ACYCLIC SUBDIGRAPHS AND LINEAR ORDERINGS: POLYTOPES, FACETS, AND A CUTTING PLANE ALGORITHM

M. Grötschel, M. Jünger & G. Reinelt

Universität Augsburg
Institut für Mathematik
Lehrstuhl für Angewandte Mathematik II
Memminger Str. 6
8900 Augsburg
West Germany

ABSTRACT

We study the acyclic subdigraph problem and the linear ordering problem from a polyhedral point of view. Insights into the facet structure of polytopes associated with these problems lead to the formulation and implementation of a cutting plane algorithm for the linear ordering problem.

I. Rival (ed.), Graphs and Orders, 217–264.

I. Introduction and Notation

The acyclic subdigraph problem (ASP) can be formulated as follows. Given a digraph D with arc weights, find a set of arcs containing no directed cycle and having maximum total weight. In Section II we investigate this NP-hard problem from a polyhedral point of view and determine several classes of facets for the associated acyclic subdigraph polytope $P_{AC}(D)$. These facets are induced by Dicycle Inequalities, Möbius Ladder Inequalities and Fence Inequalities.

In Section III we show that the separation problem for the facet defining dicycle inequalities can be solved in polynomial time. This implies that the acyclic subdigraph problem can be solved in polynomial time for weakly acyclic digraphs. Since planar digraphs are weakly acyclic this generalizes a result of Lucchesi.

The ASP is a combinatorial optimization problem with a large number of applications (triangulation of input-output matrices, archeological seriation, minimizing total weighted completion time in one-machine scheduling, aggregation of individual preferences etc.). It is often formulated as a linear ordering problem (LOP) asking for a spanning acyclic tournament of maximum total weight in the complete digraph D_n on n nodes. This problem is directly (polynomial time) equivalent to the ASP. The associated linear ordering polytope P^n_{LO} is an $\binom{n}{2}$-dimensional face of $P_{AC}(D_n)$. In Section IV we investigate which of the facets determined for $P_{AC}(D_n)$ are also facets of P^n_{LO}.

The partial knowledge of the facet structure of P^n_{LO} gives rise to the formulation of an algorithm for the linear ordering problem which is described in Section V. The main part of this algorithm is a cutting plane procedure using facet defining inequalities. This is combined with various heuristics and branch & bound techniques.

In Section VI we present some numerical results obtained with a computer implementation of our new algorithm. These data are based on optimum triangulations of a large number of input-output-matrices compiled by statistical offices of the European Community.

We conclude with a short discussion of research problems in Section VII.

The ASP often appears in the literature in an equivalent formulation under the name *feedback arc set problem (FASP)*. A *feedback arc set* in a digraph D is a subset of the arc set A intersecting each directed cycle in D. Each instance of the FASP consists of a directed graph D and a weight function on the arcs of D, and the objective is to determine a feedback arc set of minimum weight.

Clearly, if F is a feedback arc set in $D = (V,A)$, then $A \setminus F$ is acyclic, and conversely, if $B \subseteq A$ is acyclic, then $A \setminus B$ is a feedback arc set in D. In particular, an acyclic arc set $B \subseteq A$ of maximum weight determines a feedback arc set $A \setminus B$ of minimum weight and vice versa, so the ASP and the FASP are nothing but different formulations of the same problem.

We assume that the reader is familiar with the basic concepts of graph theory. We shall only consider simple graphs $G = [V,E]$ and digraphs $D = (V,A)$ (without loops or parallel edges resp. arcs) with node set V and edge set E resp. arc set A. We are mainly concerned with digraphs. The following notation will be used.

