

**stichting
mathematisch
centrum**



AFDELING MATHEMATISCHE BESLIJKUNDE
(DEPARTMENT OF OPERATIONS RESEARCH)

BW 107/79

MEI

A. KOLEN

COMPLEXITY OF LOCATION PROBLEMS ON NETWORKS

2e boerhaavestraat 49 amsterdam

Printed at the Mathematical Centre, 49, 2e Boerhaavestraat, Amsterdam.

The Mathematical Centre, founded the 11-th of February 1946, is a non-profit institution aiming at the promotion of pure mathematics and its applications. It is sponsored by the Netherlands Government through the Netherlands Organization for the Advancement of Pure Research (Z.W.O).

Complexity of location problems on networks

by

Antoon Kolen

ABSTRACT

We consider some well-known location problems, namely, the p-center problem (with mutual communication), the p-median problem (with mutual communication), the simple (capacitated) plant location problem, the location set covering problem and the maximal covering problem. We show that all these problems are NP-hard.

KEY WORDS & PHRASES: *location problems, computational complexity, NP-hardness, NP-completeness.*

1. INTRODUCTION

For some specially structured problems such as network flow and matching, polynomial-time algorithms (i.e., algorithms whose running time is bounded by a polynomial function of problem size) have been developed. However, for a large class of problems no polynomial-time algorithms are known. The theory of computational complexity showed the existence of a class of "NP-complete" problems, which are equivalent in the sense that

- (a) either all or none of them are solvable in polynomial time;
- (b) none of them is known to be solvable in polynomial time.

In addition, there is a class of "NP-hard" problems, which are, loosely speaking, as least as hard as the NP-complete problems. Many combinatorial problems that are notorious for their computational intractability such as graph coloring, set covering and travelingsalesman problems, are NP-hard. It is commonly conjectured that no polynomial-time algorithm exists for an NP-hard problem. Therefore establishing NP-hardness of a problem serves as a justification for the use of tedious enumerative optimization methods or of fast approximation algorithms.

In Section 2 we shall briefly discuss the concepts of complexity theory; this section is borrowed from the paper by LENSTRA and RINNOOY KAN [12]. The reader who is interested in details is referred to the recent book by GAREY and JOHNSON [6], which also contains an extensive listing of NP-complete problems. The reader who would like to know more about complexity theory without going into details is referred to the papers by KARP [11] and LENSTRA and RINNOOY KAN [12].

In section 3 we shall formulate some location problems on networks and discuss some relevant literature.

In Section 4 we review the complexity of the location problems formulated. The fact that the p-center problem, the p-median problem, the simple (capacitated) plant location problem, the location set covering problem and the maximal covering problem are NP-hard is well-known. This paper provides a uniform framework for the quite straightforward proofs of these results. In addition we establish NP-hardness for the p-center and p-median problems with mutual communication. The proofs of these results are more involved.

2. CONCEPTS OF COMPLEXITY THEORY [12]

A formal theory of NP-completeness would require the introduction of *Turing machines* [1] as theoretical computing devices. A *deterministic* Turing machine is a classical model for an ordinary computer, which is polynomially related to more realistic models such as the *random access machine*. It can be designed to *recognize languages*; the input consists of a *string*, which is *accepted* by the machine if and only if it belongs to the language. A *nondeterministic* Turing machine is an artificial model, which can be thought of as a deterministic one that can create copies of itself corresponding to different state transitions whenever convenient. In this case, a string is accepted if and only if it is accepted by one of the deterministic copies. P and NP are now defined as the classes of languages recognizable in polynomial time by deterministic and nondeterministic Turing machines, respectively.

For the purposes of exposition, we will expound the theory in terms of *recognition problems*, which require a yes/no answer. A string then corresponds to a problem *instance* and a language to a problem *type* or, more exactly, to the set of all its *feasible* instances. The feasibility of an instance is usually equivalent to the existence of an associated *structure*, whose size is bounded by a polynomial in the size of the instance; for example, the instance may be a graph and the structure a Hamiltonian circuit [11]. A recognition problem is in P if, for any instance, one can determine its feasibility or infeasibility in polynomial time. It is in NP if, for any instance, one can determine in polynomial time whether a given structure affirms its feasibility.

Problem P' is said to be *reducible* to problem P (notation: $P' \leq P$) if for any instance of P' an instance of P can be constructed in polynomial time such that solving the instance of P will solve the instance of P' as well. Informally, the reducibility of P' to P implies that P' can be considered as a special case of P , so that P is at least as hard as P' .

P is called *NP-hard* if $P' \leq P$ for every $P' \in NP$. In that case, P is at least as hard as any problem in NP . P is called *NP-complete* if P is NP-hard and $P \in NP$. Thus, the NP-complete problems are the most difficult problems in NP .

A polynomial-time algorithm for an NP-complete problem P could be used to solve all problems in NP in polynomial time, since for any instance of such a problem the construction of the corresponding instance of P and its solution can be both effected in polynomial time. Note the following two important observations.

- It is very unlikely that $P = NP$, since NP contains many notorious combinatorial problems, for which in spite of a considerable research effort no polynomial-time algorithms have been found so far.
- It is very unlikely that $P \in P$ for any NP-complete P , since this would imply that $P = NP$ by the earlier argument.

