C. De Simone, C. Mannino

EASY INSTANCES OF THE PLANT LOCATION PROBLEM

R. 427 Gennaio 1996

Caterina De Simone — Istituto di Analisi dei Sistemi ed Informatica, CNR, Viale Manzoni 30, 00185 Roma, Italy.

Carlo Mannino — Progetto Finalizzato Trasporti II, CNR, Viale dell'Università 11, 00185 Roma, Italy.

Abstract

In this paper we describe properties of a particular transformation of the simple plant location problem into a vertex packing problem on a corresponding undirected graph. We characterize the undirected graphs that arise from simple plant location problems and we give a necessary and sufficient condition for such graphs to be perfect. This allows us to find a new family of polynomially solvable instances of the simple plant location problem.

1. Introduction

We assume familiarity with basic notions of graph theory (see, for instance, [3]). Let G be an undirected graph; a *clique* in G is any set of pairwise adjacent vertices; a *stable set* in G is any set of pairwise nonadjacent vertices. A clique or a stable set are called *maximal* if there exists no other clique or stable set containing them. The largest size of a clique and of a stable set in G is denoted by $\omega(G)$ and $\alpha(G)$, respectively. The *vertex packing problem* on an undirected graph G, with weights on its vertices, is to find in G a stable set of weight as big as possible.

A location decision problem is to choose facilities to open, such as industrial plants or warehouses, to maximize the total profit (or minimize the total cost) of satisfying the demand for some commodity. There is some fixed cost for activating the facilities and a profit for distributing the commodities between the facilities and the clients. This problem has been extensively studied in the literature and is commonly referred to as the *plant location problem.* When no plant has capacity limitation, the problem is called the *simple plant location problem*, or in short $(SPLP)$. The study of the $(SPLP)$ is going to be the topic of this paper. The $(SPLP)$ is known to be NP-hard.

Throughout the paper, K will denote a "big" constant. Let I be a set of plants and let J be a set of clients. Let c_{ij} be the profit made for supplying client j from plant i and let q_i be the cost of activation of plant i. The *simple plant location problem* is the problem of supplying all clients so that to maximize the total profit.

To find a mathematical model of the problem, let introduce the following binary variables:

$$x_{ij} = \begin{cases} 1 & \text{if client } j \text{ is supplied from plant } i, \\ 0 & \text{otherwise;} \end{cases}$$

$$y_i = \begin{cases} 1 & \text{if plant } i \text{ is active,} \\ 0 & \text{otherwise.} \end{cases}$$

Then the simple plant location problem can be formulated as the integer linear program:

(P1)

$$\max \quad \sum_{i\in I}\sum_{j\in J} c_{ij}x_{ij} - \sum_{i\in I} q_i y_i \tag{1}$$

s.t.

$$x_{ij} \le y_i \qquad \text{for all } i \in I, j \in J, \tag{2}$$

$$\sum_{i\in I} x_{ij} = 1 \qquad \text{for all } j \in J, \tag{3}$$

$$x_{ij} \in \{0,1\} \qquad \text{for all } i \in I, j \in J, \tag{4}$$

$$y_i \in \{0,1\} \qquad \text{for all } i \in I. \tag{5}$$

Constraint (2) assures that client j is supplied by plant i only if plant i is active; constraint (3) assures that client j is supplied by exactly one plant. We shall often refer to a given instance of problem $(P1)$ as (I, J, c, q).

If $I = J$, I is the vertex set of a directed graph, and (1) and (3) are modified to $\min \sum_{i\in I}\sum_{j\in J} c_{ij}x_{ij}$ and $\sum_{i\in I} y_i = P$, then $(P1)$ is known as the *P-median problem.* The plant location problem has several applications in banking account location [5], clustering analysis [7], lock-bix location [13], etc. For a survey on location theory, see [10]. In [6] and [8] were introduced some reductions of the $(SPLP)$ to the vertex packing problem; in [1] a reduction of the P-median problem into a vertex packing problem was described.

In this paper we focus our attention on the reduction of [1]. In the following section, we shall show how to generalize this reduction to the case of the simple plant location problem. This will allows us to describe a new class of polynomially solvable instances of the $(SPLP)$.

2. Transforming the $(SPLP)$ into a vertex packing problem

The goal of this section is to show how to transform problem $(P1)$ into a vertex packing problem on a corresponding undirected graph. The reduction we are going to describe was introduced by Avella and Sassano in [1] to describe new classes of facets of the P-median polytope. In this section we shall show how it can be extended to the case of the more general $(SPLP)$.

For this purpose, consider an arbitrary instance (I, J, c, q) of problem $(P1)$. First, note that we can always assume that $q_i \geq 0$ for all $i \in I$: indeed, if $q_i < 0$ for some $i \in I$, then in every optimal solution of $(P1)$, the corresponding variable y_i will be equal to one. Next, note that, without loss of generality, we can always assume that each plant is also a client and, viceversa, each client is also a plant. To see this, consider a plant i that is not a client $(i \in I - J)$. Set $I' = I$, $J' = J \cup \{i\}$, $q'_i = q_i$ for every $i \in I'$, $c'_{hk} = c_{hk}$ for every $h \in I'$ and $k \in J' - \{i\}$, and $c'_{hi} = 0$ for every $h \in I'$. Then, it is easy to show that the original problem (I, J, c, q) is equivalent to the new problem (I', J', c', q'). Now, let j be a client that is not a plant $(j \in J - I)$. Set $I' = I \cup \{j\}$, $J' = J$, $q'_i = q_i$ for every $i \in I' - \{j\}$, $q'_j = K$, $c'_{hk} = c_{hk}$ for every $h \in I' - \{j\}$ and $k \in J'$, and $c'_{jk} = 0$ for every $k \in J'$, where K is a "big" constant (K is chosen so that plant i will never be active in any optimal solution of (I', c', q')). Again, it is easy to show that the original problem (I, J, c, q) is equivalent to the new problem (I', J', c', q'). It follows that in $(P1)$ we can always assume that $I = J$, and so we shall write (I, c, q) in place of (I, J, c, q).