If $a = (u,v)$ is an arc of the digraph $D = (V,A)$ then a is said to be *incident from* u and *incident to* v, or u is the *tail* and v is the *head* of a. We also say that the arc $a = (u,v)$ goes from u to v, and u and v are the endnodes of a. If X and Y are disjoint subsets of V then the set of arcs with tail in X and head in Y is denoted by $(X:Y)$. Two nodes $u,v \in V$ are called *adjacent* in a digraph $D = (V,A)$ if $(u,v) \in A$ or $(v,u) \in A$. A digraph $D = (V,A)$ is *complete* if for any two nodes, $u,v \in V$, $u \neq v$, the set A contains the arc (u,v) and the arc (v,u). The (up to isomorphism) unique complete digraph with n nodes is denoted by D_n. For $D = (V,A)$ and $V' \subseteq V$ we define $A(V') := \{(u,v) \in A \mid u,v \in V'\}$ and for $A' \subseteq A$ we define $V(A') := \{u,v \in V \mid (u,v) \in A'\}$.

If we take a graph $G = [V,E]$ and assign a direction to each of its edges, i. e. we define an arc set A on V such that for each edge $\{u,v\} \in E$, the set A contains exactly one of the arcs (u,v), (v,u), then $D = (V,A)$ is called an *orientation* of $G = [V,E]$.

For $v \in V$ the number $deg^-(v) := |\{(u,v) \in A \mid u \in V\}|$ of arcs entering v is called the *indegree* of v, the number $deg^+(v) := |\{(v,u) \in A \mid u \in V\}|$ of arcs leaving v is called the *outdegree* of v and the number $deg(v) = deg^-(v) + deg^+(v)$ is the degree of v.

If $D = (V,A)$ and $D' = (V',A')$ are digraphs such that $V' \subseteq V$ and $A' \subseteq A$ then we call D' a *subdigraph* of D and D a *superdigraph* of D'. We also say that D' is *contained* in D. For $D = (V,A)$ and $V' \subseteq V$, $A' \subseteq A$ the digraph $D' = (V',A(V'))$ is called a *node-induced subdigraph* of D and $D'' = (V(A'),A')$ an *arc-induced subdigraph* of D.

A set of arcs $P = \{(v_1,v_2),(v_2,v_3),\dots,(v_{n-1},v_n)\}$ is called a *dipath* (or a (v_1,v_n)-dipath) if $v_i \neq v_j$ for $i \neq j$. The *length* of a dipath is the number of its arcs. A set of arcs $C = \{(v_1,v_2),(v_2,v_3),\dots,(v_{n-1},v_n),(v_n,v_1)\}$ with $v_i \neq v_j$ for $i \neq j$ is called a *dicycle* (or *n-dicycle*). The length of a dicycle C (a dipath P) is denoted by $|C|$ $(|P|)$.

A digraph or arc set is called *acyclic* if it contains no dicycle. A *tournament* $T = (V,A)$ is a digraph containing for any two nodes $u,v \in V$ either arc (u,v) or arc (v,u) but not both. In the sequel we shall also call the arc set A a tournament (assuming that the set of nodes is given implicitly).

Clearly, an acyclic digraph D induces a partial ordering on the nodes of D, and a spanning acyclic tournament D induces a linear ordering on the nodes of D. Vice versa, every linear ordering of the nodes of the complete digraph D_n gives rise to a spanning acyclic tournament contained in D_n.

Let $D = (V,A)$ be a digraph. D is said to be *connected* if its underlying graph is connected, otherwise D is called *disconnected*. D is *strongly connected* if for each pair u,v of nodes there exist a (u,v)-dipath and a (v,u)-dipath in D. A node v is called an *articulation node* if the removal of v and all arcs incident with v disconnects the digraph.

As usual, the complete graph on n nodes is denoted by K_n. A graph $G = [V,E]$ is *bipartite* if its node set V can be partitioned into two nonempty disjoint sets V_1, V_2 with $V_1 \cup V_2 = V$ such that no two nodes in V_1 (resp. no two nodes in V_2) are connected by an edge. If $|V_1| = m$ $|V_2| = n$ and $E = \{\{i,j\} \mid i \in V_1, j \in V_2\}$ we call G the *complete bipartite graph* and denote it by $K_{m,n}$.

Finally, a graph or digraph is called *planar* if it can be drawn in the plane such that no two arcs intersect.