The first NP-completeness result is due to COOK (1971). He designed a "master reduction" to prove that every problem in NP is reducible to the SATISFIABILITY problem. This is the problem of determining whether a boolean expression in conjunctive form assumes the value *true* for some assignment of truth values to the variables; for instance, the expression $(x_1) \wedge (\bar{x}_1 x_2 \bar{x}_3) \wedge (x_3)$ is satisfied if $x_1 = x_2 = x_3 = \text{true}$. Given this result, one can establish NP-completeness of some $P \in NP$ by specifying a reduction $P' \propto P$ with P' already known to be NP-complete: for every $P'' \in NP$, $P'' \propto P'$ and $P' \propto P$ then imply that $P'' \propto P$ as well. In Section 4 we shall present several such proofs.

As far as *optimization problems* are concerned, we shall reformulate a minimization (maximization) problem by asking for the existence of a feasible solution with value at most (at least) equal to a given *threshold*. It should be noted that membership of NP for this recognition version does not immediately imply membership of NP for the original optimization problems as well. In particular, proposing a systematic search over a polynomial number of threshold values, guided by positive and negative answers to the existence, is not a valid argument. This is because a nondeterministic Turing machine is only required to give *positive* answers in polynomial time. Indeed, no *complement* of any NP-complete problem is known to be in NP !

As an obvious consequence of the above discussion, NP-completeness can only be proved with respect to a recognition problem. However, the corresponding optimization problem might be called NP-hard in the sense that the existence of a good algorithm for its solution would imply that $P = NP$.

3. FORMULATION OF LOCATION PROBLEMS

In the first part of this section we shall formulate the location models to be considered and briefly discuss some relevant literature. In order to establish NP-hardness for the location problems considered we have to prove NP-completeness for the corresponding recognition problems. In the second part of this section we shall formulate these recognition problems.

Let $G = (V, E)$ be a graph with vertex set $V = \{v_1, \dots, v_n\}$ and edge set $E = \{e_1, \dots, e_m\}$. A weight $w(v_i) \geq 0$ is associated with each vertex v_i ($i = 1, 2, \dots, n$). A length $\ell(e_j) > 0$ is associated with each edge e_j ($j = 1, 2, \dots, m$). A point x on the graph G is defined to be a vertex or a point along an edge. The distance $d(x, y)$ between two points x and y on G is defined to be the length of the shortest path between x and y . Let $X_p = \{x_1, \dots, x_p\}$ be a set of p points on G . Then we define the distance $d(z, X_p)$ between a point z on G and the set X_p as

$$d(z, X_p) = \min_{1 \leq j \leq p} d(z, x_j).$$

The *p-center problem* is to locate p objects (points) on G so as to minimize the maximum of the weighted distances between the objects and the clients (vertices) assigned to be served by them, where each client is served by the object closest to it.

Let

$$F(X_p) = \max_{1 \leq i \leq n} w(v_i) d(v_i, X_p).$$

Then the p -center problem is to find a set X_p^* on G such that

$$F(X_p^*) = \min_{X_p \text{ on } G} F(X_p).$$

The min-max objective function is often appropriate in the formulation of emergency problems, e.g., regarding police, fire and ambulance services. A vertex of a graph then corresponds to a population center. The weight of a vertex can be interpreted as a measure of its importance or the probability of an emergency occurring, and will therefore often be a function of the

size of the population. If X_p is restricted to be a subset of V we call this problem the *vertex p-center problem*. If X_p is not restricted we sometimes call it the *absolute p-center problem*. Algorithms to solve the p-center problem can be found in CHRISTOFIDES [2], HANDLER [9], and KARIV and HAKIMI [10].

The p-median problem is to locate p objects on G so as to minimize the sum of the weighted distances between the objects and the clients served by them, where each client is served by the object closest to it.

Let

$$H(X_p) = \sum_{i=1}^n w(v_i) d(v_i, X_p).$$

Then the p-median problem is to find a set X_p^* on G such that

$$H(X_p^*) = \min_{X_p \text{ on } G}$$

HAKIMI [8] has shown that there exists an optimal subset $X_p^* \subseteq V$. Therefore we can formulate the p-median problem as an integer programming problem (p) using the variables x_{ij} ($i, j = 1, 2, \dots, n$), where

$$x_{jj} = \begin{cases} 1 & \text{if an object is located at vertex } v_j, \\ 0 & \text{otherwise,} \end{cases}$$

and for $i \neq j$

$$x_{ij} = \begin{cases} 1 & \text{if client } i \text{ is assigned to vertex } v_j, \\ 0 & \text{otherwise:} \end{cases}$$

$$(p) \quad \min z = \sum_{i=1}^n \sum_{j=1}^n w(v_i) d(v_i, v_j) x_{ij}$$

$$(1) \quad \text{s.t.} \quad \sum_{j=1}^n x_{ij} = 1, \quad i = 1, 2, \dots, n,$$

$$(2) \quad x_{ij} \leq x_{jj}, \quad i, j = 1, 2, \dots, n,$$

$$\sum_{j=1}^n x_{jj} = p,$$

$$x_{ij} \in \{0,1\}, \quad i,j = 1,2,\dots,n.$$

The constraints (1) ensure that each client is assigned to a vertex. Constraints (2) ensure that client i is only assigned to a vertex where an object is located. The min-sum objective is often appropriate when one wants to minimize the total cost of a transportation network. The clients weight then correspond to the amount to be shipped between the clients and their associated (closest) object, the distance $d(v_i, v_j)$ between vertex v_i and v_j corresponds to the cost of transporting one unit between vertex v_i and v_j . The p -median problem is often solved as a linear programming by relaxation of the integer constraints, in combination with branch and bound. Solving the linear programming often results in an optimal solution with integer values of the variables. Another solution method is Lagrangean relaxation (NARULA et.al.[13]). KARIV and HAKIMI [10] give an $O(n^2 p^2)$ algorithm for finding a p -median in a tree network.

