JOURNAL OF COMBINATORIAL THEORY, Series B 28, 284–304 (1980)

On Maximal Independent Sets of Vertices in Claw-Free Graphs*

GEORGE J. MINTY

Indiana University, Bloomington, Indiana 47401

Communicated by the Editors

Received June 13, 1977

IN MEMORIAM—PROFESSOR SEYMOUR SHERMAN (1917–1977)

A graph is *claw-free* if: whenever three (distinct) vertices are joined to a single vertex, those three vertices are a nonindependent (nonstable) set. Given a finite claw-free graph with real numbers (weights) assigned to the vertices, we exhibit an algorithm for producing an independent set of vertices of maximum total weight. This algorithm is "efficient" in the sense of J. Edmonds, that is to say, the number of computational steps required is of polynomial (not exponential or factorial) order in n, the number of vertices of the graph. This problem was solved earlier by Edmonds for the special case of "edge-graphs"; our solution is by reducing the more general problem to the earlier-solved special case. Separate attention is given to the case in which all weights are $(+1)$ and thus an independent set is sought which is maximal in the sense of its cardinality.

1. Introduction and Terminology

A *graph*[1] G consists of a finite set V of undefined objects called *vertices*, together with a distinguished collection E of two-element subsets of V called *edges*. (Vertices are sometimes called *nodes*, and edges *branches*.) If $e = (x, y) \in E$, we say *e joins* x and y, or x *is joined to* y, or x *and* y *are joined*, or y *adjoins* x. The *degree* of a vertex is the number of vertices which adjoin it (the number of edges in which it appears.)

In G, an *independent set of vertices* is a subset W of V such that no two elements of W are joined; such a W is called *maximal* if there exists no independent set $W' \subset V$ containing more elements than W. (The words "maximal" and "more" refer to cardinality of sets, not "proper set-containment.")

* This research supported by National Science Foundation Grants MCS 76-07251 and MCS 78-01893. This paper was presented at the Conference in Honour of W. T. Tutte held July 5–9, 1977, at the University of Waterloo, Ontario, Canada.

[1] All graphs treated in this paper will be *finite* graphs except in Section 4.

0095-8956/80/030284–21$02.00/0

A standard problem in graph theory is: "Given a graph G, find (efficiently) a maximal independent set of vertices." (For a general discussion see [3, Chap. 13], where an independent set is called a "stable set,") The notion of "efficiency" is important in this paper, and the word implies essentially that the number of steps (elementary constructions) of the solution-algorithm should be bounded by some polynomial in n, the number of vertices of the graph. For a general discussion of the notion see Lawler [12].

Very little progress has been made on this problem for a general graph G. Let us denote by J_x the *joining-set* of a vertex x, the set of edges of G which join x to other vertices. Then the problem can be rephrased: "Find a maximal collection of disjoint joining-sets J_x in E." This is a "maximal packing problem"—see Beineke [1] for what little information exists on the solution of the general maximal packing problem in graphs.

A related problem is as follows. (For reasons of clarity, we shall adopt the "node/branch" terminology whenever we refer to this problem, however indirectly.) "Find (efficiently) a maximal set of branches such that no two have a node in common (independent)." This problem was solved by Edmonds in his classic paper [7]; for a lucid introductory exposition of his methods, see Busacker and Saaty [5], and for some improvements on the efficiency of his constructions, see Lawler [12]. We shall refer to the construction given in [7] as "Edmonds' Algorithm I."

A notion very similar to that of "graph" is *network*: two (finite) sets of objects called *vertices* and *edges* (or *nodes* and *branches*) together with a function mapping each edge into a two-element set of vertices. Every graph is essentially a network; the principal difference between the notions is that in a "network" several edges may join the same pair of vertices, whereas only one such edge is permitted in a "graph." The distinction is obviously unimportant in the "maximal independent set of vertices" problem, and a little reflection shows it is also unimportant in the "maximal independent set of branches" problem.

We make a careful distinction between "subgraph" and "subnetwork." A *subgraph* of a graph is obtained by deleting a set of vertices (and all the joining edges of these vertices); a *subnetwork* of a network (or graph) is obtained by the same kind of deletions *and* possibly deletion of a set of edges unaccompanied by deletions of vertices.

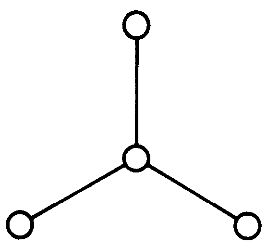
The notions of "connected graph" (or "connected network") and "component" are so standard and unambiguous that we omit their definitions.

A subgraph of the type of Fig. 1a is called a *claw*, and a graph is called *claw-free* if it contains no claw as a subgraph. (Our terminology differs slightly from that of Bose [4], who calls Fig. 1a a "claw of order 3." A claw is sometimes called a "triode" and a claw-free graph "an atriodic graph.")

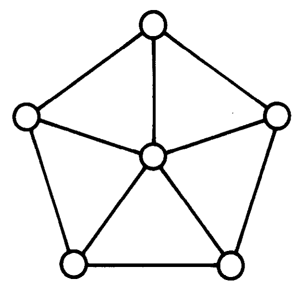
LEMMA 1. *Any subgraph of a claw-free graph is claw-free.*

The proof is left to the reader.

We introduce the notion of the *branch-graph* of a network (or graph)—see Ore [13], where it is called "edge-graph" and van Rooij and Wilf [15] where it is called "interchange graph." To each branch of a network, associate a vertex of the branch-graph; join two vertices by an edge if the two corresponding branches have a node in common.



a



b

FIGURE 1

Now, a branch-graph is claw-free, but Fig. 1b exhibits a claw-free graph which is not the branch-graph of any network. (In order to see this fact most easily, begin the attempted construction with the five "outer" vertices, noting that the five corresponding branches must form a pentagon.) The question "Which claw-free graphs are branch-graphs?" is answered in the paper of van Rooij and Wilf [14].

