# SOME POLYNOMIAL ALGORITHMS FOR CERTAIN GRAPHS AND HYPERGRAPHS

ANDRÁS FRANK
Research Institute for Telecommunication, TKI, Budapest

## 0. §
## Introduction

First a polynomial algorithm is given which finds a stable set with maximal weight in a weighted rigid circuit graph G. The algorithm also proves that G is pluperfect.

Let $\mathcal{F}$ be a family of directed subpaths of an arborescence /directed tree/ F. Polynomial algorithms are presented for the following problem :

1. Find a system of distinct representatives / SDR / of $\mathcal{F}$ . / $O(n\log n)$ steps /.
2. Capacities $c_j$ are assigned to $x_j \in F$. Find a maximal size $\mathcal{F}_c \subset \mathcal{F}$ such that every $x_j \in F$ is contained in at most $c_j$ paths of $\mathcal{F}_c$ .
3. Weights $s_i$ are assigned to $F_i \in \mathcal{F}$. Find a minimal size point set S of F such that S contains at least $s_i$ points from each $F_i \in \mathcal{F}$ .

By means of these algorithms minimax theorems are proved which are not implied directly by the unimodular property and the duality theorem of linear programming. The presented algorithms have two important properties :

1. Similarly to the Kruskal's algorithm /and not like the max-flow one/ these algorithms are not improving that is once an element has been joined to the optimal system, it will never be deleted during the later steps.

2. The results of the algorithms can directly be checked since not only the optimal structure but its optimal dual

PROC. 5TH BRITISH COMBINATORIAL
CONF. 1975, pp. 211-226.

pair is also constructed. Just as the max flow algorithm determines the min cut /of the same value/ as well.

The reader is referred to /3/ for basic terminology / e.g. stable set, clique, arborescence /.

## 1. §

## Rigid circuit graphs

Let G be a simple undirected graph. It is an open question whether there exists a polynomial algorithm for finding a stable set of maximal cardinality in G. However there are classes of graphs where the question is settled. Such classes are the bipartite, comparability and rigid circuit graphs and their complements as well. These graphs are perfect that is the cardinality of the maximal stable set equals to the minimal number of covering cliques and that is true for every induced subgraph.

An algorithm is satisfactory if it yields the maximal stable set furthermore it demonstrates somehow that the resulting set is really a maximal one. In this case one can check the validity of the algorithm on the base of the result. We can hope the existence of such an algorithm if the graph is perfect since a perfect graph has a stable set and a family of covering cliques with the same cardinality. If the algorithm finds the covering cliques, too, the stable set is surely maximal.

It has been proved by L. Lovász /4/ that a perfect graph is pluperfect, that is if we assign nonnegative integer weights to the points of the graph, the maximal weight of a stable set equals to the minimal number of cliques with the property that each point is at least as many times covered by cliques as its weight, and this holds for any induced subgraphs. /The weight of a set is the sum of the weights of its points ./



First an algorithm will be given finding a stable set of maximal weight in a rigid circuit graph.

DEFINITION

An undirected graph G is called a rigid circuit graph if any circuit C of G, $|C| \geq 4$, has a chord.

DEFINITION

A vertex v of G is called rich / simplicial / if the neighbours of v form a clique.

The algorithm requires the following known /2/ lemma presented with a new very simple proof. □

LEMMA

In every rigid circuit graph $G/V,E/$ there exists a rich vertex.

PROOF

G may be supposed to be connected. We show the stronger statement : if G has two nonadjecent vertices then there exist two nonadjecent rich points. /If G is a clique then of course every vertex is a rich one./ Let u and v be two nonadjecent vertices in G, and K is a minimal subset of V which meets every path uv. We prove that K is a clique. The deletion of K disconnects the graph G, and $C_u$ and $C_v$ denote the components containing the vertices u and v respectively. By Menger's theorem there exist $|K|$ disjoint paths between u and v, therefore for any pair x, y from K there exists a path connecting x, y in $C_u$ and another in $C_v$. If we choose the shortest ones they together form a circuit C. C has a chord, by definition, and in this situation this chord is necessarily /x,y/ .