In contrast with the p -center and p -median problem, the p -center and the p -median problem with mutual communication involve weighted distances between all pairs of clients and objects, and all pairs of objects. Let $\alpha(v_i, j)$ be the weight corresponding to client i and object j ($1 \leq i \leq n, 1 \leq j \leq p$). Let $\beta(j, k)$ be the weight corresponding to object j and object k ($1 \leq j < k \leq p$). Let

$$K(x_p) = \max \left\{ \max_{\substack{1 \leq i \leq n \\ 1 \leq j \leq p}} \alpha(v_i, j) d(v_i, x_j), \max_{1 \leq j < k \leq p} \beta(j, k) d(x_j, x_k) \right\}$$

and let

$$L(x_p) = \sum_{i=1}^n \sum_{j=1}^p \alpha(v_i, j) d(v_i, x_j) + \sum_{j=1}^p \sum_{k=j+1}^p \beta(j, k) d(x_j, x_k).$$

Then the p -center problem with mutual communication is to find a set x_p^* on G such that

$$K(x_p^*) = \min_{x_p \text{ on } G} K(x_p).$$

The p-median problem with mutual communication is to find a set X_p^* on G such that

$$L(X_p^*) = \min_{X_p \text{ on } G} L(X_p).$$

It can be shown for the p-median problem with mutual communication that there exists an optimal solution $X_p^* \subseteq V$ (FRANCIS and WHITE [5]). The p-center problem with mutual communication on a tree network can be solved by an $O((n+p)^3 \log(n+p))$ algorithm based on the result on distance constraints for tree networks given by FRANCIS et.al.[4]. The p-median problem with mutual communication on a tree network can be solved by the $O(p^3 n)$ algorithm of PICARD and RATLIFF [15]. For general networks these two problems have not yet received much attention.

The location set covering problem (LSCP) and the maximal covering problem (MCP) are problems formulated by REVELLE. Again a number of emergency facilities has to be located on vertices of a network. A vertex v_i with population $w(v_i)$ is said to be covered if there is an emergency facility within a prescribed distance S . The location set covering problem is to determine the minimal number of emergency facilities and their location such that every vertex is covered. The maximal covering problem is to determine the location of a prescribed number of p facilities such that the total population covered is maximal.

Define

$$N_i = \{v_j \mid d(v_i, v_j) \leq S\}, \quad (i = 1, 2, \dots, n)$$

$$x_j = \begin{cases} 1 & \text{if a facility is located at vertex } v_j, \\ 0 & \text{otherwise,} \end{cases}$$

$$y_i = \begin{cases} 1 & \text{if vertex } v_i \text{ is covered,} \\ 0 & \text{otherwise.} \end{cases}$$

Then (LSCP) can be formulated as

$$\min z = \sum_{j=1}^n x_j$$

$$\begin{aligned} \text{s.t. } \sum_{j \in N_i} x_j &\geq 1, & i = 1, 2, \dots, n, \\ x_j &\in \{0, 1\}, & j = 1, 2, \dots, n. \end{aligned}$$

(MCP) can be formulated as

$$\begin{aligned} \max z &= \sum_{i=1}^n w(v_i) y_i \\ \text{s.t. } \sum_{j \in N_i} x_j &\geq y_i, & i = 1, 2, \dots, n, \\ \sum_{j=1}^n x_j &= p, \\ x_j &\in \{0, 1\}, & j = 1, 2, \dots, n. \end{aligned}$$

Revelle solved problems of these types by relaxing the integer constraints. The corresponding linear programming problem frequently has an optimal solution with integer values of the variables. If this is not the case, REVELLE [16] adds a cut or uses branch and bound.

Given a set of possible locations for establishing new facilities (plants, warehouses, etc.), the plant location problem deals with the supply of a single commodity from a subset of these locations to a set of clients with a prescribed demand for the commodity. Given the cost structure, a minimum cost production/transportation plan has to be determined in terms of the number of facilities established, their location and the amount shipped from each facility to each client. We define the following data:

- m : the number of potential facilities, indexed by $i \in I = \{1, 2, \dots, m\}$,
- n : the number of clients, indexed by $j \in J = \{1, 2, \dots, n\}$,
- f_i : the nonnegative fixed cost for establishing facility i ,
- p_i : the per unit cost of operating facility i ,
- t_{ij} : the transportation cost of shipping one unit from facility i to client j ,
- D_j : the number of units demand by client j ,
- S_i : the capacity of facility i ,

and the following variables:

- s_{ij} : number of units produced at facility i and shipped to client j ,
 x_{ij} : the fraction of D_j supplied by facility i ,
 y_i : $y_i = 1$ if facility i is established and 0 otherwise.

