# Chapter 10
# Coverage of Network-Based Structures: Paths, Tours and Trees

## 10.1 Introduction

Generally speaking, location analysis and modeling is often characterized as a network based problem. Classic location books have reinforced this characterization, such as that by Handler and Mirchandani (1979) titled *Location on Networks* and that of Daskin (1995) titled *Network and Discrete Location*, as well as many other books and papers (the seminal work of Hakimi 1965, likely influenced this too). In fact, location modeling tools in ArcGIS, as an example, are part of the Network Analyst toolbox and strictly require an underlying network in order to structure and solve associated location (coverage) models. However, as reflected in much of this book, there is no inherent need or assumption that location decision making be restricted to a network, even if attribute data is derived from a corresponding network. This chapter deviates from much of the book in this respect, and assumes the underlying representation of an analysis region, including demand and potential facility locations, is based on a network. Most network location problems involve the positioning of one or more facilities at nodes or along arcs in order to optimize a level of service or access. Often, however, site selection is restricted to the nodes of the network. In the early 1980s Morgan and Slater (1980), Current (1981), and Slater (1982) broke new ground within location science when they independently suggested that facilities may be represented by some type of network structure. Examples include a path, a tree, a tour, or even the network itself. Slater (1982) suggested that facilities "... can be of an extended nature, rather than occupying a single point of a network." They suggested, for example, that facilities may be modeled as network paths representing railroad lines, pipelines, or transit routes. Slater (1982) proposed that four classes of locational problems should be considered within the context of network location: point-serves-point, point-serves-structure, structure-serves-point, and structure-serves-structure (see also Slater 1981, 1983). In this chapter we address the problem of designing a structure to serve points. One of the first examples in the literature is that of Morgan and Slater (1980) who defined

the "core" of a tree network as the shortest path connecting two endpoints of the tree, minimizing the sum of distances to all other nodes of the tree.

Current (1981) and Current et al. (1984) were the first to introduce the concept of locating a path in order to cover a set of demand nodes. This problem was called the shortest covering path problem. This concept was extended to a multi-objective path problem that maximizes path coverage while minimizing path length (Current et al. 1985). Since this early work, others have extended this concept to that of a covering salesman tour (Current and Schilling 1989), a maximal covering tour (Current and Schilling 1994) and a maximal covering tree (Hutson and ReVelle 1989, 1993). In this chapter we present the basic models of path, tour and tree covering along with a discussion of how such models might be used and applied. We also describe a more general problem of network investment/extension that is oriented towards increasing access coverage.

## 10.2   Shortest Covering Path

A considerable portion of John Current's research focused on the location of a structure within a network in order to provide service to a set of demand nodes (see Current 1981; Current et al. 1984). In much of this work, he defined service as being effective for a given node when the "structure" was located within a maximal service distance of that node. The quintessential problem of his research was the shortest covering path problem (SCPP):

*Identify the minimum cost path through a network beginning at an origin and ending at a destination where each demand node is within some predetermined coverage standard, S, of the path.*

This problem involves a prespecified origin, $p$, and destination, $q$. The objective is to find the shortest path starting at an origin and ending at a destination providing coverage to all network nodes within a maximal service standard, either based on distance or travel time. A path then is a route connecting an origin to a destination. If travel is restricted to a network of nodes and arcs, the path would consist of movement through the network, starting at the beginning node and moving along arcs through other nodes in the network until the destination node is reached. Unique is this case, however, is that the path provides coverage to other elements of the network not on the path. This clearly combines the concepts of the location set covering problem (LSCP) and a path problem. In essence the path must traverse the network in such a manner that it passes within the maximal service standard, $S$, of each node. The requirement that all nodes must be covered by the path is equivalent to the requirement that all demand (or nodes) must be covered by a set of located facilities in the LSCP. Whereas the LSCP seeks to cover all demand with the fewest facilities, the objective of the SCPP is to minimize the length of the path. In a network-based context, one can classify the LSCP as a point-serves-point problem, where facility sites and demand areas are nodes, and the SCPP can be classified as a

structure-serves-point problem, where the path is the structure and points are nodes representing demand.

Assume that we have a network of nodes and arcs. In the original work of Current et al. (1984) the network was considered undirected. To refine this somewhat, we will assume here that the network consists of a mix of undirected arcs and directed arcs. Without loss of generality, if two nodes $i$ and $j$ are connected by an arc that is undirected, it can be replaced by two directed arcs, one for each direction. Our formulation is based upon the following notation, much of which is consistent with previous chapters:

$N$ = set of network nodes
$i,j$ = indices used to refer to specific nodes of the network
$p$ = origin node of the path
$q$ = destination node of the path
$V$ = any subset of $N$
$A$ = { $(i,j)$| nodes $i$ and $j$ are connected by an arc that can be traversed}
$T_k$ = { $i \mid (i,k) \in A$}, set of nodes where an arc can be traversed directly to node $k$
$F_k$ = { $j \mid (k,j) \in A)$ }, the set of nodes that can be reached directly from node $k$
$d_{ij}$ = proximity attribute for travel between nodes $i$ and $j$, where $(i,j) \in A$
$S$ = maximal service distance or travel time standard
$C_k = \{j \mid d_{jk} \leq S \text{ where } (i,j) \in A\}$
$x_{ij} = \begin{cases} 1, & \text{if path includes arc } (i,j) \\ 0, & \text{otherwise} \end{cases}$

Using this notation, we can formulate the Shortest Covering Path Problem (SCPP) as follows:

$$\text{SCPP1}: \qquad Minimize \sum_{i \in N} \sum_{j \in F_i} d_{ij} x_{ij} \qquad (10.1)$$

Subject to:

$$\sum_{j \in F_p} x_{pj} = 1 \qquad (10.2)$$

$$\sum_{i \in T_q} x_{iq} = 1 \qquad (10.3)$$

$$\sum_{i \in T_k} x_{ik} - \sum_{j \in F_k} x_{kj} = 0 \quad \forall k \in N \mid k \neq p,q \qquad (10.4)$$

$$\sum_{j \in C_k} \sum_{i \in T_j} x_{ij} \geq 1 \quad \forall k \in N \mid k \neq p,q \qquad (10.5)$$

$$\sum_{i \in V} \sum_{j \in F_i \cap V} x_{ij} \leq |V| - 1 \quad \forall V \subset N \mid 2 < |V| < n - 1 \tag{10.6}$$

$$x_{ij} \in \{0, 1\} \quad \forall (i, j) \in A \tag{10.7}$$

The objective (10.1) of the SCPP minimizes the total length of all arcs selected for the path. The path starts at node $p$ and terminates at node $q$. Constraint (10.2) specifies that the path uses an arc which leaves the origin node $p$. Constraint (10.3) specifies that an arc is selected entering node $q$, ensuring the path reaches the intended destination. Constraints (10.4) are associated with all intermediate nodes of the network. These constraints establish that an intermediate node cannot be an origin or a destination. That is, Constraints (10.4) require that if an arc is traversed to an intermediate node $k$, then an arc must be selected that leaves node $k$. The opposite is also true. If no arc is selected that enters node $k$, then no arc can be selected which leaves node $k$. Constraints (10.5) establish that the path must pass through at least one node that is within the maximum service standard of each intermediate node $k$. Given that the path starts at node $p$ and ends at node $q$, means that those two nodes are automatically covered. Thus, these two nodes are not needed as the proximity requirements are already satisfied by their inclusion on the path. Constraints (10.6) deal with possible subtours, a sub-path that forms a loop (or cycle[1]) without connection to the origin-destination path. Subtours maintain constraints (10.2–10.5), but do represent an infeasibility. Constraints (10.6) prevent a subtour from being part of any solution. They were originally formulated by Dantzig et al. (1954) in an integer programming formulation of the travelling salesman problem. Finally, the path variables are restricted to be binary integer in constraints (10.7).

