Composite Regions of Feasibility for Certain Classes of Distance Constrained Network Location Problems

BARBAROS Ç. TANSEL

Department of Industrial Engineering, Bilkent University, Bilkent 06533 Ankara, Turkey

GÜLCAN N. YEŞİLKÖKÇEN

Michael DeGroote School of Business, McMaster University, Hamilton, Ontario, L8S 4M4 Canada

Distance constrained network location involves locating \( m \) new facilities on a transport network \( G \) so as to satisfy upper bounds on distances between pairs of new facilities and pairs of new and existing facilities. The problem is \( \mathcal{NP} \)-complete in general, but polynomially solvable for certain classes. While it is possible to give a consistency characterization for these classes, it does not seem possible to give a global description of the feasible set. However, substantial geometrical insights can be obtained on the feasible set by studying its projections onto the network. The \( j \)-th projection defines the \( j \)-th composite region which is the set of all points in \( G \) at which new facility \( j \) can be feasibly placed without violating consistency. We give efficient methods to construct these regions for solvable classes without having to know the feasible set and discuss implications on consistency characterization, what if analysis, and recursive solution constructions.

The location problem studied in this paper involves locating several new facilities on a network, such as a transport network, so as to satisfy upper bounds on distances between pairs of new and existing facilities and pairs of new facilities. The existing facilities (demand points) are at the nodes of the network. The new facilities can be located anywhere on the network including nodes and interiors of edges. If a distance bound is imposed on a pair of facilities, those facilities are said to interact. Not all facility pairs need to interact, but those that do must be placed so as not to violate the imposed upper bounds. Such constraints are relevant in a wide range of location problems when service quality becomes unacceptable beyond certain critical distances. For example, it is appropriate that emergency service facilities be within a critical driving time of potential demand sites to avoid fatalities, damage to human life, or excessive property losses. Service units with distinguishable but complementary service characteristics (e.g. ambulances, hospitals, fire stations) are expected to be not too far from one another. In military contexts, response units may be required to be within reasonable distances from each other as well as from their supply bases. Distance constraints may also be appropriate in manufacturing to avoid excessive delays, inventory buildup, and scheduling difficulties that may arise from large material handling distances between machining centers. In telecommunication networks, it is often necessary to place switching stations or repeaters within technologically defined distances to receive, store, and reroute information. Other motivating examples can also be found in the relevant literature (e.g. Francis, Lowe, and Ratliff (1978), Tansel, Francis, and Tamir (1980, 1982), Erkut, Francis, and Tamir (1992), Tansel and Yeşilköçen (1993)). Also, the solution of distance con...
constraints is of direct utility in the analysis of minimax location problems. For further information on network location, the reader may consult TANSEL, FRANCIS, and LOWE (1983a, 1983b), BRANDEAU and CHIU (1989).

The problem is \( \mathcal{NP} \)-complete in general (KOLEN, 1986). Polynomial time solvable cases are in two classes: (C1) the transport network (location space) is a tree network with arbitrary interactions between facilities (FRANCIS, LOWE, and RATLIFF, 1978); (C2) the transport network is arbitrary while new facility interactions induce a tree structure (TANSEL and YESILKOKCEN, 1993). Similarly structured optimization forms have also been solved efficiently (CHAHJED and LOWE, 1991 and 1992) when new facility locations are restricted to nodes and new facility interactions induce a series-parallel graph or a \( k \)-tree structure, but these do not relate to our work directly.

Our focus is on classes (C1) and (C2). We use the existing theory and algorithms to derive properties of the solution set. In particular, we define the notion of composite region of feasibility for each new facility and give methods to construct these regions. The \( j \)-th region identifies the set of all points in the network at which new facility \( j \) can be feasibly placed so as to allow a feasible placement of all remaining new facilities. These regions provide geometric insights, lead to recursive solution methods, and have potential applications in sensitivity analysis.

The problem of how to construct these regions has not been addressed in the location literature except for the single facility case. For that special case, there is only one region to be constructed, which is the composite neighborhood discussed in FRANCIS, LOWE, and RATLIFF (1978) for trees and extended recently in TANSEL and YESILKOKCEN (1993) to general networks.

In the multifacility case, the \( j \)-th composite region corresponds to the projection of the feasible set onto the network in the \( j \)-th coordinate. In this sense, the definition is a conceptually good construct but not an operational one computationally (unless we already know the feasible set in which case there would be little or no need to worry about its projections). There are algorithms in the existing location literature that construct solutions on a need basis (SLP of FRANCIS, LOWE, and RATLIFF, 1978, and SEIP of TANSEL and YESILKOKCEN, 1993), but such algorithmic constructions cannot generate all elements of the feasible set since the set is uncountably infinite in general. The only remaining possibility seems to be to construct the projections without having to know the feasible set so as to obtain insights on the global structure of all solutions. Our primary focus in the paper is to develop computationally efficient procedures that achieve this objective.

Now we give an overview of the paper. In Section 1, we provide definitions and problem statement. In Section 2, we introduce the notion of composite regions. In Sections 3–6, we focus on the construction of composite regions. Section 3 considers the case where the location space is a tree and the structure of the interaction between new facilities is arbitrary. Sections 4–6 consider the case where the location space is a general (cyclic) network and the structure of the interaction between new facilities is a tree. Analysis in Section 3 basically relies on separation conditions of FRANCIS, LOWE, and RATLIFF (1978), and analysis in Sections 4–6 relies on expand/intersect method of TANSEL and YESILKOKCEN (1993). Finally, we conclude the paper in Section 7 with a brief summary of the results.

1. DEFINITIONS, PROBLEM STATEMENT

Suppose we are given \( G \), an embedded undirected connected network having positive edge lengths. A point \( x \in G \) is either a node or an interior point of some embedded edge. Let \( V = \{v_1, \ldots, v_n\} \) be the node set of \( n \) distinct nodes. For any two points \( x, y \in G \), the distance \( d(x, y) \) is the length of a shortest path connecting \( x \) and \( y \). \( d \) satisfies the properties of nonnegativity, symmetry, and triangle inequality and \( G \) with distance \( d \) is a metric space. If \( G \) is a tree, we write \( T \) instead of \( G \).