To be able to apply methods of linear algebra to graph theory we associate vectors to arc sets in the following way: Let $D = (V,A)$ be a digraph. If $|A| = m$ we denote by $\mathbb{R}^A$ the m-dimensional real vector space, for which the components of the vectors $x \in \mathbb{R}^A$ are indexed by the arcs $(i,j) \in A$. For convenience we denote a component by x_{ij} or x_e if $e = (i,j)$. The *incidence vector* $x^B \in \mathbb{R}^A$ of an arc set $B \subseteq A$ is defined by setting $x^B_{ij} = 1$ if $(i,j) \in B$ and by setting $x^B_{ij} = 0$ otherwise. Incidence vectors for edge sets of undirected graphs are defined in the same way.

If $c : A \to \mathbb{R}$ is a *weight function* on the arcs of a digraph $D = (V,A)$, the weight of a set of arcs $B \subseteq A$ is

$$c(B) = \sum_{(i,j)\in B} c_{ij} .$$

Similarly if we associate a variable x_{ij} to each arc (i,j) we denote by $x(B)$ the formal sum of the variables belonging to the arcs of B.

A polyhedron $P \subseteq \mathbb{R}^m$ is the intersection of finitely many half-spaces in $\mathbb{R}^m$. A *polytope* is a bounded polyhedron or equivalently the convex hull of finitely many points. We denote the convex hull of a set $S \subseteq \mathbb{R}^m$ by conv(S). The dimension of a polyhedron P, denoted by *dim P* is the maximum number of affinely independent points in P minus one.

If $a \in \mathbb{R}^m \setminus \{0\}$, $a_0 \in \mathbb{R}$, then the inequality $a^T x \leq a_0$ is said to be *valid* with respect to a polyhedron $P \subseteq \mathbb{R}^m$ if $P \subseteq \{x \in \mathbb{R}^m \mid a^T x \leq a_0\}$. We say that a valid inequality $a^T x \leq a_0$ defines a *face* of P if $\emptyset \neq P \cap \{x \mid a^T x = a_0\} \neq P$. A valid inequality $a^T x \leq a_0$ defines a *facet* of P if it defines a face of P and if there exist $\dim P$ affinely independent points in $P \cap \{x \mid a^T x = a_0\}$. Two face-defining inequalities $a^T x \leq a_0$, $b^T x \leq b_0$ are *equivalent* if $P \cap \{x \mid a^T x = a_0\} = P \cap \{x \mid b^T x = b_0\}$.

A polyhedron $P \subseteq \mathbb{R}^m$ is called *full-dimensional* if $\dim P = m$. For every full-dimensional polyhedron there exists an inequality system $Ax \leq b$ with $P = \{x \mid Ax \leq b\}$ which is unique up to multiplication by a positive constant. If P is not full-dimensional then P is contained in the intersection of hyperplanes, i. e. P has a representation of the form $P = \{x \mid Ax \leq b,\ Dx = d\}$.

If $P = \{x \mid Ax \leq b,\ Dx = d\}$ then we say that the system $Ax \leq b$, $Dx = d$ is *complete* for P. If D has full rank and $\{x \mid Dx = d\}$ is the affine space spanned by P, then $Dx = d$ is called a *minimal equation system* for P. A complete system $Ax \leq b$, $Dx = d$ for P is called *nonredundant* with respect to P if $Dx = d$ is a minimal equation system and if the deletion of any inequality of $Ax \leq b$ results in a polyhedron larger than P. It is known that in such a case for every facet of P the system $Ax \leq b$ contains exactly one inequality defining it, i. e. every inequality of the system $Ax \leq b$ defines a facet of P and no two inequalities are equivalent.

When dealing with polyhedra, especially algorithmically, we are naturally interested in minimal and nonredundant descriptions by linear equations and inequalities.