### 10.2.1   Subtour Issues

Solving the above formulation for any application instance beyond a very small network is generally out of the question as the number of constraints (10.6) increases exponentially with an increase in the number of nodes. Complete enumeration would result in nearly $2^n$ constraints, where $|N| = n$. This represents the various combinations of nodes that exist to uniquely connect a subtour, expressed as:

$$\binom{n}{2} + \binom{n}{3} + \ldots + \binom{n}{n-2} + \binom{n}{n-1} \tag{10.8}$$

---

[1]Formally defined, a cycle is a path with the same beginning and ending node.
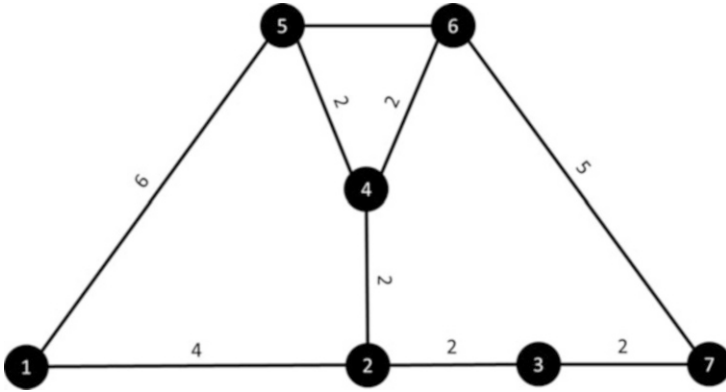
**Fig. 10.1**  Network of seven nodes and nine arcs (undirected)

This can be a very big number. For example, for a network with 15 nodes, $n = 15$, there would be 32,737 possible subtour constraints. A network of 50 nodes, $n = 50$, on the other hand would have over 1.1 quadrillion possible subtour constraints.

Current et al. (1994) formulated the SCPP problem with constraints (10.6) given above,[2] although the original work, Current et al. (1984), did not include them. The reason is that Current et al. (1984) implied in their discussion that such constraints would be used when necessary. They demonstrated that subtours do exist when solving the problem without constraints (10.6) and then used as few constraints (10.6) as possible to identify an optimal solution without any subtours.

To demonstrate the issue at hand, suppose that we apply this model without constraints (10.6) to an 8 node, 13 arc problem. This is depicted in Fig. 10.1. Assume that each arc can be traversed in either direction, with node 1 as the origin, node 8 as the destination, and a maximal service standard set at zero distance. This means that the shortest covering path needs to start at node 1, end at node 8, and visit each node. Solving the problem without constraints (10.6) yields the solution depicted in Fig. 10.2. Note that there is an unconnected loop or subtour, nodes 3 to 4 to 6 to 3. It is not connected to the path that travels from nodes 1 to 2 to 5 to 7 to 8. This is an optimal solution to the model without constraints (10.6), but it is not a feasible pathway. Thus, some or all of the constraints (10.6) are necessary to prevent infeasible subtours from occurring in a solution. This would mean up to 239 additional constraints to address subtour issues in this case.

There are three possible approaches to resolve/address the possibility of subtours occurring in a solution:

1. Include the entire set of constraints (10.6) in the model and solve

---

[2]Current et al. (1994) suggested that these constraints be written for all subsets $V$ of $N$ including $N$ itself. But, a cycle comprised of the entire set of $N$ nodes should never exist when solving the SCPP.
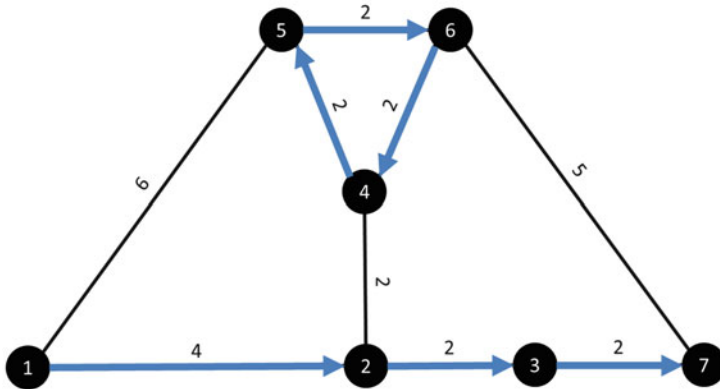
**Fig. 10.2** A solution to SCPP1 when applied to the seven node, nine arc network [Constraints (10.6) not used]

2. Include only select constraints (10.6) in order to force a particular solution(s) to form feasible pathways without subtours
3. Reformulate the problem in a manner that constraints (10.6) are not necessary

The first approach is not practical as the number of constraints (10.6) grows exponentially as a function of the number of nodes, as noted above. Even for a very small network of 50 nodes the result is over 1.1 quadrillion constraints. Clearly, this would limit the size of any problem that could be solved. Current et al. (1984) realized this fact and suggested using the second approach. Basically, they first solve a problem without any constraints (10.6). For the eight node problem in Fig. 10.1, using the model without constraints (10.6) results in the solution containing a subtour, nodes 3 to 4 to 6 to 3 (Fig. 10.2). To prevent this subtour from occurring in a solution, the following constraint would be added:

$$x_{3,4} + x_{4,6} + x_{6,3} \leq 2 \qquad (10.9)$$

This constraint prevents no more than two of the three arc selection variables $x_{3,4}$, $x_{4,6}$, and $x_{6,3}$ from being one in value. This, in effect, would eliminate the occurrence of this specific loop in a solution. The basic idea is that after solving the model without constraints (10.6) either an optimal, feasible covering path is found or a solution is found comprised of a pathway along with one or more subtours. If any subtours appear in the solution, then one or more constraints (10.6) are added to prevent their occurrence and then the model is resolved. If this second solution contains subtours, then more constraints (10.6) are added, and the model is resolved. Essentially, one continues this process until the resulting solution is a connected path without any subtours. By solving this problem in an iterative manner, only those constraints (10.6) that are needed would be used.

One should note that constraint (10.9) is not an exact form of constraints (10.6), but rather a simplified form. Figure 10.3 depicts a hypothetical cycle of six nodes:
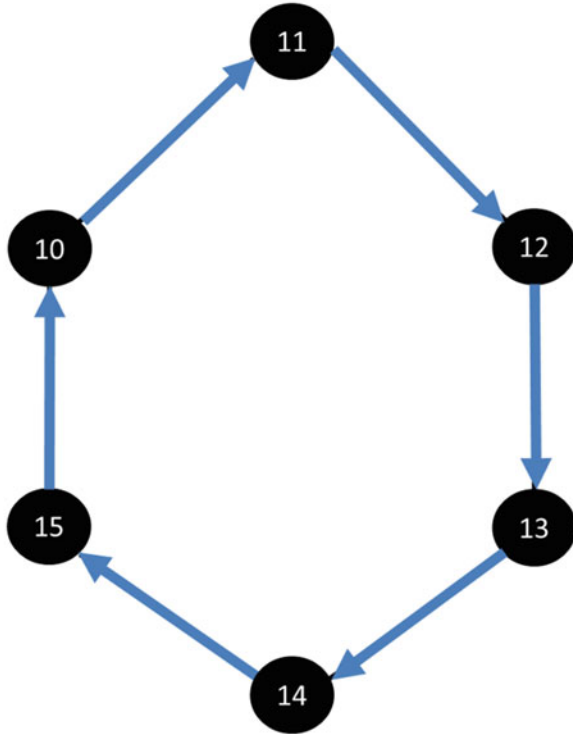
10 to 11 to 12 to 13 to 14 to 15 to 10. If this cycle was encountered in a solution, the
Current et al. (1984) approach would involve adding the following constraint:

$$x_{10,11} + x_{11,12} + x_{12,13} + x_{13,14} + x_{14,15} + x_{15,10} \leq 5 \qquad (10.10)$$

and then resolving the model. Of course, resolving this problem, without other
stipulations, would probably result in a solution involving the same loop of nodes,
but where the direction of the cycle would be reversed. Technically, this would
require another constraint, namely:

$$x_{10,15} + x_{15,14} + x_{14,13} + x_{13,12} + x_{12,11} + x_{11,10} \leq 5 \qquad (10.11)$$