Two less interesting classes of claw-free graphs are the following: (1) On the real line, consider a finite system of *intervals* such that no interval completely contains another of the system; represent each interval by a vertex, and connect two vertices if the corresponding intervals overlap (see [10]). (2) Consider an arbitrary finite graph. Construct a new graph whose vertices represent the *branches and nodes* of the given graph; connect two vertices by an edge if the two corresponding branches (resp. branch/node pair) are incident on each other. The present paper contributes nothing substantial to the maximal-independent-set-of-vertices problems for classes (1) and (2) of graphs, since these problems can be solved by already-existing methods.

One of the objects of this paper is to give an efficient construction for a maximal independent set of vertices in a claw-free graph. Edmonds' Algorithm I solves this problem for a branch-graph, since a maximal independent set of vertices in a branch-graph obviously corresponds to a maximal independent set of branches in its "underlying graph."

A second, more general, problem solved by Edmonds concerns graphs (or networks) "with weighted branches"—to each branch is assigned a real number called its *weight*, and the problem is to find an independent set of branches of maximum total weight. We shall refer to the construction of Edmonds [6] as "Edmonds' Algorithm II," and note that it solves the problem of Edmonds' Algorithm I as a special case—by taking all branches to have weight (+1). (For a more complete discussion, see Lawler [12] or Edmonds and Pulleyblank [8].) In this paper, we give also an efficient construction for the problem "in a claw-free graph with real-valued weights assigned to the vertices, find an independent set of vertices of maximum total weight."

In one sense, the results of this paper are much more far-reaching than earlier work on "maximal independent sets of vertices." Chapter 13 of Berge [3] states, "No good algorithm is known for determining the stability number of a graph." In the same place may be found the solution of the problem for various special graphs, but there is no good test for determining whether a given graph (given, say, in the form of its incidence matrix) is isomorphic to one of these. On the other hand, the problem of determining whether a graph is claw-free (or a branch-graph, following van Rooij and Wilf) is certainly solved by an obvious polynomially bounded algorithm, so that a graph is easily tested to determine whether the methods of this paper are applicable.

The results of this paper are also "best possible" in a sense. Namely, if it were possible to solve the corresponding problems for graphs free of "claws of order 4" (rather than "claws of order 3"), then one could solve, as a special case, the well-known Three-Dimensional Assignment Problem, which is NP-complete (see Lawler [12]) and for which it is authoritatively conjectured that there is no polynomially bounded algorithm.

The methods proposed in this paper are easily seen to be "polynomially bounded" because the solution of a "maximal independent set of vertices" problem is carried out by solving a small number (polynomially many) of "maximal independent set of branches" problems by means of Edmonds' algorithms, which are known to be polynomially bounded. (A cautionary note: complexity-estimates in the problems of the present paper are to be made in terms of the number of *vertices* of the graph, but our vertices correspond roughly to *branches* in Edmonds' problem.)

2. Preliminary Definitions and Lemmas

By a *Basic Structure* we mean a pair (G, W) consisting of a *claw-free* graph taken together with an independent set of vertices, called the *black vertices*; the remaining vertices are called the *white vertices*. (The same "black/white" convention will be used in our illustrative figures.)

Lemma 2. *A Basic Structure cannot contain either of the structures of Fig. 2 as subnetworks.*

The proof is left to the reader.

By Lemma 2 (Fig. 2a) the white vertices of a Basic Structure can be classified according to the number of black vertices adjoining them; we call a white vertex *bound, free*, or *superfree* according as this number is 2, 1, or 0.

In a Basic Structure, an *alternating path* is a connected subgraph whose white vertices are an independent set. No vertex of an alternating path can have degree exceeding 2 (see Lemma 2 and Fig. 2a for a hint as to the proof). If there is a vertex of degree 1, then there is exactly one other of degree 1; these two vertices are called the *termini* of the alternating path. If there is a vertex of degree 0, then it constitutes the entire alternating path, and references to "the termini" will refer to this vertex with no further apology for the use of the plural form.

Note that a white vertex of an alternating path which is not a terminus cannot adjoin a black vertex which is not in the path, by Lemma 2, Fig. 2a. The reader is invited to draw some pictures and classify alternating paths into convenient categories.

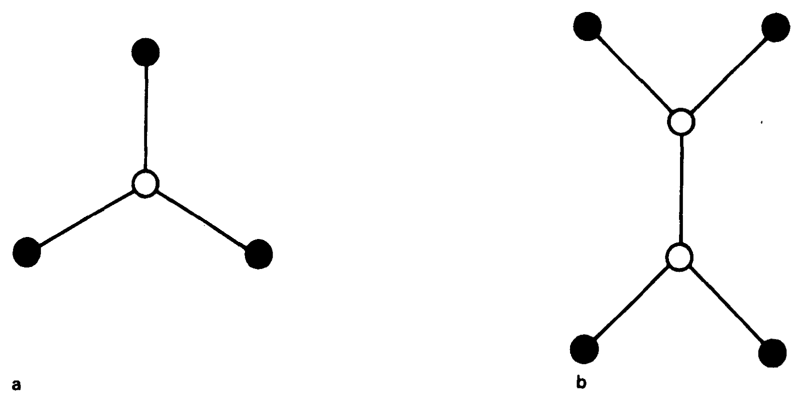


FIGURE 2

A *white alternating path* is an alternating path whose termini are white. Note that in a white alternating path the number of white vertices exceeds the number of black vertices by 1. An *augmenting path* is a white alternating path whose white vertices are joined to no black vertices outside the path—of course, sufficient for a white alternating path to be an augmenting path is that its *termini* are joined to no black vertices outside the path.

We now state a key Lemma.

LEMMA 3. *Consider a Basic Structure (G, W) in which the set of independent vertices W is not maximal. Then there exists an augmenting path in the Basic Structure.*

(Compare with Theorem 2 of Chapter 13 of Berge [3].)