Of special interest for our purposes are *0-1-polytopes* which are polytopes that can be defined by $P = \text{conv}(S)$, $S \subseteq \{0,1\}^n$. We call facets of a polyhedron $P \subseteq \mathbb{R}^n$ which can be defined by an inequality

$x_i \geq \beta$ or $x_i \leq \alpha$ for some $i \in \{1,2,\ldots,n\}$ and $\alpha,\beta \in \mathbb{R}$ the *trivial facets of* P, and the inequalities defining them *trivial inequalities*.

Introductions to the kind of polyhedral theory we are interested in, can be found in BACHEM and GRÖTSCHEL (1982), SCHRIJVER (1984), PULLEYBLANK (1983).

We shall not discuss the theory of computational complexity here. An excellent introduction and at the same time comprehensive survey of computational complexity theory is the book GAREY and JOHNSON (1979).

Most results presented in the following sections are stated without proofs. These can be found in GRÖTSCHEL, JÜNGER & REINELT (1982a,b), JÜNGER (1984), REINELT (1984).

II. The Acyclic Subdigraph Polytope $P_{AC}(D)$

We shall now study the facial structure of the class of acyclic subdigraph polytopes $P_{AC}(D)$ which are polytopes associated with digraphs $D = (V,A)$ such that the vertices of $P_{AC}(D)$ correspond bijectively to the acyclic arc sets $B \subseteq A$. In particular, we shall derive several classes of facets of these polytopes. To avoid some degenerate situations we assume throughout the following that all digraphs considered contain at least one arc.

We define

(II.1) $$P_{AC}(D) := \text{conv}\{x^B \in \mathbb{R}^A \mid B \subseteq A \text{ is an acyclic arc set in } D = (V,A)\} .$$

With this definition we can formulate the Acyclic Subdigraph Problem as the linear program

(II.2) $$\text{maximize} \quad c^T x \qquad \text{subject to} \quad x \in P_{AC}(D)$$

where $c \in \mathbb{Z}^A$ is a weight function.

As pointed out in Section I, our aim is to derive large classes of facets of $P_{AC}(D)$ in order to be able to apply linear programming techniques. First we shall summarize some trivial properties of $P_{AC}(D)$.

$P_{AC}(D)$ contains the zero vector (the incidence vector of the empty set) and all unit vectors (the incidence vectors of single arcs) in $\mathbb{R}^A$. These $|A| + 1$ vectors are clearly affinely independent, so $\dim(P_{AC}(D)) = |A|$ for any digraph $D = (V,A)$.

It is immediate from the definition that $0 \leq x \leq y \in P_{AC}(D)$ implies $x \in P_{AC}(D)$. From this and the full-dimensionality we can conclude that there is a matrix A and a vector b such that $P_{AC}(D) = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ and each inequality $d^T x \leq d_0$ in the system $Ax \leq b$ defines a facet of $P_{AC}(D)$ and $d^T x \leq d_0$ is unique up to multiplication by a positive constant. Moreover, if $d^T x \leq d_0$ defines a nontrivial facet of $P_{AC}(D)$ then we additionally have $d \geq 0$ and $d_0 > 0$.

A straightforward observation is the following. Let $D = (V,A)$ be a digraph and let the inequality $d^T x \leq d_0$ define a facet of $P_{AC}(D)$ for some $d \in \mathbb{R}^A_+$, $d_0 > 0$. Define $A' := \{(u,v) \mid (v,u) \in A\}$ and $D' = (V,A')$. Let $d' \in \mathbb{R}^A_+$, $d'_{(u,v)} := d_{(v,u)}$. Then $d'^T x \leq d_0$ defines a facet of $P_{AC}(D')$. This is trivial, since $\{x \in P_{AC}(D) \mid d^T x = d_0\} = \{x \in P_{AC}(D') \mid d'^T x = d_0\}$.

For a digraph $D = (V,A)$ and a vector $d \in \mathbb{R}^A_+$ we define the *support* of d in D by $D_d = (V(A_d),A_d)$ where $A_d := \{a \in A \mid d_a > 0\}$. The following lemma states some properties of facet defining inequalities for $P_{AC}(D)$.

