_t,i_u,i_v} \leq \sum_{j=t}^{v-1} |p_j p_{j+1}|, \tag{16}$$

where $p_{k+1} = p_0$. Aggregating all of the above inequalities yields $2z^R_{LB1} \leq 2z^R_{LB2} \leq 2l^*$. Therefore, $z_{LB2} \leq z^R_{LB2} \leq l^*$. Now let $(\hat{\mathbf{s}}, \hat{\mathbf{y}})$ be any feasible solution to (14). Corresponding to $(\hat{\mathbf{s}}, \hat{\mathbf{y}})$, we define a unique feasible solution $(\hat{\mathbf{x}}, \hat{\mathbf{y}})$ to (3), where $\hat{x}_{ij} = \sum_{k=1}^n \hat{s}_{ijk}$, for all $0 \leq i, j \leq n$. Using $l_{ij} + l_{jk} \leq e_{ijk}$, it is straightforward to show that $\hat{z}_{LB1} \leq \hat{z}_{LB2}$, where $\hat{z}_{LB1}$ and $\hat{z}_{LB2}$ denote the objective function value of $(\hat{\mathbf{x}}, \hat{\mathbf{y}})$ and $(\hat{\mathbf{s}}, \hat{\mathbf{y}})$, respectively. Therefore, $z_{LB1} \leq z_{LB2}$, which concludes the proof. $\square$

4.2 Subtour Elimination

To separate (14e), we can use the same procedure specified for the separation of (3e), with the modification of setting the capacity of arc (i, j) as $\sum_{k=0}^n \overline{s}_{ijk}$. For the expanded model, though, we also consider an alternative form of subtour elimination constraints as follows.

Proposition 6. Consider the following set of subtour elimination constraints for (14).

$$\sum_{i\in\mathcal{S}} \sum_{j\in\mathcal{S}} \sum_{k\in\mathcal{S}} s_{ijk} - \sum_{i\notin\mathcal{S}} \sum_{k\notin\mathcal{S}} s_{ivk} \leq \sum_{j\in\mathcal{S}\setminus\{v\}} y_j - y_v,$$
$$\forall \mathcal{S} \subset \{1, \ldots, n\} : 3 \leq |\mathcal{S}| \leq |C| - 3 \text{ and } v \in \mathcal{S}. \tag{17}$$

Let

$$Y^1 = \{(\mathbf{s}, \mathbf{y}) : (\mathbf{s}, \mathbf{y}) \text{ is feasible to (14)}\},$$
$$Y^2 = \{(\mathbf{s}, \mathbf{y}) : (\mathbf{s}, \mathbf{y}) \text{ is feasible to (14) with (14e) substituted by (17)}\}.$$

Moreover, let $\overline{Y}^1$ be the LP relaxation polytope of (14) and define $\overline{Y}^2$ as the LP relaxation polytope of (14) with (17) instead of (14e). Then, $Y^1 = Y^2$ and $\overline{Y}^2 \subseteq \overline{Y}^1$.

Proof. We first prove that $\overline{Y}^2 \subseteq \overline{Y}^1$ by showing that every (possibly fractional) solution to $\overline{Y}^2$ satisfies (14e) as well. By rearranging (17) we obtain

$$\sum_{i\in\mathcal{S}} \sum_{j\in\mathcal{S}} \sum_{k\in\mathcal{S}} s_{ijk} + 2y_v \leq \sum_{j\in\mathcal{S}} y_j + \sum_{i\notin\mathcal{S}} \sum_{k\notin\mathcal{S}} s_{ivk}. \tag{18}$$

Note that $\sum_{i \notin \mathcal{S}} \sum_{k \notin \mathcal{S}} s_{ivk} \leq \sum_{i \notin \mathcal{S}} \sum_{j \in \mathcal{S}} \sum_{k \notin \mathcal{S}} s_{ijk}$ and $\sum_{j \in \mathcal{S}} y_j = \sum_{i=0}^{n} \sum_{j \in \mathcal{S}} \sum_{k=0}^{n} s_{ijk}$. Therefore, we have

$$2y_v \leq \sum_{i=0}^{n} \sum_{j \in \mathcal{S}} \sum_{k=0}^{n} s_{ijk} + \sum_{i \notin \mathcal{S}} \sum_{j \in \mathcal{S}} \sum_{k \notin \mathcal{S}} s_{ijk} - \sum_{i \in \mathcal{S}} \sum_{j \in \mathcal{S}} \sum_{k \in \mathcal{S}} s_{ijk}. \tag{19}$$

The above can be restated as

$$2y_v \leq \sum_{i \notin \mathcal{S}} \sum_{j \in \mathcal{S}} \sum_{k \in \mathcal{S}} s_{ijk} + \sum_{i \in \mathcal{S}} \sum_{j \in \mathcal{S}} \sum_{k \notin \mathcal{S}} s_{ijk} + 2 \sum_{i \notin \mathcal{S}} \sum_{j \in \mathcal{S}} \sum_{k \notin \mathcal{S}} s_{ijk} \tag{20a}$$

$$= \sum_{i \notin \mathcal{S}} \sum_{j \in \mathcal{S}} \sum_{k=0}^{n} s_{ijk} + \sum_{i=0}^{n} \sum_{j \in \mathcal{S}} \sum_{k \notin \mathcal{S}} s_{ijk}. \tag{20b}$$

Note that the first term of (20b) represents the number of times a (possibly fractional) tour enters set $\mathcal{S}$, and the second term represents the number of times a tour exits set $\mathcal{S}$. By (14b), these values must be equal, and so (20b) reduces to $2 \sum_{i=0}^{n} \sum_{j \in \mathcal{S}} \sum_{k \notin \mathcal{S}} s_{ijk}$. But since

$$\sum_{i=0}^{n} \sum_{j \in \mathcal{S}} \sum_{k \notin \mathcal{S}} s_{ijk} = \sum_{i \in \mathcal{S}} \sum_{j \notin \mathcal{S}} \sum_{k=0}^{n} s_{ijk}, \tag{21}$$

we have that (20) implies (14e).

We now prove that $Y^1 = Y^2$. We showed above that if a solution satisfies (17), it also satisfies (14e). Therefore, $Y^2 \subseteq Y^1$. Now consider an integer solution $(\bar{\mathbf{s}}, \bar{\mathbf{y}})$ in Y^1, which represents a tour over a subset of C that intersects p_0. We show that $(\bar{\mathbf{s}}, \bar{\mathbf{y}})$ satisfies (17) as well. Suppose $\mathcal{S} \subset C$ is chosen such that $0 \notin \mathcal{S}$ and $v \in \mathcal{S}$. Consider the following cases.

- $\sum_{i \in \mathcal{S}} \sum_{j \in \mathcal{S}} \sum_{k \in \mathcal{S}} \bar{s}_{ijk} \geq 1$: Note that

$$\sum_{j \in \mathcal{S}} \bar{y}_j = \sum_{i \in \mathcal{S}} \sum_{j \in \mathcal{S}} \sum_{k \in \mathcal{S}} \bar{s}_{ijk} + \sum_{i \notin \mathcal{S}} \sum_{j \in \mathcal{S}} \sum_{k \in \mathcal{S}} \bar{s}_{ijk} + \sum_{i \in \mathcal{S}} \sum_{j \in \mathcal{S}} \sum_{k \notin \mathcal{S}} \bar{s}_{ijk} + \sum_{i \notin \mathcal{S}} \sum_{j \in \mathcal{S}} \sum_{k \notin \mathcal{S}} \bar{s}_{ijk}. \tag{22}$$

  Moreover, we have

$$\sum_{i \notin \mathcal{S}} \sum_{j \in \mathcal{S}} \sum_{k \in \mathcal{S}} \bar{s}_{ijk} \geq 1 \tag{23}$$

$$\sum_{i \in \mathcal{S}} \sum_{j \in \mathcal{S}} \sum_{k \notin \mathcal{S}} \bar{s}_{ijk} \geq 1 \tag{24}$$

$$\sum_{i \notin \mathcal{S}} \sum_{j \in \mathcal{S}} \sum_{k \notin \mathcal{S}} \bar{s}_{ijk} \geq 0. \tag{25}$$

  Therefore,

$$\sum_{i \in \mathcal{S}} \sum_{j \in \mathcal{S}} \sum_{k \in \mathcal{S}} \bar{s}_{ijk} \leq \sum_{j \in \mathcal{S}} \bar{y}_j - 2 \leq \sum_{j \in \mathcal{S}} \bar{y}_j - 2y_v. \tag{26}$$

  and (17) holds.