The existing facilities are at nodes \( v_1, \ldots, v_n \) and \( m \) new facilities are to be located at points \( x_1, \ldots, x_m \in G \). Let \( I_C, I_B \) be given sets of index pairs for which distance bounds are of interest. The distance constraints (DC) are as follows:

\[
\begin{align*}
\text{DC.1:} & \quad d(x_j, x_k) \leq b_{jk}, \quad (j, k) \in I_B \\
\text{DC.2:} & \quad d(x_j, v_i) \leq c_{ji}, \quad (j, i) \in I_C
\end{align*}
\]

Note that \( I_B \subseteq \{(j, k) : 1 \leq j < k \leq m\} \) and \( I_C \subseteq \{(j, i) : 1 \leq j \leq m, 1 \leq i \leq n\} \) with \( c_{ji}, b_{jk} \) finite positive constants for the given index pairs.

We represent the data of the problem by forming an auxiliary network, called LN (Linkage Network), with node set \( \{N_1, \ldots, N_n\} \cup \{E_1, \ldots, E_m\} \) and edge set \( A_B \cup A_C \) where \( A_B = \{(N_j, N_k) : (j, k) \in I_B\} \) and \( A_C = \{(N_j, E_i) : (j, i) \in I_C\} \). Edges \( (N_j, N_k) \in A_B \) have lengths \( b_{jk} \) and edges \( (N_j, E_i) \in A_C \) have lengths \( c_{ji} \). Let \( LN_B \) be the subgraph of LN consisting of nodes \( N_1, \ldots, N_m \) and edges in \( A_B \). We assume LN and \( LN_B \) are both connected, otherwise the problem decomposes into independent subproblems corresponding to components.
DC is said to be consistent if there is at least one 
\((x_1, \ldots, x_m)\) that satisfies (DC.1) and (DC.2). Earlier work focused on characterization of consistency and construction of a feasible solution for classes (C1) and (C2). Note that (C1) is identified with \(G\) being a tree \(T\) and \(LN_B\) arbitrary while (C2) is identified with \(LN_B\) being a tree and \(G\) arbitrary. In both cases, no assumptions are made on \(A_c\).

Define \(G^m = \{(x_1, \ldots, x_m) : x_j \in G, j = 1, \ldots, m\}\), the \(m\)-fold Cartesian product of \(G\) with itself and let \(N(x, r) = \{y \in G : d(x, y) \leq r\}\) for any point \(x\) in \(G\) and \(r \geq 0\). \(N(x, r)\) is the neighborhood of \(x\) with radius \(r\).

2. COMPOSITE REGIONS

The idea behind the notion of composite regions is to identify the set of all alternate locations in the network at which a given new facility can be feasibly placed. For the case of a single facility, the notion coincides with that of the feasible set.

In the multifacility case, the notion coincides with projections of the feasible set (which is a subset of \(G^m\)) onto \(G\). The projections can be displayed on \(G\) and provide good geometric insights on the feasible set that may not be revealed by algebraic description alone.

We now define the notion. Let \(F\) be the set of \(X = (x_1, \ldots, x_m)\) in \(G^m\) that satisfy (DC.1) and (DC.2). \(F\) is called the feasible set. For \(j \in J = \{1, \ldots, m\}\), define the set

\[ L_j = \{y \in G : \exists X = (x_1, \ldots, x_m) \text{ in } F \text{ such that } x_j = y\}. \]

We call \(L_j\) the composite region for new facility \(j\). \(L_j\) consists of \(j\)-th components of all feasible location vectors.

In the sequel, we give methods to construct the composite regions \(L_1, \ldots, L_m\). This has a number of important consequences.

First, observe that either \(F, L_1, \ldots, L_m\) are all nonempty or all are empty. This allows to resolve the consistency question in the following way: compute (somehow), say, \(L_1\). \(DC\) is consistent if and only if \(L_1\) is nonempty. Hence, if \(L_1\) (or any other \(L_j\)) is efficiently computable, then a yes or no answer is available to the recognition problem DC.

Second, observe that every point \(y\) in \(L_1\) is a feasible choice for new facility \(j\) since the definition implies there exists a vector \(X\) in \(F\) whose \(j\)-th component is equal to \(y\). In this sense, \(L_j\) specifies the set of all alternate locations in the network at which new facility \(j\) can be placed without causing a violation in \(DC\). This has direct use in what if analysis. If a feasible location vector \(X\) is found to be inappropriate later due to factors not considered initially, then its components may be moved around in their composite regions to obtain a new feasible solution that is admissible. Some care is required in doing this since moving a facility to a new location in its composite region affects the composite regions of other ones conditional on the fixed location of the moved new facility. Nevertheless, knowing \(L_1, \ldots, L_m\) gives significant flexibility in choosing alternate locations.

A third important consequence is the fact that knowing a composite region gives the ability to construct a feasible vector recursively. To illustrate, suppose \(L_1\) is computed. Place new facility 1 at an arbitrary point \(y\) in \(L_1\) and change its status to an existing facility. The resulting \(DC\) has now \(m - 1\) unknowns and \(n + 1\) fixed locations. We may construct \(L_2\) with respect to the reduced system and fix the location of \(x_2\) in its composite region conditional on \(x_1\). Continuing in this way, this gives a procedure that eliminates new facilities one at a time from \(DC\) and changing their status to existing facilities in subsequent steps.

Apart from these considerations, the composite regions are important because their availability allows to construct as many feasible vectors as desired by using the recursion idea described above. Hence, even if \(F\) cannot be fully described algebraically, as many feasible location vectors can be generated as desired when the composite regions are available.

Lastly, the availability of \(L_1, \ldots, L_m\) may be useful for solving optimization problems over \(F\). For example, distance constrained multicenter and multimedian problems require optimization over \(F\). The theory of the composite regions may lead to algorithms that solve these problems.