Now, let (x, y) be any feasible solution of $(P1)$. Note that constraint (2) implies that: if $x_{ij} = 1$ then $y_i = 1$ (if plant i supplies client j then plant i must be active). In particular, if $x_{ii} = 1$ then $y_i = 1$. Is the contrary true? In other words, is it true that $y_i = 1$ implies $x_{ii} = 1$? Clearly, in general, the answer is "no".

We shall say that the instance (I, c, q) satisfies the *median hypothesis* if there exists an optimal solution (x, y) of the corresponding integer linear programming formulation $(P1)$ such that $x_{ii} = 1$ whenever $y_i = 1$, for every $i \in I$. To put it differently, the median hypothesis assures that, if plant i is active then it supplies itself. (Recall that we are assuming that every plant is also a client.) As observed in [1], the median hypothesis is always satisfied in the case of the P-median problem - which justifies its name. Now, a sufficient condition for the median hypothesis to be satisfied for a given plant i is:

$$c_{ii} \geq c_{hi} \qquad\qquad \text{for all } h \in I. \tag{6}$$

Observe that condition (6) has a straightforward interpretation: when plant i is active, the convenience of supplying plant i from itself is not less then the convenience of supplying it from some other plant.

Condition (6) seems quite reasonable in practice. In fact, we shall show that we can always assume its validity. For this purpose, assume there exists a plant i for which the median hypothesis does not hold, and so it violates (6). Set $L_i = \max\{c_{hi} : h \in I\}$ and consider the following instance of the simple plant location problem, (I', c', q') in which $I' = (I - \{i\}) \cup \{i_1, i_2\}$, and

$$c'_{hk} = \begin{cases} 0 & \text{if } h \in I' \text{ and } k = i_1, \text{ or if } h = i_2 \text{ and } k \in I' - \{i_2\} \\ c_{ii} & \text{if } h = i_1 \text{ and } k = i_2, \\ c_{ik} & \text{if } h = i_1 \text{ and } k \in I' - \{i_1, i_2\}, \\ c_{hi} & \text{if } h \in I' - \{i_1, i_2\} \text{ and } k = i_2, \\ L_i & \text{if } h = k = i_2, \\ c_{hk} & \text{otherwise;} \end{cases}$$

$$q'_h = \begin{cases} q_i & \text{if } h = i_1, \\ K & \text{if } h = i_2, \\ q_h & \text{otherwise.} \end{cases}$$

The "big" constant K is chosen so that plant i_2 will never be active in any optimal solution of (I', c', q') (it is sufficient to choose $K >> L_i$). Now, it is easy to show that, with every optimal solution of (I, c, q), we can associate an optimal solution of (I', c', q') of the same value, and that, viceversa, with every optimal solution of (I', c', q') we can associate an optimal solution of (I, c, q) of the same value. If we assume that plant i was the only one that did not satisfy the median hypothesis, then by construction, every plant in problem (I', c', q') satisfies condition (6), and so it satisfies the median hypothesis. If there were more than one plant that did not satisfy the median hypothesis, we can clearly apply the above transformation several times. Hence, we can always assume that the median hypothesis holds, and so

$$y_i = x_{ii} \qquad \text{for all } i \in I. \tag{7}$$

We are now ready to show how to transform the $(SPLP)$ into a vertex packing problem. For this purpose, substitute each variable x_{ii} in $(P1)$ with the variable y_i. Problem $(P1)$ becomes:

(P2)

$$\max \sum_{i,j\in I, i\neq j} c_{ij}x_{ij} - \sum_{i\in I}(q_i - c_{ii})y_i \tag{8}$$

s.t.

$$x_{ij} \leq y_i \qquad \text{for all } i,j \in I, j \neq i, \tag{9}$$

$$y_j + \sum_{i\in I-\{j\}} x_{ij} = 1 \qquad \text{for all } j \in I, \tag{10}$$

$$x_{ij} \in \{0,1\} \qquad \text{for all } i,j \in I, j \neq i, \tag{11}$$

$$y_i \in \{0,1\} \qquad \text{for all } i \in I. \tag{12}$$

Now, from (10) we obtain $y_j = 1 - \sum_{i\in I-\{j\}} x_{ij}$, for all $j \in I$, and so substituting into (8) and (9) and deleting the constant term $(-\sum_{j\in I}(q_j - c_{jj}))$ in (8), we get:

(P3)

$$\max \sum_{i,j\in I, j\neq i} (c_{ij} - c_{jj} + q_j)x_{ij} \tag{13}$$

s.t.

$$x_{ij} + \sum_{k\in I-\{i\}} x_{ki} \leq 1 \qquad \text{for all } i,j \in I,\ j \neq i, \tag{14}$$

$$x_{ij} \in \{0,1\} \qquad \text{for all } i,j \in I, j \neq i. \tag{15}$$

Clearly, $(P3)$ is an equivalent integer linear programming formulation of our problem $(P1)$ that involves only the variables x_{ij}, with $i \neq j$. Indeed, for every optimal solution x^* of $(P3)$, we can easily build an optimal solution $(\bar{x}, \bar{y})$ of $(P1)$ by simply setting: $\bar{x}_{ij} = x^*_{ij}$ for all $i, j \in I, i \neq j$, and $\bar{x}_{jj} = \bar{y}_j = 1 - \sum_{i \in I - \{j\}} x^*_{ij}$, for all $j \in I$.