Now the subgraph $G_1$ induced by $C_u \cup K$ is a rigid circuit graph. If $G_1$ is a clique then it contains a rich point

---

□ Made together with P. Kas.

/ of $G_1$ / in $C_u$ otherwise, by induction, there exist two nonadjecent rich points of $G_1$, one of these is in $C_u$. This rich point of $G_1$ is rich in G,too. We receive another rich point in $C_v$ by means of a similar argument. □ □ □

By the previous lemma we can order the vertices $x_1$, $x_2,\ldots x_n$ of G so that $x_i$ is a rich point of the subgraph $G_i$ induced by the subset $x_i, x_{i+1},\ldots x_n$. During the process we go through the vertices of G in this order and mark some of them with red colour, and alter the weights of some vertices. More exactly, in the i-th step we test whether or not the current weight $s_i'$ of $x_i$ is positive. If it is not positive then i=i+1 . Otherwise mark $x_i$ with red colour and reduce the weights of those points of $G_i$ by $s_i'$, which are joined to $x_i$ and let $s_i'=0$. If any weight became negative so alter it to 0.

Finally go back on the red vertices and mark a red one with blue if it is non adjecent to any of the previously marked /blue/ vertices.

THEOREM 1.

A rigid circuit graph is pluperfect. The blue set constructed above is a stable set with maximal weight.

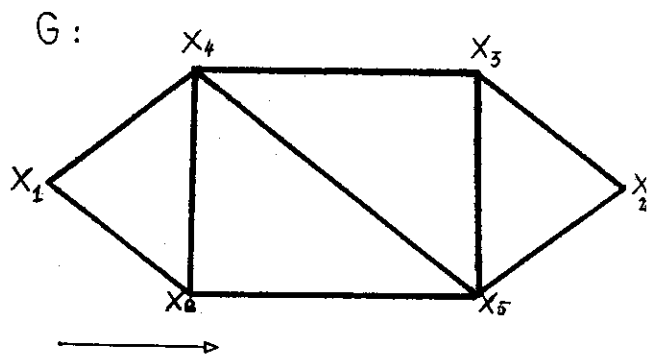
PROOF

The original weight of the blue set is s. We have to verify that there exist s cliques so that these cover every vertex $x_i$ of G $s_i$ times. The vertex $x_i$ and its neighbours of $G_i$ form a clique $C_i$ . We say that $C_i$ belongs to $x_i$ . The sought covering clique system consists of the cliques belonging to the red vertices, taking $s_i'$ times every clique $C_i$, where $s_i'$ is the last positive current weight of the point $x_i$ . This cliqe system covers every point $x_i$ at least $s_i$ times because during the process the current weight became 0. The number of these cliques is s, since a blue point $x_i$ with weight $s_i$ is in exactly



$s_i$ cliques assigning these ones to $x_i$-s, we receive every clique. /If the clique $C_i$ had not been assigned to a blue point then the vertex $x_i$ itself would have been blue. /

**EXAMPLE**




| $s_1$ | $s_2$ | $s_3$ | $s_4$ | $s_5$ | $s_6$ |
|---|---|---|---|---|---|
| 1 | 2 | 2 | 2 | 2 | 4 |
| [0] | 2 | 2 | 1 | 2 | 3 |
| | [0] | 0 | 1 | 0 | 3 |
| | | | [0] | 0 | 2 |
| | | | | | [0] |

The maximal stable set : $\{x_2, x_6\}$ , its weight is 6.

The minimal clique cover : $(x_1,x_4,x_6)$ , $(x_2,x_3,x_5)$ , $(x_2,x_3,x_5)$ , $(x_4,x_5,x_6)$ , $(x_6)$ $(x_6)$

its cardinal is 6.

We note that F. Gavril /1/ gave algorithms to find a clique with maximal weight and a stable set with maximal cardinality in a rigid circuit graph. The first one based on the simple fact that every clique is contained in such a clique which belongs to a vertex. So we have to test only the weights of n cliques belonging to the n vertices of G and choose the maximal one among these.