- $\sum_{i\in\mathcal{S}}\sum_{j\in\mathcal{S}}\sum_{k\in\mathcal{S}}\overline{s}_{ijk}=0$ and $\overline{y}_v=0$: (17) clearly holds in this case.
- $\sum_{i\in\mathcal{S}}\sum_{j\in\mathcal{S}}\sum_{k\in\mathcal{S}}\overline{s}_{ijk}=0$ and $\overline{y}_v=1$: In this case, if $\sum_{j\in\mathcal{S}\setminus\{v\}}y_j\geq 1$ then (17) is valid. If $\sum_{j\in\mathcal{S}\setminus\{v\}}y_j=0$, then $\sum_{i\notin\mathcal{S}}\sum_{k\notin\mathcal{S}}s_{ivk}=1$, and (17) remains valid. $\square$

Note that since $Y_1=Y_2$, the same separation procedure that generates (14e) can also be used to generate (17).

We conclude this section by presenting stronger subtour elimination constraints for (14). Suppose $\mathcal{S}\subset\{1,\ldots,n\}$ and let $\overline{\mathcal{S}}=\{0,\ldots,n\}\setminus\mathcal{S}$. Define $\widetilde{M}(\mathcal{S})=M\setminus\{m\in M:\overline{\mathcal{S}}\cap N(m)\neq\emptyset\}$, i.e., $\widetilde{M}(\mathcal{S})$ is the set of targets not covered by a cell in $\overline{\mathcal{S}}$. Let $\widetilde{C}(\mathcal{S})$ be a smallest cardinality subset of $\mathcal{S}$ that covers all targets in $\widetilde{M}(\mathcal{S})$. Then

$$\sum_{i\in\mathcal{S}}y_i\geq|\widetilde{C}(\mathcal{S})| \tag{27}$$

is valid for (14). If $|\widetilde{C}(\mathcal{S})|\geq 1$, then (27) implies the validity of the following stronger subtour elimination constraint for (14):

$$\sum_{i\notin\mathcal{S}}\sum_{j\in\mathcal{S}}\sum_{k=0}^{n}s_{ijk}\geq 1. \tag{28}$$

Note that (28) is a modified version of similar connectivity inequalities for the CTP (Gendreau et al., 1997). When $|\widetilde{C}(\mathcal{S})|\geq 2$, (28) can be further strengthened as follows.

$$\sum_{i\notin\mathcal{S}}\sum_{j\in\mathcal{S}}\sum_{k\in\mathcal{S}}s_{ijk}+\sum_{i\in\mathcal{S}}\sum_{j\in\mathcal{S}}\sum_{k\notin\mathcal{S}}s_{ijk}+\sum_{i\notin\mathcal{S}}\sum_{j\in\mathcal{S}}\sum_{k\notin\mathcal{S}}s_{ijk}\geq 2. \tag{29}$$

Therefore, when an inequality (17) is generated in our separation routine, we first check to see if can be strengthened into the form (28) or (29) before adding the cut to the model.

4.3 Alternative Formulation

Formulations (3) and (14) contain $O(n^2)$ and $O(n^3)$ binary variables, respectively, which possibly makes them intractable for CETSP-partitionings having a large number of cells. We develop an alternative MIP formulation whose number of binary variables does not depend on n.

Note that any feasible solution to (3) can be characterized by two sets of decisions: (a) the order in which the neighborhood sets S_m are visited, and (b) for each S_m, which cell represents target m on the tour. Given any particular order of visiting the neighborhood sets, the problem of optimally identifying a representative cell for each target can be solved by solving a shortest path problem as follows. For a CETSP-partitioning $C = \{C_0, \ldots, C_n\}$ define set Q as

$$Q = \{(i, m) : C_i \in C,\ m \in M,\ i \in N(m)\} \cup \{\psi, \tau\}, \tag{30}$$

where ψ and τ respectively serve as source and destination nodes in the graph. An ordered pair (i, m) belongs to Q for each cell $C_i \in C$ and every $m \in M$ such that $C_i \cap S_m \neq \emptyset$. For convenience, we also describe the elements of Q by Greek letters, with components $\hat{i}(\bullet)$ and $\hat{m}(\bullet)$, where $\hat{i}(\psi) = \hat{i}(\tau) = 0$ and $\hat{m}(\psi) = \hat{m}(\tau) = p_0$. The distance between any two elements δ and σ in Q is defined as $d_{\delta\sigma} = l_{\hat{i}(\delta),\hat{i}(\sigma)}$. Let u_{lm} be a binary variable that indicates whether target l is visited immediately before (or simultaneously with) target m. We seek a shortest tour that starts from the depot, visits a cell in every neighborhood set (in the order determined by the u-variables), and returns to the depot. This is equivalent to finding a shortest path (with respect to arc costs d) from ψ to τ in a graph $G(\mathbf{u}) = (Q, A(\mathbf{u}))$, where $A(\mathbf{u}) = \{(\delta, \sigma) : \delta, \sigma \in Q, \text{ and } u_{\hat{m}(\delta),\hat{m}(\sigma)} = 1\}$. To model this shortest path, we introduce nonnegative flow variables $f_{\delta\sigma}$ along with the necessary flow-balance constraints. The corresponding formulation is stated below, where $M' = M \cup \{p_0\}$.

$$\text{Min} \quad \sum_{\delta \in Q} \sum_{\sigma \in Q} d_{\delta\sigma} f_{\delta\sigma} \tag{31a}$$

$$\text{s.t.} \quad \sum_{m \in M'} u_{ml} = \sum_{m \in M'} u_{lm} = 1 \quad \forall l \in M' \tag{31b}$$

$$\sum_{l \in \mathcal{S}} \sum_{m \in \mathcal{S}} u_{lm} \leq |\mathcal{S}| - 1, \quad \forall \mathcal{S} \subset M',\ 2 \leq |\mathcal{S}| \leq |M'| - 2 \tag{31c}$$

$$f_{\delta\sigma} \leq u_{\hat{m}(\delta),\hat{m}(\sigma)} \quad \forall \delta \in Q,\ \sigma \in Q \tag{31d}$$

$$\sum_{\sigma \in Q} f_{\sigma\delta} - \sum_{\sigma \in Q} f_{\delta\sigma} = 0, \quad \forall \delta \in Q \setminus \{\psi, \tau\} \tag{31e}$$

$$\sum_{\sigma \in Q} f_{\sigma\tau} = 1, \tag{31f}$$

$$f_{\delta\sigma} \geq 0, \quad \forall \delta \in Q,\ \sigma \in Q \tag{31g}$$

$$u_{lm} \in \{0, 1\}, \quad \forall l \in M',\ m \in M'. \tag{31h}$$

In the above formulation, constraints (31b), (31c), and (31h) impose a TSP tour over set M', while (31d)–(31g) model the corresponding CETSP tour as a shortest path from ψ to τ in $G(\mathbf{u})$. Note that when $|C^1| = |C^2| = 1$, an equivalent form of (12) can be applied to formulation (31) by adding $f_{\delta\sigma} = f_{\sigma\delta} = 0$ to the model, where $\hat{i}(\delta) \in C^1$ and $\hat{i}(\sigma) \in C^2$.