Note that $(P3)$ is nothing but the integer linear programming formulation of the *set packing problem* which, in turn, is well known to be equivalent to a vertex packing problem on the intersection graph G of $(P3)$: G is the undirected graph whose vertices are in one-to–one correspondence with the variables of $(P3)$ and there exists an edge joining two vertices of G if and only if the corresponding variables of $(P3)$ appear in a same constraint; the weights of the vertices of G are simply equal to the coefficients of the corresponding variables in the objective function of $(P3)$. Hence, solving a simple plant location problem amounts to find a stable set of maximum weight in the corresponding undirected graph. In the following section we shall give a combinatorial interpretation of problem $(P3)$ in terms of a directed graph D, and we shall show that the intersection graph G of $(P3)$ can be seen as a "line graph" of D.

3. Line graphs of digraphs

In the previous section we have seen that we can formulate the simple plant location problem as the integer linear problem $(P3)$. Now, problem $(P3)$ can be interpreted as a combinatorial problem. To see this, set

$$w_{ij} = c_{ij} - c_{jj} + q_j. \tag{16}$$

First, note that, if $w_{ij} \leq 0$ for some $i, j \in I$, then there always exists an optimal solution of $(P3)$ in which the corresponding variable x_{ij} will be zero. Indeed, consider the original problem $(P1)$; the fact that $c_{ij} \leq c_{jj} - q_j$ simply means that the convenience in supplying client j from plant i is not more then the convenience in opening plant j and supplying plant j from itself.

Now, set $N = I$ and $A = \{(i,j) : i,j \in I, i \neq j, w_{ij} > 0\}$ and consider the directed graph $D = (N, A)$. To each arc (i, j) of D assign the positive weight w_{ij}. Note that D is simple (no loops and no multiple arcs) and is not necessarily complete. Call a subset F of arcs of D *nice* if it satisfies the following two properties:

(i) for every node i of D, there exists in F at most one arc entering in i;

(ii) if $(i, j) \in F$, then F contains no arc entering node i and it contains no arc leaving node j.

With this terminology, it is easy to see that solving problem $(P3)$ amounts to find a nice subset F of arcs of D of maximum weight.

Now, we shall show that the intersection graph of problem $(P3)$ can be seen as a *line graph* of the directed graph $D = (N, A)$.

For this purpose, we shall first recall the definition of the line graph of an undirected graph. Let $H = (W, P)$ be an undirected simple graph (no loops and no multiple edges) and let e and f be two arbitrary distinct edges of H. The edges e and f are called *dependent* if they are incident in a same vertex. Let Q denote the family of all pairs of dependent edges of H. Then, the graph $LH = (P, Q)$ is called the *line graph* of H. Note that there exists a vertex of LH for every edge of H and two vertices of LH are adjacent if and only if the corresponding edges of H are dependent. Furthermore, if H has weights on its edges then the vertices of LH will have correspondingly the same weights. Clearly, there exists a one–to–one correspondence between the sets of independent edges of H (*matchings*) and the sets of independent vertices of LH (stable sets). Hence, finding a matching in H of maximum weight is equivalent to finding in LH a stable set of maximum weight. It is well known that both problems are polynomially solvable [9].

Now, consider a simple directed graph (no loops and no multiple arcs) $D = (N, A)$. As in the undirected case, we could define some "dependency" between two arcs of D; assume that this is done. Then, denoting by F the family of all pairs of "dependent" arcs of D, we can define the *line graph* of

D as the undirected graph $LD = (A, F)$, where, as in the previous case, two vertices of LD are adjacent if and only if the corresponding arcs of D are "dependent". Again, there exists a one–to–one correspondence between sets of "independent" arcs of D and sets of independent vertices of LD (which again are stable sets). If D had weights assigned on its arcs, then LD has correspondingly the same weights assigned on its vertices. Let (Q_1) be the problem of finding in D a set of "independent" arcs of maximum weight and let (Q_2) be the problem of finding in LD a stable set of maximum weight. Clearly, problems (Q_1) and (Q_2) are equivalent. Clearly, there are several ways of defining a "dependency" between two arcs of D; for each choice there will be a correspondent problem (Q_1). In the following we shall see three different definitions.

For this purpose, when a denotes the arc (i, j), we shall write $i = t(a)$ (tail) and $j = h(a)$ (head). Now, let a and b be two arbitrary arcs of D. A natural definition of "dependency" is the following: a and b are dependent if they share a common node, i.e. t(a)=t(b) or t(a)=h(b) or h(a)=h(b). In this case, a set of independent arcs of D is nothing but the natural generalization of a matching in a directed graph, and hence problem (Q_1) (and so (Q_2)) is again easy: solving (Q_1) amounts to find a matching of maximum weight in the undirected graph underlying D.