Proof. If W is not maximal, there exists an independent set W' of greater cardinality. Consider the subgraph consisting of the vertices of $W \Delta W'$, the symmetric difference of W and W'. (Otherwise expressed: delete from the graph all vertices which belong to neither set or both sets.) In this subgraph (which need not be connected) the number of vertices of W' exceeds the number of vertices of W, so there is a component in which this inequality holds. This component is the desired augmenting path.

(We shall frequently have occasion, in this paper, to form the symmetric difference of a pair of independent sets, and it is important to notice that the components of the symmetric difference are alternating paths.)

This lemma is the natural generalization to claw-free graphs of a lemma of Berge [2] which was used by Edmonds in his solution of the "maximal independent set of branches" problem. The proof given here is an obvious adaptation of the proof in Busacker and Saaty [5] attributed by these authors to Edmonds [7].

We now explain the terminology "augmenting path." By exchanging the colorations (white and black) of the vertices of an augmenting path, we obtain an independent set of black vertices which has one more black vertex than W has, and which thus brings us one step closer to a maximal independent set of vertices. Edmonds' first contribution to the "maximal independent set of branches" problem was an efficient algorithm (Edmonds' Algorithm I) for finding an augmenting path of branches in that situation.

LEMMA 3. (a) *A single white vertex is an augmenting path if and only if it is superfree (adjoins no black vertex).*

(b) *For $n \geqslant 1$: let $(x_0, x_1, \ldots, x_{2n})$ be a sequence of distinct vertices in which the even-indexed vertices are white and the odd-indexed vertices are black, and which each vertex is joined to its neighbors in the sequence. Then necessary and sufficient that it constitute an augmenting path is: its terminal vertices are an unjoined pair of free vertices, and no two white vertices whose indices differ by 2 are joined. Moreover, any augmenting path with more than one vertex is of this type.*

(c) *Given an augmenting path P in a Basic Structure S, let Basic Structure S' be obtained from S by deletion of white vertices not on P. Then P is an augmenting path in S'. Conversely, if S' is obtained from S by deletion of white vertices, then any augmenting path in S' is an augmenting path in S.*

Parts (a) and (c) are obvious. Part (b) is easy, with the help of Lemma 2, and we leave proof to the reader. We urge him to check carefully the assertions that the white vertices are an independent set and that no white vertex of the path is joined to a black vertex outside the path.

Remark 1. Generally speaking, it is possible to visualize an augmenting path (assume it rather long) in numerous different ways as a sequence (T_0, x_1, T_1,..., x_n, T_n), where the x_i are black vertices and the T_i are white alternating paths. We shall synthesize an augmenting path in this fashion. The following lemma is obvious:

LEMMA 5. *Let* $(T_0, x_1, T_1, x_2, \ldots, x_n, T_n)$ *be a sequence of black vertices* x_i *and white alternating paths* T_i *(note: some* T_i *may be single white vertices) in which*

(A) *the* x_i *are all distinct, no* x_i *appears in any* T_i, *and the* T_i *are pairwise mutually disjoint,*

(B) *the initial terminus of* T_0 *and the final terminus of* T_n *are free vertices not joined to each other,*

(C) *each* x_i *is joined to the final terminus of its predecessor in the sequence and the initial terminus of its successor, and these two termini are not joined to each other.*

Then the sequence is (in an obvious sense) an augmenting path.

We now introduce a technical device for the construction of white alternating paths. A *black alternating trail* is a sequence (x_{-1}, x_0, x_1,..., x_{2n+1}) in which $n \geqslant 0$; all vertices of the sequence are distinct; the even-indexed vertices are white and the odd-indexed vertices are black; each vertex is joined to its predecessor and successor in the sequence; and no two white vertices of the sequence are joined if their indices differ by 2.

LEMMA 6. *The sequence obtained from a black alternating trail by deleting* x_{-1} *and* x_{2n+1} *is a white alternating path.*

The proof is left to the reader.

3. THE "REDUCED BASIC STRUCTURE"

We shall see later that we can give a very simple description of the way an augmenting path "enters and leaves" a black vertex, for all black vertices except the two which are adjacent to the termini of the augmenting path. For these two termini we shall have to resort to a very inelegant trick.

The problem is: "How to construct an augmenting path in a Basic Structure, or alternatively, verify that none exists." It is an easy task to find augmenting paths which consist of one (white) vertex or of three vertices, so let us assume none exist. Now, an augmenting path terminates in an unjoined pair of free white vertices, so if there exists no such pair there is no augmenting path. Assuming such pairs exist, we *select one such pair and attempt to find an augmenting path whose termini are this pair.* We now delete all remaining free (white) vertices and all white vertices adjoining either of the selected pair, since none of these can occur in the sought-for augmenting path.

We are thus led to the concept of a *Reduced Basic Structure*, or *RBS*: a Basic Structure in which (1) there are no superfree white vertices, and precisely two free white vertices, which free pair are not joined to each other, all remaining white vertices being bound, (2) the two free vertices are of degree 1, and (3) the two (black) vertices to which they are joined are distinct. (It is helpful to keep in mind that in a RBS, two white vertices are joined *only if they are bound vertices joined to a common black vertex.*)

We digress on the innocent-looking phrase "select two unjoined free vertices." What this phrase really means is "generate a collection of problems to be solved, each problem corresponding to one such pair." We assure the reader that there will be no further proliferation of problems of this kind—it is the concatenation of proliferations, in the fashion "The problem generates n subproblems, each of which generates $(n-1)$ subproblems, each of which in turn generates $(n-2)$ subproblems ..." which leads to grossly inefficient (factorially many steps) algorithms. Even if each problem leads to only two subproblems, n concatenations leads to the figure 2^n, which is quite unacceptable in the spirit of the present paper.

A number of other efficiency-reductions are possible here but we shall *not* insist that they be made. For example, if the graph of the Reduced Basic Structure is not connected and the two free vertices are in different components, then there is no augmenting path; if they are in the same component, we can delete all other components. If there are two black vertices x and y such that all white vertices joined to x are also joined to y, then x and all the mentioned white vertices can be deleted. (Actually, this reduction will be made later.) We do not make such reductions now because we do not wish to complicate the description of a Reduced Basic Structure.