Proposition 7. Suppose that the optimal CETSP tour length is l^*, and consider an optimal solution $(\mathbf{u}^*, \mathbf{f}^*)$ to (31) with objective function value z_{LB3}. Define the set of turn cells as $I' = \{\delta \in Q \setminus \{\psi, \tau\} : \sum_{\sigma \in Q} f^*_{\delta\sigma} = 1\}$. Then,

$$z_{LB3} \leq l^* \leq z_{LB3} + \sum_{\delta \in I'} h'_\delta, \tag{32}$$

where

$$h'_\delta = \begin{cases} 2r_{\hat{i}(\delta)} \sin\left(\frac{\alpha_{\hat{i}(\delta)}}{2}\right) & \text{if } C_{\hat{i}(\delta)} \subseteq S_{\hat{m}(\delta)}, \\ 4r_{\hat{i}(\delta)} \sin\left(\frac{\alpha_{\hat{i}(\delta)}}{2}\right) & \text{if } C_{\hat{i}(\delta)} \nsubseteq S_{\hat{m}(\delta)}, \end{cases} \tag{33}$$

for an arc-based CETSP-partitioning, and

$$h'_\delta = 2\sqrt{{w_{1,\hat{i}(\delta)}}^2 + {w_{2,\hat{i}(\delta)}}^2}, \tag{34}$$

for a grid-based CETSP-partitioning.

Proof. Consider an optimal CETSP tour characterized by an ordered set of turn points $(p_0, \ldots, p_k)$ with a corresponding solution $(\mathbf{x}^R, \mathbf{y}^R)$ as constructed in the proof of Proposition 3. Suppose that this solution corresponds to a tour $C_0 \to \ldots \to C_r \to C_0$. For each $i = 1, \ldots, r$ define $M^i = \{m \in M : m \notin \cup_{h=1}^{i-1} M^h$ and $\exists j \in \{1, \ldots, k\}$ such that $p_j \in S_m \cap C_i\}$. Note that $M^i \neq \emptyset$ for all $i = 1, \ldots, r$ or otherwise, a shorter CETSP tour can be obtained by excluding all turn points in C_i from T. Now suppose $M^i = \{m^i_1, \ldots, m^i_{|M^i|}\}$, $\forall i = 1, \ldots, r$, and consider the following feasible solution to (31).

- $f_{\psi,(1,m^1_1)} = 1$,
- $f_{(i,m^i_j),(i,m^i_{j+1})} = 1$ for all $1 \leq i \leq r$, $j = 1, \ldots, |M^i| - 1$,
- $f_{(i,m^i_{|M^i|}),(i+1,m^{i+1}_1)} = 1$ for $1 \leq i \leq r - 1$,
- $f_{(r,m^r_{|M^r|}),\tau} = 1$,
- $u_{lm} = 1$ for $l, m \in M'$, if and only if $l \neq m$ and there exist δ and σ in Q such that $\hat{m}(\delta) = l$, $\hat{m}(\sigma) = m$, and $f_{\delta\sigma} = 1$.

The objective function value of this solution equals that of $(\mathbf{x}^R, \mathbf{y}^R)$. Therefore, $z_{LB3} \leq z^R \leq l^*$. Validity of the upper bound follows from a similar discussion to that of Proposition 3, noting that in this case every visit to C_i requires a visit to at most one middle point. $\square$

Proposition 7 implies that the optimal objective function value of (31) can be used as a lower bound on the optimal CETSP tour length. Note that regardless of the size n of the underlying CETSP-partitioning, problem (31) contains $O(|M|^2)$ binary variables.

We next describe a Benders decomposition algorithm for solving (31). To that end, suppose $\overline{d}_{lm}$ equals the minimum distance between two neighborhood sets S_l and S_m, for $l, m \in M'$. Formulation (31) can be equivalently stated as

$$\text{Min} \quad \sum_{l\in M'}\sum_{m\in M'} \overline{d}_{lm}u_{lm} + \sum_{\delta\in Q}\sum_{\sigma\in Q}(d_{\delta\sigma} - \overline{d}_{\hat{m}(\delta),\hat{m}(\sigma)})f_{\delta\sigma} \tag{35a}$$

$$\text{s.t.} \quad \text{Constraints (31b)–(31h).} \tag{35b}$$

We reformulate (35) as:

$$\text{Min} \quad \sum_{l\in M'}\sum_{m\in M'} \overline{d}_{lm}u_{lm} + \theta(\mathbf{u}) \tag{36a}$$

$$\text{s.t.} \quad \text{Constraints (31b), (31c), and (31h),} \tag{36b}$$

where

$$\theta(\mathbf{u}) = \text{Min} \quad \sum_{\delta\in Q}\sum_{\sigma\in Q}(d_{\delta\sigma} - \overline{d}_{\hat{m}(\delta),\hat{m}(\sigma)})f_{\delta\sigma} \tag{37a}$$

$$\text{s.t.} \quad \text{Constraints (31d)–(31g).} \tag{37b}$$

In the Benders decomposition framework, we let (36) serve as a *master problem* in which $\theta(\mathbf{u})$ is replaced by a value function variable θ. Then, given a fixed value of $\mathbf{u}$, (37) serves as the *subproblem*. Solving (37) either proves the optimality of $\mathbf{u}$, or yields a Benders (optimality) inequality. Given $\overline{\mathbf{u}}$, define π as a $|Q|$-vector that contains the dual values associated with this shortest path problem, i.e., π_δ denotes the dual value associated with constraint (31e) for $\delta \in Q \setminus \{\psi, \tau\}$, π_τ is the dual value associated with constraint (31f), and $\pi_\psi \equiv 0$. Such dual values indicate the length of a shortest path from ψ to any other node in $G(\overline{\mathbf{u}})$. For any $\delta \in Q$ and $\sigma \in Q$, define $-\gamma_{\delta\sigma}$ as the dual value associated with constraint (31d). Hence, $\gamma_{\delta\sigma} = \max\{0, \pi_\sigma - \big(d_{\delta\sigma} - \overline{d}_{\hat{m}(\delta),\hat{m}(\sigma)}\big) - \pi_\delta\}$, and let

$$\eta_{lm} = \sum_{\delta\in Q:\hat{m}(\delta)=l}\ \sum_{\sigma\in Q:\hat{m}(\sigma)=m} \gamma_{\delta\sigma}. \tag{38}$$

Note that η_{lm} is nonnegative and can take positive values only for $(l, m) \in U^0 = \{(l, m) : l \in M', m \in M', \text{ and } \overline{u}_{lm} = 0\}$. Hence, the corresponding Benders inequality can be written as

$$\theta + \sum_{(l,m)\in U^0} \overline{\eta}_{lm}u_{lm} \geq \pi_\tau, \tag{39}$$

where $\overline{\eta}_{lm} = \min\{\eta_{lm}, \pi_\tau\}$. These reduced coefficients are valid due to the nonnegativity of η_{lm}, $\forall(l,m) \in U^0$, and the fact that $\theta \geq 0$. This concept can be used to further tighten (39). Noting that (39) is valid at $\mathbf{u} = \overline{\mathbf{u}}$, regardless of the coefficients of the u-variables in U^0, we need to ensure that (39) is valid for $\mathbf{u}' \neq \overline{\mathbf{u}}$. Note that $\mathbf{u}'$ contains at least two variables in U^0 that equal one. Let $\rho = \min_{(l,m)\in U^0}\{\overline{\eta}_{lm}\}$. Suppose U^{01} denotes a subset of U^0 whose elements satisfy $\overline{\eta}_{lm} \geq \pi(\tau) - \rho$, and define $U^{02} = U^0 \setminus U^{01}$. If $\rho \leq \frac{\pi_\tau}{2}$, then (39) can be strengthened as

$$\theta + \sum_{(l,m)\in U^{01}} (\pi_\tau - \rho)u_{lm} + \sum_{(l,m)\in U^{02}} \eta_{lm}u_{lm} \geq \pi_\tau; \tag{40}$$

otherwise, the following inequality is valid for (36).

$$\theta + \sum_{(l,m)\in U^0} \frac{\pi_\tau}{2} u_{lm} \geq \pi_\tau. \tag{41}$$

5. Computational Experiments

To examine the efficiency of the proposed formulations and algorithms, we generate random CETSP instances on which we obtain upper and lower bounds for the optimal CETSP tour length. All algorithms are implemented in C++ programming language and compiled using Microsoft Visual Studio 2008. Integer programming instances are solved using CPLEX version 12.2 via ILOG Concert Technology 2.9 on a PC with an Intel Core2 Quad processor Q9500 and 4 GB of memory, running Windows 7.