But what if we define the dependency between two arcs of D in a different way? Is it true that both problems (Q_1) and (Q_2) are still polynomial? The answer is in general "no".

One example is suggested by Chvátal and Ebenegger [4]. They call a and b dependent if $h(a) = t(b)$. In this case, a set of independent arcs of D is a *directed max-cut*, and so problem (Q_1) is NP-hard [11].

One other example is given by Balas [2] who defines a and b dependent if $h(a) = h(b)$ or if $t(a) = t(b)$ or if $t(a) = h(b)$ and $h(a) = t(b)$. In this case, it is easy to show that problem (Q_1) is nothing but a relaxation of the monotone asymmetric traveling salesman problem, and so is still NP-hard [16].

Our definition of dependency between two arcs is the following: arcs a and b are said to be *dependent* if $h(a) = h(b)$ or if $h(a) = t(b)$; see Figure 1.

The "line graph" of a directed graph D that arises this way, will be called the *M-graph* of D.

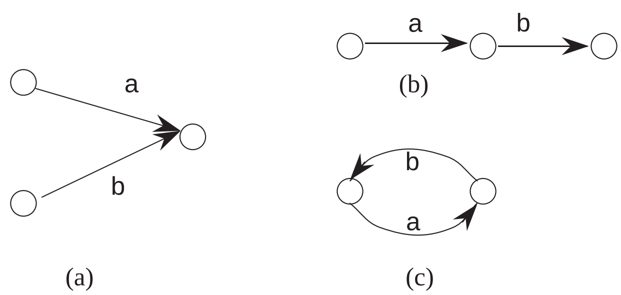

Figure 1: Dependence between arcs

Since every subset F of independent arcs in D satisfies both properties (i) and (ii), problem $(P3)$ is equivalent to problem (Q_1), and so (Q_1) is again NP-hard. Hence, the simple plant location is equivalent to the problem $(Q2)$ of finding a stable set of maximum weight in the line graph of D (M-graph of D).

Now, problem (Q_2) amounts to solve a corresponding integer linear program. To write such a program, let $G = (V, E)$ denote the M-graph of the directed graph $D = (N, A)$. Let:

$$z_k = \begin{cases} 1 & \text{if vertex } k \text{ belongs to a maximal stable set of } G, \\ 0 & \text{otherwise.} \end{cases}$$

Then, finding in G a stable set of maximum weight is to solve:

(P4)

$$\max \sum_{k \in V} w_k z_k \tag{17}$$

s.t.

$$\sum_{k \in C} z_k \leq 1 \qquad \text{for all maximal cliques } C \text{ in } G, \tag{18}$$

$$z_k \in \{0,1\} \qquad \text{for every node } k \text{ of } G. \tag{19}$$

Since $(P1)$ is equivalent to $(P3)$, which, in turn, is equivalent to $(P4)$, it follows that we can always formulate any instance of the $(SPLP)$ as the integer linear program $(P4)$.

Now, let $(P4)^R$ denote the linear problem obtained from $(P4)$ by relaxing the integrality constraints. We are interested in determining when all optimal solutions of problem $(P4)^R$ are integral, for every choice of the vector $w = (w_k)$. It is well known that a necessary and sufficient condition for this to happen is that the graph G is *perfect* (for a definition of a perfect graph, see Section 5). Hence, every instance of the simple plant location problem whose corresponding undirected graph is perfect, can be solved by simply solving the linear program $(P4)^R$. And this can be accomplished in polynomial time by the ellipsoid method [12]. But, the ellipsoid method is not useful in practice, and so one could object that having transformed the simple plant location problem into problem $(P4)$ was of no use; moreover, the linear program $(P4)^R$ has, in general, an exponential number of constraints (since, in general, a graph has an exponential number of maximal cliques).

In fact, the graphs in the formulation $(P4)$ are M-graphs and in the following section we shall show that every M-graph has only a polynomial number of maximal cliques. Hence, for all M-graphs G that are perfect, an optimal solution of $(P4)$ can be found by solving the easy linear program $(P4)^R$. It follows that every instance of the simple plant location problem whose corresponding M-graph is perfect can be solved by simply finding an optimal solution of a corresponding LP. It is then interesting to find necessary and sufficient conditions for an M-graph to be perfect. For this purpose, in section 4, we shall first give a characterization of the M-graphs. In the following of the paper, we shall often denote the arc of D corresponding to an arbitrary vertex u of the M-graph of D, by a_u.