If each facility has unlimited capacity we refer to the problem as the simple plant location problem (SPLP). If each facility has a finite capacity we refer to the problem as the capacitated plant location problem (CPLP). (SPLP) can be formulated as

$$\begin{aligned}
 \min z &= \sum_{i \in I} \sum_{j \in J} (p_i + t_{ij}) D_j x_{ij} + \sum_{i \in I} f_i y_i \\
 \text{s.t.} \quad &\sum_{i \in I} x_{ij} = 1, \quad j \in J, \\
 &x_{ij} \leq y_i, \quad i \in I, j \in J, \\
 &x_{ij} \geq 0, \quad i \in I, j \in J, \\
 &y_i \in \{0,1\}, \quad i \in I.
 \end{aligned}$$

The constraints $x_{ij} \leq y_i$ ensure that $x_{ij} = 0$ if facility i is not established. Note that there always exists an optimal solution with $x_{ij} \in \{0,1\}$.

(CPLP) can be formulated as

$$\begin{aligned}
 \min z &= \sum_{i \in I} \sum_{j \in J} (p_i + t_{ij}) s_{ij} + \sum_{i \in I} f_i y_i \\
 \text{s.t.} \quad &\sum_{i \in I} s_{ij} = D_j, \quad j \in J, \\
 &\sum_{j \in J} s_{ij} \leq S_i y_i, \quad i \in I, \\
 &s_{ij} \geq 0, \quad i \in I, j \in J, \\
 &y_i \in \{0,1\}, \quad i \in I.
 \end{aligned}$$

(SPLP) can be considered as a special case of (CPLP) with all capacities large enough. Therefore we have only to show that (SPLP) is NP-hard. Simple plant location problems are usually solved efficiently by the algorithm of

ERLENKOTTER [3]. An extension of this algorithm given by GUIGNARD and SPIELBERG [7] can also handle mixed plant location problems, i.e. plant location problems where some facilities have capacity constraints and others do not. The (CPLP) can also be solved using Lagrangean relaxation (NAUSS [14]).

We conclude this section by formulating the recognition problems corresponding to the above location problems.

MIN-MAX MULTICENTER

INSTANCE: A graph $G = (V, E)$, a weight $w(v) \in \mathbb{Z}^+$ for each $v \in V$, a length $\ell(e) \in \mathbb{Z}^+$ for each $e \in E$, a positive integer $p \leq |V|$, a positive rational number B .

QUESTION: Is there a set X_p of p points on G such that

$$w(v)d(v, X_p) \leq B \text{ for all } v \in V?$$

MIN-SUM MULTICENTER

INSTANCE: A graph $G = (V, E)$, a weight $w(v) \in \mathbb{Z}^+$ for each $v \in V$, a length $\ell(e) \in \mathbb{Z}^+$ for each $e \in E$, a positive integer $p \leq |V|$, a positive rational number B .

QUESTION: Is there a set X_p of p vertices on G such that

$$\sum_{v \in V} w(v)d(v, X_p) \leq B?$$

MIN-MAX MULTICENTER WITH MUTUAL COMMUNICATION

INSTANCE: A graph $G = (V, E)$ with vertex set $V = \{v_1, \dots, v_n\}$, a positive integer $p \leq |V|$, weights $\alpha(v_i, j) \in \mathbb{Z}^+$ ($1 \leq i \leq n, 1 \leq j \leq p$), $\beta(j, k) \in \mathbb{Z}^+$ ($1 \leq j < k \leq p$), a length $\ell(e) \in \mathbb{Z}^+$ for each $e \in E$, a positive rational number B .

QUESTION: Is there a set $X_p = \{x_1, \dots, x_p\}$ of p points on G such that

$$\alpha(v_i, j)d(v_i, x_j) \leq B \text{ (} 1 \leq i \leq n, 1 \leq j \leq p \text{),}$$

$$\beta(j, k)d(x_j, x_k) \leq B \text{ (} 1 \leq j < k \leq p \text{)?}$$

MIN-SUM MULTICENTER WITH MUTUAL COMMUNICATION

INSTANCE: A graph $G = (V, E)$ with vertex set $V = \{v_1, \dots, v_n\}$, a positive integer $p \leq |V|$, weights $\alpha(v_i, j) \in \mathbb{Z}^+$ ($1 \leq i \leq n, 1 \leq j \leq p$), $\beta(j, k) \in \mathbb{Z}^+$ ($1 \leq j < k \leq p$), a length $\ell(e) \in \mathbb{Z}^+$ for each $e \in E$, a positive rational

number B.

QUESTION: Is there a set $X_p = \{x_1, \dots, x_p\}$ of p vertices on G such that

$$\sum_{i=1}^n \sum_{j=1}^p \alpha(v_i, j) d(v_i, x_j) + \sum_{j=1}^p \sum_{k=j+1}^p \beta(j, k) d(x_j, x_k) \leq B?$$

LOCATION SET COVERING

INSTANCE: A graph $G = (V, E)$, a positive integers $p \leq |V|$ and S.