4. A Useful Theorem

We make a digression at this point to prove a useful theorem in the theory of symmetric relations. (For the motivations see Figs. 4 and 5 and Lemma 7 of Section 5.)

Let S be a nonempty set, R a symmetric relation on S; we write xRy if x is related to y, otherwise $xR'y$. We shall say that *the* T-*property* (*triangle-property*) *holds* for a triple (x, y, z) of elements of S provided an odd number of the statements "xRy, yRz, xRz" hold, i.e., exactly one of these statements holds or all three hold. The idea is obviously independent of the ordering of the triple.

If the T-property holds for *all* triples of elements of S, then it is easy to show that R is an equivalence-relation on S and that there are at most two equivalence-classes. (Reflexivity is shown by consideration of the triple (x, x, x).)

Now let $\{S_\alpha: \alpha \in A$, an index-set$\}$ be a (fixed) partitioning of S into nonempty subsets called "wings." We denote the *principal part* of a symmetric relation R on S by $\bar{R}$, where $x\bar{R}y$ provided x, y are in two different wings and xRy. We note that $\bar{R}$ is symmetric (since R is symmetric) but is not in general reflexive or transitive. ($\bar{R}$ has the function of "discarding all information about R which is relevant only to a single wing.")

We shall say that *R has the* PT-*property* (partition/triangle property) provided the T-property holds for any triple (x, y, z) in three different wings, and note that it is essentially a property of $\bar{R}$.

THEOREM 1. *Let S be a set, $\{S_\alpha\}$ a partitioning of S into at least three (nonempty) subsets, and L a symmetric relation on S enjoying the PT-property.*

Then there exists a unique relation R on S such that (1) $\bar{R} = \bar{L}$, *and* (2) *R has the T-property for every triple (x, y, z), hence is an equivalence-relation partitioning S into at most two equivalence-classes.*

Proof. For each wing S_α and $z \notin S_\alpha$, we define a relation $L_{\alpha,z}$ on S_α as follows: $xL_{\alpha,z}y$ if and only if:

$$\text{either } x\bar{L}z \text{ and } y\bar{L}z \text{ or } x\bar{L}'z \text{ and } y\bar{L}'z$$

(that is to say, by demanding that the T-property hold for the triples (x, y, z)).

The next step is to show that $L_{\alpha,z_1} = L_{\alpha,z_2}$ if z_1, z_2 are in different wings. Several cases must be considered, each of which is treated by two successive applications of the PT-property.

Next, we observe that $L_{\alpha,z_2} = L_{\alpha,z_2}$ even if z_1, z_2 are in the same wing, as follows: x, y are in one wing and z_1, z_2 in another; choose w in still a third wing and observe $L_{\alpha,z_1} = L_{\alpha,w} = L_{\alpha,z_2}$. Thus we can drop the second subscript and refer to the relation as simply L_α.

Note that each relation L_α is symmetric; it is a routine matter to show that it enjoys the T-property, hence is an equivalence-relation on S_α.

We now define $R = \bar{L} \cup (\bigcup_\alpha L_\alpha)$. Obviously R is reflexive (since each L_α is reflexive) and symmetric, and $\bar{L} = \bar{R}$. We pause to point out also that the relation R "was forced on us" at every point of the construction, from which fact the "uniqueness" assertion follows.

Now consider any triple (x, y, z). If the three points are in three different wings, the T-property holds because the PT-property holds for $R = L$. If, say, x and y are in one wing S_α while z is in another wing, it holds by definition of $L_\alpha = L_{\alpha,z}$. If all three are in one wing, it holds because L_α enjoys the T-property. Q.E.D.

A statement virtually equivalent to Theorem 1 is as follows: *With S, $\{S_\alpha\}$ as above, consider the "complete multipartite graph" whose vertex-set is S and in which two vertices are connected by an edge if and only if they are in different sets of the partition. Suppose the edges colored "red" and "green" in such a way that every triangle (note: its vertices are in three different sets of the partition) has an even number of green edges. Then the vertices can be colored with two colors (say, "blue" and "yellow") such that green edges connect unlike-colored vertices and red edges connect like-colored vertices. The coloring is unique (except obviously for interchange of "blue" and "yellow").*

In this form, the theorem is seen to be a relative of theorems of Harary [11] on "signed graphs." L. Lovasz has pointed out that the essence of Theorem 1 is "the triangles generate the circuits."

A note on application: given the truth of the theorem, S can be partitioned into the required equivalence-classes by a simple successive-labeling procedure applied to its elements. (The classes are the connected components of "the graph with red edges.")

Our use for the theorem is based on the fact of the impossibility of the two structures of Fig. 3 as subgraphs of a RBS; the application is easily followed by paying attention to Fig. 4, the corresponding "possible structures."

FIGURE 3

FIGURE 4

5. Classification of Black Vertices

Let us now consider a Reduced Basic Structure (RBS). We shall classify the black vertices.

Consider a black vertex which appears in an augmenting path of length 5 or more. The two adjoining white vertices must themselves be unjoined. We ignore this requirement for the moment and concentrate on another requirement, leading to a concept of "the wing-structure of a black vertex," as follows.

We partition the white vertices adjoining the black vertex under consideration into (nonempty) classes called *wings*. A free (white) vertex is a wing. The remaining adjoining white vertices are partitioned according to which *other* black vertex they are joined to. For any wing (except a free vertex) of the black vertex x, the black vertex y which determines the wing will be called *the tip of the wing*. Note that (a) if x is the tip of a wing of y, then y is the tip of a wing of x, and the two wings coincide; (b) an augmenting path must "enter a black vertex through one wing and leave through another"—an even stronger statement is true if one of the wings is a free white vertex.