4. M-Graphs

In the previous section we claimed that every M-graph G of a directed graph $D = (N, A)$ has a polynomial number of maximal cliques. To see this, consider an arbitrary maximal clique C of G. Write $C = \{1, 2, \ldots, n\}$ and let $a_1, a_2, \ldots, a_n$ denote the corresponding arcs in D. First, assume that $n \geq 4$. Since G is the line graph of D and since all vertices in C are pairwise adjacent, it follows that at least $n-1$ arcs a_i must share the same head; without loss of generality, we can assume that these are the first $n-1$ (i.e., $h(a_1) = h(a_2) = \ldots = h(a_{n-1})$). But then, either $h(a_n) = h(a_1)$ (see Figure 2 (a)) or $t(a_n) = h(a_1)$ (see Figure 2 (b)); in the last case, the head of a_n may coincide with the tail of precisely one arc a_i, with $1 \leq i \leq n-1$. Note that we have at most $|A|$ such cliques, one for each arc of D. Next, assume that $n = 3$. Then, either the three arcs a_i share the same head (see Figure 3 (a)); or $h(a_1) = h(a_2) = t(a_3)$ (with the possibility that the head of a_3 coincides with the tail of precisely one arc a_1, a_2) (see Figure 3 (b)); or $h(a_1) = t(a_2)$, $h(a_2) = t(a_3)$, and $h(a_3) = t(a_1)$ (see Figure 3 (c)). Note that we have at most $\binom{|N|}{3}$ such cliques. But then, since $|A| = O(|N|^2)$, it follows that G has $O(|N|^3)$ maximal cliques.

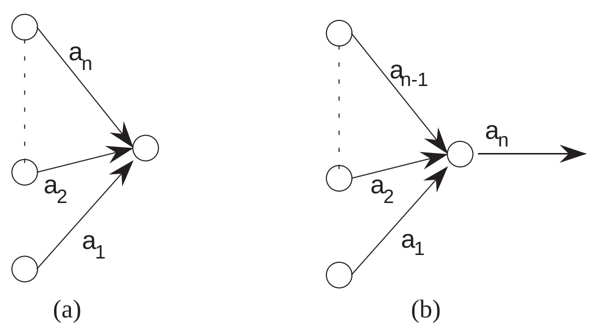

Figure 2: Large cliques of an M-graph

In the following we shall give a characterization of the M-graphs. For this purpose, let $C = (V(C), E(C))$ be an undirected graph satisfying the following three properties:

- $V(C) = T(C) \cup H(C)$ with $T(C) \cap H(C) = \emptyset$;

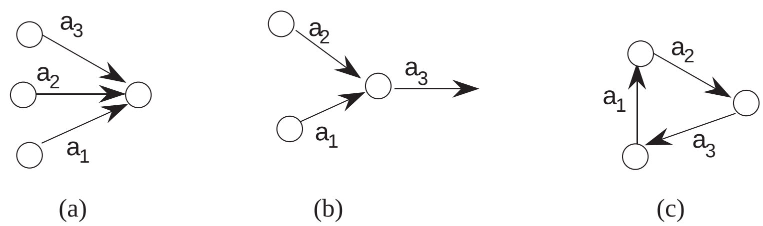

Figure 3: Small cliques of an M-graph

- $T(C)$ induces a clique in C and $H(C)$ induces a stable set in C;
- every vertex in $T(C)$ is adjacent to every vertex in $H(C)$.

The graph C is called a *crown*; the set $T(C)$ is the *tail* of C and $H(C)$ is the *head* of C. In Figure 4 is shown a crown C with $|T(C)| = 4$ and $|H(C)| = 3$.

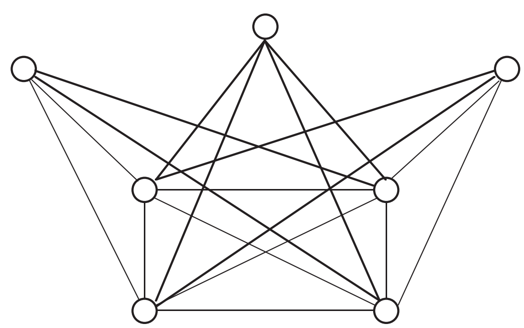

Figure 4: A crown.

Now, let $\mathcal{C}$ be a family of crowns in an arbitrary undirected graph G that satisfies the following two properties:

(1) each edge of G is contained in at least one crown in $\mathcal{C}$;

(2) each vertex v of G is contained in exactly two crowns in $\mathcal{C}$, say C_1 and C_2, and $v \in H(C_1) \cap T(C_2)$ or $v \in H(C_2) \cap T(C_1)$.

A family $\mathcal{C}$ of crowns in G satisfying both properties (1) and (2) is called a *Krausz-type covering* of the graph G. Note that, in general, a Krausz-type covering does not necessarily exist in an arbitrary undirected graph. In fact, we can show that the graphs that have a Krausz-type covering are precisely the M-graphs.

Theorem 1. *An undirected graph is an M-graph if and only if there exists a Krausz-type covering of it.*

Proof.

If Let $G = (V, E)$ be an arbitrary undirected graph and let $\mathcal{C} = \{C_1, \ldots, C_l\}$ be a family of crowns in a Krausz-type covering of G. We only need show that G is the M-graph of some directed graph. For this purpose, associate with $\mathcal{C}$ a digraph D in the following way: there exists in D a node x_i for each crown C_i in $\mathcal{C}$, and there exists in D an arc (x_i, x_j) if some vertex of G is in the head of C_i and in the tail of C_j.

We claim that G is the M-graph of D. To prove the claim, first observe that the number of arcs of D is equal to the number of vertices of G: indeed, for every two crowns in $\mathcal{C}$, say C_i and C_j, there exists at most one vertex of G which is contained in $H(C_i) \cap T(C_j)$, and so for every arc (x_i, x_j) of D there exists precisely one vertex in V; viceversa, for every vertex u in V there exist precisely two crowns in $\mathcal{C}$, say C_i and C_j, such that $u \in H(C_i) \cap T(C_j)$ (by property (2)), and so there exists in D precisely one arc (the arc (x_i, x_j)).