QUESTION: Is there a subset $V' \subseteq V$, $|V'| \leq p$, such that for every vertex $v \in V$ there is a vertex $u \in V'$ such that

$$d(u, v) \leq S?$$

MAXIMAL COVERING

INSTANCE: A graph $G = (V, E)$, positive integers $p \leq |V|$ and S, a weight $w(v) \in \mathbb{Z}^+$ for all $v \in V$, a positive rational number B.

QUESTION: Is there a set $V' \subseteq V$, $|V'| = p$, such that $\sum_{v \in T} w(v) \geq B$, where T is the set of vertices with the property that $d(v, V') \leq S$ for all $v \in T$?

SIMPLE PLANT LOCATION

INSTANCE: A complete bipartite graph $K_{m, n}$ with vertex sets V_1 ($|V_1| = m$) and V_2 ($|V_2| = n$), nonnegative integers c_{ij}, f_i ($1 \leq i \leq m, 1 \leq j \leq n$), a positive rational number B.

QUESTION: Are there a non-empty subset $I \subseteq \{1, 2, \dots, m\}$ and subsets $J_i \subseteq \{1, 2, \dots, n\}$ for all $i \in I$ such that $\{J_i \mid i \in I\}$ forms a partition of $\{1, 2, \dots, n\}$ and $\sum_{i \in I} \sum_{j \in J_i} c_{ij} + \sum_{i \in I} f_i \leq B$?

4. COMPLEXITY OF LOCATION PROBLEMS

As indicated in Section 2 we have to reduce some known NP-complete problem Q to problem P in order to prove that P is NP-complete. The NP-complete problems we shall use here for this purpose are the following.

1. VERTEX COVER

INSTANCE: A graph $G' = (V', E')$ and a positive integer $k \leq |V'|$.

QUESTION: Is there a vertex cover of size at most k, i.e., is there a subset $V'' \subseteq V'$, $|V''| \leq k$, such that for every edge $(u, v) \in E'$ u or v belongs to V'' ?

2. CLIQUE

INSTANCE: A graph $G' = (V', E')$ and a positive integer $k \leq |V'|$.

QUESTION: Is there a clique of size at least k , i.e., is there a subset $V'' \subseteq V'$, $|V''| \geq k$, such that for all $u, v \in V''$ $(u, v) \in E'$?

3. DOMINATING SET

INSTANCE: A graph $G' = (V', E')$ and a positive integer $k \leq |V'|$.

QUESTION: Is there a dominating set of size at most k , i.e., is there a subset $V'' \subseteq V'$, $|V''| \leq k$, such that for every $u \in V' \setminus V''$ there is a $v \in V''$ such that $(u, v) \in E'$?

We shall assume that $G' = (V', E')$ does not contain self-loops. All forthcoming NP-completeness proofs have the same structure. Given an instance of one of the problems 1, 2, and 3 we shall present a reduction which defines an instance of the problem under consideration such that solving the latter instance will solve the former instance. We shall leave it to the reader to prove that the recognition problems we will consider belong to NP, as well as that the reductions we present are polynomial bounded.

THEOREM 1. MIN-MAX MULTICENTER is NP-complete.

PROOF. Let an instance of DOMINATING SET be given by $G' = (V', E')$ and k . The corresponding instance of MIN-MAX MULTICENTER is defined by $V = V'$, $E = E'$, $w(v) = 1$ for all $v \in V$, $\ell(e) = 1$ for all $e \in E$, $p = k$, $B = 1$.

Let V'' be a dominating set of size at most k for $G' = (V', E')$. Then $d(v, V'') \leq 1$ for all $v \in V$ in the graph $G = (V, E)$.

Let $X_k = \{x_1, \dots, x_k\}$ be a set of k points on $G = (V, E)$ such that $d(v, X_k) \leq 1$ for all $v \in V$. Moving each point x_j ($1 \leq j \leq k$) to the vertex closest to it gives a set V'' of vertices of size at most k such that $d(v, V'') \leq 1$ for all $v \in V$. Hence V'' is a dominating set of size at most k in $G' = (V', E')$. \square

THEOREM 2. MIN-SUM MULTICENTER is NP-complete.

PROOF. Let an instance of DOMINATING SET be given by $G' = (V', E')$ and k . The corresponding instance of MIN-SUM MULTICENTER is defined by $V = V'$,

$E = E'$, $w(v) = 1$ for all $v \in V$, $\ell(e) = 1$ for all $e \in E$, $p = k$, $B = |V| - k$.

Let V'' be a dominating set of size at most k for $G' = (V', E')$. Then

$$\sum_{v \in V'} d(v, V'') \leq |V| - k.$$

Let X_k be a set of k vertices on $G = (V, E)$ with $\sum_{v \in V} d(v, X_k) \leq |V| - k$. Since for all $v \in V \setminus X_k$ we have $d(v, X_k) \geq 1$ we conclude that $\sum_{v \in V} d(v, X_k) = |V| - k$ and $d(v, X_k) = 1$ for all $v \in V \setminus X_k$.

Hence X_k is a dominating set of size k in $G' = (V', E')$. \square

We note that both MIN-MAX and MIN-SUM MULTICENTER are NP-complete even for $w(v) = 1$ for all $v \in V$ and $\ell(e) = 1$ for all $e \in E$. The reductions used in Theorem 1 and Theorem 2 can also be found in KARIV and HAKIMI [10].