The proposed form of subtour breaking constraints (10.6) are somewhat more
complex and would prevent both forward and reverse cycles as well as other
possibilities for the set $V$, comprised in this case of nodes 10 through 15. Figure 10.4
depicts the cycle of nodes 10 through 15 along with all arcs that exist which connect
any pair of nodes within this of six nodes. Given a set $V = \{10, 11, 12, 13, 14, 15, 16\}$,
constraint (10.6) in this case would be:

$$x_{10,11} + x_{10,15} + x_{11,10} + x_{11,12} + x_{11,14} + x_{11,15} + x_{12,11} +$$
$$x_{12,13} + x_{12,14} + x_{12,15} + x_{13,12} + x_{13,14} + x_{14,11} + x_{14,12} + \qquad (10.12)$$
$$x_{14,13} + x_{14,15} + x_{15,10} + x_{15,11} + x_{15,12} + x_{15,14} \quad \le \quad 5$$

Essentially, this constraint allows at most 5 arcs to be used in a solution among all of the arcs which interconnect the six nodes in the set $V$. Thus, this constraint will cover both conditions (10.10) and (10.11) as well as other combinations, including sets of possible subtours spanning the six node set. For example, Fig. 10.5 depicts two subtours, 10 to 11 to 15 to 10 and 12 to 13 to 14 to 12, which would be prevented from simultaneously occurring as well. Any subtour or combination of subtours which spans the set $V$ will be prevented in this case. This includes, for example, the subtour 10 to 11 to 14 to 13 to 12 to 15 to 10, or the three subtours 10 to 11 to 10, 12 to 13 to 12, and 14 to 15 to 14. Consequently, it makes sense to use the general form of constraints (10.6) for subtour elimination rather than the simplified form used in Current et al. (1984).



**Fig. 10.4** Modified network of arcs
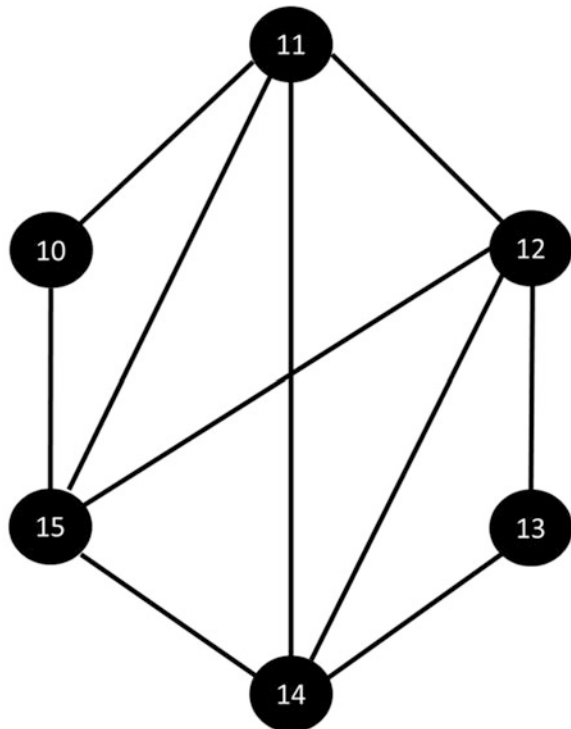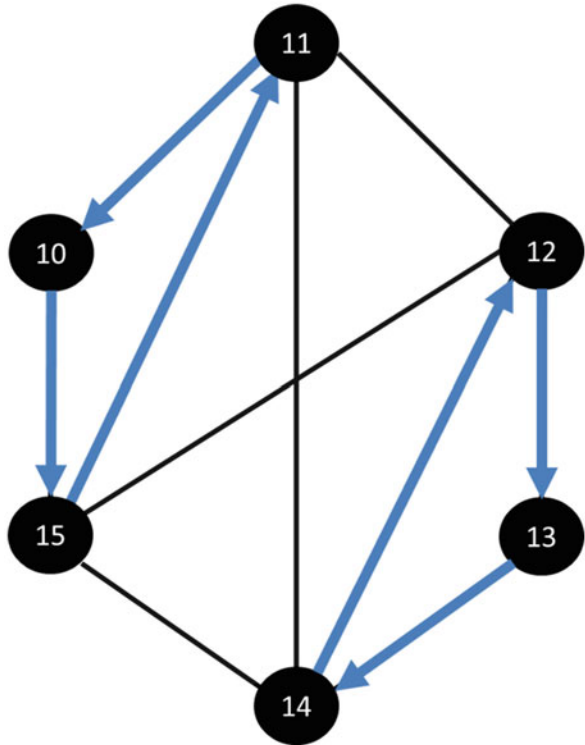
## 10.2.2   Alternatives for Eliminating Subtours

There are alternate forms in which the shortest path covering problem can be cast in
order to eliminate the need for constraints (10.6). One of the ways is to use an
approach based upon a construct of Vajda (1961) originally developed for the
travelling salesman problem. Consider the following decision variable representing
the order in which arcs are used along a route:

$$x_{ijt} = \begin{cases} 1, & \text{if the } t^{th} \text{ arc in the path travels from } i \text{ to } j \\ 0, & \text{otherwise} \end{cases}$$

Using this decision variable, it is possible to formulate a variant of the SCPP,
inspired in part by the related work of Curtin and Biba (2011). The formulation
follows:

$$\text{SCPP2}: \quad \textit{Minimize} \ \sum_{i} \sum_{j \in F_i} \sum_{t=1}^{n-1} d_{ij} x_{ijt} \tag{10.13}$$

*Subject to:*

$$\sum_{j \in F_p} x_{pj1} = 1 \tag{10.14}$$

$$\sum_{i \in T_q} \sum_{t=1}^{n-1} x_{iqt} = 1 \tag{10.15}$$

$$\sum_{i \in T_k} \sum_{t=1}^{n-1} x_{ikt} \leq 1 \quad \forall k \tag{10.16}$$

$$\sum_{j \in F_k} \sum_{t=1}^{n-1} x_{kjt} \leq 1 \quad \forall k \tag{10.17}$$

$$\sum_{i} \sum_{j \in F_i} x_{ijt} \leq 1 \quad \forall t < n \tag{10.18}$$

$$\sum_{i \in T_k} x_{ikt} - \sum_{j \in F_k} x_{kjt+1} = 0 \quad \forall k \neq p, q \& t < n \tag{10.19}$$

$$\sum_{j \in C_k} \sum_{i \in T_j} \sum_{t=1}^{n-1} x_{ijt} \geq 1 \quad \text{for all} \quad k \in N \quad \text{where} \quad k \neq p, q \tag{10.20}$$

$$x_{ijt} \in \{0, 1\} \quad \forall (i, j) \in A, t < n \tag{10.21}$$

The objective (10.13) of SCPP2 minimizes path distance based upon the arcs selected for the path, equivalent to the SCPP1. Each arc $(i, j)$ is represented by a set of variables, $x_{ijt}$. If a given arc is selected, $x_{ijt} = 1$, then the path travels from node $i$ to node $j$ along arc $(i, j)$ as the $t^{th}$ arc in the path. Accordingly, the order of the arcs encountered along the path is tracked by the index $t$. As the path must begin at the origin $p$, then the first arc chosen in the path must leave node $p$. This property is maintained in constraint (10.14). The path must eventually arrive at the destination node $q$. Since we do not know in advance how many arcs are to be used in the path, we do not know the order nor the number needed. Consequently, constraint (10.15) ensures that the path reaches the destination node $q$ using some order $t$, but that is not fixed in value. Constraints (10.16) maintain that each node is entered at most once over all possible levels of path order. Constraints (10.17) maintain that a path can leave a node at most once. Constraints (10.18) maintains that at most one arc is chosen for a given order. That is, no two arcs can be assigned the same order in the path. Constraints (10.19) are key in this model variant. It establishes that if a specific intermediate node $k$ is entered on the $t$ step or order, there must be an arc selected to

leave that node using an arc selected with order $t + 1$. This constraint ensures that the path is a continuous and that the order of arcs selected along the path increases in increments of one. There is no requirement that all sequence orders be used. Rather, when the path reaches the destination at a specific sequence order, $t = q$, the decision variable values associated with $t + 1$ or higher will be zero. Constraints (10.20) ensure that all nodes other than the origin and destination must be covered or served by the path. Finally, constraints (10.21) present the binary integer restrictions on the decision variables.