Lemma II.3. *Let $D = (V,A)$ be a digraph and let the inequality $d^T x \leq d_0$ define a nontrivial facet of $P_{AC}(D)$ for some $d \in \mathbb{R}^A_+$, $d_0 > 0$. Let $D_d = (V(A_d),A_d)$ be the support of d in D.*

(i) D_d is strongly connected.

(ii) D_d is connected and contains no articulation node (i. e. is 2-connected). ▯

We shall now state two lemmata which are very helpful for proving that the inequalities to be studied in the following define facets of $P_{AC}(D)$ for certain digraphs $D = (V,A)$.

If $a = (u,w) \in A$ is an arc of $D = (V,A)$ and $v \notin V$, then the digraph $D' = (V',A')$ where $V' = V \cup \{v\}$ and $A' = (A \setminus \{(u,w)\}) \cup \{(u,v),(v,w)\}$ is called the digraph obtained from D by subdividing the arc $a \in A$.

Lemma II.4. (The Subdivision Lemma) *Let $D = (V,A)$ be a digraph and $d^T x \le d_0$ be a nontrivial facet defining inequality for $P_{AC}(D)$. Let $D' = (V',A')$ be the digraph obtained from D by subdividing the arc $(i,k) \in A$ into the arcs $(i,j),(j,k) \in A'$. Set*

$$d'_{uv} := d_{uv} \quad \text{for all} \quad (u,v) \in A \cap A' ,$$

$$d'_{ij} := d'_{jk} := d_{ik} ,$$

$$d'_0 := d_0 + d_{ik} .$$

Then the inequality $d'^T x \le d'_0$ defines a facet of $P_{AC}(D')$. □

The Subdivision Lemma provides a method to derive facets of $P_{AC}(D)$ whose defining inequalities have arbitrarily large support. An immediate question arises, namely, is a "converse" statement true, i. e. is it possible to derive a method to obtain facets of $P_{AC}(D')$ from nontrivial facets of $P_{AC}(D)$ by "contracting" arcs in D ? It turns out that this is indeed the case.

Given a digraph $D = (V,A)$ and $a = (u,v) \in A$ we say that the digraph $D' = (V',A')$ obtained from D by identifying the nodes u and v and then removing loops and parallel arcs is obtained from D by *contracting* the arc $a \in A$.

Let $D = (V,A)$ and $v \in V$ satisfy $\deg^-(v) = \deg^+(v) = 1$. First of all, it is clear that for any nontrivial facet of $P_{AC}(D)$ defined by an inequality $d^T x \le d_0$ we must have $d_{uv} = d_{vw}$, where u and w are the unique nodes adjacent to v. To see this, suppose without loss of generality that $d_{uv} > d_{vw}$. Since $d^T x \le d_0$ defines a nontrivial facet of $P_{AC}(D)$, we know $d_{vw} \ge 0$. Furthermore, there must be an acyclic arc set $B \subseteq A$ whose incidence vector x^B satisfies $d^T x^B = d_0$, $x^B_{vw} = 1$ and $x^B_{uv} = 0$, because $x^B_{uv} = x^B_{vw} = 0$ would imply that $d^T x \le d_0$ is not valid for $P_{AC}(D)$ since $B' = B \cup \{(u,v)\}$ is an acyclic arc set in D with $d^T x^{B'} > d_0$, and $x_{uv} = 1$ for all $x \in P_{AC}(D)$

satisfying $d^Tx = d_0$ would contradict the facet defining property of $d^Tx \leq d_0$. But now the set $\hat{B} \subseteq A, \hat{B} := (B \setminus \{(v,w)\}) \cup \{(u,v)\}$ is also acyclic and $d^Tx^{\hat{B}} = d_0 - d_{vw} + d_{uv} > d_0$, contradicting the validity of $d^Tx \leq d_0$. Thus we have established $d_{uv} = d_{vw}$.

Now we can formulate a counterpart to the Subdivision Lemma.