With these motivating considerations, we now focus on the computation of composite regions for classes (C1) and (C2).

3. TREE NETWORKS, ARBITRARY INTERACTIONS

In this section we consider class (C1). We assume \(G\) is a tree \(T\). No assumptions are made on the linkage network \(LN\) other than connectivity. To compute the composite regions, we will use the Separation Conditions of FRANCIS, LOWE, and RATTLIFF (1978) which are known to be necessary and sufficient for consistency of \(DC\). First, we state these conditions.

Let \(d(E_j, E_k)\) be the length of a shortest path in \(LN\) connecting nodes \(E_j\) and \(E_k\), \(1 \leq j < k \leq n\). The
Separation Conditions are the $n(n - 1)/2$ inequalities
\[ d(v_j, v_k) < \mathcal{L}(E_j, E_k), \quad 1 \leq j < k \leq n. \]

$DC$ is consistent if and only if the Separation Conditions hold (Francis, Lowe, and Ratliff, 1978).

Define $r_{ji} = \mathcal{L}(N_j, E_i)$ to be the length of a shortest path in $LN$ connecting nodes $N_j$ and $E_i$, $1 \leq j \leq m, 1 \leq i \leq n$.

The next theorem identifies each composite region as the intersection of neighborhoods centered at nodes.

**Theorem 3.1.** If separation conditions hold, then
\[ L_j = \bigcap_{i=1}^{n} N(v_i, r_{ji}) \neq \emptyset \text{ for } j = 1, \ldots, m. \]
Otherwise, $L_j = \emptyset \forall j$.

The otherwise part of the theorem is a direct consequence of the fact that violation of separation conditions implies $F = \emptyset$ which implies $L_j = \emptyset \forall j$.

The proof of the nontrivial part is a consequence of Properties 3.1 and 3.2 which we give next. Property 3.1 gives necessary conditions for a point to belong to a composite region.

**Property 3.1.** For any $j \in \{1, \ldots, m\}$, if $y \in L_j$ then $y \in \bigcap_{i=1}^{n} N(v_i, r_{ji})$. □

The property is a direct consequence of the fact that there exists a feasible solution $\bar{X} = (\bar{x}_1, \ldots, \bar{x}_m)$ to $(DC)$ with $\bar{x}_j = y$ so that repeated use of the triangle inequality and aggregation of constraints along a shortest path between $N_j$ and $E_i$ gives $d(\bar{x}_j, v_i) \leq r_{ji}$ for each $i$. We omit the details.

**Remark 3.1.** The property holds for general networks as well as other metric spaces since triangle inequality is the only essential feature needed in the proof. Hence, necessity is true for all metric spaces.

The next property gives the sufficient conditions for a point to belong to a composite region.

**Property 3.2.** Assume separation conditions hold. For any $q \in \{1, \ldots, m\}$, if $y \in \bigcap_{i=1}^{n} N(v_i, r_{qi})$ then $y \in L_q$.

**Proof.** Let $q \in \{1, \ldots, m\}$ and $y \in \bigcap_{i=1}^{n} N(v_i, r_{qi})$. To show $y \in L_q$, we will construct a location vector $\bar{X} = (\bar{x}_1, \ldots, \bar{x}_m)$ such that $\bar{x}_q = y$ and $\bar{X} \in F$. Fix the location of new facility $q$ at $y$ and rewrite the distance constraints in the following form with $x_q = y$ separated from the rest of the variables (put $b_{jq} = b_{jq}$ for $q < j$):
\[ d(x_j, x_k) \leq b_{jk}, \quad (j, k) \in I_B, \quad j \neq q \]
\[ d(x_j, v_i) \leq c_{ji}, \quad (j, i) \in I_C, \quad j \neq q \]

First we show that (4) is satisfied, then we show there is a feasible solution to (1)–(3) in the variables $x_j, j \neq q, j \in \{1, \ldots, m\}$.

To show (4) is satisfied, observe that $y \in \bigcap_{i=1}^{n} N(v_i, r_{qi})$ implies $d(y, v_i) \leq r_{qi}, i = 1, \ldots, n$. But $r_{qi} \leq c_{qi}$ since $r_{qi}$ is the shortest path length between $N_q$ and $E_i$ (if $(q, i) \notin I_C$ then $c_{qi}$ can be taken as $\infty$). Hence, (4) is satisfied.

Let $DC$ be the distance constraints (1)–(3). Observe that with $y$ being a fixed location we may take new facility $q$ as an existing facility. Let $LN$ be the linkage network corresponding to $DC$ obtained from $LN$ by declaring $N_q$ as an $E$-node (say, $n + 1$st $E$-node) and deleting all edges of the form $(N_q, E_i)$ from $A_C$. All remaining edges still have their old lengths. This modification of $LN$ clearly produces the correct $LN$ corresponding to $DC$. Consider now the separation conditions corresponding to $DC$. With $\mathcal{L}(F_s, F_t)$ denoting the shortest path length between any two nodes $F_s, F_t$ of $LN$, the separation conditions for $DC$ (with $N_q$ being the $n + 1$st $E$-node) are:
\[ d(v_j, v_k) \leq \mathcal{L}(E_j, E_k), \quad 1 \leq j < k \leq n \]
\[ d(v_j, y) \leq \mathcal{L}(E_j, N_q), \quad 1 \leq j \leq n. \]