THEOREM 3. SET LOCATION COVERING is NP-complete.

THEOREM 4. MAXIMAL COVERING is NP-complete.

The proofs of Theorems 3 and 4 are left to the reader as exercises (hint: use DOMINATING SET).

THEOREM 5. MIN-MAX MULTICENTER WITH MUTUAL COMMUNICATION is NP-complete.

PROOF. Let an instance of CLIQUE be given by $G' = (V', E')$ and k . The corresponding instance of MIN-MAX MULTICENTER WITH MUTUAL COMMUNICATION is defined by $V = \{s\} \cup \{\bar{s}\} \cup \{\bar{\bar{s}}\} \cup V' \cup \bar{V}' \cup \bar{\bar{V}}'$, where $\bar{V}' = \{\bar{v}_i \mid v_i \in V'\}$ and $\bar{\bar{V}}' = \{\bar{\bar{v}}_i \mid v_i \in V'\}$, $E = \{(s, v_i), (\bar{s}, \bar{v}_i), (\bar{\bar{s}}, \bar{\bar{v}}_i), (v_i, \bar{v}_i), (\bar{v}_i, \bar{\bar{v}}_i) \mid i = 1, 2, \dots, n\} \cup \{(v_i, \bar{\bar{v}}_j) \mid (v_i, v_j) \in E'\}$, $p = 3k$, $[\alpha(s, j) = 1, \alpha(\bar{s}, j+k) = 1, \alpha(\bar{\bar{s}}, j+2k) = 1, \beta(j, j+k) = 1, \beta(j+k, j+2k) = 1$ for all j ($j = 1, 2, \dots, k$)], $\beta(i, j+2k) = 1$ for all i, j ($1 \leq i < j \leq k$), all other weights are zero, $\ell(e) = 1$ for all $e \in E$, $B = 1$. The graph $G = (V, E)$ is illustrated in Fig. 1.

Let x_1, \dots, x_{3k} be points on $G = (V, E)$ satisfying the distance constraints (DC1) and (DC2):

$$(DC1) \quad \begin{cases} d(s, x_j) \leq 1, d(\bar{s}, x_{j+k}) \leq 1, d(\bar{\bar{s}}, x_{j+2k}) \leq 1, d(x_j, x_{j+k}) \leq 1, \\ d(x_{j+k}, x_{j+2k}) \leq 1 \quad \text{for all } j \text{ (} j = 1, 2, \dots, k \text{),} \end{cases}$$

$$(DC2) \quad \{d(x_i, x_{j+2k}) \leq 1 \quad \text{for all } i, j \text{ (} 1 \leq i < j \leq k \text{)}.$$

Since $d(s, \bar{s}) = d(\bar{s}, \bar{\bar{s}}) = 3$ it follows from (DC1) that $x_j \in V'$, $x_{j+k} \in \bar{V}'$, $x_{j+2k} \in \bar{\bar{V}}'$ and that $x_{j+k} = \bar{x}_j$, $x_{j+2k} = \bar{\bar{x}}_j$ for all j ($j = 1, 2, \dots, k$). From now on we will write \bar{x}_j and $\bar{\bar{x}}_j$ instead of x_{j+k} and x_{j+2k} respectively. Suppose $x_j = x_k (=v)$ ($j < k$) in V' . Then also $\bar{\bar{x}}_j = \bar{\bar{x}}_k$. It follows from (DC2) that $(x_j, \bar{\bar{x}}_k) \in E$ or equivalently $(v, v) \in E'$. Since $G' = (V', E')$ does not contain self-loops we conclude that all points $x_1 \dots x_k$ are different vertices v_{q_1}, \dots, v_{q_k} in V' .

It follows from (DC2) that $(x_i, \bar{\bar{x}}_j) \in E$ ($1 \leq i < j \leq k$) or equivalently $(v_{q_i}, v_{q_j}) \in E'$. Hence v_{q_1}, \dots, v_{q_k} constitute a k -clique in $G' = (V', E')$.

Let v_{q_1}, \dots, v_{q_k} constitute a k -clique in $G' = (V', E')$. Then define $x_j = v_{q_j}$, $x_{j+k} = \bar{v}_{q_j}$ and $x_{j+2k} = \bar{\bar{v}}_{q_j}$ for all j ($j = 1, 2, \dots, k$). It is easily seen that $x_1 \dots x_{3k}$ satisfy the distance constraints (DC1) and (DC2).