All CETSP instances are generated according to the following procedure. The $|M|$ target locations and the depot are chosen randomly on a rectangle of length 16 and width 10. The neighborhood set for all targets are discs of identical radius r (which we vary in our computational tests). For each disc, we specify the intersection of its boundary and the convex hull of the target locations and the depot, and divide this intersection into N' smaller arcs, which are initial elements of the partitioning set C. Therefore, the initial CETSP-partitioning contains $|M|N' + 1$ cells.

We first provide a comparison between the performance of different methods of obtaining a lower bound on the optimal CETSP tour length. For brevity, we will refer to formulations (3), (14) (with subtour elimination constraints (17)), and (35) as LB1, LB2, and LB3, respectively. The subtour elimination constraints for all three models are initially relaxed. At each node of the branch-and-bound tree, we use the separation procedure explained in

Section 3 to find violated subtour inequalities. If the violation magnitude is greater than or equal to 0.5 for a subtour elimination constraint, we add that inequality to the linear programming relaxation of the corresponding node.

The results of running all three models over a set of ten small test instances are reported in Tables 1 and 2. These instances contain six target locations with the neighborhood set of each location defined as a disc of radius $r = 0.25$, and each arc is partitioned into $N' = 4$ smaller ones. A 1500-second limit is imposed on the running time of all methods. The **Lower Bound** columns report the lower bound that is obtained via each method, while the **Upper Bound** columns report the upper bound provided by (4) (for LB1 and LB2) and (32) (for LB3). We also report the average absolute gap (difference) between the lower and upper bounds of all methods in the **Avg. Gap** row. We observe that while LB2 always provides the best lower bound, it generally requires a significantly longer time to solve than the other two methods. Note that LB1 consumes roughly 30 times the CPU time required to solve LB3, and it provides a lower bound that is dominated by that of LB3. The fast running time of LB3 is evidently due at least in part to the low number of subtour cuts that are required to solve the problem.

Table 3 compares the running time of formulations LB1 and LB3 for ten 8- and 10-node instances for which LB2 is too large to be practically useful. Overall, these results demonstrate that LB3 is significantly easier to solve than LB1, and once again, there appears to be direct correlation between the number of subtour cuts generated (many fewer for LB3 than for LB1). The lower bounds obtained by these algorithms are not displayed in the table, because all lower bounds were the same except for three instances for $r = 0.25$ and six for $r = 0.5$. For eight out of these nine instances we found that LB3 provides a slightly better lower bound than LB1. In one instance, the lower bound obtained from LB1 was better than that of LB3 with a difference of 0.052.

Table 4 demonstrates the efficiency of the Benders decomposition algorithm for LB3 in solving 80 random instances. For each value of $|M|$, we solve two instances associated with setting $r = 0.25$ and $r = 0.5$. The number of arcs per neighborhood set is set to be $N' = 4$ for each instance. We present the running time (in CPU seconds) required by the Benders decomposition algorithm, and for solving formulation (35) directly using CPLEX. We report the number of added subtour elimination constraints for both methods. For the Benders decomposition algorithm, we also report the number of Benders inequalities needed to solve the master problem to optimality. (Note that the average CPU times factor in 1500

Table 1: Comparing Lower Bound Formulations on Six-Node Instances

	LB1		LB2		LB3	
Instance	Subtour Cuts	Time (s)	Subtour Cuts	Time (s)	Subtour Cuts	Time (s)
CETSP-6-01	261	4.6	145	263.3	27	0.8
CETSP-6-02	357	22.7	205	442.3	21	0.8
CETSP-6-03	448	26.3	195	378.4	14	1.6
CETSP-6-04	127	15.4	0	21.6	14	0.5
CETSP-6-05	156	17.5	0	22.6	12	1.2
CETSP-6-06	173	20.2	0	22.5	7	0.8
CETSP-6-07	233	23.7	216	364.7	10	0.7
CETSP-6-08	269	21.2	226	429.0	23	1.5
CETSP-6-09	115	12.5	0	25.9	19	0.8
CETSP-6-10	496	28.8	372	767.2	28	0.9
Averages	**263.5**	**19.2**	**135.9**	**273.7**	**17.5**	**0.9**

Table 2: Lower and Upper Bounds Obtained on Six-Node Instances

	LB1		LB2		LB3	
Instance	Lower Bound	Upper Bound	Lower Bound	Upper Bound	Lower Bound	Upper Bound
CETSP-6-01	33.1604	34.2768	33.4994	34.6158	33.1604	34.2768
CETSP-6-02	27.0611	27.8568	27.3521	28.1474	27.0615	27.8568
CETSP-6-03	24.7048	25.6439	25.0509	25.99	24.7048	25.6439
CETSP-6-04	36.1009	37.0838	36.4295	37.4124	36.1009	37.0838
CETSP-6-05	21.3865	22.5598	21.5978	22.771	21.3865	22.9133
CETSP-6-06	27.1195	28.1387	27.5114	28.5306	27.1195	28.1387
CETSP-6-07	33.9453	35.0102	34.3707	35.4329	33.948	35.0102
CETSP-6-08	23.4709	24.514	23.8133	24.8564	23.4709	24.514
CETSP-6-09	27.8231	29.0243	28.137	29.3381	27.8231	29.0243
CETSP-6-10	33.6742	34.9533	34.1952	35.4743	33.6742	34.9533
Avg. Gap	**1.0614**		**1.0611**		**1.0965**	

seconds for the case in which a problem instance is not solved within the time limit.) The results demonstrate that using Benders decomposition dramatically improves the model's solvability. Using (33), we can also obtain upper bounds on the optimal CETSP tour length for all instances. The average difference between the upper and lower bounds for different instance sizes are specified below, where the first, second, and third components denote $|M|$, r and the average absolute gap, respectively: $(14, 0.25, 3.80)$, $(14, 0.5, 7.78)$, $(16, 0.25, 4.38)$, $(16, 0.5, 9.20)$, $(18, 0.25, 5.20)$, $(18, 0.5, 11.04)$, $(20, 0.25, 5.79)$, and $(20, 0.5, 12.68)$. Note that a few instances were not solved within the time limit by either algorithm. The lower bound that we report for these instances is given by the lower bound on z_{LB3} after 1500 seconds. The upper bound for these instances employs Proposition 7, with z_{LB3} given by the objective corresponding to the best feasible solution found for LB3 after 1500 seconds.