Technically speaking, constraints (10.17) are redundant considering constraints (10.16). This is due to the fact that if each node can be entered at most once along the path based upon constraints (10.16), then constraints (10.19) will prevent a node from being left more than once. Collectively, the constraints prevent any subtours from occurring in a solution. Thus, subtour elimination constraints are not needed in SCPP2. Further, the path is also not permitted to cross itself either, a condition that is imposed in SCPP1 as well.

The size of SCPP2, in terms of constraints is on the order of $n^2 + n$, which is considerably smaller than that of SCPP1. Unfortunately, the SCPP2 model contains $n - 1$ times more integer variables than SCPP1. Thus, the cost of reducing the number of constraints in SCPP1 is significant. There exist several alternate ways in which SCPP can be formulated based upon approaches relied upon in structuring the traveling salesman problem. They include the use of a flow-based structure to ensure no subtours (Gavish and Graves 1978), a two-commodity flow model (Finke et al. 1983), and variables established to track node sequencing (Miller et al. 1960). Virtually, all models suffer from dimensionality issues and/or lack of a strong linear programming relaxation. Orman and Williams (2004) have tested a number of formulations for the travelling salesman problem and conclude that the best approach appears to be the original approach of Dantzig et al. (1954), where subtour breaking constraints are included as needed. They suggest solving a problem sequentially, adding constraints when one or more subtours arise, and continuing until no subtours exist. The resulting solution is then feasible, and an optimal traveling salesman problem solution. Their results lend credence to the approach adopted by Current (1981) and Current et al. (1984), suggesting that it is likely the best method to formulate and solve the classic form of the SCPP.

## 10.3   Salesman and Tour Coverage

In a previous section we provided two relatively different formulations for the SCPP. Both formulations are based in part on past travelling salesman problem research. Current and Schilling (1989) proposed a counterpart to the travelling salesman problem called the Covering Salesman Problem (CSP). They describe the CSP as follows:

*Identify the minimum cost tour of a subset of n cities such that every city not on the*
   *tour is within some predetermined covering distance standard, S, of a city that is*
   *on the tour.*

Solutions to the classic travelling salesman problem must visit each city on the tour. The CSP therefore represents a relaxation of this assumption and instead requires that the tour come sufficiently close to each city. A tour then is a special type of path that must visit each node in the network. Application of the CSP could involve finding the shortest route for a mobile health clinic to travel within a rural region of villages, where the route stops at villages that are sufficiently close to all remaining villages.

Formulating this problem is quite straight forward given the previous discussion of the SCPP1. In particular, constraints (10.2) and (10.3) are eliminated and constraints (10.4) and (10.5) are modified to hold for all nodes $k \in N$ as there is no pre-specified origin or destination. Solving this problem will yield the shortest cycle covering all nodes. It is important to note that the SCPP2 cannot be easily modified in order to solve for a covering salesman tour because it is not known a priori how many arcs will be needed for the covering salesman tour.

The CSP does not require that a specific node be a starting (and ending) location, and Current and Schilling (1989) anticipated that in many cases this might be desirable. They also suggested there might be other nodes/cities required to be part of the tour. To accomplish this, all that is required is to add a constraint (10.2) for each such node. While the SCPP2 cannot be easily modified to solved for a covering tour, if a starting city is known (e.g., node $p$), then a covering route can be found by modifying constraints (10.15) to hold for node $q = p$ and by enforcing constraints (10.19) and (10.20) for all nodes except $p$.

Current and Schilling (1989) also suggested that in some cases the cost of a route may be more than the sum of the distances of the arcs that are traversed, as it may well be a function of the number of stops on the route. Such a possibility can be easily addressed as a second objective. This second objective would equal the number of arcs chosen for the path, multiplied by the cost of stopping. Current and Schilling (1989) indicated that one could then generate a tradeoff between the number of stops and the length of the route. Note that solving a covering salesman problem where the route consists of exactly $T$ stops can be also found using a modified form of SCPP2, where constraints (10.14) and (10.15) are dropped, constraints (10.18) are converted into equalities and invoked for all orders from 1 to $T$, constraints (10.19) are imposed for all orders 1 to $T - 1$, constraints (10.20) are enforced for all nodes, and the following constraint is added:

$$\sum_{i \in T_k} x_{ikT} - \sum_{j \in F_k} x_{kj1} = 0 \quad \forall k \tag{10.22}$$

This constraint forces the arc used in order $T$ to enter the node left in order 1, thereby making a loop. The modified model is structured so that it is a path that starts at some endogenously determined starting node, uses exactly $T$ arcs, and

returns to that starting node using the path order $T$ arc. When solving this model for a range of values $T$ one can generate a tradeoff between salesman route length and the number of nodes/stops used in the route.

ReVelle and Laporte (1993) proposed the traveling circus problem, which is related to the covering salesman problem. This was later called the covering tour problem (CTP) in Gendreau et al. (1997), and this naming convention is adopted here. A formal description is as follows:

*Identify the minimum cost tour that visits a prespecified set of nodes and covers a prespecified set of nodes.*

Formal specification of CTP is facilitated by the use of the following three sets of nodes (or vertices in a graph):

$V$ = the set of nodes that can be visited along the route or tour
$\Omega$ = set of nodes that must be visited along the route/tour
$\Phi$ = set of nodes that must be covered by the route/tour

The CTP seeks to find the minimum length tour that visits all nodes $\Omega$ and covers all nodes $\Phi$. It should be noted that such tours may not exist for a given graph. If sets $V$ and $\Phi$ include all nodes, $N$, and the set $\Omega$ is empty, then the CTP is equivalent to the CSP. If $\Omega$ is not empty, then the CTP is equivalent to a special case of the CSP where certain nodes must be visited. Gendreau et al. (1997) provide an alternative approach for formulating the CTP, differing from the two models given above, and propose a branch and cut algorithm as well as a heuristic for solution. It should be noted that there are several related problems that have appeared in the literature. They include the tour cover problem on a graph involving node costs and not arc costs (Arkin et al. 1993), the geometric covering salesman problem where the vertex set of possible tour points is not a discrete set of points but planar regions (Arkin and Hassin 1994), the prize collecting travelling salesman problem (Fischetti and Toth 1988), the selective travelling salesman problem (Laporte and Martello 1990), and the generalized covering salesman problem (Golden et al. 2012).

## 10.4   Maximal Coverage Variants

Given coverage path equivalents for the LSCP, extension to account for partial coverage of demand areas (nodes) should not be surprising. Current et al. (1985) proposed an extended path problem that allowed for some nodes not to be covered. Whereas the shortest covering path must cover all nodes, this extension allowed for some nodes not to be covered. This will be referred to here as the Maximal Covering–Shortest Path Problem (MCSPP). It is described as follows:

*Given a path origin and destination, find the path that starts at the origin and terminates at the destination that minimizes path distance and maximizes path coverage.*

This is a two objective problem, formulated by extending the SCPP. To demonstrate how this is done, consider the following additional notation:

$a_k$ = demand at node $k$

$y_k = \begin{cases} 1, & \text{if demand at node } k \text{ is covered within standard } S \\ 0, & \text{otherwise} \end{cases}$

$Z_1$ = objective function value corresponding to total coverage of the path

$Z_2$ = objective function value indicating the total distance of the path

Given this notation, the MCSPP is formulated as follows:

$$\text{MCSPP}: \quad \text{Maximize } Z_1 = \sum_{\substack{k \in N \\ k \neq p,q}} a_k y_k \quad\quad (10.23)$$

$$\text{Minimize } Z_2 = \sum_{i \in N} \sum_{j \in F_i} d_{ij} x_{ij} \quad\quad (10.24)$$

Subject to:

$$\sum_{j \in C_k} \sum_{i \in T_j} x_{ij} \geq y_k \quad \forall k \neq p,q \quad\quad (10.25)$$

$$y_k \in \{0,1\} \quad \forall k \neq p,q \quad\quad (10.26)$$

and constraints (10.2, 10.3, 10.4, 10.6, and 10.7).