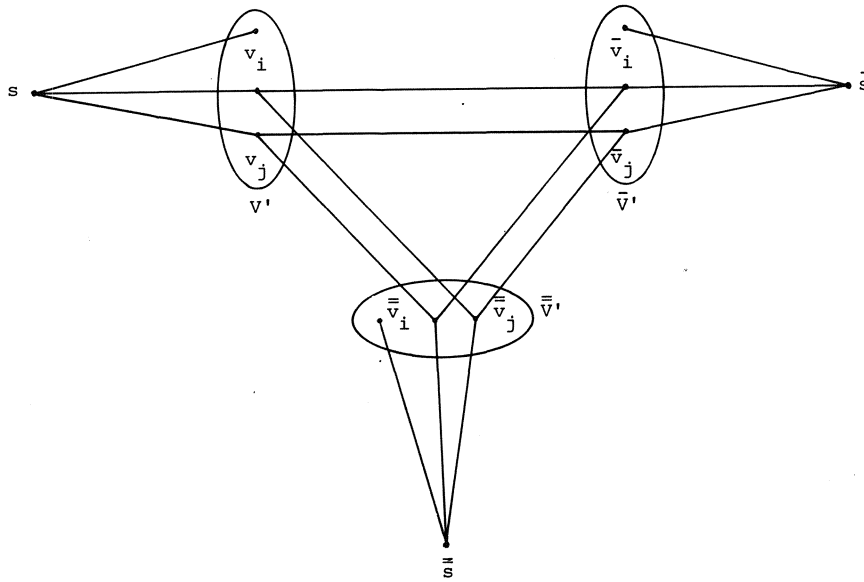


Fig.1. The graph $G = (V, E)$ corresponding to Theorem 5 and Theorem 6.

THEOREM 6. MIN-SUM MULTICENTER WITH MUTUAL COMMUNICATION is NP-complete.

PROOF. Let an instance of CLIQUE be given by $G' = (V', E')$ and k . The corresponding instance of MIN-SUM MULTICENTER WITH MUTUAL COMMUNICATION is defined by $V = \{s\} \cup \{\bar{s}\} \cup V' \cup \bar{V}' \cup \bar{\bar{V}}'$, where $\bar{V}' = \{\bar{v}_i \mid v_i \in V'\}$ and

$\bar{V}' = \{\bar{v}_i \mid v_i \in V'\}$, $E = \{(s, v_i), (\bar{s}, \bar{v}_i), (\bar{s}, \bar{v}_i), (v_i, \bar{v}_i), (\bar{v}_i, \bar{v}_i) \mid i = 1, 2, \dots, n\} \cup \{(v_i, \bar{v}_j) \mid (v_i, v_j) \in E'\}$ (see Fig.1), $p = 3k$, $[\alpha(s, j) = p - j + 1, \alpha(\bar{s}, j+k) = 2, \alpha(\bar{s}, j+2k) = j, \beta(j, j+k) = 1, \beta(j+k, j+2k) = 1$ for all j ($j = 1, 2, \dots, k$)], $\beta(i, j+2k) = 1$ for all i, j ($1 \leq i < j \leq k$), all other weights are zero, $\ell(e) = 1$ for all $e \in E$, $B = 3/2(k^2 + 3k)$.

Let $x_1 \dots x_{3k}$ by vertices of $G = (V, E)$ such that

$$\sum_{j=1}^k [(p-j+1)d(s, x_j) + 2d(\bar{s}, x_{j+j}) + jd(\bar{s}, x_{j+2k}) + d(x_j, x_{j+k}) + d(x_{j+k}, x_{j+2k})] + \sum_{i=1}^k \sum_{j=i+1}^k d(x_i, x_{j+2k}) \leq 3/2(k^2 + 3k). \quad (1)$$

We have the following inequalities:

$$d(s, \bar{s}) \leq d(s, x_j) + d(x_j, x_{j+k}) + d(x_{j+k}, \bar{s}) \text{ for all } j \text{ (} j = 1, 2, \dots, k \text{)}, \quad (2)$$

$$d(\bar{s}, \bar{s}) \leq d(\bar{s}, x_{j+k}) + d(x_{j+k}, x_{j+2k}) + d(x_{j+2k}, \bar{s}) \text{ for all } j \text{ (} j = 1, 2, \dots, k \text{)}, \quad (3)$$

$$d(s, \bar{s}) \leq d(s, x_i) + d(x_i, x_{j+2k}) + d(x_{j+2k}, \bar{s}) \text{ for all } i, j \text{ (} 1 \leq i < j \leq k \text{)}. \quad (4)$$

Summing inequalities (2) and (3) for all j ($j = 1, 2, \dots, k$), inequality (4) for all i, j ($1 \leq i < j \leq k$) and adding gives

$$\sum_{j=1}^k [(p-j+1)d(s, x_j) + 2d(\bar{s}, x_{j+k}) + jd(\bar{s}, j+2k) + d(x_j, x_{j+k}) + d(x_{j+k}, x_{j+2k})] + \sum_{i=1}^k \sum_{j=i+1}^k d(x_i, x_{j+2k}) \geq k[d(s, \bar{s}) + d(\bar{s}, \bar{s})] + \frac{k(k-1)}{2} d(s, \bar{s}) = 3/2(k^2 + 3k) \quad (5).$$

We conclude from (1) and (5) that equality must hold in (1) and (5). Since equality holds in (5) equality holds in (2), (3) and (4).

Equality holds in (2), (3) and (4) if and only if $x_j \in V'$, $x_{j+k} \in \bar{V}'$, $x_{j+2k} \in \bar{V}'$, $x_{j+k} = \bar{x}_j$, $x_{j+2k} = \bar{x}_j$ for all j ($j = 1, 2, \dots, k$) and $(x_i, \bar{x}_j) \in E$ for all i, j ($1 \leq i < j \leq k$). Analogously to Theorem 5 we can prove that x_1, \dots, x_k are different vertices in V' which constitute a k -clique in $G' = (V', E')$.