Figure 7 illustrates the convergence of lower and upper bounds for an instance of size $|M| = 12$ and $r = 0.25$. In each iteration, we obtain lower and upper bounds on the optimal

Table 3: Comparing Formulations LB1 and LB3

		$r = 0.25$				$r = 0.5$			
		LB1		LB3		LB1		LB3	
Instance	$\lvert M \rvert$	Subtour Cuts	Time (s)	Subtour Cuts	Time (s)	Subtour Cuts	Time (s)	Subtour Cuts	Time (s)
CETSP-8-01	8	603	25.9	35	1.2	346	14.7	38	1.9
CETSP-8-02	8	700	37.1	49	2.5	760	31.9	83	6.5
CETSP-8-03	8	210	8.1	15	0.5	314	11.4	15	0.7
CETSP-8-04	8	619	19.4	46	2.7	189	3.5	34	3.0
CETSP-8-05	8	1596	64.4	73	2.5	663	25.0	67	4.6
CETSP-8-06	8	232	9.0	33	1.0	341	12.9	65	3.7
CETSP-8-07	8	249	11.7	54	1.0	196	5.6	61	1.2
CETSP-8-08	8	387	22.3	52	1.2	917	35.2	60	1.7
CETSP-8-09	8	254	12.7	42	0.8	137	6.1	44	1.2
CETSP-8-10	8	379	17.3	36	2.1	374	21.3	62	8.3
Averages		**522.9**	**22.8**	**43.5**	**1.5**	**423.7**	**16.8**	**52.9**	**3.3**
CETSP-10-01	10	1231	92.2	30	1.6	1906	192.3	41	3.7
CETSP-10-02	10	972	97.1	93	5.7	1698	215.3	122	12.5
CETSP-10-03	10	938	52.5	68	5.6	419	45.0	79	12.7
CETSP-10-04	10	1812	231.5	50	4.1	1008	78.2	51	4.8
CETSP-10-05	10	890	73.7	89	1.9	638	53.3	166	19.6
CETSP-10-06	10	687	56.7	59	1.6	1887	227.0	68	13.2
CETSP-10-07	10	415	24.5	38	2.5	564	63.3	38	6.3
CETSP-10-08	10	2598	196.3	50	7.2	530	53.5	104	18.4
CETSP-10-09	10	279	15.4	34	1.4	385	24.6	77	4.7
CETSP-10-10	10	509	52.3	24	1.2	309	23.7	90	18.2
Averages		**1033.1**	**89.2**	**53.5**	**3.3**	**934.4**	**97.6**	**83.6**	**11.4**

CETSP tour length using the Benders decomposition algorithm that was proposed in Section 4.3. Figure 7 also shows the solution time of each iteration for the upper and lower bound problems.

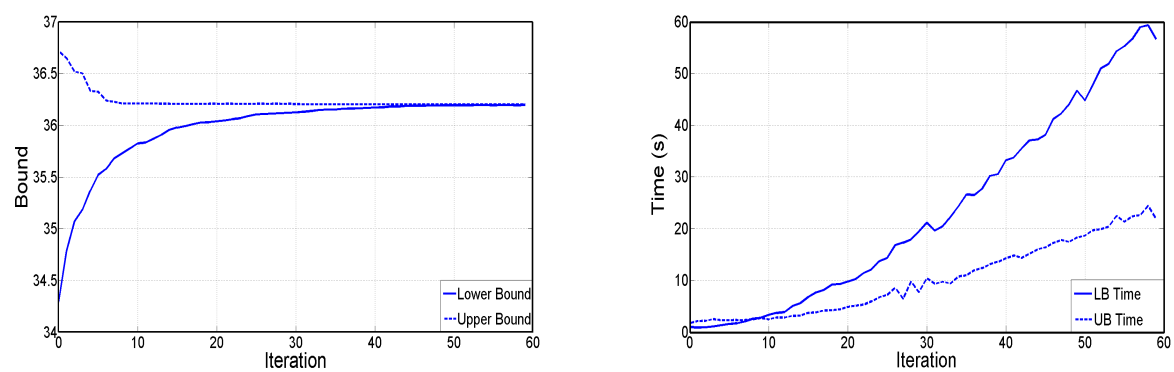

Figure 7: Results for the iterative solution method

6. Conclusions and Future Research

In this paper, we presented several methods of obtaining lower and upper bounds on the optimal objective function value of an important geometric variant of the TSP, called the CETSP. The ability to obtain tight lower bounds on the optimal CETSP tour length is

Table 4: Performance of Benders Decomposition

		$r = 0.25$						$r = 0.5$						
		Monolithic			Benders			Monolithic			Benders			
Instance	$\lvert M \rvert$	Time (s)	Gap	Subtour Cuts	Time (s)	Benders Cuts	Subtour Cuts	Time (s)	Gap	Subtour Cuts	Time (s)	Gap	Benders Cuts	Subtour Cuts
CETSP-14-01	14	49.7	0.0%	148	2.5	88	150	353.7	0.0%	216	5.3	0.0%	257	202
CETSP-14-02	14	10.7	0.0%	80	1.5	21	92	24.1	0.0%	160	2.2	0.0%	49	88
CETSP-14-03	14	10.5	0.0%	209	6.4	48	140	76.4	0.0%	199	11.7	0.0%	223	213
CETSP-14-04	14	10.0	0.0%	139	2.6	28	154	103.9	0.0%	360	4.5	0.0%	137	243
CETSP-14-05	14	5.6	0.0%	130	2.1	20	98	70.3	0.0%	211	3.3	0.0%	69	174
CETSP-14-06	14	17.2	0.0%	115	3.7	7	141	77.4	0.0%	183	2.5	0.0%	52	213
CETSP-14-07	14	5.3	0.0%	66	2.4	8	65	71.0	0.0%	251	2.8	0.0%	37	106
CETSP-14-08	14	21.3	0.0%	110	6.3	26	167	71.0	0.0%	277	7.4	0.0%	242	196
CETSP-14-09	14	11.8	0.0%	94	2.9	14	97	21.1	0.0%	106	1.9	0.0%	42	111
CETSP-14-10	14	7.9	0.0%	168	3.4	21	148	27.1	0.0%	177	6.0	0.0%	154	174
Averages		**15.0**	**0.0%**	**125.9**	**3.4**	**28.1**	**125.2**	**89.6**	**0.0%**	**214.0**	**4.8**	**0.0%**	**126.2**	**172.0**
CETSP-16-01	16	89.3	0.0%	355	5.0	166	231	709.4	0.0%	323	8.3	0.0%	294	239
CETSP-16-02	16	22.7	0.0%	228	1.8	18	193	1060.8	0.0%	652	8.7	0.0%	238	302
CETSP-16-03	16	11.7	0.0%	153	6.7	12	175	111.2	0.0%	198	9.7	0.0%	168	244
CETSP-16-04	16	23.5	0.0%	258	3.0	25	289	250.3	0.0%	460	13.8	0.0%	384	460
CETSP-16-05	16	61.0	0.0%	311	3.2	47	209	577.4	0.0%	554	8.4	0.0%	202	255
CETSP-16-06	16	17.1	0.0%	173	4.0	47	251	802.6	0.0%	480	14.2	0.0%	337	334
CETSP-16-07	16	40.2	0.0%	272	7.5	30	172	213.3	0.0%	297	15.6	0.0%	450	248
CETSP-16-08	16	7.8	0.0%	149	2.0	8	145	29.3	0.0%	176	7.2	0.0%	50	196
CETSP-16-09	16	218.0	0.0%	262	11.6	203	246	1500.9	73.7%	1046	520.1	0.0%	5365	521
CETSP-16-10	16	52.0	0.0%	162	5.2	71	158	331.4	0.0%	357	15.8	0.0%	339	199
Averages		**54.3**	**0.0%**	**232.3**	**5.0**	**62.7**	**206.9**	**558.7**	**7.4%**	**454.3**	**62.2**	**0.0%**	**782.7**	**299.8**
CETSP-18-01	18	296.0	0.0%	556	8.1	145	458	1500.8	15.2%	745	35.6	0.0%	790	545
CETSP-18-02	18	269.2	0.0%	353	12.8	321	265	1500.8	4.2%	467	202.0	0.0%	2597	413
CETSP-18-03	18	20.7	0.0%	182	9.1	66	230	1500.8	17.4%	622	102.9	0.0%	1536	550
CETSP-18-04	18	49.9	0.0%	201	2.7	15	127	382.5	0.0%	291	11.0	0.0%	132	146
CETSP-18-05	18	121.7	0.0%	205	4.7	65	165	862.7	0.0%	427	28.0	0.0%	483	230
CETSP-18-06	18	1500.9	10.0%	596	13.4	259	444	1501.5	80.1%	884	173.9	0.0%	2142	718
CETSP-18-07	18	55.7	0.0%	301	4.2	39	292	140.4	0.0%	283	11.4	0.0%	108	211
CETSP-18-08	18	75.7	0.0%	293	22.0	533	295	1501.0	96.1%	912	1501.0	6.7%	7068	693
CETSP-18-09	18	85.0	0.0%	252	5.7	61	198	442.1	0.0%	294	38.4	0.0%	680	274
CETSP-18-10	18	31.8	0.0%	272	4.7	68	306	176.5	0.0%	392	36.2	0.0%	506	307
Averages		**250.7**	**1.0%**	**321.1**	**8.8**	**157.2**	**278.0**	**950.9**	**21.3%**	**531.7**	**214.0**	**0.7%**	**1604.2**	**408.7**
CETSP-20-01	20	625.0	0.0%	286	11.7	221	230	1500.9	7.4%	367	556.4	0.0%	4397	656
CETSP-20-02	20	36.8	0.0%	209	11.8	175	320	1500.9	4.6%	493	135.4	0.0%	1211	609
CETSP-20-03	20	18.4	0.0%	339	8.3	98	772	1501.0	17.7%	948	41.5	0.0%	586	814
CETSP-20-04	20	36.9	0.0%	254	9.2	122	252	1501.4	13.6%	528	84.6	0.0%	992	311
CETSP-20-05	20	709.7	0.0%	309	8.2	95	309	1501.7	81.4%	842	716.9	0.0%	4628	987
CETSP-20-06	20	366.6	0.0%	464	15.0	120	306	1501.2	3.1%	567	177.3	0.0%	1734	707
CETSP-20-07	20	73.0	0.0%	238	15.8	247	251	1501.2	76.9%	1071	1501.3	3.3%	6872	741
CETSP-20-08	20	1501.4	2.5%	417	39.4	632	481	1500.8	10.0%	719	1501.0	4.3%	6100	1233
CETSP-20-09	20	91.0	0.0%	322	17.4	169	441	476.5	0.0%	515	57.0	0.0%	746	387
CETSP-20-10	20	20.4	0.0%	299	10.1	33	292	359.4	0.0%	623	21.4	0.0%	289	346
Averages		**347.9**	**0.2%**	**313.7**	**14.7**	**191.2**	**365.4**	**1284.5**	**21.5%**	**667.3**	**479.3**	**0.8%**	**2755.5**	**679.1**