## 2. §

### Subtrees of tree

Now let F be a tree and $\mathcal{F} = \{F_1, F_2, \ldots F_k\}$ a system of subtrees of F. It is known and easy statement that the line graph $G_{\mathcal{F}}$ of F is a rigid circuit graph. /$G_{\mathcal{F}}$ has k points and two points $y_i$ and $y_j$ are incident in $G_{\mathcal{F}}$ iff $F_i$ and $F_j$ has a point in common. / Assign to every $F_i$ a non-negative integer weight $s_i$ and seek a subsystem of disjoint subtrees of $\mathcal{F}$ with maximal weight. The above algorithm is suitable for this aim, however it has drawbacks : It is needed to produce the line graph and the ordering of points described above. The following algorithm solves the problem directly.

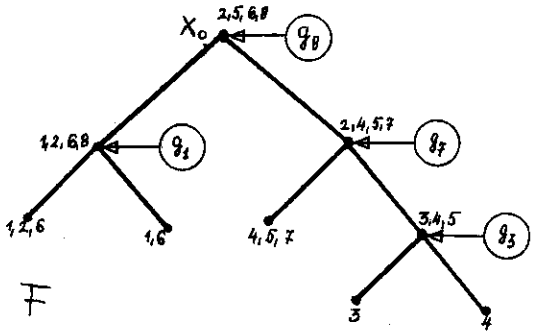
Let $x_0$ be an arbitrary vertex of F and $g_i \in F_i$ the the nearest point to $x_0$. /If $x_0 \in F_i$ then $g_i = x_0$./ During the process we colour some $g_i$ with red and modify the weights. More exactly, let us consider the trees $F_i$ with positive current weight, and choose that one which is the farthest from $x_0$. We colour the $g_i$ of this tree $F_i$ by red and reduce by $s_i'$ the weight of the trees containing $g_i$ . If the weight of any tree is negative thus we alter it to 0.

When every current weight is equal to 0, we go through



on the red points in reversed order and construct the optimal subsystem $\mathcal{F}_1$ as follows. $F_i$ belongs to $\mathcal{F}_1$ if $g_i$ is not contained by any of those $F_j$ 's which have been previously joined to $\mathcal{F}_1$ . The weight of $\mathcal{F}_1$ is s. The s points covering every $F_i$ $s_i$ times are the following : every red point $g_i$ taken $s_i'$ times where $s_i'$ is the last positive current weight of $F_i$.

EXAMPLE




$F_i$ consists of the points indiced by i.

| | $s_1$ | $s_2$ | $s_3$ | $s_4$ | $s_5$ | $s_6$ | $s_7$ | $s_8$ |
|---|---|---|---|---|---|---|---|---|
| | 2 | 3 | [3] | 3 | 4 | 2 | 1 | 3 |
| $g_3$ | 2 | 3 | 0 | 0 | 1 | 2 | [1] | 3 |
| $g_7$ | [2] | 2 | 0 | 0 | 0 | 2 | 0 | 3 |
| $g_1$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | [1] |
| $g_8$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

The disjoint trees : $F_8$ , $F_7$ , $F_3$
their weight : 3 , 1 , 3

The sum of the weights :7

The covering points: $3g_3$ , $g_7$ , $2g_1$ , $g_8$

Their number :7

## 3. §
## Subpaths of arborescence

In the following three problems $F$ is an arborescence whose points are $x_1, x_2, \ldots x_n$ and its root is the vertex $x_1$. Let $\mathcal{F} = \{F_1, F_2, \ldots F_k\}$ a system of directed subpaths of $F$. The initial endpoint of $F_i$ is denoted by $b_i$ and the distance $(x_1, x_i)$ by $d(x_i)$. The indices of points and of subpaths are such that $d(x_i) > d(x_j)$ implies $i > j$ and $d(b_i) > d(b_j)$ implies $i > j$ ! /This condition is very important at the our algorithms/

We mention the subsystem of trees in general is a normal hypergraph while in this special case is a unimodular one, and there are no known algorithms for the following two problems in the first case.

### 3.1 c-independent paths

Assign a nonnegative integer capacity $c_i$ to every point $x_i$ of F, find a c-independent subsystem $\mathcal{F}_c$ of $\mathcal{F}$ for which $|\mathcal{F}_c|$ is maximal .