If we show (5) and (6) are satisfied, then $\overline{DC}$ is consistent. Observe that $\mathcal{L}(F_s, F_t) \leq \mathcal{L}(F_s, F_q)$ for any two nodes $F_s, F_t$ in $LN$ since $LN$ is identical to $LN$ except some edges have been removed (so that every path in $LN$ is also in $LN$). By assumption, separation conditions for $DC$ are satisfied so that $d(v_j, v_k) \leq \mathcal{L}(E_j, E_k) \leq \mathcal{L}(E_j, E_q), 1 \leq j < k \leq n$. Hence, (5) is satisfied. Furthermore, $y \in \bigcap_{i=1}^{n} N(v_i, r_{qi})$ implies $d(y, v_i) \leq r_{qi} = \mathcal{L}(N_q, E_i) \leq \mathcal{L}(N_q, E_i)$ for $1 \leq i \leq n$. Hence, (6) is also satisfied. It follows that there exists a feasible solution $\bar{x}_j, j \neq q, j \in \{1, \ldots, m\}$ to $\overline{DC}$ so that inserting $\bar{x}_q = y$ in the $q$-th position of this vector gives a feasible solution $\bar{X} = (\bar{x}_1, \ldots, \bar{x}_m)$ which satisfies (1)–(4). Hence, $\bar{X} \in F, \bar{x}_q = y$ so that $y \in L_q$. □

Observe that the proof uses the separation conditions to conclude that the reduced system $\overline{DC}$ is consistent. Hence, the property is true for tree networks as well as the cases with Tchebychev distance in $R^k$ ($k \geq 2$) and rectilinear distance in $R^2$. Separation conditions are necessary and sufficient for consistency of $DC$ in all of these cases (Francis, Lowe, and Ratliff, 1978).

Theorem 3.1 is justified now. Property 3.1 implies $L_j \subseteq \bigcap_{i=1}^{n} N(v_i, r_{ji}), j = 1, \ldots, m$ while Property 3.2 implies $\bigcap_{i=1}^{n} N(v_i, r_{ji}) \subseteq L_j$ for $j = 1, \ldots, m$. 

\[ d(x_j, y) \leq b_{jq}, \quad (j, q) \in I_B, \quad j \neq q \]
\[ d(y, v_i) \leq c_{qi}, \quad (q, i) \in I_C. \]
under the assumption separation conditions hold. It follows that, if separation conditions hold, then $L_j = \bigcap_{i=1}^n N(u_i, r_{ji})$, $j = 1, \ldots, m$.

Computation of the values $r_{ji}$ can be done in $O(m(m + n)^2)$ time by applying Dijkstra’s shortest path algorithm on $LN$ once for each new facility node $N_j$, $1 \leq j \leq m$. Once all $r_{ji}$, $1 \leq j \leq m$, $1 \leq i \leq n$ are computed, we can use Sequential Intersection Procedure (SIP) of Francis, Lowe, and Ratliff (1978) to compute each composite region $L_j$ in $O(n^2)$ time, giving a total effort of $O(mn^2)$. It is also possible to use a modified version of the Sequential Location Procedure (SLP) of Francis, Lowe, and Ratliff (1978), with $m = 1$, to reduce the time bound of constructing one composite region to $O(n)$, but we find the details of this modification tangential to the main development of this paper and omit them. Thus, computing the values $r_{ji}$ dominates the effort to construct the composite regions.

In Figure 1, we provide an example to illustrate the composite regions. Square nodes represent the three new facilities and circle nodes represent the six existing facilities in the linkage network. The numbers next to edges in $LN$ give the bounds $b_{jk}$ and $c_{ji}$ on the separation of facility pairs. The appropriate radii are given in the matrix $R$ in the figure. (a) shows the feasible regions $S_j = \bigcap \{N(u_i, c_{ji}) : i$ such that $(j, i) \in I_G\}$ of new facilities with respect to existing facilities alone (i.e. the bounds on the distances between new facility pairs are relaxed). (b) shows the composite regions of feasibility for all new facilities. In constructing the sets $S_j$ and $L_j$, $1 \leq j \leq m$, we used SIP. The reader can verify the given sets by constructing the neighborhoods around all nodes by moving $c_{ji}$ (or $r_{ji}$) units from node $u_i$ in all possible directions and finding the intersection of all neighborhoods for a given new facility.

4. GENERAL NETWORKS, TREE TYPE INTERACTIONS

We now focus on class (C2). No assumptions are made on $G$ (other than it be connected with no parallel edges and no self loops). We assume $LN_B$ is a tree network after all redundant edges (corresponding distance constraints) have been eliminated from $LN$ (from DC). An edge $(F_p, F_q)$ in $LN$ is redundant if its deletion from the edge set does not increase the shortest path length $L(F_p, F_q)$ and does not disconnect $LN$. Constraints corresponding to redundant edges can be deleted from DC without changing the feasible set (Francis, Lowe, and Ratliff, 1978). This is justified by repeated use of the triangle inequality and is true for any metric space.

Even though we present our analysis in the context of embedded networks, everything we say in this section except the complexity discussion is also true for an arbitrary metric space with a well defined distance. Hence, $G$ may be taken as any metric space with distance $d$.

Our method of computing the composite sets is based on the notions of expansion and intersection defined in Tansel and Yesilkokcen (1993). First we give the necessary definitions. For any nonempty subset $S$ of $G$ and $b \geq 0$, define

$N(S, b) = \{x \in G : \exists y \in S$ such that $d(x, y) \leq b\}$.

We call $N(S, b)$ the expansion of $S$ by $b$. It includes all points of $G$ that are reachable from at least one point of $S$ within $b$ distance units. An equivalent definition is $N(S, b) = \bigcup_{y \in S} N(y, b)$. For example, if $S$ is the interval $[0, 1]$ in $\mathbb{R}$, its expansion by $b$ is the interval $[-b, 1 + b]$ in $\mathbb{R}$.
Associated with each new facility \( j \) \((j = 1, \ldots, m)\), define \( S_j = \cap_{i \in I_j} N(v_i, c_{ji}) \) where \( I_j \) is the set of existing facility indices \( i \in \{1, \ldots, n\} \) for which \((j, i) \in I_c\). If \( I_j \) is empty, take \( S_j = G \). An equivalent statement of \( DC \) is as follows:

\[
d(x_j, x_k) \leq b_{jk}, \quad (j, k) \in I_B \quad \text{(DC.1)}
\]

\[
x_j \in S_j, \quad j = 1, \ldots, m. \quad \text{(DC.2')}
\]