Next, observe that for every pair of adjacent vertices of G, the corresponding arcs of D are dependent, that is they have a common head or the head of the first is the tail of the second. To see this, let u and v be two arbitrary adjacent vertices of G; let a_u and a_v denote the corresponding arcs of D. By property (1), edge uv must be contained in some crown C_k of $\mathcal{C}$. Clearly, not both u and v are in $H(C_k)$ (because they are adjacent), and so either both u and v are in the tail of C_k or u is in the tail of C_k and v is in the head of C_k. In the first case, property (2) implies that there exist in $\mathcal{C}$ two different crowns, say C_i and C_j, such that $u \in H(C_i)$ and $v \in H(C_j)$ (with $i \neq j$, since $uv \in E$). But then, the arcs a_u and a_v are dependent since $a_u = (x_i, x_k)$ and

$a_v = (x_j, x_k)$, and we are done. In the second case, again property (2) implies that there exist two crowns in $\mathcal{C}$ (not necessarily distinct), say C_i and C_j, such that $u \in H(C_i)$ and $v \in T(C_j)$. But then, the arcs a_u and a_v are dependent since $a_u = (x_i, x_k)$ and $a_v = (x_k, x_j)$, and again we are done.

Hence, to prove the claim, we only need show that for every pair of dependent arcs of D the corresponding vertices of G are adjacent. For this purpose, let a_u and a_v be two arbitrary dependent arcs of D; let u and v denote the corresponding vertices of G. Without loss of generality, we can assume that $u \in H(C_i) \cup T(C_j)$, and so $a_u = (x_i, x_j)$. Since a_u and a_v are dependent, either x_i is the head of a_v or x_j is the head of a_v or x_j is the tail of a_v. In the first case, we can write $a_v = (x_k, x_i)$ (k may coincide with j), and so $v \in H(C_k) \cap T(C_i)$; but then, since $u \in H(C_i)$, u and v are adjacent, and we are done. In the second case, we can write $a_v = (x_k, x_j)$, and so $v \in H(C_k) \cap T(C_j)$; but then, since $u \in T(C_j)$, u and v are adjacent, and we are done. In the third case, we can write $a_v = (x_j, x_k)$ (k may coincide with i), and so $v \in H(C_j) \cap T(C_k)$; but then, since $u \in T(C_j)$, u and v are adjacent, and again we are done. Hence, the claim is proved.

<u>Only-if</u> Let $D = (N, A)$ be an arbitrary directed graph and let x be an arbitrary node of D. Let $G = (V, E)$ denote the M-graph of D. Let $\delta^+(x)$ and $\delta^-(x)$ denote the set of all arcs leaving node x (outgoing arcs) and entering node x (ingoing arcs), respectively. Set $\delta(x) = \delta^+(x) \cup \delta^-(x)$. Let $A(x)$ and $B(x)$ denote the set of vertices of G that correspond to all arcs in $\delta^+(x)$ and $\delta^-(x)$, respectively. Note that the set $A(x) \cup B(x)$ induces a crown C_x in G with $H(C_x) = A(x)$ and $T(C_x) = B(x)$: indeed, $A(x)$ induces a stable set in G and $B(x)$ induces a clique in G, and every vertex in $A(x)$ is adjacent to every vertex in $B(x)$.

Now, let $\mathcal{C} = \{C_x : x \in N\}$. To prove the theorem, we only need show that $\mathcal{C}$ is a Krausz-type covering of G, i.e. it satisfies both properties (1) and (2). For this purpose, let u and v be any two adjacent vertices of G; let a_u and a_v denote the corresponding arcs of D. Since a_u and a_v are dependent, they must share at least one node, say x, such that the crown C_x in $\mathcal{C}$ contains the edge uv, and so property (1) is satisfied. Now, let u be an arbitrary vertex of G

corresponding to the arc a_u of D. Write $a_u = (x, y)$. Since $a_u \in \delta^+(x) \cap \delta^-(y)$, it follows that u is contained in exactly two crowns of $\mathcal{C}$, namely C_x and C_y; furthermore, $u \in H(C_x) \cap T(C_y)$. Hence $\mathcal{C}$ satisfies also property (2). The theorem follows. ∎

5. When an M-graph is perfect

In section 3 we have seen that an arbitrary instance of the simple plant location problem is completely characterized by a corresponding directed graph $D = (N, A)$ with weights on its arcs. Furthermore, if the corresponding M-graph (line graph of D) is perfect, then we can solve this instance by simply finding an optimal solution of a linear program that has a polynomial number of constraints (problem $(P4)^R$). Hence, it is interesting to characterize all perfect M-graphs.

The goal of this section is give such a characterization. For this purpose, we need to recall some results on perfect graphs. A graph $G = (V, E)$ is called *perfect* if, for each of its induced subgraphs F, $\omega(F)$ is equal to the minimum number of colors in a coloration of V so that adjacent vertices get different colors. A graph is *minimal imperfect* if it is not perfect but all of its proper induced subgraphs are perfect. In the early sixties, Claude Berge conjectured that every minimal imperfect graph is a chordless cycle whose number of vertices is odd and at least five (*odd hole*) or its complement (*odd anti-hole*). This conjecture is known as the *Strong Perfect Graph Conjecture*; it remains unsettled.