vital in evaluating the quality of any heuristic solution to the problem. We proved several properties of an optimal CETSP tour, and used them to establish a way of partitioning the continuous solution space. This partitioning scheme is then used in formulating three different integer programming problems that yield lower and upper bounds on the optimal CETSP tour length. In particular, we formulated a lower bounding problem that is a special instance of the CTP, and described an alternative formulation that yields a tighter lower bound. We also described a way of reformulating the underlying integer program that makes it amenable to Benders decomposition. The subproblem in the decomposition scheme finds a shortest path in a special directed graph with an expanded node set. We observed that this reformulation yields a lower bound that is not dominated by that of the original formulation, while greatly enhancing the running time via Benders decomposition. An iterative refining of the underlying partitioning scheme ensures that the lower bounds obtained from the three methods converge to the optimal CETSP tour length.

The framework presented in this paper can be extended to many other problems with continuous solution space. Candidates include the lawn mowing problem (Arkin et al., 2000) and the the polygon exploration problem (Hoffmann et al., 2002). In some applications, a turning cost may be present based on the angle that is formed at each turn point of the tour (see, e.g., Arkin et al. 2006). Formulation (14) may, in particular, be extended to address practical instances of geometric tour problems with turn costs. Another interesting line of future research would perform a comprehensive polyhedral study on the proposed integer programming problems formulated in this paper.

Acknowledgments

Dr. Smith gratefully acknowledges the support of the Defense Threat Reduction Agency under Grant HDTRA1-10-1-0050 and the National Science Foundation under Grant CMMI-1100765.

References

Applegate, D. L., R. E. Bixby, V. Chvatal, W. J. Cook. 2006. *The Traveling Salesman Problem: A Computational Study*. Princeton University Press, Princeton, NJ.

Arkin, E. M., M. A. Bender, E. D. Demaine, S. P. Fekete, J. S. B. Mitchell, S. Sethia. 2006. Optimal covering tours with turn costs. *SIAM Journal on Computing* **35** 531–566.

Arkin, E. M., S. P. Fekete, J. S. B. Mitchell. 2000. Approximation algorithms for lawn mowing and milling. *Computational Geometry* **17** 25–50.

Arkin, E. M., R. Hassin. 1994. Approximation algorithms for the geometric covering salesman problem. *Discrete Applied Mathematics* **55** 197–218.

Balas, E. 1989. The prize collecting traveling salesman problem. *Networks* **19** 621–636.

Balas, E. 2004. The prize collecting traveling salesman problem and its applications. D. Du, P. M. Pardalos, G. Gutin, A. Punnen, eds., *The Traveling Salesman Problem and its Variations*, vol. 12. Springer US, 663–695.

Ciullo, D., G. D. Celik, E. Modiano. 2010. Minimizing transmission energy in sensor networks via trajectory control. *Proceedings of the 8th International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt) 2010*. Avignon, France, 132–141.

Concorde. 2011. http://www.tsp.gatech.edu/concorde.html.

Current, J. R., D. A. Schilling. 1989. The covering salesman problem. *Transportation Science* **23** 208–213.

Dumitrescu, A., J. S. B. Mitchell. 2003. Approximation algorithms for TSP with neighborhoods in the plane. *Journal of Algorithms* **48** 135–159.

Fischetti, M., J. S. Gonzalez, P. Toth. 1997. A branch-and-cut algorithm for the symmetric generalized traveling salesman problem. *Operations Research* **45** 378–394.

Garey, M. R., D. S. Johnson. 1979. *Computers and Intractability: A Guide to the Theory of NP-completeness*. W. H. Freeman & Co., New York.

Gendreau, M., G. Laporte, F. Semet. 1997. The covering tour problem. *Operations Research* **45** 568–576.

Gendreau, M., G. Laporte, F. Semet. 1998. A branch-and-cut algorithm for the undirected selective traveling salesman problem. *Networks* **32** 263–273.

Gulczynski, D. J., J. W. Heath, C. C. Price. 2006. The close enough traveling salesman problem: A discussion of several heuristics. *Perspectives in Operations Research* **36** 271–283.

Hoffmann, F., C. Icking, R. Klein, K. Kriegel. 2002. The polygon exploration problem. *SIAM Journal on Computing* **31** 577–600.

Jozefowiez, N., F. Semet, E. Talbi. 2007. The bi-objective covering tour problem. *Computers & Operations Research* **34** 1929–1942.

Mata, C. S., J. S. B. Mitchell. 1995. Approximation algorithms for geometric tour and network design problems. *SCG '95: Proceedings of the Eleventh Annual Symposium on Computational Geometry*. New York, 360–369.

Mennell, W. K. 2009. Heuristics for solving three routing problems: Close-enough traveling salesman problem, close-enough vehicle routing problem, sequence-dependent team orienteering problem. Ph.D. thesis, The Robert H. Smith School of Business, University of Maryland, College Park, MD.

Padberg, M., G. Rinaldi. 1991. A branch-and-cut algorithm for the resolution of large-scale symmetric traveling salesman problems. *SIAM Review* **33** 60–100.

Shuttleworth, R., B. L. Golden, S. Smith, E. Wasil. 2008. Advances in meter reading: Heuristic solution of the close enough traveling salesman problem over a street network. B. L. Golden, S. Raghavan, E. A. Wasil, eds., *The Vehicle Routing Problem: Latest Advances and New Challenges*, vol. 43. Springer, 487–501.

Valle, C. A., A. S. da Cunha, G. R. Mateus, L. C. Martinez. 2009. Exact algorithms for a selective vehicle routing problem where the longest route is minimized. *Electronic Notes in Discrete Mathematics* **35** 133–138.

Yuan, B., M. Orlowska, S. Sadiq. 2007. On the optimal robot routing problem in wireless sensor networks. *IEEE Transactions on Knowledge and Data Engineering* **19** 1252–1261.