We now give an algorithm to compute the composite regions. We call the algorithm \( \text{SEIP-CR} \) \((\text{Sequential Expand/Intersect Procedure-Composite Region})\). The algorithm takes the sets \( S_1, \ldots, S_m \) as input and works directly with \( LN_B \) one edge at a time. Phase 1 constructs the composite region for the root node which is, by definition, the last node processed at the end of Phase 1. Although composite regions for other nodes can be obtained by repeated use of Phase 1 with different root nodes, Phase 2 more efficiently constructs the composite regions for all nodes beginning with the root node. Phase 2 is initiated only if the composite region for the root node is found to be nonempty. Otherwise, \( DC \) is inconsistent and all composite regions are null. We note that the first phase of the algorithm \( \text{SEIP-CR} \) is an equivalent statement of the first phase of \( \text{SEIP} \) \((\text{Sequential Expand/Intersect Procedure})\) given in TANSEL and YESILKOKCEN \((1993)\).

In the algorithm, the green tree is the subtree that spans all green colored nodes. There is a brown subtree rooted at every tip node of the green tree that is a maximal subtree that spans brown colored nodes and that tip node.

**SEIP-CR**

**Phase 1** \((\text{Input}: S_1, \ldots, S_m, LN_B \text{ with edge lengths } b_{jk}, (N_j, N_k) \in A_B. \text{ Define } b_{jk} = b_{kj} \forall j > k.)\)

**Initial:** Color all nodes of \( LN_B \) green. Define \( F_j = S_j \forall j \). \( F_j \) is the set associated with node \( N_j \) \((j = 1, \ldots, m)\).

(1) Choose a tip node \( N_t \) of the green tree and let \( N_{a(t)} \) be the unique green colored node adjacent to it.

(2) Construct the expansion \( N(F_t, b_{t,a(t)}) \), then construct the intersection \( N(F_t, b_{t,a(t)}) \cap F_{a(t)} \). Assign \( F_{a(t)} \leftarrow N(F_t, b_{t,a(t)}) \cap F_{a(t)} \).

(3) If \( F_{a(t)} \) is null, go to infeasible termination. Otherwise, color \( N_t \) brown. If exactly one green colored node remains (which is \( N_{a(t)} \)), go to feasible termination, else return to (1).

**Infeasible Termination:** Terminate with \( \bar{L}_j = \emptyset \forall j \). \( DC \) is inconsistent.

**Feasible Termination:** Save the index of the last green colored node. Let \( r \) be this index. Go to Phase 2 with output sets \( F_1, \ldots, F_m \).

**Phase 2** \((\text{Input}: F_1, \ldots, F_m \text{ all nonempty, } N_r \text{ is green colored.})\)

**Initial:** Assign \( \bar{L}_j = F_j \forall j \).

(1) Choose any brown colored node adjacent to a green colored node. Let \( N_t \) be the brown colored node chosen and let \( N_{a(t)} \) be the unique green colored node adjacent to it.

(2) Construct the expansion \( N(F_{a(t)}, b_{t,a(t)}) \), then construct the intersection \( N(F_{a(t)}, b_{t,a(t)}) \cap \bar{L}_t \). Assign \( \bar{L}_t \leftarrow N(F_{a(t)}, b_{t,a(t)}) \cap \bar{L}_t \).

(3) Color \( N_{a(t)} \) green. If no brown colored node remains, go to termination. Otherwise, return to (1).

**Termination:** Output \( \bar{L}_1, \ldots, \bar{L}_m \).

The next theorem asserts that the output sets \( \bar{L}_j, j = 1, \ldots, m \) are in fact the composite regions.

**THEOREM 4.1.** Let \( \bar{L}_1, \ldots, \bar{L}_m \) be the output sets from the algorithm \( \text{SEIP-CR} \). Then \( L_j = \bar{L}_j, j = 1, \ldots, m \).

The proof of the theorem will be given in Section 6. First we demonstrate the procedure via an example. Consider the example \( LN_B \) in Figure 2 with six nodes. Initially all nodes are green and \( F_j = S_j, j = 1, \ldots, 6 \). A legal sequence of coloring nodes brown is \( N_1, N_2, N_3, N_4, N_5 \) which leaves the root node \( N_6 \) which remains green colored at the end of Phase 1. Figure 2 gives the constructed sets in each iteration. Some commenting on the complexity of the algorithm is in order. Clearly, both phases perform the expand/intersect operation \( O(m) \) times. The amount
of work done per operation depends on the metric space under consideration. For general embedded networks \( G \), it is shown in TANSEL and YEŞILKÖKÇEN (1993) that each input set \( S_j \) is in general a disconnected set consisting of up to \( n + 1 \) segments per edge and \( O(|E|n) \) disjoint parts on the entire network. The expand/intersect operation can be performed on each edge of \( G \) separately. An expansion operation on a given edge can increase the number of segments of the input set by at most two. Intersecting an expanded set with another set produces a new set whose number of segments is at most the total number of segments in both sets less one. With these considerations, TANSEL and YEŞILKÖKÇEN (1993) gives a detailed algorithm for Phase 1 whose time bound is \( O(|E|mn(m + n)) \). Since Phase 2 operations are essentially the same as Phase 1 operations in post order, it is direct to show that Phase 2 complexity is bounded by the same order. Hence, SEIP-CR is an \( O(|E|mn(m + n)) \) algorithm for constructing composite regions \( L_1, \ldots, L_m \) on general networks.

Next we provide an example of SEIP-CR applied on a network.

Consider the example network \( G \) shown in Figure 3. The numbers next to edges are the edge lengths and the distance matrix is given. The distance bounds \( c_{ij} \) and \( b_{jk} \) are given in the matrices \( C \) and \( B \) in the same figure. This data defines the linkage network \( LN \) and its subgraph \( LNB \). The sets \( S_1, S_2, S_3 \) shown in (1), (2), and (3), respectively, represent the feasible regions of each new facility with respect to existing facilities alone. Phase 1 processes nodes of \( LNB \) in the order 1-2-3 (node 3 is the root) leaving node 3 green colored at the end. That is, the expansion \( N(F_1, b_{12}) \) is constructed first (see (4) in Fig. 3), then intersected with \( F_2 \) which is initially equal to \( S_2 \) (see (5) in Fig. 3) and node \( N_1 \) is colored brown in \( LN_B \). Next, the expansion \( N(F_2, b_{23}) \) is constructed and intersected with \( F_3 \) (see (6)–(7) in Fig. 3) after which node \( N_2 \) is colored brown in \( LN_B \).