We shall show that the Strong Perfect Graph Conjecture holds for the M-graphs. In fact, we shall prove more: an M-graph is perfect if and only if it does not contain the odd anti-hole with seven vertices and it does not contain odd holes. In order to prove this result, we shall rely on the following properties of a minimal imperfect graph $G = (V, E)$ with n vertices:

(a) G has precisely n stable sets of size $\alpha(G)$ and precisely n cliques of size $\omega(G)$; [14]:

(b) every vertex of G is in precisely $\omega(G)$ cliques of size $\omega(G)$ and in precisely

$\alpha(G)$ stable sets of size $\alpha(G)$ [14];

(c) the n stable sets can be enumerated as $S_1, S_2, \ldots, S_n$ and the n cliques can be enumerated as $C_1, C_2, \ldots, C_n$ in such a way that $S_i \cap C_j = \emptyset$ if and only if $i = j$ [14];

(d) for every node v of G there exists a unique partition of $G - v$ into $\alpha(G)$ cliques of size $\omega(G)$; in particular, if $S_1, S_2, \ldots, S_{\alpha(G)}$ are the stable sets containing v, then $C_1, C_2, \ldots, C_{\alpha(G)}$ partition $G - v$ [14];

(e) if $\omega(G) \leq 3$ then G is an odd hole or the odd anti-hole with seven vertices [15].

Lemma 1. *Let $G = (V, E)$ be a minimal imperfect graph and let M denote an arbitrary clique in G of size $\omega(G) - 1$. Then there exist at most two vertices in $V - M$ that are adjacent to every vertex in M.*

Proof. Assume the contrary: there exist three vertices adjacent to every vertex in M, say u, v, and w. Clearly, u, v and w are pairwise nonadjacent. Write $\alpha = \alpha(G)$ and $\omega = \omega(G)$.

First, note that there exist precisely $\alpha - 1$ stable sets of size α that contain both u and v. To see this, let $\mathcal{S}_u$ and $\mathcal{S}_v$ denote the family of all α stable sets of size α that contain vertex u and vertex v, respectively. Property (d) implies that $\mathcal{S}_u \neq \mathcal{S}_v$. If $|\mathcal{S}_u \cap \mathcal{S}_v| \neq \alpha - 1$ then $|\mathcal{S}_u \cap \mathcal{S}_v| \leq \alpha - 2$, and so there exist in $\mathcal{S}_u$ two distinct stable sets, say S' and S'', such that $v \notin S' \cup S''$. Now, since $u \in S' \cup S''$, $S' \cap M = \emptyset$ $S'' \cap M = \emptyset$. But then, S' and S'' are two distinct maximum stable sets having empty intersection with the maximum clique $\{v\} \cup M$, contradicting property (c). Similarly, there exist precisely $\alpha - 1$ stable sets of size α that contain both u and w, and there exist precisely $\alpha - 1$ stable sets of size α that contain both v and w.

Now, let $C_1, C_2, \ldots, C_\alpha$ denote the α cliques of size ω in the partition of $G - u$. Since the pair u, v belongs to exactly $\alpha - 1$ stable sets of size α, without loss of generality, we can assume that these are $S_1, S_2, \ldots, S_{\alpha-1}$, and so $G - v$ is partitioned into $C_1, C_2, \ldots, C_{\alpha-1}, C_{\alpha+1}$, and so $v \in C_\alpha$ and $u \in C_{\alpha+1}$. Since the pairs u, w and v, w belong to exactly $\alpha - 1$ stable sets of size α, it follows that $G - w$ is partitioned into $C_1, C_2, \ldots, C_{\alpha-2}, C_\alpha, C_{\alpha+1}$, and so $w \in C_{\alpha-1}$. Note that the two sets $K_1 = C_\alpha - \{v\}$ and $K_2 = C_{\alpha+1} - \{u\}$ are the same. But then, since the two cliques C_α and $C_{\alpha+1}$ are in the partition of $G - w$, it follows that $K_1 = K_2 = \emptyset$, and so $\omega = 1$, which is impossible. ■

Finally, before proving the main result of this section, we need the following property of the M-graphs. For this purpose, consider the graph H in Figure 5, where the edge joining vertices 5 and 6 may not be present.

Lemma 2. *The graph H in Figure 5 is not an M-graph.*

Proof. Assume the contrary: H is an M-graph, and so there exists a directed graph D such that H is the M-graph of D. Let a_i denote the arc of D corresponding to the vertex i of H, $i = 1, 2, \ldots, 6$. Since the set $\{1, 2, 3, 4\}$ is a maximal clique in G, as observed at the beginning of the previous section, at least three arcs in $\{a_1, a_2, a_3, a_4\}$ share the same head. Without loss of generality, we can assume that $h(a_1) = h(a_2) = h(a_3)$. But then, it is impossible that arc a_5 is dependent on both arcs a_1 and a_2, and is independent of arc a_3. ■

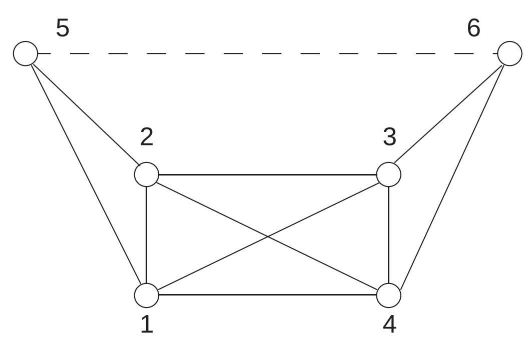

Figure 5: The graph H

Since every induced subgraph of an M-graph is also an M-graph, and since every odd anti-hole with more than seven vertices contains H as induced subgraph, an instant consequence of Lemma 2 is:

Corollary 1. *No M-graph contains odd anti-holes with more than seven vertices.*

Now we are ready to prove the main result of this section.