This model is essentially the same as the SCPP1 except that the possibility has been added that a given node and its demand may not be covered by the path. This is done with the use of the coverage variable $y_k$ introduced above. Essentially the path has to be within $S$ of a given node $k$ in order for the coverage variable $y_k$ to be equal to one. This is established in constraint (10.25). Current et al. (1985) proposed solving this model without constraints (10.6) in the same manner as the shortest covering path problem. When doing so, however, it is to be expected that subtours will occur, just as was the case in the SCPP solution process. This means that constraints would be necessary like those of constraints (10.6) to prevent subtours. So, as new subtours are encountered, additional subtour elimination constraints are added and the problem resolved. Eventually, a solution will be obtained that is optimal and feasible (i.e. containing no subtours). This approach is exactly the same as that described for the SCPP previously. An optimal MCSPP solution is given in Fig. 10.6 involving the City of Richardson, TX. This solution was generated with emphasis on coverage as compared to path distance. Thus, the path is less direct between the origin and the destination nodes and tends to meander as a means of increasing path coverage.

A complicating feature of the MCSPP deserving some attention is the fact that is has two objective functions, (10.23) and (10.24). The implication is that there may be many different solutions that could be considered optimal with respect to some
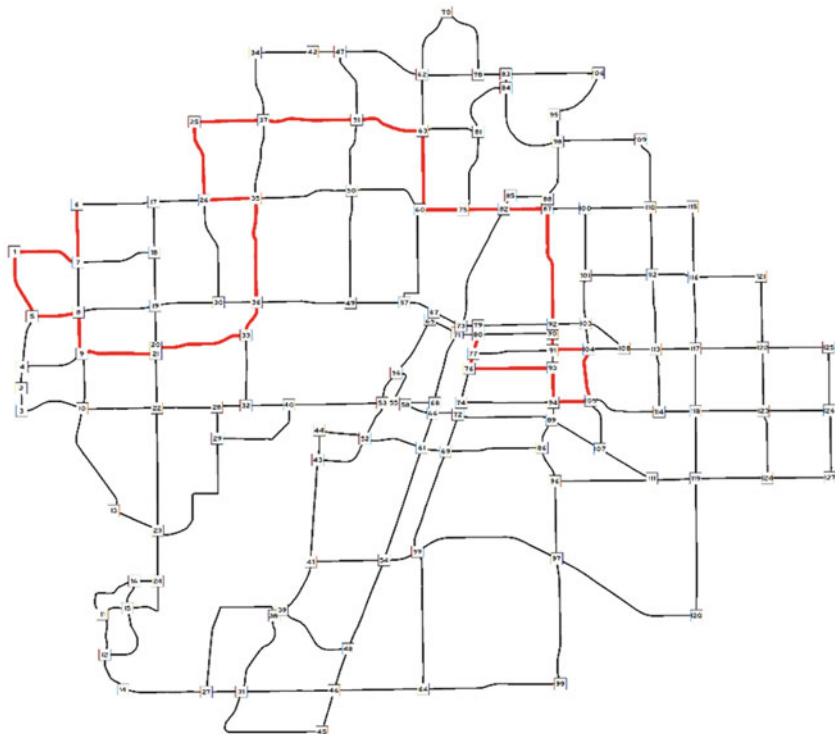
**Fig. 10.6** A solution to MCSPP applied to the City of Richardson, TX. This path emphasizes coverage over that of path distance, where the origin node is on the outside boundary of Richardson and the destination node represents the transit center near the center of the city

combination of preference ratings of one objective over the other. Such solutions are known as Pareto optima. Further, it may be possible to summarize these alternative optima using a graphic that is often termed objective space. This is a plot of all solutions in terms of their values of objective (10.23) and objective (10.24), with one of the objectives represented by the x-axis and other on the y-axis. Altogether these solutions represent a tradeoff that exists between the two objectives. The challenge of course is identifying these tradeoff solutions. Cohon (1978) discusses methods for solving multiobjective optimization problems, noting that two options are the weighting method and the constraint method. Rather than get into the details of solution methods at this point, we will leave the discussion at merely highlighting that solving models that involve two or more objectives is itself a challenging task, and the subject of much ongoing research. Details for generating tradeoffs for path problems can be found in Medrano and Church (2014, 2015).

While the MCSPP is challenging to solve on many different fronts, one issue affecting this has to do with the number of associated decision variables. Under certain conditions, reformulation with fewer decision variables may be possible. For example, suppose that the covering standard is zero, meaning that the path needs to

travel to each node that it covers. The design of a light rail train route connecting villages is a good example of when such an extension might be necessary, where light rail does not effectively serve a village unless it stops there. Current et al. (1985) called such a problem extension the maximum population shortest path problem and created a special formulation for this case. They reasoned that a pathway would enter a node at most once on its way to the destination. If we know that a pathway enters a node at most once, then that node is reached and "covered" when an arc is chosen entering that node.

Using the notation described above for the SCPP and the MCSPP, a variant is possible. For consistency purposes, this is simply referred to as the MCSPP2:

$$\text{MCSPP2}: \qquad Maximize \ Z_1 = \sum_{k \neq p,\, q} \sum_{i \in T_k} a_k x_{ik} \qquad (10.27)$$

$$Minimize \ Z_2 = \sum_{i \in N} \sum_{j \in F_i} d_{ij} x_{ij} \qquad (10.28)$$

Subject to:

$$\sum_{i \in T_k} x_{ik} \leq 1 \quad \forall k \neq p, q \qquad (10.29)$$

and constraints (10.2), (10.3), (10.4), (10.6), and (10.7)

Consistent with MCSPP, the MCSPP2 has two objectives: maximize coverage, (10.27), and minimize path length, (10.28). Constraints (10.29) ensure that the path enters each node (not the origin or destination) at most once. This allows us to eliminate the need for the coverage variable $y_k$, and reconstruct the objective (10.27) solely in terms of the path variables, $x_{ij}$. Thus, this is a more compact model than when the maximal service distance is greater than 0 in value. One may wonder why this construct cannot be used for the MCSPP as well. The reason why is that the maximal covering path may travel close to a node more than once, and thus cover that node several times. If constraints (10.29) are modified for non-zero coverage distances, if would take the following form:

$$\sum_{j \in C_k} \sum_{i \in T_j} x_{ij} \leq 1 \quad \forall k \neq p, q \qquad (10.30)$$

This, in essence, would prevent a path from coming close to a given node $k$ more than once. If any arc is less than the maximal service standard, then condition (10.30) would actually prevent that arc from ever being used in a solution. That is, using that arc would cover the node by its entry, and be covered by the fact the nearby neighboring node was entered as well in a previous stage or order.

Another variant to be detailed is associated with a tour instead of a path. Current et al. (1994) referred to this as the maximal covering tour problem (MCTP). This problem can be described as follows:

*Given a network of n nodes, where each node has an associated amount of demand and each arc has a traversal cost, find the route which visits exactly p-nodes while minimizing route traversal cost and maximizing route coverage.*

They chose to restrict the tour to visit exactly $p$-nodes, as this was a special case of what has been termed the median tour problem. The median tour problem involves finding a route which visits exactly $p$-nodes while minimizing the route cost as well as the sum of the shortest distances of those nodes not selected for the route in travelling to their closest node on the route.