Once, \( F_1, F_2, F_3 \) are available, Phase 2 begins by initiating \( F_j = L_j, 1 \leq j \leq m \). Then similar expand intersect operations are performed in the order 3-2-1 of new facility nodes in \( LNB \) (see (8)–(11) in Fig. 3). We also give a feasible solution shown in (12) of Fig. 3.

5. PROPERTIES OF OUTPUT SETS FROM PHASE 1

In this section we prove a theorem which reveals an interesting feature of the expand/intersect procedure: that it constructs composite regions for relaxations of DC corresponding to brown subtrees that arise in Phase 1. An important consequence of this is the fact that the output set \( F_1 \) is the same as the composite region \( L_r \) for the root node, a key result which we use to justify SEIP-CR.

**Theorem 5.1.** During some iteration of Phase 1, let \( N_r \) be the tip node selected of the current green tree, \( B_t \) be the brown subtree rooted at \( N_r \), and DC\(_j\) be the distance constraints

\[
d(x_j, x_k) \leq b_{jk}, \quad (N_j, N_k) \text{ is an edge in } B_t
\]

\[
x_j \in S_j, \quad N_j \text{ is a node in } B_t.
\]

Denote by \( L_t(\text{DC}_j) \) the composite region for new facility \( t \) with respect to \( \text{DC}_j \). Then

\[
L_t(\text{DC}_j) = F_t
\]

where \( F_t \) is the output set computed for new facility \( t \) in Phase 1.

**Proof.** Let \( k \) be the iteration index. We use induction on \( k \). \( N_t \) is the node selected in iteration \( k \).

For \( k = 1 \), \( N_t \) is the only node in \( B_t \) so \( DC_t \) consists of one constraint: \( x_t \in S_t \). For this constraint the solution set is \( S_t \) so that \( L_t(\text{DC}_t) = S_t \). Note also that \( S_t = F_t \) due to initialization in Phase 1.

Assume now the theorem holds for nodes selected in iterations 1, \ldots, \( k - 1 \) (\( k > 1 \)). We must show that \( N_t \), the node selected in iteration \( k \), satisfies \( L_t(\text{DC}_t) = F_t \).

Let \( R \) be the set of indices of nodes in \( B_t \) that are adjacent to \( N_t \). If \( R = \emptyset \), then \( N_t \) is the only node in \( B_t \) so the justification given for \( k = 1 \) is also valid here. Assume now \( R \neq \emptyset \). All nodes in \( R \) are already brown colored in iterations earlier than \( k \) so that the induction assumption gives

\[
L_t(\text{DC}_j) = F_j \quad \forall j \in R
\]

where \( DC_j \) refers to the constraints corresponding to the brown subtree \( B_t \) that was rooted at \( N_t \) in some earlier iteration. Since \( B_t \) is the union of (disjoint) subtrees \( B_{j', j} \in R \), with the additionally appended node \( N_t \) and edges \( (N_t, N_j), j \in R \), we may rewrite \( DC_t \) in partitioned form as follows:

\[
d(x_t, x_j) \leq b_{tj}, \quad j \in R,
\]

\[
x_t \in S_t
\]

\[
DC_{j'}, j \in R.
\]

To show \( L_t(\text{DC}_j) \subseteq F_t \), let \( y \in L_t(\text{DC}_j) \). Then there is a feasible solution \( \hat{X} = \{\hat{x}_j : N_j \in B_t\} \) to \( DC_t \) such that \( \hat{x}_t = y \). Feasibility implies \( \hat{x}_j \in L_j(\text{DC}_j) \subseteq L_j(\text{DC}_j) = F_j \forall j \in R \) where the equality follows
Fig. 3. Construction of composite regions on general networks.

This content downloaded from 139.179.72.51 on Sat, 15 Dec 2018 17:13:00 UTC
All use subject to https://about.jstor.org/terms
(5) $F_2 \leftarrow N(F_1, b_{12}) \cap F_2$

(6) $N(F_2, b_{23})$ with $b_{23} = 4$

(7) $\bar{L}_3 = F_3 \leftarrow N(F_2, b_{23}) \cap F_3$

(8) $N(L_3, b_{23})$

(9) $\bar{L}_2 \leftarrow N(L_3, b_{23}) \cap \bar{L}_2$

(10) $N(L_2, b_{12})$

(11) $\bar{L}_1 \leftarrow N(L_2, b_{12}) \cap \bar{L}_1$

(12) A feasible solution.

Fig. 3. Continued.
from (7). Feasibility also implies (8) and (9) are satisfied so that

\[ \hat{x}_i \subseteq \bigcap_{j \in R} N(\hat{x}_j, b_{ij}) \cap S_i \]

where \( \subseteq \) follows from \( \hat{x}_j \in F_j \) \( \forall j \in R \) and equality follows from the construction of \( F_t \). Hence \( y = \hat{x}_t \in F_t \).

To prove \( F_t \subseteq L_t(DC_t) \), let \( y \in F_t \). It suffices to construct a feasible solution \( X = \{x_j : j \in J\} \) to \( DC \) such that \( \hat{x}_t = y \). We do this construction now.