If one wing of a black vertex is a free (white) vertex, it will be called a *regular* (black) *vertex of the first kind*. If the vertex is not of this type and has three or more wings, it will be called a *regular vertex of the second kind*. If it has exactly two wings, it will be called an *irregular vertex*, and if it has one wing (or no wings!) it will be called a *useless vertex*. Fig. 5 suggests the structure of "a four-winged regular vertex of the second kind." (Edges joining white vertices are not shown.)

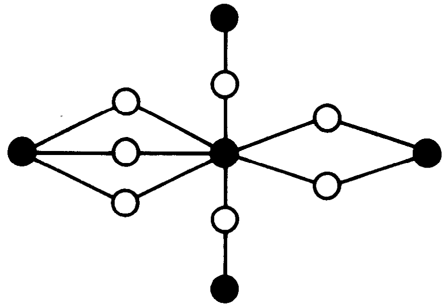


Figure 5

(A note on "increasing the efficiency of our construction," which should properly be a postscript: a "useless vertex" and all adjoining white vertices can be deleted from the RBS. This operation can be iterated until there are no more useless (black) vertices, leaving a simpler RBS.)

(A colleague has encouraged me to use the term "petals" rather than "wings"; I have successfully resisted his suggestion, as it would cause unnecessary confusion in view of Edmonds' preexisting definition of "blossom"!)

LEMMA 7. *Given three (white) vertices in three different wings of a regular vertex of the second kind, the number of joined pairs of these three is odd, and hence the number of unjoined pairs is even.*

Verification is left to the reader. (See Figs. 3 and 4.)

For each *regular* vertex, we now partition the adjoining (white) vertices into two subsets called *node-classes*, as follows.

For a regular vertex of the first kind, put the free vertex into one node-class and all the other adjoining white vertices into the other node-class. For a regular vertex of the second kind: let S be the set of adjoining vertices, and use the wings as the sets S_α of Theorem 1. It is easily verified (see Fig. 4) that the "is joined to" relation on S satisfies the hypotheses of Theorem 1; let the node-classes be the equivalence-classes of the conclusion of the theorem. If, now, any regular vertex has only one node-class, we take "an empty set" conventionally as "its other node-class" (but regard two empty node-classes as distinct if they belong to distinct regular vertices).

(The significance of the partitioning by Theorem 1 is as follows. In the construction of an augmenting path, it is uninteresting to consider whether two white vertices in the same wing are joined or not—they cannot both be in an augmenting path, as already remarked. The node-classes codify another requirement—for *regular vertices*, an augmenting path must "enter through one node-class and leave by the other," since an augmenting path must enter and leave a regular vertex of the second kind through an unjoined pair of white vertices in different wings—see Lemma 4(b). The introduction of $\bar{L}$ signifies: "we *do not care* whether two white vertices in a single wing are joined or not.")

(Efficiency note: a regular vertex with an empty node-class, together with all adjoining white vertices, may be deleted, etc.)

6. IRREGULAR WHITE ALTERNATING PATHS

We next turn our attention to the question "how does an augmenting-path progress from one regular vertex to another?" The mechanism is quite simple, and can be shown by an easy labeling process. An *irregular white alternating path* (IWAP) is a white alternating path all of whose black vertices are irregular vertices, obtained from a black alternating trail with regular termini by deletion of the termini. (See Lemma 6.)

Construct a sequence of vertices and wings as follows. Designate a regular vertex as x_0 and one of its wings as W_1. Call the tip of this wing x_1, and the other wing of x_1, W_2. Call the tip of this wing x_2, and so on until a regular vertex or a one-winged vertex is discovered. If it is a one-winged vertex, clearly the entire structure is useless for the construction of augmenting paths and can be deleted from the RBS (but do not delete x_0). On the other hand, if a regular vertex x_n is discovered, then we have a sequence (x_0, W_1, x_1,..., W_n, x_n), and clearly any IWAP leading from x_0 to x_n whose initial terminus is in W_1 is in a certain sense "contained in" this sequence.

We caution that perhaps $x_0 = x_n$, in which case clearly $n > 2$. This case is of no interest, and the entire structure (except x_0) may be deleted from the RBS.

If $n = 1$, then each (white) vertex in W_1 is itself an IWAP.

If $n > 1$, to determine the terminal vertices of all IWAPs whose initial vertex is in W_1, we construct an Auxiliary Graph: its vertices are the (white) vertices of W_1,..., W_n and two vertices *are* joined by an edge provided (a) they are in successive wings of the sequence, and (b) they *are not* joined by an edge in the RBS (the graph G). Clearly an IWAP with initial vertex in W_1 corresponds to a sequence (w_1,..., w_n) of vertices, where $w_i \in W_i$ and which constitutes a path through the Auxiliary Graph.

Thus for any vertex w_1 joined to a regular vertex, we can discover all vertices w_n such that w_1 and w_n are the termini of an IWAP by discovering "which of the vertices of W_n are accessible from w_1 by such a path through the Auxiliary Graph." The solution of this problem is well known to students of PERT networks. For other readers, we suggest "put an arrow on each edge of the Auxiliary Graph directed from W_i to W_{i+1} and find which vertices of W_n are accessible from w_1 through the resulting network of one-way streets," a problem familiar to students of the Shortest-Route Problem (see Lawler [12]). (The arrows prevent construction of zigzag, i.e., "backtracking" paths.)

7. Construction of the Edmonds'-Graph

We now use the material of the preceding three sections, together with Lemma 5, to find an augmenting path in a RBS by the device of the *Edmonds'-Graph* and Edmonds' Algorithm I.

The *Edmonds'-Graph* is constructed as follows. Assume the RBS has N regular vertices. We form a graph with $(2N + 2)$ nodes and N black branches, each black branch joining two nodes, leaving two isolated nodes. Each black branch is identified with a regular vertex, and its two nodes are to be identified with the two "node-classes" introduced in Section 5 (which were induced on regular vertices of the second kind by application of Theorem 1).

Introduce two "white branches," identified with the two free vertices, joining the two isolated nodes, respectively, to the nodes representing the node-classes of the regular (black) vertices of the first kind to which these free vertices belong.