Appendix: Material to Appear in Online Supplement

A-1. Proof of Proposition 2

First, define $D = conv(M \cup \{p_0\})$. The dimension of D must be at least one (or else $D = \{p_0\}$ and thus $|M| = 0$). If the dimension of D is one, the proposition follows by noting that an optimal tour traverses the length of D from p_0 toward each of its two endpoints, stopping once all neighborhood sets are visited near each end of D and then back to p_0. Now consider the case in which D is two-dimensional, and by contradiction, suppose that T is an optimal tour that contains points outside D. Let q' be a point on T that intersects the boundary of D, such that the trajectory of T immediately leaves D (i.e., $\forall \varepsilon$ such that $0 < \varepsilon \leq \overline{\varepsilon}$, point $q' + \varepsilon d$ is on T but is not in D, for some trajectory vector d and a positive $\overline{\varepsilon}$). Because q' is on the boundary of D, it must be a convex combination of two points c_1 and c_2 in $M \cup \{p_0\}$. Consider the (infinite) line intersecting c_1 and c_2 and define H as the halfspace with respect to this line that contains D; also, define $\overline{H}$ as the halfspace consisting of this same line and points on the opposite side of D (so that $H \cap \overline{H}$ is just the line intersecting c_1 and c_2). Following the trajectory of T after it leaves D at q', suppose that T re-enters H at q''. This point must exist because the tour must return to $p_0 \in H$.

Now consider an alternative tour $\hat{T}$, which is the same as T except that the segment of T between q' and q'' (all of which lies outside of H) is replaced with a segment that lies completely on the line spanning c_1 and c_2. Let $\overline{M} = \{m \in M : T$ visits the neighborhood set of m on the segment between q' and q'' that lies in $\overline{H}\}$. If $\overline{M} = \emptyset$, then replacing the segment of T with the straight segment between q' and q'' retains the feasibility of the tour while reducing its length; hence, T could not be optimal (Figure 8). Otherwise, suppose that $\overline{M} \neq \emptyset$. Because the line segment intersecting c_1 and c_2 induces a facet of D, all targets in $\overline{M}$ can be visited by traversing this segment (or else, some $m \in \overline{M}$ would not belong to D). Suppose that $[\lambda_{\min}, \lambda_{\max}]$ is the smallest interval such that the set of all points $q' + \lambda(q'' - q')$ for $\lambda \in [\lambda_{\min}, \lambda_{\max}]$ intersects neighborhood set for each $m \in \overline{M}$. Define $\overline{\lambda}_{\min} = \min\{0, \lambda_{\min}\}$ and $\overline{\lambda}_{\max} = \max\{1, \lambda_{\max}\}$. The unique shortest tour segment τ that is completely contained in $\overline{H}$, starts at q', intersects the neighborhood set of each $m \in \overline{M}$, and ends at q'' is described as follows. Segment τ starts at the point on the line segment where $\lambda = 0$, moves to the point where $\lambda = \overline{\lambda}_{\min}$, then to the point where $\lambda = \overline{\lambda}_{\max}$, and returns to the point where $\lambda = 1$ (Figure 9). Now, suppose that we create a new tour $\hat{T}$, which is the same as T except

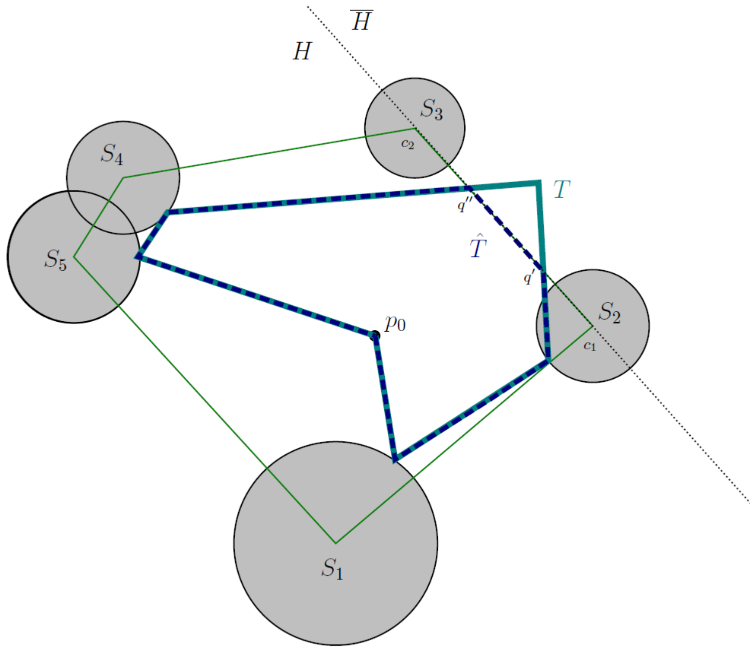

Figure 8: Illustration of the convex hull of target points

where τ replaces the segment from q' to q'' in T. Because the segment in T joining q' and q'' is contained in $\overline{H}$ and is not identical to τ, its distance is longer than that of τ. Thus, tour $\hat{T}$ is still feasible but is shorter than T, which contradicts the optimality of T. This completes the proof. $\square$

A-2. Calculation of Distance Values

We describe how to calculate the cost coefficients l_{ij} and e_{ijk} for the cells of a given CETSP-partitioning. Two cells C_i and C_j are said to be *regularly placed* with respect to each other if both of the following conditions hold:

1. $a_i + w_{1i} \leq a_j$ or $a_i \geq a_j + w_{1j}$, and

2. $b_i + w_{2i} \leq b_j$ or $b_i \geq b_j + w_{2j}$.

If C_i and C_j are not regularly placed with respect to each other and $a_j - w_{1i} < a_i < a_j + w_{1j}$, then $l_{ij} = \max\{b_i - b_j - w_{2j}, b_j - b_i - w_{2i}\}$. If $b_j - w_{2i} < b_i < b_j + w_{2j}$, then $l_{ij} = \max\{a_i - a_j - w_{1j}, a_j - a_i - w_{1i}\}$. On the other hand, when C_i and C_j are regularly placed with respect to each other, it can be easily shown that there exists a vertex v of

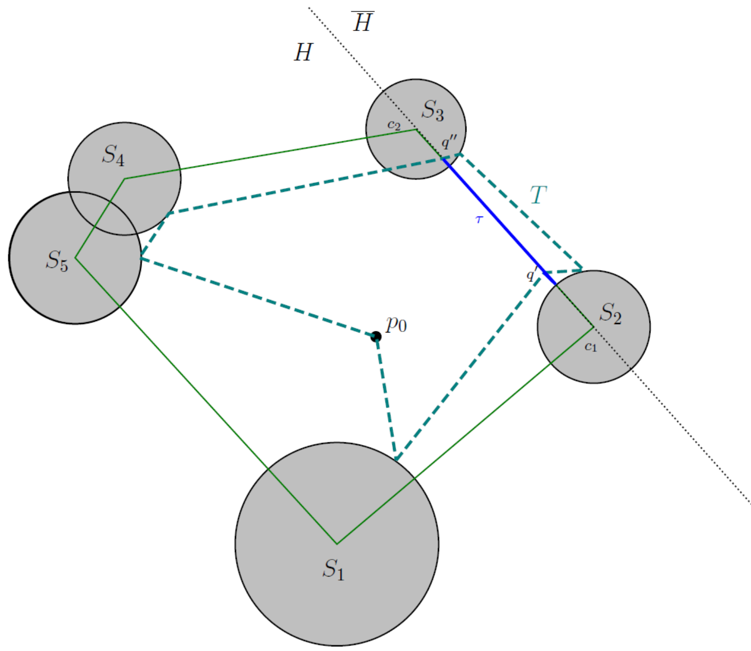

Figure 9: Illustration of the convex hull argument in proof of Proposition 2

C_i and a vertex v' of C_j such that the line segment (vv') is the shortest line segment that connects C_i and C_j and so $l_{ij} = |vv'|$. As a result, calculating each l_{ij}-value can be done in $O(1)$ time.