Put \( \hat{x}_t = y \). For \( j \in R \), select an arbitrary point \( y_j \) in the nonempty set \( N(y_j, b_{ij}) \cap F_j \) and put \( x_j = y_j \) \( \forall j \in R \). The mentioned set is nonempty because \( y \in F_t \) implies \( y \in N(F_j, b_{ij}) \forall j \in R \) which implies there exists a point \( y_j \in F_j \) such that \( d(y_j, y) \leq b_{ij} \) for such \( j \). We observe now the portion of \( DC \) corresponding to (8) and (9) is satisfied by the partially constructed solution \( \{x_j : j \in J\} \). We construct the remaining components of \( X \) by making use of the induction hypothesis. For fixed \( j \in R \), the fact that \( y_j \in F_j \) implies \( y_j \in L_t(DC_t) \) (from (7)) so that there is a feasible solution to \( DC_t \) for which the location of new facility \( j \) is fixed at \( y_j (= \hat{x}_j) \). Let \( \bar{X}(j) = \{\hat{x}_i : \hat{x}_i \neq j, N_i \in B_j\} \cup \{\hat{x}_j\} \) be such a feasible solution. Clearly, \( \bar{X}(j) \) satisfies (10) for fixed \( j \in R \). It follows that \( \bar{X} = [\bigcup_{j \in R} \bar{X}(j)] \cup \{\hat{x}_j\} \) is feasible to (8, 9, 10). Hence, \( \hat{x}_t = y_t \in L_t(DC_t) \).

\( DC_t \) in Theorem 5.1 is a relaxation of \( DC \). That is, \( L_t \subseteq L_t(DC_t) \). This gives:

**Corollary 5.1.** \( L_j \subseteq F_j \) \( \forall j \in J \).

The next result is simply a specialization of Theorem 5.1 to the case \( t = r \).

**Theorem 5.2.** \( L_r = F_r \) for the root index \( r \).

We now have a characterization of consistency for \( DC \).

**Theorem 5.3. (Consistency Theorem)** Assume \( LN_B \) is a tree. \( DC \) is consistent if and only if \( F_r \neq \emptyset \) for the root node \( N_r \).

**Proof.** Clearly, \( DC \) is consistent if and only if the composite region \( L_r \neq \emptyset \). With \( L_r = F_r \), the result follows.

Observe that the consistency characterization of \( DC \) via composite sets \( L_j \) is always true. That is, either the sets \( F, L_1, \ldots, L_m \) are all nonempty or they are all empty and so \( DC \) is consistent if and only if \( L_j \neq \emptyset \) for an arbitrary \( j \) in \( J \). This claim is valid regardless of the structure of \( LN_B \). However, the characterization is of little use unless we have a way of computing at least one of the sets \( L_1, \ldots, L_m \). The assumption of tree structure on \( LN_B \) does precisely that: it allows us to construct the set \( F_r \) which happens to be the set \( L_r \). Hence, Theorem 5.3 gives an operational (computable) test for consistency. In fact, \( F_r \) is computable in \( O(|E|mn(m + n)) \) time for class (C2) (TANSEL and YESILKOKCEN, 1993), and so, a yes or no answer is available for any instance of \( DC \) in class (C2) in polynomial time.

**6. JUSTIFICATION OF SEIP-CR**

In this section we justify the second phase of SEIP-CR. First, we have the following lemma.

**Lemma 6.1.** Let \( (p, q) \in I_B \). If \( DC \) is consistent then

\[ L_q \subseteq N(L_p, b_{pq}). \]  

**Proof.** The assumption of consistency implies \( L_p, L_q \neq \emptyset \). Let \( y \in L_q \). Then for some \( \bar{X} \in F \), we have \( \hat{x}_q = y \). Feasibility of \( \bar{X} \) implies

\[ d(\hat{x}_q, \hat{x}_p) \leq b_{pq} \]  

\[ \hat{x}_p \in L_p. \]  

(12) gives \( \hat{x}_q \in N(\hat{x}_p, b_{pq}) \) while (13) implies \( N(\hat{x}_p, b_{pq}) \subseteq N(L_p, b_{pq}) \). Thus, \( y = \hat{x}_q \in N(L_p, b_{pq}) \) completing the proof.

We remark that due to symmetry we also have \( L_p \subseteq N(L_q, b_{pq}) \) in the above lemma. We further remark that the proof does not require the assumption of a tree structured \( LN_B \). That is, the lemma is valid for any set of distance constraints in any type of metric space.

For each new facility \( q \in J \), define \( J_q \) to be the set of indices \( p \in J \) such that \( (N_p, N_q) \) is an edge in \( LN_B \). We now have the following property.

**Property 6.1.** Assume \( DC \) is consistent. \( \forall q \in J \), we have

\[ L_q \subseteq \bigcap_{p \in J_q} N(L_p, b_{pq}). \]  

**Proof.** Any point \( y \in L_q \) is in each of the sets \( N(L_p, b_{pq}) \), \( p \in J_q \), due to Lemma 6.1. Hence, (14) follows.

The property simply asserts that a composite region for a given new facility is in the intersection of the expansions of the composite regions of all new facilities that are related to it via a distance bound.

We remark that Property 6.1 is true regardless of the structure of \( LN_B \) since Lemma 6.1 holds for
arbitrary DC. If we now assume \( LN_B \) is a tree, then we have the following result:

**PROPERTY 6.2.** Assume \( LN_B \) is a tree and DC is consistent. Let \( F_q \) be the output set for node \( N_q \) from any application of Phase 1. For any node \( N_p \) adjacent to \( N_q \), we have:

\[
L_q \subseteq N(L_p, b_{pq}) \cap F_q. \tag{15}
\]

**Proof.** Lemma 6.1 implies \( L_q \) is a subset of \( N(L_p, b_{pq}) \) while \( L_q \subseteq F_q \) due to Corollary 5.1. Hence, every point \( y \) in \( L_q \) is in both of the sets \( N(L_q, b_{pq}) \) and \( F_q \), completing the proof. \( \Box \)

We may now prove a much stronger assertion than (15): that (15) holds as a set equality if node \( N_q \) is the brown colored node selected in some iteration of Phase 2 of SEIP-CR and \( N_p \) is the unique green colored node which is adjacent to \( N_q \). This is essentially all that is needed to justify Phase 2 of SEIP-CR.