Now consider two nodes from among the $2N$ nodes which are identified with node-classes. Join them by a white branch if there exists an IWAP (which might consist of a single white vertex) whose terminal vertices are contained in these two node-classes, respectively, and "identify" the white branch with any such IWAP.

The construction of the Edmonds'-Graph is complete.

Now, it is clear that an augmenting path in the RBS can be regarded as a sequence of the kind discussed in Remark 1 and Lemma 5: $(T_0, x_1, T_1, \ldots, x_n, T_n)$, where the x_i are precisely the regular vertices of the augmenting path, T_0 and T_n are the free white vertices, the other T_i are IWAPs, and for each x_i, the predecessor and successor of x_i in the sequence have their final and initial termini, respectively, in the two different node-classes of x_i. Thus, if an augmenting path exists, there exists an augmenting path of branches in the Edmonds'-Graph.

Now, suppose we have discovered an augmenting path of branches in the Edmonds'-Graph. It generates (by the prescribed "identifications" with objects in the RBS) a sort of path through the RBS, which we claim is an augmenting path. We take up the conditions of Lemma 5 point by point.

Condition (B) is clear from the nature of the terminal-branches of Edmonds' augmenting path of branches and the choice of two unjoined free vertices in the definition of RBS.

Condition (C) is clear from the description of the terminal vertices of an IWAP and the fact that Edmonds' augmenting path enters and leaves a black edge through different node-classes as constructed by Theorem 1.

Condition (A) is clear because Edmonds' augmenting path does not pass twice through any black edge, the T_i contain only irregular black vertices while the x_i are regular, and no two white edges connecting the same pair of black edges of the Edmonds'-Graph can appear in Edmonds' augmenting path.

All the above assertions are rather obvious except the very last assertion of (A) above—we urge the reader to check carefully both the truth and the relevance of the assertion.

The main point of this paper is not theorem-proving, but a theorem can serve the purpose of summing up what has been accomplished:

THEOREM 2 (Main Theorem). *Given a RBS: there exists a graph (the*

Edmonds'-Graph) and an independent set of branches in it in 1–1 correspondence with the RBS' regular vertices, and a (in general many-to-one) mapping of the set of augmenting paths of vertices in the RBS onto (!) the set of augmenting paths of branches of the Edmonds'-Graph.

(Note that the domain of this mapping is the *entire* class of augmenting paths in the RBS.)

8. Finding Independent Vertex-Sets of Maximal Total Weight in Claw-Free Graphs with Weighted Vertices

(I am indebted to W. Cunningham of Carleton University and R. Giles of the University of Kentucky, especially the former, for an idea whose exploitation resulted in a radical improvement of this section.)

A *graph with weighted vertices* (or "weighted graph") is a graph taken together with a function assigning a real number (*weight*) to each vertex. We now attack the problem "find (efficiently) an independent set of vertices, the sum of whose weights is greatest possible, in a weighted claw-free graph G," thus generalizing the corresponding problem for branch-graphs (or "graphs with weighted branches") as solved by Edmonds' Algorithm I [6].

(The reader may prefer to delete from the graph all vertices of negative weight, since these cannot appear in a "heaviest independent set," and also those of zero weight, which are in a similarly obvious sense "redundant to the problem." In the discussion to follow, it is not necessary to insist that this reduction have been made, however.)

For this problem, we need to modify our earlier definitions. A *Basic Structure* consists of a weighted claw-free graph together with an independent set of vertices, the "black vertices." An *augmenting path* is a connected subgraph in which the white vertices are an independent set and no white vertex is joined to a black vertex outside the path, and whose total weight (the weight-sum of the white vertices minus the weight-sum of the black vertices) is (strictly) positive.

The number of white vertices need not exceed the number of black vertices, and may in fact be smaller. In an augmenting path, just as before, every vertex is of degree 0, 1, or 2, and we leave to the reader the task of classifying augmenting paths.

Lemma 8. *In a Basic Structure: if the weight-sum of the black vertices is not maximal, there exists an augmenting path.*

Proof. Proceed as in the proof of Lemma 3, letting W' be an independent set of greater total weight, and select a component of $W \Delta W'$ of positive total weight.

We shall need a lemma which is analogous to Lemma 8 but somewhat more detailed:

LEMMA 9. *Let W and W' be two (distinct) independent sets of vertices in a claw-free graph, referred to as "the black vertices" and "the purple vertices," respectively, and let $C_1,..., C_m$ be the components of the subgraph whose vertices are the symmetric difference $W \Delta W'$. Then by exchange of colorations (black and purple) in any subsystem $C_1,..., C_k$ $(k \leqslant m)$ of these components, one obtains new "black" and "purple" sets, each of which is an independent set. The sum of the weights of the C_i $(i = 1,..., m)$, where each weight is computed as "the weight-sum of the purple vertices minus the weight-sum of the black vertices," is the weight-sum of W' minus the weight-sum of W. Moreover, the corresponding statements are true with all vertices taken as having weight* $(+1)$.

The proof, and the translation of the last sentence of the lemma into statements about "numbers of vertices," are left to the reader. The lemma is actually valid in *any* graphs (not necessarily claw-free).

We now let N be the smallest number such that there is an independent set of vertices in G of maximal total weight and having cardinality N, and let w_0, $w_1,...,$ w_N be the sequence of real numbers such that $w_n(n - 1,..., N)$ is the weight of a maximal-weighted set of vertices having cardinality n. (Of course, $w_0 = 0$.) We shall construct a sequence of independent sets W_0, $W_1,...,$ W_N of vertices having weights $w_0,..., w_N$, respectively.