Lemma II.5. (The Contraction Lemma) *Let $D = (V,A)$ be a digraph, $(i,j),(j,k) \in A$, $i \neq k$, $deg^-(j) = deg^+(j) = 1$ and suppose $d^Tx \leq d_0$ defines a nontrivial facet of $P_{AC}(D)$. Let $D' = (V',A')$ be the digraph obtained from D by contracting the arc (i,j) (or equivalently the arc (j,k)). Set*

$$d'_{uv} := d_{uv} \quad \text{for all } (u,v) \in A \cap A'$$

$$d'_{ik} := d_{ij} \quad (=d_{jk})$$

$$d'_0 := d_0 - d_{ij}.$$

Then the inequality $d'^Tx \leq d'_0$ defines a facet of $P_{AC}(D')$. □

We are now ready to present four classes of facet defining inequalities for $P_{AC}(D)$ all of whose coefficients are either 0 or 1.

First of all, it is easy to determine which of the trivial inequalities of the form $x_a \geq 0$, $x_a \leq 1$ for $a \in A(D)$ define facets of $P_{AC}(D)$.

Theorem II.6. (Trivial Facets) *Let $D = (V,A)$ be a directed graph.*

(i) The inequality $x_a \geq 0$ defines a facet of $P_{AC}(D)$ for all $a \in A$.

(ii) The inequality $x_a \leq 1$ defines a facet of $P_{AC}(D)$ for $a = (u,v) \in A$ if and only if $\bar{a} = (v,u) \notin A$.

□

By definition, an acyclic arc set $B \subset A$ of a digraph $D = (V,A)$ contains no dicycle, in other words, $|B \cap C| \leq |C| - 1$ for every dicycle $C \subseteq A$. This immediately implies that the inequality

$$\text{(II.7)} \qquad x(C) := \sum_{(i,j) \in C} x_{ij} \leq |C| - 1$$

is a valid inequality for $P_{AC}(D)$ for each dicycle $C \subseteq A$. If C is a k-dicycle in D we call the inequality $x(C) \leq k-1$ a *k-Dicycle Inequality*. The validity of the k-Dicycle Inequalities for $P_{AC}(D)$ for $k \geq 2$ implies the inclusion

(II.8) $$P_{AC}(D) \subseteq P_C(D) := \{x \in \mathbb{R}^A \mid 0 \leq x_{ij} \leq 1 \text{ for all } (i,j) \in A, \; x(C) \leq |C| - 1 \text{ for all dicycles } C \subseteq A \text{ in } D = (V,A)\}.$$

More importantly, but trivial to prove, we have

(II.9) $$P_{AC}(D) = \text{conv}\{x \in P_C(D) \mid x \in \{0,1\}^A\}$$

which means that the problem

(II.10) $$\begin{aligned} &\text{maximize} && c^T x \\ &\text{subject to} && x(C) \leq |C| - 1 \quad \text{for all dicycles } C \subseteq A \\ &&& 0 \leq x \leq 1 \\ &&& x \text{ integral} \end{aligned}$$

is an integer programming formulation of the Acyclic Subdigraph Problem. This fact plays a central role in algorithmic approaches to the Acyclic Subdigraph Problem and the Linear Ordering Problem. We shall return to this in Section IV.

Theorem II.11. (Dicycle Inequalities) *Let C be a dicycle in a digraph $D = (V,A)$. Then the Dicycle Inequality $x(C) \leq |C| - 1$ defines a facet of $P_{AC}(D)$.*

Proof. We have already observed that $x(C) \leq |C| - 1$ is valid for $P_{AC}(D)$. Now suppose that C is a k-dicycle in D. It is trivial to see that the k dipaths obtained from C by removing one arc from D form a collection of acyclic arc sets whose incidence vectors in $\mathbb{R}^A$ are linearly independent and satisfy $x(C) \leq k-1$ with equality.