The MCTP can be formulated by modifying the MCSPP. As the MCSPP is bi-objective, involving the minimization of path distance and the maximization of coverage, these are the same objectives of the MCTP. Formulation requires the removal of the origin and destination constraints, (10.2) and (10.3), in the MSCPP and enforce the path continuation constraints (10.4) for all nodes. In addition, it is necessary to add the following constraint:

$$\sum_i \sum_{j \in F_i} x_{ij} = p \qquad (10.31)$$

This constraint ensures that exactly $p$ nodes will be selected for the tour. Thus, the entire model can be formulated as bi-objectives, (10.23) and (10.24) subject to constraints (10.4), for all nodes, (10.6), (10.7), (10.25), (10.26), and (10.31).

### 10.4.1 Trees

Thus far, network structures have been reviewed consisting of paths and tours. Recall that a path is a route from an origin to a destination, traveling along arcs in the network through nodes. A tour is a particular type of path where the origin is also the destination, and all nodes in the network must be visited. Differing somewhat from a path or tour is a network structure known as a tree. Formally defined, a tree is a sub-network that is connected and contains no cycles (or loops). The problem of locating structures that provide coverage was extended to tree graphs in Hutson and ReVelle (1989). They defined two types of coverage, direct and indirect. In direct coverage, only nodes connected to the tree are considered served. That is, the service distance standard is assumed to be zero. In the indirect coverage case it is assumed that nodes relatively close to the tree structure, but not attached to it, are considered served. For example, if the tree structure represents a hydrogen fuel distribution network, then the nodes of the tree are refilling stations. Accordingly, hydrogen vehicles from nearby nodes of the tree can access the refueling system as can those nodes that are part of the tree structure. Indirect coverage can therefore be defined for maximal covering standards greater than 0. The distinction here is somewhat akin to the difference between the MCSPP2 (i.e., direct coverage) and the MCSPP (i.e., combined direct and indirect coverage). Hutson and ReVelle (1989) also suggested

that indirect coverage could also include those nodes not on the tree but are within a specified distance of an arc on the tree. Worth noting as well is that indirect coverage provided by arcs may be applicable to problems where system access is based upon all tree structure elements and not just the nodes of the tree (see Curtin and Biba 2011).

Covering tree location problems are bi-objective, unless complete coverage must be provided. In the case of finding a complete direct covering tree on a graph, it makes sense to find the least cost tree that reaches all nodes. Since the tree must cover (reach) all demand, and assuming that the cost of the tree is the sum of the distances of the arcs selected for the tree, then this problem reduces to one of finding the minimal spanning tree of a graph. This can be done by very efficient algorithms (Kruskal 1956). However, if a discrete set of points distributed across a planar region is to be covered by a direct covering tree, then the problem is that of finding the optimal Steiner tree on the plane. Thus, the graph version of this problem is quite simple, and the planar version of the problem is quite complex, computationally.

Hutson and ReVelle (1993) suggested that minimal cost spanning trees could serve as a "backbone" network for service delivery problems. For example, tree structures have been applied for the provision of communication, water, and limited service road networks (Church and Current 1993). Given an efficient spanning tree, Hutson and ReVelle (1989, 1993) proposed two problem variants:

1. minimal cost covering subtree on a spanning tree backbone—*Identify those arcs which form a connected subtree on a tree graph, which minimizes cost and covers all nodes within a specified maximal service distance*
2. maximal covering subtree on a spanning tree backbone—*Identify those arcs which form a connected subtree on a tree graph, which simultaneously minimizes cost and maximizes demand coverage*

The more general form of the subtree design/selection problem is the latter case, and will be formulated as the maximal direct covering subtree problem (MDCSTP). We can rely on notation already used, although simplification is possible through reference to each arc on the tree being numbered $j = 1, 2, 3, \ldots, m$. Consider then the following notation:

$x_j = \begin{cases} 1, & \text{if arc } j \text{ has been selected for the maximal covering tree} \\ 0, & \text{otherwise} \end{cases}$

$I_{h,i} = \{j | \text{ arc } j \text{ is on the path between nodes } h \text{ and } i\}$

$P_i = \{j | \text{ arc } j \text{ can cover node } i \}$

The MDCSP was originally formulated by Hutson and ReVelle (1989) as follows:

$$\text{MDSCP}: \quad \textit{Minimize } Z_1 = \sum_j c_j x_j \quad\quad (10.32)$$

$$Maximize \ Z_2 = \sum_i d_i y_i \tag{10.33}$$

*Subject to:*

$$y_i \le \sum_{j \in P_i} x_j \tag{10.34}$$

$$y_h + y_i \le x_j + 1 \quad \forall i, h, j \in I_{h,i} \tag{10.35}$$

$$x_j = \{0, 1\} \quad \forall j \tag{10.36}$$

$$y_i = \{0, 1\} \quad \forall i \tag{10.37}$$

Objective (10.32) minimizes the costs of the arcs selected for the covering subtree. Objective (10.33) maximizes the demand covered by the selected subtree. Constraints (10.34) establish whether a given demand has been covered. Constraints (10.35) ensure that the arcs that have been selected produce a subtree that is connected. Constraints (10.36) and (10.37) impose binary integer restrictions on decision variables. The key constraint in this model is (10.35). The basic idea that the path between any two nodes on a spanning tree is unique, otherwise there would be a cycle within the tree graph and the graph would not be a tree. So, if two nodes, $h$ and $i$, are covered by the subtree, then for the subtree to be connected, all arcs on the path between nodes $h$ and $i$ must be selected for the subtree, otherwise the subtree will not be connected. Essentially, if both nodes are covered, $y_i = 1$ and $y_h = 1$, this condition forces any $x_j = 1$ where arc $j$ is on the path between nodes $h$ and $i$.

This model is deceptively simple, but the number of constraints (10.35) that are necessary to ensure that the subtree is connected can be quite large. Hutson and ReVelle (1989) solved this problem using a weighting approach, where a positive weight is used for the coverage objective and a negative weight is used for the tree cost objective. By varying the weights, and solving this model for each set of weights, they were able to generate a tradeoff between subtree coverage and subtree cost. Because the number of constraints (10.35) can be very large, Hutson and ReVelle (1989) formulated a model that could optimally add to an existing subtree, requiring a limited number of constraints. They suggested that this smaller model could be solved $n - 1$ times, where the existing subtree was one of the $n - 1$ arcs. By solving for each of the arcs as an existing tree on which to build an optimal subtree, the globally best covering subtree can then be identified as the best of these $n - 1$ solutions.

Church and Current (1993) developed a different formulation for the maximal direct and indirect covering subtree problems. Rather than using a variable to represent the selection of an arc for the subtree, they employed variables that represented a clipping/trimming off process of portions/branches of the initial tree graph in order to reveal the optimal subtree. Clipping a branch of a subtree was based upon the direction in which a branch was clipped. Their model involved only $2n - 2$ binary variables and $n - 1$ constraints, which eliminate the need for a large number

of constraints (10.35). They also developed a polynomial bounded algorithm (on the order of $n^2$) that could solve both the indirect and direct covering tree problems, eliminating the need for solving an integer programming problem.