**DEFINITION**

An $\mathcal{F}_c$ is called c-independent if any point $x_i$ is in at most $c_i$ subpaths from $\mathcal{F}_c$ .

**DEFINITION**

A covering system consists of some vertices of F and all the paths of $\mathcal{F}$ not covered by these vertices. The weight of a covering system is defined as the sum of capacities of its points plus the number of its paths.

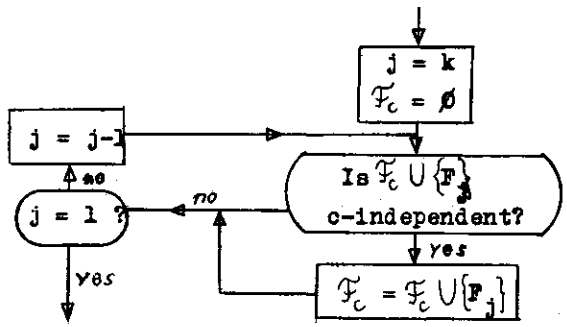
**THEOREM 2**

The maximal cardinal of a c-independent system is equal to the minimal weight of a covering system.

**PROOF**

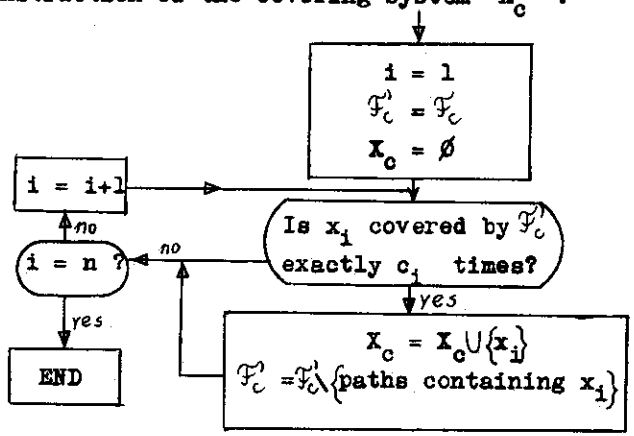
We verify the nontrivial part of the theorem by means of the algorithm.

The construction of the maximal c-independent system :




We go through on the paths of $\mathcal{F}$ in the reversed order and choose all paths such that the choosen paths form a c-independent system.

The construction of the covering system $X_c$ :



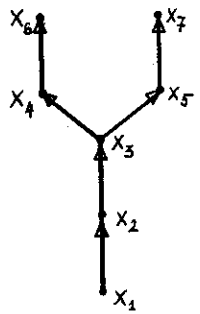

We go through on the points of F in the original order and join one to $X_c$ if it is saturated by those paths of $\mathcal{F}_c$ which are disjoint to $X_c$.

By the construction we have to verify only that the subset $X_c$ covers every path from $\mathcal{F} \setminus \mathcal{F}_c$. Suppose indirectly that $F_i \in \mathcal{F} \setminus \mathcal{F}_c$ is not covered by $X_c$. Then there exist certain points of $F_i$ saturated by $\mathcal{F}_c$ . Let z denote that vertex

among these ones for which $d(z)$ is maximal. The initial endpoints of those paths of $\mathcal{F}_c$ which saturate z are on the path $[b_1, z]$ by the construction of $\mathcal{F}_c$ . /Since $F_i \in \mathcal{F}_c$ would hold otherwise./ But then z should have been in $X_c$. □

EXAMPLE

F:




$F_1 = x_1, x_2, x_3$
$F_2 = x_2, x_3, x_4$
$F_3 = x_3, x_5, x_7$
$F_4 = x_3, x_4$
$F_5 = x_3, x_4, x_6$
$F_6 = x_4, x_6$
$F_7 = x_5, x_7$