Let v_{q_1}, \dots, v_{q_k} constitute a k -clique in $G' = (V', E')$. Then define $x_j = v_{q_j}$, $x_{j+k} = \bar{v}_{q_j}$, $x_{j+2k} = \bar{v}_{q_j}$ for all j ($j = 1, 2, \dots, k$). It is easily seen that equality holds in (2), (3) and (4). Therefore (1) is satisfied with equality. \square

THEOREM 7. SIMPLE PLANT LOCATION is NP-complete.

PROOF. Let an instance of VERTEX COVER be given by $G' = (V', E')$ and k . The corresponding instance of SIMPLE PLANT LOCATION is defined by $V_1 = V'$, $V_2 = E'$, $c_{ij} = 1$ if vertex v_i is incidently with edge e_j , $c_{ij} = M$ (M very large) otherwise, $f_i = 1$ for all i , $|B| = |E| + k$.

Let I be a non-empty subset of $\{1, 2, \dots, |V|\}$ and let J_i ($i \in I$) be subsets which form a partition of $\{1, 2, \dots, |E|\}$ such that $\sum_{i \in I} \sum_{j \in J_i} c_{ij} + \sum_{i \in I} f_i \leq |E| + k$.

Then $c_{ij} \neq 0$ for all $i \in I, j \in J_i$. Hence $c_{ij} = 1$ or equivalently edge e_j is incident with vertex v_i for all $i \in I, j \in J_i$. Therefore $V'' = \{v_i \mid i \in I\}$ is a vertex cover of $G' = (V', E')$. Since $\sum_{i \in I} \sum_{j \in J_i} c_{ij} = |E|$, V'' is a vertex cover of size at most k .

Let $V'' = \{v_1 \dots v_p\}$ be a vertex cover of $G' = (V', E')$ with $|V''| = p \leq k$. Then each edge is incident with at least one vertex of V'' . We assign each edge to a vertex of V'' incident with the edge. This defines a partition $J_1 \dots J_p$ of $\{1, 2, \dots, |E|\}$.

Hence $\sum_{i \in I} \sum_{j \in J_i} c_{ij} + \sum_{i \in I} f_i = |E| + p \leq |E| + k$. \square

ACKNOWLEDGEMENT

I would like to thank JAN KAREL LENSTRA for his careful reading of the manuscript and a number of suggestions which greatly improved the contents of this paper.

REFERENCES

- [1] AHO, A.V., J.E. HOPCROFT, J.D. ULLMAN (1974), *The Design and Analysis of Computer Algorithms*, Addison-Wesley, Reading, Mass.
- [2] CHRISTOFIDES, N., (1975), *Graph Theory - An Algorithmic Approach*, Academic Press, New York.
- [3] ERLKOTTER, D., *A dual-based procedure for uncapacitated facility location*, *Operations Res.*26 (1978), 992-1009.
- [4] FRANCIS, R.L., T.J. LOWE, H.D. RATLIFF, *Distance constraints for tree network multifacility location problems*, *Operations Res.*26 (1978), 570-596.
- [5] FRANCIS, R.L. & J.A. WHITE (1974), *Facility layout and location*, Prentice-Hall, Inc. Englewood Cliffs, New Jersey.

- [6] GAREY, M.R. & D.S. JOHNSON (1979), *Computers and Intractability: a Guide to the Theory of NP-Completeness*, Freeman, San Francisco.
- [7] GUIGNARD, M. & K. SPIELBERG, *A dual method for the mixed plant location*, University of Pennsylvania, the Wharton School, Technical Report no. 20 (1977).
- [8] HAKIMI, S.L., *Optimal locations of switching centers and the absolute centers and medians of a graph*, *Operations Res.*13 (1965), 462-475.
- [9] HANDLER, G.Y., *Complexity and efficiency in minimax network location*, (To appear in *Combinatorial Optimization* (ed. N. Christofides et. al., John Wiley)). (1977).
- [10] KARIV, O. & S.L. HAKIMI, *An algorithmic approach to network location problems*, part I: *The p-center*, part II: *The p-median*, (submitted to *SIAM J. Applied Math.*) (1976).
- [11] KARP, R.M., *On the computational complexity of combinatorial problems*, *Networks* 5 (1975) 45-68.
- [12] LENSTRA, J.K. & A.H.G. RINNOOY KAN, *Computational complexity of discrete optimization problems*, *Interfaces between computer science and operations research* ed. J.K. Lenstra, A.H.G. Rinnooy Kan, P. van Emde Boas, *Mathematical Centre Tract 99*, Amsterdam.
- [13] NARULA, S.C., U.I. OGBU, H.M. SAMUELSSON, *An algorithm for the p-median problem*, *Operations Res.*25 (1977) 709-713.
- [14] NAUSS, R.M., *An improved algorithm for the capacitated facility location problem*, *Research Report University of Missouri-St. Louis*.
- [15] PICARD, J.C. & H.D. RATLIFF, *A cut approach to the rectilinear distance facility location problem*, *Operations Research* 26 (1978), 422-433.
- [16] REVELLE, C., C. TOREGAS, R. SWAIN, *Reply to Rao's note on the location of emergence service facilities*, *Operations Res.*22 (1974), 1262-1267.

ONTVANGEN 2 JULI 1979