## 10.5   Arc Improvement

Murawski and Church (2009) extended the concept of coverage to a network of specific node locations as well as arcs when dealing with a problem of health clinic access in Ghana. Their concern was directed to the fact that existing health clinics were often inaccessible in the wet season as roads were of poor quality and were impassible, an issue raised in Oppong (1996). Whereas Oppong (1996) had proposed to relocate health clinics to increase clinic access by taking into account road quality, Murawski and Church (2009) proposed improving roads in order to increase access to existing clinics. Their problem of infrastructure investment can be defined as:

*Maximize the total demand who can be provided with service coverage by determining which network arcs are to be improved, subject to a limited road investment budget.*

This problem is called the maximal covering network improvement problem (MC-NIP). We can formulate this model with some redefined notation:

$i,j,k$ = indices of network nodes
$c_{ij}$ = cost of upgrading arc $(i,j) \in A$
$U = \{ (i,j)|\ (i,j) \in A\ $ and $c_{ij} > 0 \}$, the set of arcs that are not upgraded
$B$ = budget for arc improvement
$F = \{$set of existing facility sites$\}$
$D = \{$set of nodes without upgraded access to a facility$\}$
$F_i = \{\ j\ |\ d_{ij} \leq S$ where $j \in F\ \}$
$a_i$ = population of node $i$
$\Omega_{ij} = \{$ set of arcs on possible path between nodes $i$ and $j \}$
$$H_{ij} = \begin{cases} 1, \text{if arc } (i,j) \text{ is upgraded} \\ 0, \text{otherwise} \end{cases}$$
$$y_{ij} = \begin{cases} 1, & \text{if node } i \text{ and facility } j \text{ are connected by an upgraded arc} \\ 0, \text{otherwise} \end{cases}$$
$$P_{i'j'}^{ij} = \begin{cases} 1, \text{if upgraded route between } i \text{ and } j \text{ uses arc } (i',j') \\ 0, \text{otherwise} \end{cases}$$

The above variables are defined for only those cases which apply. For example, if a village is already provided all-season access to a health clinic services within the maximum coverage distance, then it is unnecessary to have a coverage variable for that village. The same is true for road links that are already considered all-season passable. Murawski and Church (2009) note that the choice of direction (clinic to

village coverage via paved road segments) is somewhat arbitrary and that the model could be formulated using path directions from villages to clinics.

The model formulation is as follows:

$$\text{MC-NIP}: \quad \textit{Maximize} \; \sum_{i \in D} \sum_{j \in F_i} a_i y_{ij} \tag{10.38}$$

*Subject to:*

$$\sum_{(i',i) \in \Omega_{ij}} P_{i'i}^{ij} = y_{ij} \quad \forall i \in D, j \in F_i \tag{10.39}$$

$$\sum_{\substack{(i',k) \in \Omega_{ij} \\ (k,i') \in \Omega_{ij}}} P_{ak}^{ij} - \sum_{\substack{(k,j') \in \Omega_{ij} \\ (j',k) \in \Omega_{ij}}} P_{kb}^{ij} = 0 \quad \forall i \in D, j \in F_i \, k \neq i \& k \neq j \tag{10.40}$$

$$P_{i'j'}^{ij} + P_{j'i'}^{ij} \leq H_{i'j'} \quad (i',j') \in \Omega_{ij} \mid i \in D, j \in F_i, i \neq i', i \neq j', j \neq i', j \neq j' \tag{10.41}$$

$$\sum_{(i',j') \in \Omega_{ij}} \left( P_{i'j'}^{ij} + P_{j'i'}^{ij} \right) d_{i'j'} \leq S y_{ij} \quad \forall i \in D, j \in F_i \tag{10.42}$$

$$\sum_{j \in F_i} y_{ij} \leq 1 \quad \forall i \in D \tag{10.43}$$

$$\sum_{(i,j) \in U} c_{ij} H_{ij} \leq B \tag{10.44}$$

$$H_{ij} \in \{0,1\} \quad \forall (i,j) \in U \tag{10.45}$$

$$0 \leq P_{i'j'}^{ij} \leq 1 \& 0 \leq P_{j'i'}^{ij} \leq 1 \quad \forall i \in D, j \in F_i, (i',j') \in \Omega_{ij} \tag{10.46}$$

$$y_{ij} \in \{0,1\} \quad \forall i \in D, j \in F_i \tag{10.47}$$

Objective (10.38) seeks to maximize the number of people that are provided upgraded service coverage by virtue of arc investment decisions. Access coverage is provided to a node by the existence of a service facility and an upgraded arc path from that facility to a specific node, as long as it is with the service coverage standard. The facility nodes already have "all-season" access and others might already have such access due to existing upgraded arcs. The remaining nodes can be classified as to whether they can be provided upgraded coverage access through arc investments, $D$, or whether they are too far from a facility to ever be considered accessible independent of the arc condition. The model addresses only those nodes in $D$. Constraints (10.43) ensure that a demand node is considered to be covered no more than once, even though it may be possible to travel two different upgraded access routes from a facility to that demand node. Constraint (10.44) limits the investment in all upgrading to no more than an allowable budget. The bulk of the remaining constraints track which nodes are provided coverage by virtue of which

arcs are selected for upgrading. To achieve this it is necessary to define if a path exists between a facility and a node that is no longer than S and is comprised entirely of upgraded arcs. Path variables, $P_{i'j'}^{ij}$, are used to define a feasible path between node $i$ and facility $j$. Such variables represent traveling along an arc connecting $i'$ and $j'$. When $P_{i'j'}^{ij} = 1$, then all-season accessible travel from nodes $i'$ and $j'$ from facility $j$ to demand node $i$ is possible. If an accessible path connects node $i$ to facility $j$, then constraint (10.39) allows $y_{ij}$ to equal 1. If this is not the case, $y_{ij}$ will remain at zero. Constraints (10.40) maintain that paths are connected through the typical node-balance constraints used in previously defined models. At every feasible intermediate node between node $i$ and facility $j$, when a path enters an intermediate node $k$, the path must exit that same node $k$. Altogether, a feasible path between node $i$ and facility $j$ starts at node $j$, and must reach and stop at demand node $i$. Whenever a feasible path does not exist between $i$ and $j$, then all path variables $P_{i'j'}^{ij}$ associated with $i$ and $j$ are zero in value. Path variables are not restricted in value when traversing any existing all season road links. But, path variables are restricted to be zero, when their associated arcs need upgrading and they have not been upgraded This is established in constraint (10.41). That is, a path from facility $j$ to node $i$ can use a specific arc $(i',j')$ only if it is already upgraded, or has been upgraded. The last significant constraint (10.42) restricts the distance of the path selected between demand $i$ and facility $j$ to be less than or equal to the standard $S$, otherwise the path is not feasible. Finally, constraints (10.45), (10.46), and (10.47) indicate restrictions on decision variables.

Collectively, the MC-NIP allows no more than one feasible path comprised of upgraded arcs between any facility and node $i \in D$. Any such path must be no longer than the service coverage standard, $S$. This model was applied to the same region that was modeled by Oppong (1996), and enabled substantial increases in coverage to be provided by modest investments in network improvements.

## 10.6   Applications

The covering path problems are particularly appealing in that they represent design goals found in transit routing, light rail routing and rail station location, emergency/ disaster relief supply routes where the nodes along the route are distribution points for emergency supplies, and mobile clinics for vaccination and health screening. The area of transit design has been addressed by a number of researchers starting with the conceptual models first proposed by Current et al. (1984). Boffey and Narula (1998) and Wu and Murray (2005) have suggested multi-path maximal covering shortest route models that optimize several routes simultaneously in order to cover a region. The maximal covering shortest path structure has also been expanded in a number of ways for extending transit routes, maximizing coverage of origin-destination travel, stop location, and arc coverage (see Murray and Wu 2003; Church et al. 2005; Matisziw et al. 2006; Curtin and Biba 2011).

The covering tour problems also capture the essence of design problems as planning security patrol routes and rural postal delivery routes using clustered/sets of mail boxes. For instance, suppose an industrial plant and grounds were divided into small sectors, and the objective was to determine a security route which was able to view each of the areas as stops along the route. In some communities, mail boxes for neighborhoods are being clustered and located at a specific point. One can think of the shortest covering tour problem as one that captures the essence of the route moving from cluster box locations and returning to the starting office, while ensuring that each customer is no further than a certain distance from their closest location of clustered mailboxes. The same type of problem involves drop box locations for banks, where drop boxes are placed so that customers can make large cash deposits after bank hours. There are also a number of potential applications for the generalized covering tour problem where a return frequency differs among nodes or areas.