The free capacities in the steps :

| | $F_7$ | $F_6$ | $F_5$ | $F_3$ | $F_1$ |
|---|---|---|---|---|---|
| $c_1=2$ | 2 | 2 | 2 | 2 | 1 |
| $c_2=2$ | 2 | 2 | 2 | 2 | 1 |
| $c_3=3$ | 3 | 3 | 2 | 1 | 0 |
| $c_4=2$ | 2 | 1 | 0 | 0 | 0 |
| $c_5=3$ | 2 | 2 | 2 | 1 | 1 |
| $c_6=2$ | 2 | 1 | 0 | 0 | 0 |
| $c_7=2$ | 1 | 1 | 1 | 0 | 0 |

The c-independent system : $F_7, F_6, F_5, F_3, F_1$
Its cardinal : 5

The covering system : $x_3, F_6, F_7$
Its weight /$c_3+2$ / : 5

---

□ I. Ruzsa gave independently a construction of $\mathcal{F}_c$ which is similar to the first part of our one in the special case when F is itself a path.



3.2 s-covering points

Let F and $\mathcal{F}$ have the same meaning. Assign nonnegative integer numbers $s_i$ /= $s(F_i)$/ to every $F_i$ such that $s_i \leq |F_i|$. Determine the minimal size S /which is called a minimal s-covering system of points/ for which $S \cap F_i \geq s_i$, if $1 \leq i \leq k$. How many points are in S ? Let $\mathcal{F}_1$ be a subset of $\mathcal{F}$ . If the paths in $\mathcal{F}_1$ are disjoint the sum of their weights is a lower bound for |S|. Otherwise in order to obtain a valid lower bound the above sum must be reduced as follows.

DEFINITION

Let us define the weight of $\mathcal{F}_1 \subset \mathcal{F}$ by

$$s(\mathcal{F}_1) = \sum_{F \in \mathcal{F}_1} s(F) - \sum_X (c_x - 1)$$

where the second summation runs over those points x of F which are covered $c_x$ /> 1/ times by paths from $\mathcal{F}_1$ .

THEOREM 3

$$\max s(\mathcal{F}_1) = \min |S|$$

where $\mathcal{F}_1 \subset \mathcal{F}$ is arbitrary and S is an s-covering.

PROOF

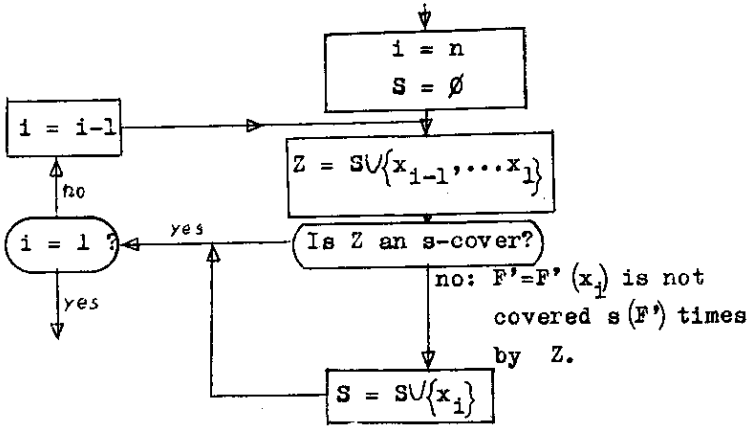
An obvious enumeration leads to max ≤ min. To verify the reversed inequality we have to prove that there exists an s-covering subset S and $\mathcal{F}_1 \subset \mathcal{F}$ such that the following three conditions hold.

1. $|F_i \cap S| = s_i$ when $F_i \in \mathcal{F}_1$.
2. $x \in F_i \cap F_j$ , $\{F_i, F_j\} \subset \mathcal{F}_1$ implies $x \in S$.
3. Every point of S is in some paths from $\mathcal{F}_1$ .

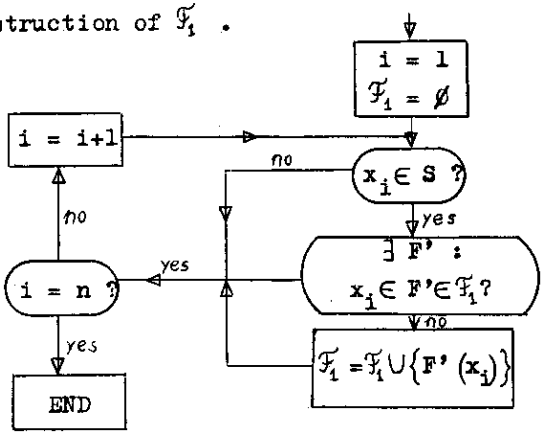
ALGORITHM

The construction of the s-covering S.