Now let $(i,j) \in A$ be an arc not in C. If both nodes i,j are in C (in this case we call (i,j) a *chord* of C), then remove from C the arc (j,k) whose tail is j to obtain a (k,j)-dipath P_{ij}. If one of the endnodes of (i,j) is not in C, let P_{ij} be any dipath of length $k-1$ contained in C. It is obvious that each of the arc sets $P_{ij} \cup \{(i,j)\}$ is acyclic and satisfies $x(C) = k-1$. Moreover, the incidence vectors of all arc sets constructed above (k dipaths and $|A| - k$ dipaths plus an arc) are linearly independent. This proves the theorem. ▯

Clearly, we could prove Theorem II.11 in a less direct way by observing that each 2-Dicycle Inequality of the form $x_{ij} + x_{ji} \leq 1$ for some 2-dicycle $\{(i,j),(j,i)\} \subseteq A$ defines a facet of $P_{AC}(D)$ and applying the Subdivision Lemma. In connection with the 2-Dicycle Inequalities we can derive a simple but interesting result which further reduces the class of valid inequalities to be examined for their facet defining property.

Lemma II.12. *Suppose the inequality $d^T x \leq d_0$ defines a facet of $P_{AC}(D)$ for $D = (V,A)$ and both $d_{ij} > 0$ and $d_{ji} > 0$ for two nodes $i,j \in V$ and $(i,j),(j,i) \in A$. Then $d^T x \leq d_0$ is equivalent to the 2-Dicycle Inequality $x_{ij} + x_{ji} \leq 1$ with respect to $P_{AC}(D)$.* ▯

We shall now present a very rich class of facet defining inequalities.

Definition II.13. *Let $C_1, C_2, \ldots, C_k$ be a sequence of different dicycles in a digraph $D = (V,A)$ such that the following holds:*

(M1) *$k \geq 3$ and k odd.*

(M2) *C_i and C_{i+1}, $i \in \{1,2,\ldots,k-1\}$ have a directed path P_i in common, C_1 and C_k have a dipath P_k in common.*

(M3) *Given any dicycle C_j, $j \in \{1,2,\ldots,k\}$, set $J = \{1,2,\ldots,k\} \cap (\{j-2,j-4,j-6,\ldots\} \cup \{j+1,j+3,j+5,\ldots\})$. Then every set $(\bigcup_{i=1}^{k} C_i) \setminus \{e_i \mid i \in J\}$ contains exactly one dicycle (namely C_j), where e_i, $i \in J$, is any arc contained in the dipath P_i.*

(M4) *The largest acyclic arc set in $\bigcup_{i=1}^{k} C_i$ has cardinality $|\bigcup_{i=1}^{k} C_i| - \frac{k+1}{2}$.*

Then we call the arc set $M = \bigcup_{i=1}^{k} C_i$ *a* <u>*Möbius-Ladder.*</u> *For convenience we say that the dicycles* C_i , C_{i+1} , $i \in \{1,2,\ldots,k-1\}$, *and* C_1 , C_k *are adjacent (with respect to* M*).*

Axiom (M4) implies immediately that for any Möbius Ladder M contained in a digraph D the *Möbius Ladder Inequality*

(II.14) $x(M) \leq |M| - \frac{k+1}{2}$

is valid with respect to $P_{AC}(D)$.

The requirements (M1), ..., (M4) are of course not easy to check for a given arc set M. They are however precisely those assumptions which make a certain proof method work. (M4) implies the validity of (II.14) and (M3) implies that the sets $M \setminus \{e_i \mid i \in J\}$ minus any arc in C_j are maximum cardinality acyclic arc sets and that there are enough acyclic arc sets of this kind to find $|M|$ whose incidence vectors are linearly independent. For even k the construction does not give anything interesting. We might in fact also consider single dicycles as Möbius Ladders for k = 1.

A "general" Möbius Ladder is depicted in Figure II.15.