## 10.7   Summary and Concluding Comments

Slater (1982) classified location problems based upon points serving points, points serving structures, structures serving points, and structures serving structures. This chapter highlighted the fact that the shortest covering path problem, as an example, can be conceived to be a structure (i.e., path in this case) serving a set of points (i.e., nodes in this case). The emergency warning siren location problem described in Chaps. 1 and 8 involves structures (i.e., siren coverage areas defined as a polygon) serving a structure (a region defined as a polygon). In this chapter we have discussed problems on a network that involve a structure (path, tour, tree and arc upgrading) that serve discrete points of demand represented as nodes of a network. Other problems can be derived and defined within this framework, many of which have not yet been explored in any formal manner. That is, they remain open research problems in terms of how best to structure and solve in support of planning and management decision making.

The models that have been formulated in this chapter for optimizing maximal covering paths and maximal covering tours have been based upon an assumption that optimal paths or routes never backtrack or cross over themselves to form cycles/loops. This assumption has been made explicit in Curtin and Biba (2011) as they ensure that a covering path never crosses over itself by constraining the path so that it enters a node no more than once. The use of sub-tour elimination constraints in the covering path models of Current et al. (1984, 1985) also imposes that loops will not occur for feasible solutions. However, Niblett (2016) and Niblett and Church (2016) have questioned such an assumption, highlighting that the shortest covering path problem of Current et al. (1984) may result in an infeasible solution, when in fact a feasible solution exists, or even result in finding a sub-optimal solution. That is, there are instances where the optimal solution in fact requires a cycle to achieve maximum coverage. Niblett and Church (2016) have presented a new, revised formulation for the shortest covering path problem capable of identifying the true optimal solution,

whether it involves a cycle or not. They have also shown how to reformulate the node-arc path covering model of Curtin and Biba (2011) so that it can find optimal solutions when optimal solutions may involve an associated cycle.

Our emphasis in this chapter has been to describe the fundamental problems that have been developed to date associated with the location of a network structure, like a path, tour or tree that is designed to cover some of all of the nodes of the network. We have also presented integer-linear programming formulations for these fundamental models. It should be noted heuristics and optimal algorithms have been developed for some of these problems, and remains an active research area.

# References

Arkin EM, Hassin R (1994) Approximation algorithms for the geometric covering salesman problem. Discret Appl Math 55:197–218

Arkin EM, Haldorsson NM, Hassin R (1993) Approximating tree and tour covers of a graph. Inf Process Lett 47:275–282

Boffey B, Narula SC (1998) Models for multi-path covering-routing problems. Ann Oper Res 82:331–342

Church R, Current J (1993) Maximal covering tree problems. Nav Res Logist 40:129–142

Church R, Noronha, V, Lei T, Corrigan W, Burbidge S, Marston J (2005) Spatial and temporal utility modeling to increase transit ridership: final report to Caltrans. VITAL, University of California, Santa Barbara, CA

Cohon JL (1978) Multiobjective programming and planning. Academic, New York

Current JR (1981) Multiobjective design of transportation networks. PhD dissertation, The Johns Hopkins University, Baltimore, MD

Current JR, Schilling DA (1989) The covering salesman problem. Transp Sci 23:208–213

Current JR, Schilling DA (1994) The median tour and maximal covering tour problems: formulations and heuristics. Eur J Oper Res 73:114–126

Current J, ReVelle C, Cohon J (1984) The shortest covering path problem: an application of locational constraints to network design. J Reg Sci 24:161–184

Current JR, ReVelle CS, Cohon JL (1985) The maximum covering/shortest path problem: a multiobjective network design and routing formulation. Eur J Oper Res 21:189–199

Current J, Pirkul H, Rolland E (1994) Efficient algorithms for solving the shortest covering path problem. Transp Sci 28(4):317–327

Curtin KM, Biba S (2011) The transit route arc-node service maximization problem. Eur J Oper Res 208:46–56

Dantzig G, Fulkerson R, Johnson S (1954) Solution of a large-scale traveling-salesman problem. J Oper Res Soc Am 2:393–410

Daskin MS (1995) Network and discrete location: models, algorithms, and applications. Wiley, New York

Finke GA, Claus A, Gunn E (1983) A two commodity network flow approach to the travelling salesman problem. In Proceedings of the 14th South Eastern conference on combinatorics, graph theory and computing, Atlantic University, FL

Fischetti M, Toth P (1988) An additive approach for the optimal solution the prize-collecting salesman problem. In: Golden BL, Assad AA (eds) Vehicle routing: methods and studies. North Holland, Amsterdam

Gavish B, Graves SC (1978) The travelling salesman problem and related problems. Working Paper OR-078-78. Operations Research Center, MIT Press, Cambridge, MA

Gendreau M, Laporte G, Semet F (1997) The covering tour problem. Oper Res 45:568–576

Golden B, Naji-Azimi Z, Raghavan S, Salari M, Toth P (2012) The generalized covering salesman problem. INFORMS J Comput 24:534–553

Hakimi SL (1965) Optimum distribution of switching centers in a communication network and some related graph theoretic problems. Oper Res 13(3):462–475

Handler GY, Mirchandani PB (1979) Location on networks: theory and algorithms. MIT Press, Cambridge, MA

Hutson VA, ReVelle C (1989) Maximal direct covering tree problems. Transp Sci 23:288–289

Hutson VA, ReVelle C (1993) Indirect covering tree problems on spanning tree networks. Eur J Oper Res 65:20–32

Kruskal JB Jr (1956) On the shortest spanning subtree of a graph and the travelling salesman problem. Proc Am Math Soc 7:48–50

Laporte G, Martello S (1990) The selective travelling salesman problem. Discret Appl Math 26:193–207

Matisziw TC, Murray AT, Kim C (2006) Strategic route extension in transit networks. Eur J Oper Res 171:661–673

Medrano FA, Church RL (2014) Corridor location for infrastructure development: a fast bi-objective shortest path method for approximating the Pareto frontier. Int Reg Sci Rev 37(2):129–148

Medrano FA, Church RL (2015) A parallel computing framework for finding the supported solutions to a biobjective network optimization problem. J Multicrit Decis Anal 22(5–6):244–259

Miller CE, Tucker AW, Zemlin RA (1960) Integer programming formulation of travelling salesman problems. J ACM 3:326–329

Morgan CA, Slater PJ (1980) A linear algorithm for a core of a tree. J Algorithm 1:247–258

Murawski L, Church RL (2009) Improving accessibility to rural health services: the maximal covering network improvement problem. Socio Econ Plan Sci 43:102–110

Murray AT, Wu X (2003) Accessibility tradeoffs in public transit planning. J Geogr Syst 5:93–107

Niblett TJ (2016) On the development of a new class of covering-path models. PhD dissertation, UCSB, Santa Barbara, CA

Niblett TJ, Church RL (2016) The shortest covering path problem a new perspective and model. Int Reg Sci Rev 39:131–151

Oppong JR (1996) Accommodating the rainy season in third world location-allocation applications. Socio Econ Plan Sci 30:121–137

Orman AJ, Williams HP (2004) A survey of different integer programming formulations of the travelling salesman problem. Working Paper No. LSEOR 04.67. London School of Economics and Political Science, London

ReVelle CS, Laporte G (1993) New directions in plant location. Stud Locational Anal 5:32–58

Slater PJ (1981) On locating a facility to service areas within a network. Oper Res 29:523–531

Slater PJ (1982) Locating central paths in a graph. Transp Sci 16:1–18

Slater PJ (1983) Some definitions of central structures. Lect Notes Math 1073:169–178

Vajda S (1961) Mathematical programming. Addison-Wesley, London

Wu C, Murray AT (2005) Optimizing public transit quality and system access: the multiple-route, maximal covering/shortest-path problem. Environ Plann B Plann Des 32:163–178