LEMMA 10. *The sequence $w_0, w_1,..., w_N$ is monotone (strictly) increasing.*

Proof. For $n = 0, 1,..., (N - 1)$, we now prove that $w_n < w_{n+1}$ by induction on n. Assume $w_0 < w_1 < \cdots < w_{n-1} < w_n$. Now let W_n be a set of vertices ("the black vertices," all other vertices being "white") with cardinality n and weight w_n, and let W_N' have cardinality N and weight w_N. By the minimality of N we have $w_N > w_n$. Among the components of the subgraph whose vertices are $W_n \triangle W_N'$, there can be none having positive weight (weight of white vertices minus weight of black vertices) in which the number of black vertices is greater than or equal to the number of white vertices, for then by the exchange of colorations (black and white) in this augmenting path we would produce an independent set of $(n - 1)$ or n "black" vertices with weight exceeding w_n, in contradiction with either the induction hypothesis or the maximality of w_n. But by Lemma 8 there exists an augmenting path, in which, therefore, the number of white vertices exceeds (by one) the number of black vertices ("augmenting path with white termini"). It follows that exchange of colorations in this augmenting path produces a "black" independent set W_{n+1}' having $(n + 1)$ vertices and with weight exceeding w_n (but less than or equal to w_{n+1}, by definition of w_{n+1}).

LEMMA 11. *For $n = 0, 1, \ldots, N-1$: suppose W_n is an independent set of "black" vertices with cardinality n and weight w_n. Then there exists an augmenting path P with white termini such that exchange of colorations in P produces a set of "black" vertices W_{n+1} of weight w_{n+1} (and having cardinality $n+1$).*

Proof. Choose an independent set W'_{n+1} of cardinality $(n+1)$ and weight w_{n+1}. The components of $W_n \triangle W'_{n+1}$ can be classified into nine classes, according as their weights are negative, zero, or positive, and according as the number of black vertices is one greater than, equal to, or one less than the number of white vertices.

Four of these classes are necessarily empty, as follows. There can be no component in which there are more black vertices than white and having positive or zero weight, for such a component could be used (as in the proof of Lemma 10) to contradict the inequality $w_n > w_{n-1}$ of Lemma 10. There can be no component in which the numbers of white and black vertices are equal and of nonzero weight; for if such were of positive weight it would be an augmenting path for W_n, in contradiction with the maximality of w_n, and if it were of negative weight it could be similarly applied to W'_{n+1} to contradict the maximality of w_{n+1}.

Furthermore, if a component is of zero weight and has equal numbers of black and white vertices, it is of no interest for the argument to follow. There remain four interesting classes of components:

(a) one more black vertex than white, negative weight,
(b) one fewer black vertex than white, negative weight,
(c) one fewer black vertex than white, zero weight,
(d) one fewer black vertex than white, positive weight.

Let A, B, C, D be the numbers of components in each of these classes, respectively. Then clearly $B + C + D = A + 1$, since in the union of these components there is one more white vertex than black. Now, by Lemma 9 there is a component of positive weight, so there is at least one component of class (d). Call any such component P, and pair off the remaining components of classes (b), (c), and (d) with components of class (a) in any desired manner.

Now, there can be no (a,b) pair or (a,c) pair, since in such a pair the number of black vertices would equal the number of white vertices and the total weight would be negative, and the two components of the pair could be used simultaneously by exchange of coloration ("black and purple") on the "purple" vertices of W'_{n+1} to contradict the maximality of w_{n+1}. It follows that classes (b) and (c) are in fact empty, and that $D = A + 1$.

In similar manner, we see that the total weight of any (a,d) pair must be negative or zero, or the pair could be used to contradict the maximality of

w_n. (We remark parenthetically that such a weight must be positive or zero, or the pair could be used to contradict the maximality of w_{n+1}; hence all such pairs have weight zero.) But the sum of these pair-weights and the weight of P is $(w_{n+1} - w_n)$; thus the weight of P is greater than or equal to $(w_{n+1} - w_n)$, and P is the desired augmenting path. Q.E.D.

(Further parenthetical remarks: P must have weight precisely $(w_{n+1} - w_n)$; but P was an arbitrary component of class (d), hence all components of class (d) have the same weight. Combining this with the earlier parenthetical remark, we see also that all components of class (a) have the same—negative—weight, namely, $(w_n - w_{n+1})$.)

We are now in a position to state the algorithm.

We take W_0, of course, as the empty set. If there is no vertex of positive weight, $W_0 = W_N$ and the algorithm terminates; otherwise, take W_1 as being a single vertex of largest positive weight.

The general recursion-step is now highly analogous to that of the earlier algorithm for the "unweighted case." To construct W_{n+1} from W_n, we seek out the heaviest augmenting path with white termini. Such augmenting paths with one or three vertices are easily listed. To discover candidates with five or more vertices: select a pair of unjoined free vertices joined to distinct black vertices and make the same deletions of white vertices as earlier (all other free vertices, and all white vertices adjoining either of the selected free pair) but in addition delete all superfree white vertices, thus creating a RBS.

For each RBS so created, construct the Edmonds'-Graph as before. For each branch, it is obvious which RBS-entity and weight should be assigned to it, except for the branches corresponding to IWAPs. For each such branch, choose the heaviest possible corresponding IWAP and assign that IWAP and its weight to the branch. (Again, students of PERT or the Shortest-Route Problem—or more properly, the Longest-Route Problem—will have no trouble determining the appropriate IWAP—see Lawler [12].)

Now apply Edmonds' Algorithm II to determine, in each Edmonds'-Graph, a heaviest-possible independent set of branches. If the weight of no such set exceeds w_n and there is no augmenting path of length 1 or 3, then clearly $n = N$ and the algorithm can be terminated with the discovery of $W_n = W_N$, for if $n < N$, the augmenting path whose existence is guaranteed by Lemma 11 corresponds to an augmenting path of branches in the Edmonds'-Graph.

By the same remark, if the heaviest independent set of branches in any Edmonds'-Graph is no heavier than the heaviest independent set of vertices producible in the Basic Structure by use of an augmenting path of length 1 or 3, then this latter set can be taken as W_{n+1}.