**THEOREM 6.1.** Assume \( LN_B \) is a tree and DC is consistent. Let \( F_q \) be the output set for node \( N_q \) from any application of Phase 1 for which the root node is not \( N_q \). Let \( N_p \) be the unique node adjacent to node \( N_q \) which is processed in Phase 1 subsequent to the computation of \( F_q \). Then

\[
L_q = N(L_p, b_{pq}) \cap F_q. \tag{16}
\]

**Proof.** First, we note that, since the root node \( N_r \) is different from \( N_q \), there is a unique node \( N_p \), which is the first encountered node distinct from \( N_q \) when we walk on the path connecting \( N_q \) to \( N_r \). Clearly then, among all nodes adjacent to \( N_q \), \( N_p \) is the only one that remains green just after \( N_q \) is brown colored (see Fig. 4) in Phase 1. It follows that, in Phase 2, since the green tree grows from \( N_r \), \( N_p \) will be added to the green tree prior to \( N_q \).

We now proceed with the proof of (16).

Let \( B_q \) be the brown subtree rooted at \( N_q \) when \( N_q \) is the selected tip node of the green tree in the application of Phase 1 stated in the theorem and let \( F_1, \ldots, F_m \) be the output sets from the same application. With \( J_q \) being the set of indices of \( N_j \) that are adjacent to \( N_q \), we know \( p \) is in \( J_q \). \( F_p \) is computed subsequent to \( F_q \), and all nodes \( N_j, j \in J_q - \{p\} \) are in the brown subtree \( B_q \) so that

\[
F_q = \bigcap_{j \in J_q - \{p\}} N(F_j, b_{jq}) \cap S_q. \tag{17}
\]

Define \( Q \) to be the right hand side of (17). Consider now a second application of Phase 1 with root node \( N_q \). Let \( \tilde{F}_1, \ldots, \tilde{F}_m \) be the output sets from application #2. Because \( N_q \) is the root node in application #2, Theorem 5.2 implies

\[
\tilde{F}_q = L_q. \tag{18}
\]

Observe that all nodes \( N_j \) in \( B_q, j \neq q \), are processed prior to \( N_q \) in both applications of Phase 1 so that the resulting brown subtrees \( B_j \) rooted at these nodes were the same in both applications. This implies

\[
\tilde{F}_j = F_j \ \forall j \text{ such that } N_j \text{ is in } B_q \text{ and } j \neq q \tag{19}
\]

(Theorem 5.1 implies \( F_j \) and \( F_q \) are both equal to the same partially induced composite region \( L(DC_j) \) corresponding to the brown subtrees \( B_j \) rooted at these nodes, thus justifying (19)).

The definition of \( Q \) and (19) imply

\[
Q = \bigcap_{j \in J_q - \{p\}} N(\tilde{F}_j, b_{jq}) \cap S_q. \tag{20}
\]

Since \( N_q \) is the root node in application #2, we have

\[
\tilde{F}_q = N(\tilde{F}_p, b_{pq}) \cap Q. \tag{21}
\]

We now have:

\[
L_q \subseteq N(L_p, b_{pq}) \cap F_q \quad \text{(from Property 6.2)}
\]

\[
\subseteq N(\tilde{F}_p, b_{pq}) \cap \tilde{F}_q \quad \text{(from } L_p \subseteq \tilde{F}_p, \text{ i.e. Corollary 5.1)}
\]

\[
= N(\tilde{F}_p, b_{pq}) \cap Q \quad \text{(from (17) and definition of } Q) \]

\[
= \tilde{F}_q \quad \text{(from (21))}
\]

\[
= L_q. \quad \text{(from (18))}
\]

Hence, all set inclusions are satisfied as set equalities which proves (16). \( \Box \)

We now have the concluding theorem.
THEOREM 6.2. Let \( \bar{L}_j, j \in J \), be the output sets from SEIP-CR. Then

\[
L_j = \bar{L}_j \quad \forall j \in J.
\]  

Proof. If \( \bar{L}_j = \emptyset \) \( \forall j \) due to infeasible termination, the assertion is true since Phase 1 terminates infeasible if and only if \( DC \) is inconsistent (Theorem 5.3). Suppose now Phase 1 terminated feasible. Let \( r \) be the root index. Then \( \bar{L}_r = F_r \neq \emptyset \) and Theorem 5.2 implies \( L_r = \bar{L}_r \). Theorem 6.1 implies \( L_q = \bar{L}_q \) \( \forall q \in J_r \), since \( \bar{L}_q = N(\bar{L}_r, b_{rq}) \cap F_q \). Hence (22) holds \( \forall j \in J_r \). Let now \( p \in J_r \) and consider all nodes \( N_j, j \in J_p, j \neq r \). Clearly, (22) holds for all such nodes again due to Theorem 6.1. The inductive structure of the proof exhausts all indices in \( J \) in this way, thus completing the proof. \( \square \)

7. SUMMARY AND CONCLUSION

THE COMPOSITE REGION for new facility \( j \) is the set of all points on the network at which new facility \( j \) can be safely placed without causing a violation of distance constraints. These regions give an alternate characterization of consistency, provide geometrical insights on the feasible set, enable recursive constructions of as many feasible solutions as desired, and have potential applications in sensitivity analysis.

We gave efficient methods to construct these regions for two classes of distance constraints without having to know the feasible set. In one class, the transport network is a tree, and in the other class the transport network is arbitrary but new facility interactions are of a special type.

ACKNOWLEDGMENT

WE WOULD LIKE to thank an anonymous referee for suggesting an induction based proof of the material in Section 5 which was originally based on a proof that was too much dependent on an earlier algorithm of the authors.

REFERENCES


TANSEL, B. Ç, AND G. N. YEŞILKOÇKEN, Polynomially Solvable Cases of Multifacility Distance Constraints on Cyclic Networks, Research Report IEOR-9311, Department of Industrial Engineering, Bilkent University, Ankara, Turkey, 1993.

(Received: February 1994; revision received: October 1994; accepted: November 1994)