Now let (v_1v_2) be the shortest line segment connecting two regularly placed cells C_i and C_j and (v_3v_4) be the shortest line segment connecting cells C_j and C_k, which are also regularly placed with respect to each other. Hence, $l_{ij} = |v_1v_2|$ and $l_{jk} = |v_3v_4|$. To evaluate e_{ijk}, we consider the following three cases.

- *Case 1*: v_2 and v_3 are the same vertex of C_j. In this case, $e_{ijk} = l_{ij} + l_{jk}$.

- *Case 2*: v_2 and v_3 are on the same edge of C_j. Let $v_i = (\omega_{1i}, \omega_{2i})$ for $i = 1, \ldots, 4$. Since v_2 and v_3 are on the same side of cell C_j, either $\omega_{12} = \omega_{13}$ or $\omega_{22} = \omega_{23}$. Without loss of generality, suppose $\omega_{22} = \omega_{23} = \overline{\omega}$. The minimum distance from any point in C_i to any point in C_j and then to any point in C_k is obtained by going from v_1 to some point $(\mu, \overline{\omega})$ on the line segment between v_2 and v_3 and from there to v_4. Therefore, the corresponding optimization problem is as follows.

$$\min_{\omega_{12} \leq \mu \leq \omega_{13}} \left((\mu - \omega_{11})^2 + (\overline{\omega} - \omega_{21})^2\right)^{\frac{1}{2}} + \left((\mu - \omega_{14})^2 + (\overline{\omega} - \omega_{24})^2\right)^{\frac{1}{2}} \tag{42}$$

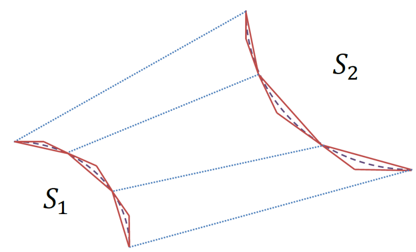

Figure 10: Obtaining a lower bound on the distance between two arcs

Since the objective function is strictly convex, the optimal value of μ is either ω_{12}, ω_{13}, or where μ is the unconstrained minimizer of (42), such that

$$\frac{\mu - \omega_{11}}{\omega_{14} - \mu} = \left|\frac{\overline{\omega} - \omega_{21}}{\overline{\omega} - \omega_{24}}\right|, \text{ assuming } \overline{\omega} \neq \omega_{24}.$$

- *Case 3*: v_2 and v_3 are diagonally opposite each other in C_j. In this case, if the line segment (v_1, v_4) passes through C_j, then $e_{ijk} = |v_1 v_4|$. Otherwise, $e = |v_1 v'| + |v' v_4|$, where v' is one of two vertices of C_j other than v_2 and v_3 (whichever is a minimizer of distance), because this path dominates all other paths that do not pass through a vertex of C_j. Therefore, calculating e_{ijk}-coefficients can be done in $O(1)$ time.

Unlike the grid-based CETSP-partitioning, computing objective function coefficients for an arc-based CETSP-partitioning does not appear to be easy. This is because the corresponding distance minimization problem in general is not convex. One way to deal with this difficulty is to numerically calculate all the minimum distances between any two arcs prior to solving the mathematical programming problems. We can also use an easily-computable valid lower bound on the minimum distance. To obtain this lower bound, one can calculate the minimum distance between two minimal triangles (outer-approximations) that contain the two arcs. This is equivalent to solving a convex optimization problem. The corresponding triangles can be obtained using the tangent lines at the two endpoints, plus the arc's chord. Moreover, we can divide each of the two arcs into several smaller portions to obtain a collection of smaller triangles, as illustrated in Figure 10. The minimum distance between any pair of triangles representing different arcs provides a valid lower bound on the minimum distance between the two arcs. This lower bound can become arbitrarily tight by increasing the number of triangles.

A-3. NP-completeness Proofs for SP1 and SP2

First, we prove that SP1 is strongly NP-complete by a reduction from the exact cover by 3-sets problem (X3C), defined as follows (Garey and Johnson, 1979).

Definition 2. Problem X3C: Let $G = \{g_1, \ldots, g_{3q}\}$ and consider a collection $F = \{F_1, \ldots, F_p\}$ of 3-element subsets of G. Is there a subset $F' \subseteq F$ such that every element of G occurs in exactly one member of F'?

SP1 clearly belongs to NP, because an input guess can be verified to satisfy the SP1 conditions in $O(n)$ time. To show that SP1 is NP-complete, we reduce any arbitrary instance (G, F) of X3C into an instance of SP1 as follows. Let $M = \{m_1, \ldots, m_{3q+p}\}$ and $C = \{C_0, \ldots, C_{p+1}\}$. For each C_i, $i = 1, \ldots, p$, define $V_i = \{m_j : g_j \in F_i\} \cup \{m_{3q+i}\}$ and let $V_{p+1} = \{m_1, \ldots, m_{3q}\}$. Also, let $\overline{y}_i = \frac{q+\epsilon}{q+1}$ for $i = 1, \ldots, p+1$, where $0 < \epsilon < \frac{1}{q+2}$. We assert that there exists an exact cover by 3-sets of G if and only if there exist subsets C^1 and C^2 of C that solve the above instance of SP1.

$\Longrightarrow$ Suppose there exists a subset $F' \subseteq F$ such that $|F'| = q$ and every element of G occurs in exactly one member of F'. Let $C^1 = \{C_{p+1}\}$ and $C^2 = \{C_i \in C : F_i \in F'\}$. Then,

$$\sum_{i \in C^1 \cup C^2} \overline{y}_i - |C^1| - |C^2| = (q+1)\left(\frac{q+\epsilon}{q+1}\right) - q - 1 \tag{43}$$

$$= \epsilon - 1 > -1. \tag{44}$$

$\Longleftarrow$ Suppose that there exist disjoint subsets C^1 and C^2 that solve SP1. Then we have that $\sum_{i \in C^1 \cup C^2} \overline{y}_i > |C^1| + |C^2| - 1$, i.e. $\sum_{i \in C^1 \cup C^2} \frac{1-\epsilon}{q+1} < 1$. Note that C^1 cannot contain any of $C_1, \ldots, C_p$ or otherwise $V(C^1) \nsubseteq V(C^2)$. Therefore, $C^1 = \{C_{p+1}\}$ and C^2 must cover all elements in $\{m_1, \ldots, m_{3q}\}$, which requires $|C^2|$ to be at least q. Hence we have that $(|C^2| + 1)\frac{1-\epsilon}{q+1} < 1$, or equivalently, $|C^2| < \frac{q+\epsilon}{1-\epsilon}$. Because $\epsilon < \frac{1}{q+2}$, we conclude that $|C^2| \leq q$. Consequently, $|C^2| = q$ and elements of C^2 correspond to an exact cover by 3-sets of G. This completes the proof that SP1 is NP-complete.

Next, SP2 clearly belongs to NP. To show that SP2 is NP-complete, we establish a reduction from X3C to SP2 as follows. Define $M = \{m_1, \ldots, m_{3q+p+1}\}$ and $C = \{C_0, \ldots, C_{p+3}\}$.

For each C_i, $i = 1, \dots, p$, define $V_i = \{m_j : g_j \in F_i\} \cup \{m_{3q+i}\}$ and let

$$V_{p+1} = \{m_1, \dots, m_{3q+p}\}, \tag{45}$$

$$V_{p+2} = \{m_{3q+1}, \dots, m_{3q+p+1}\}, \tag{46}$$

$$V_{p+3} = \{m_{3q+p+1}\}. \tag{47}$$

Also, let $\overline{y}_i = \frac{q+1+\epsilon}{q+3}$ for $i = 1, \dots, p+3$, where $0 < \epsilon < \frac{2}{q+4}$. There exists an exact cover by 3-sets of G if and only if there exist two non-empty disjoint subsets C^1 and C^2 of C that solve the above instance of SP2, where $C^1 = \{C_{p+1}, C_{p+3}\}$ and C^2 consists of C_{p+2} and the elements of C corresponding to an element F_i in an X3C solution for G. The remaining details follow from the proof for SP1.