In the remaining case, let B'_{n+1} be the former set, and note its weight is (by the same remark) at least w_{n+1}. Form a set of vertices W'_{n+1} in the RBS consisting of (a) the (black regular) vertices corresponding to the black branches of B'_{n+1}, (b) the black (irregular) vertices appearing in IWAPs corresponding to white branches *not* appearing in B'_{n+1}, (c) the white vertices which appear in IWAPs corresponding to white branches appearing in B'_{n+1}, and (d) those free white vertices in the RBS which correspond to white branches appearing in B'_{n+1}.

It is routine but somewhat tedious to verify that W'_{n+1} is an independent set of vertices (keepsing in mind that B'_{n+1} is an independent set of branches, the black vertices of a RBS are an independent set, and two white vertices of a RBS are joined only if both are bound vertices joined to a common black vertex).

The weight of W'_{n+1} is calculated, by formation of $W_n \triangle W'_{n+1}$ and application of Lemma 9, to be the same as the weight of B'_{n+1}. Now, by Lemma 10 (monotonicity of the sequence $\{w_n\}$) W'_{n+1} must contain at least $(n + 1)$ vertices; but it cannot contain more than this number, for otherwise there would be at least two components of $W'_{n+1} \triangle W_n$ with white termini which would be free vertices. Thus, by the maximality of w_{n+1}, the weight of W'_{n+1} must be precisely w_{n+1}.

Clearly we can now choose W_{n+1} as W'_{n+1} or (if it is desired to form W_{n+1} from W_n by the use of a single augmenting path with white termini) by constructing an augmenting path with white termini following the construction implicit in the proof of Lemma 11, and using it to augment W_n.

(Efficiency note: it is a consequence of Lemma 11 that *all* the efficiency-reductions made in the "unweighted case"—such as rejection of RBSs containing no augmenting path with white termini, deletion of "useless" parts of the RBS resp. Edmonds'-Graph, etc.—can also be made in the "weighted case.")

An interesting modification of the above algorithm is as follows. Number the vertices of the graph $1, \ldots, M$, and suppose the weights are $y_1, \ldots, y_M$. Replace these weights by $y_1 + \varepsilon$, $y_2 + \varepsilon^2, \ldots, y_M + \varepsilon^M$, where ε is to be thought of as an "infinitesimally small" positive number. (Alternatively, replace by $y_1 - \varepsilon$, $y_2 - \varepsilon^2, \ldots, y_M - \varepsilon^M$.) The weight of any set of vertices is now "an ordinary number plus a formal polynomial in ε"; thus in comparing two such weights, one first compares the "ordinary numbers" and then, if they are equal, compares the formal polynomials.

The consequences of this modification are left for the reader to work out. (Note that no set of vertices has weight "zero" and that no two sets have the same weight.) Hints on convenient exploitation (replacing the formal polynomials by a lexicographic ordering of n-tuples) can be found in the book of Charnes *et al.* [6].

In conclusion, we now attack the problem of "finding a maximal-weight independent set of vertices containing a prescribed number of vertices." (This number might be larger than N, the number of vertices in a maximal-weight independent set.)

Although vertices of negative weight and zero weight could be deleted before beginning in the solution of the earlier problem, we must now take them into consideration.

The problem is very easily solved by adding a very large number M to the weights of all the vertices, so that *any* set of n vertices is heavier than any set of $(n-1)$ vertices, and proceeding as before. Just as in the "ε-method" described above, the weight of any set of vertices can now be considered to be "a formal (linear) polynomial in M," and weight-comparisons of sets of vertices are to be made in an obvious way, ordering weights "first by the coefficients of M, and secondarily by the constant terms."

Note added in proof. Since this paper was written, the writer has learned that Mlle. Najiba Sbihi of the University of Grenoble (France) has independently solved the problem of this paper (in the unweighted case only) by different methods, and without introduction of the Edmonds'-Graph. See [14].

References

1. L. W. Beineke, A survey of packings and coverings in graphs, *in* "The Many Facets of Graph Theory" (G. Chartrand and S. Kapoor, Eds.), pp. 45–53, Springer-Verlag, Berlin/New York, 1969.
2. C. Berge, Two theorems in graph theory, *Proc. Nat. Acad. Sci. USA* **43** (1957), 842–844.
3. C. Berge, "Graphs and Hypergraphs," North-Holland, Amsterdam, 1973 (translation and revision of "Graphes et Hypergraphes," Dunod, Paris, 1970).
4. R. C. Bose, Strongly regular graphs, partial geometries and partially balanced designs, *Pacific J. Math.* **13** (1963), 389–419.
5. R. G. Busacker and T. L. Saaty, "Finite Graphs and Networks," McGraw–Hill, New York, 1965.
6. A. Charnes, W. W. Cooper, and A. Henderson, "An Introduction to Linear Programming," Wiley, New York, 1953.
7. J. Edmonds, Paths, trees, and flowers, *Canad. J. Math.* **17** (1965), 449–467.
8. J. Edmonds and W. Pulleyblank, "Optimum Matchings," Johns Hopkins Univ. Press, Baltimore, Md., in press.
9. J. Edmonds, Maximum matching and a polyhedron with 0, 1-vertices, *J. Res. Nat. Bur. Standards Sect. B* **69** (1965), 125–130.
10. P. C. Gilmore and A. J. Hoffman, A characterization of comparability graphs, *Canad. J. Math.* **16** (1964), 539–548.
11. F. Harary, On the notion of balance in a signed graph, *Michigan Math. J.* **2** (1953), 143–146.
12. E. L. Lawler, "Combinatorial Optimization—Networks and Matroids," Holt, Rinehart & Winston, New York, 1976.
13. O. Ore, "Theory of Graphs," Colloq. Publ. No. 38, Amer. Math. Soc., Providence, R.I., 1962.

14. N. SBIHI, Algorithme de recherche d'un stable de cardinalité maximum dans un graphe sans étoile, *Discrete Math.*, in press.
15. A. C. M. VAN ROOIJ AND H. WILF, The interchange graph of a finite graph, *Acta Math. Acad. Sci. Hungar.* **16** (1965), 263–269.