We go through on the points in the reversed order and join a vertex $x_i$ into S only if it is required by all means i.e. if not even $Z = S \cup \{x_{i-1}, \ldots x_1\}$ is an s-covering /since there were a path $F' = F'(x_i)$ covered less then $s(F')$ times by Z.




The construction of $\mathcal{F}_1$ .

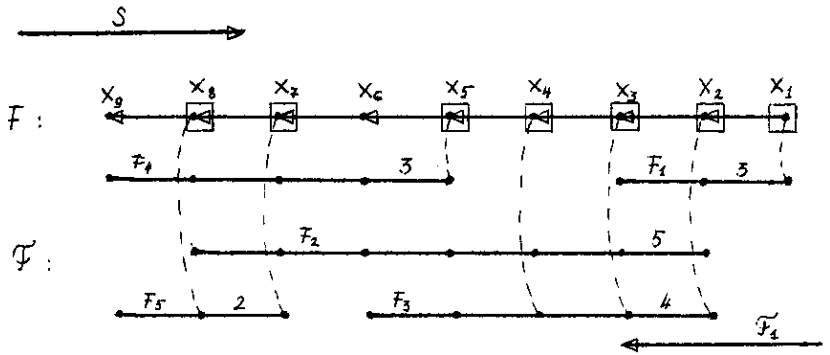



To receive $\mathcal{F}_1$ we go through on the points in the original order and seek the following point $x_i$ of S not covered by paths of $\mathcal{F}_1$ . We take $F'(x_i)$ into $\mathcal{F}_1$ .

It is easy to check the condition 1,2,3 the S and $\mathcal{F}_1$ presented by the above algorithm.



EXAMPLE




The given point is in S because of the path indicated by vertical dotted line .

| | $F_1$ | $F_2$ | $F_3$ | $F_4$ | $F_5$ |
|---|---|---|---|---|---|
| initial point: | $x_1$ | $x_2$ | $x_2$ | $x_5$ | $x_7$ |
| terminal point: | $x_3$ | $x_8$ | $x_6$ | $x_9$ | $x_9$ |
| $s(F_i)$ : | 3 | 5 | 4 | 3 | 2 |

$S = \{x_8, x_7, x_5, x_4, x_3, x_2, x_1\}$ $\quad |S| = 7$

$\mathcal{F}_1 = \{F_1, F_3, F_5\}$ , $s(\mathcal{F}_1) = s_1 + s_3 + s_5 - (c_{x_2} - 1 + c_{x_3} - 1) = 7$

3.3 distinct representatives

Finally we present an algorithm to obtain a system of distinct representatives /SDR/ of the above system $\mathcal{F}$ . This question is completely settled in the case of an arbitrary hypergraph both theoretically and algorithmically. /Hall's theorem and alternating paths./ However the application of this general solution in case of the above special hypergraph was unsatisfactory in certain cases arising from the design of /electrical/ printed circuit boards.

Here we give an $O(n \log n)$ algorithm which significantly differs from the solution of the general problem. In fact, the general algorithm does not have property 1. described

in the introduction.

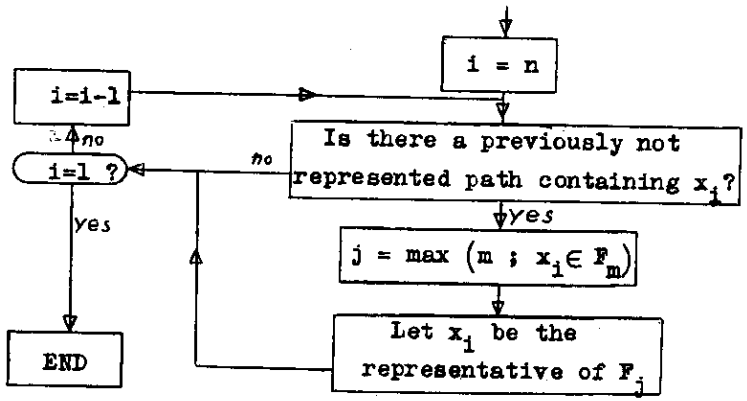
Hall's theorem states : An arbitrary hypergraph $\mathcal{H}/V,E/$ has an SDR iff for any $X \subset V$ the number of hyperedges in X is at most $|X|$. In our case the seemingly weaker condition will also be sufficient.