Theorem 2. *An M-graph is perfect if and only if it does not contain the odd anti-hole with seven vertices and it does not contain odd holes.*

Proof. Let $G = (V, E)$ be an arbitrary M-graph and let D be a directed graph such that G is the M-graph of D. The "only if part" is obvious.

To prove the "if part", assume the contrary: G is not perfect. Then, without loss of generality, we can assume that G is the smallest (with respect to the number of vertices) such a graph, and so G is minimal imperfect. By assumption, G does not contain the odd anti-hole with seven vertices and it does not contain odd holes. Since G does not contain odd anti-holes with more than seven vertices, property (e) implies that $\omega(G) \geq 4$. Now, let u be an arbitrary vertex of G and let $a_u = (x, y)$ be the corresponding arc of D. If the arc $a_v = (y, x)$ exists in D, denote by v the corresponding vertex of G. Let A denote the set of vertices of G corresponding to all arcs of D that enter node x that are different from the arc a_v; let B denote the set of vertices of G corresponding to all arcs of D that enter node y that are different from the arc a_u; and let C denote the set of vertices of G corresponding to all arcs of D that exit node y that are different from the arc a_v. Note that: the sets A and B induce cliques in G and the set C induces a stable set in G; every vertex in C is adjacent to every vertex in B and to at most one vertex in A; every vertex in B is adjacent to no vertex in A; vertex u is adjacent to every vertex in $A \cup B \cup C \cup \{v\}$; and vertex v is adjacent to every vertex in $A \cup B$ and to no vertex in C. Now, let $E = C \cup \{v\}$ if the vertex v exists in G (i.e., if the arc (y, x) exists in D), $E = C$ otherwise. Now, every maximal clique of G, of size at least four, that contains u can only be of two types: the clique $A \cup \{u, v\}$ and the clique $B \cup \{u, w\}$ where w is any vertex in E. Since $\omega(G) \geq 4$, it follows that $|E| \geq 3$ and that the clique $M = B \cup \{u\}$ has size $\omega(G) - 1$ (for otherwise, there would exist at most three maximum cliques containing the vertex u, contradicting property (b)). But then, M along with any three vertices in E violate Lemma 1. Hence the theorem follows. ■

References

[1] P. Avella and A. Sassano, On the P-median polytope, Technical Report 22.94, Università degli Studi di Roma "La Sapienza", Roma, Italy (1994).

[2] E. Balas, The asymmetric assignment problem and some new facets of the travelling salesman polytope on a directed graph, Siam J. Discr. Math. 2 (1989) 425–599

[3] C. Berge, Graphs and Hypergraphs, (North-Holland, Amsterdam, 1973).

[4] V. Chvátal and C. Ebenegger, A note on line digraphs and the directed max-cut problem, Discrete Appl. Math. 29 (1990) 165–170.

[5] G. Cournejols, M.L. Fisher, and G.L. Nemhauser, Location of bank accounts to optimize float: an analytic study of exact and approximate algorithms, Manag. Scie. 23 (1977) 789–810.

[6] G. Cournejols and J.M. Thizy, Some facets of the simple plant location polytope, Math. Progr. 23 (1982) 50–74.

[7] H.L. Crowder and J.M. Mulvey, Cluster analysis: an application of lagrangean relaxation, Manag. Scie. 25 (1979) 329–340.

[8] P.M. Dearing, P.L. Hammer, and B. Simeone, Boolean and graph theoretic formulation of the simple plant location problem, Transp. Scie. 26 (1992) (138–148.

[9] J. Edmonds. Paths, trees and flowers, Canad. J. Math. 17 (1965) 449–467.

[10] R.L. Francis and P.B. Mirchandani, Discrete Location Theory, (Wiley-Interscience, New York, 1989).

[11] M.R. Garey and D.S. Johnson, Computer and Intractability: a Guide to the Theory of NP-Completeness, (Freeman, New York, 1979).

[12] M. Grötschel, L. Lovász, and A. Schrijver, Polynomial Algorithms for Perfect Graphs, in: C. Berge and V. Chvátal eds., Topics on Perfect Graphs, Ann. of Discrete Math. 21 (1984) 325–356.

[13] A. Kraus, C. Janssen, and A. McAdams, The lock-box location problem, a class of fixed charge location problems, J. of Bank Research 1 (1970) 51–58.

[14] M. Padberg. Perfect zero-one matrices, Math. Progr. 6 (1974) 180–196.

[15] A. Tucker, The validity of the perfect graphs conjecture for K_4-free graphs, in: C. Berge and V. Chvátal eds., Topics on Perfect Graphs, Ann. of Discrete Math. 21 (1984) 149–157.

[16] O. Vornberger, Komplexität von Wegeproblemen in Graphen, Ph.D. Thesis, Gesamthochschule Paderborn, Germany, 1980.