THEOREM 4

The hypergraph $\mathcal{F}$ has an SDR iff for any subarborescence F' of F the number of paths in F' is at most $|F'|$.

PROOF

The necessity is trivial. The following algorithm yields either the SDR or the subarborescence violating the condition.



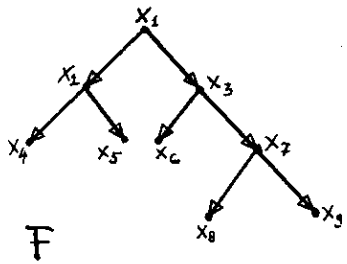

If at the end of this part of the algorithm is a nonrepresented path $F_i$ of $\mathcal{F}$ then we assigned every point of $F_i$ to such a path whose initial endpoint is not nearer to the root $x_1$ than $b_i$ is. /Here, as usual $b_i$ denote the initial endvertex of $F_i$./

Now we construct a subarborescence F' violating the condition. The root of F' will be $b_i$. We consider all the branches beginning at $b_i$ and delete from each branch the first vertex v with the property that either v does not represent any path or it represents such a path whose



initial endvertex is nearer to $x_1$ than $b_i$ is. The F' consists of vertices which can be reached from $b_i$. It is easy to check that F' contains the path $F_i$ and all paths of $\mathcal{F}$ represented by its vertices.

EXAMPLE



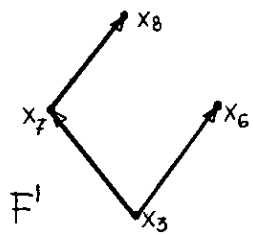

| | $x_1$ | $x_2$ | $x_3$ | $x_4$ | $x_5$ | $x_6$ | $x_7$ | $x_8$ | $x_9$ |
|---|---|---|---|---|---|---|---|---|---|
| $F_1$ | □ | | □ | | | | □ | □ | |
| $F_2$ | □ | | □ | | | | □ | | □ |
| $F_3$ | | □ | | | □ | | | | |
| $F_4$ | | □ | | □ | | | | | |
| $F_5$ | | | □ | | | | □ | | |
| $F_6$ | | | □ | | | □ | | | |
| $F_7$ | | | □ | | | | □ | □ | |
| $F_8$ | | | | | | | □ | □ | |
| $F_9$ | | | | | | | □ | □ | |

The incidence matrix of $\mathcal{F}$ .

The SDR given by the algorithm :

| $x_9$ | $x_8$ | $x_7$ | $x_6$ | $x_5$ | $x_4$ | $x_3$ | $x_1$ |
|---|---|---|---|---|---|---|---|
| $F_2$ | $F_9$ | $F_8$ | $F_6$ | $F_3$ | $F_4$ | $F_7$ | $F_1$ |

The path $F_5$ is not represented. The arborescence F' violating the condition :




$|F'| = 4$ and F' contains five paths : $F_9, F_8, F_6, F_7, F_5$

REFERENCES


1. F. Gavril, " Algorithms for minimum coloring, maximum clique, minimum covering by cliques, and maximum independent set of a chordal graph ", SIAM J. Comput.Vol.1,No.2. 1972
2. G.A. Dirac," On rigid circuit graphs ", Abh.Math.Sem. Univ.Hamburg 25 /1961/ ,71-76.
3. C. Berge ," Graphs and Hypergraphs ", NORTH-HOLLAND 1973
4. L. Lovász, " Normal hypergraphs and the perfect graph conjecture ", Discrete Math, 1972. Vol.1 (3)
5. I. Ruzsa , Oral communication



TKI, 1026 Budapest
Gàbor Aron u. 65
HUNGARY