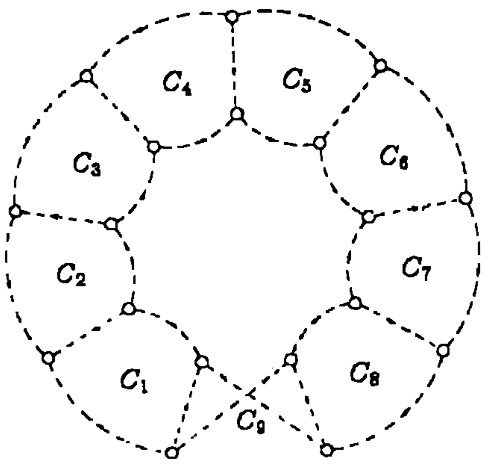

Figure II.15

If $C_1, C_2, \ldots, C_k$ is a sequence of directed cycles satisfying (M1) and (M2) and if no two different nonadjacent dicycles C_i, C_j have a node in common, then the union of these dicycles clearly forms a Möbius Ladder. Such a situation is depicted in Figure II.15. It may however well be that different nonadjacent dicycles have a node or even a dipath in common, cf. Figures II.16 and II.17.

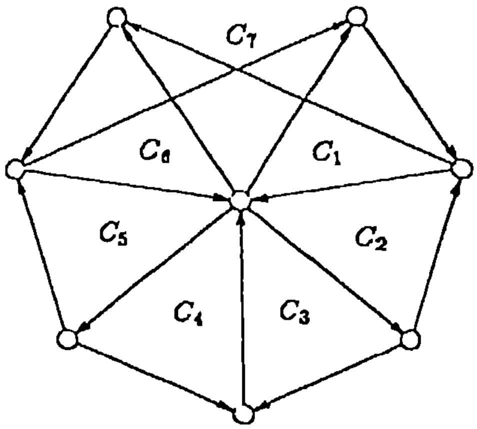

Figure II.16

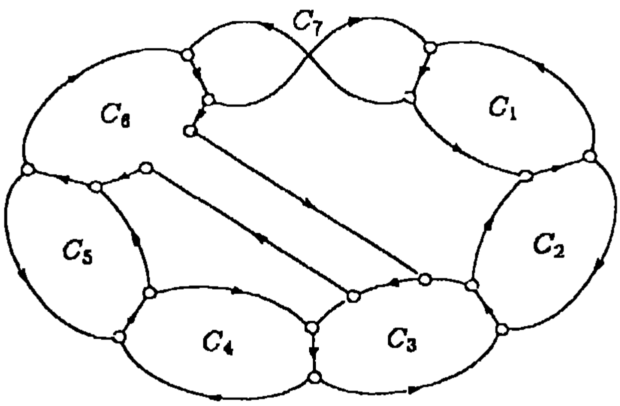

Figure II.17

It should be clear how to generate large classes of Möbius Ladders from the examples shown in Figures II.15, II.16, II.17.

Although this is not obvious at the first glance, there are Möbius Ladders all of whose defining dicycles are 3-dicycles, see Figure II.18 for an example.

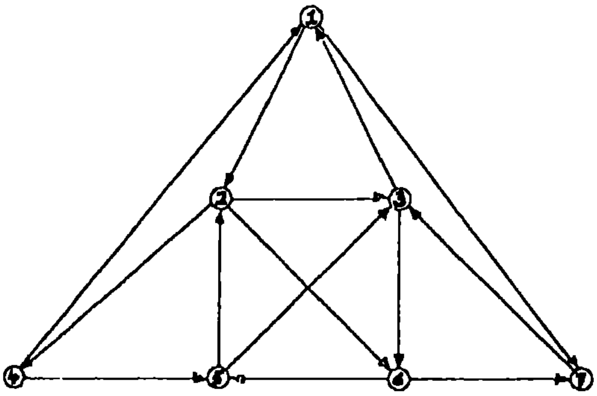

Figure II.18

The defining dicycles are induced by the seven node sets {1,2,3}, {3,1,7}, {7,3,6}, {3,6,5}, {6,5,2}, {5,2,4} and {2,4,1}. We leave it to the reader to verify that the digraph displayed in Figure II.18 is indeed a Möbius Ladder.

Unfortunately, Möbius Ladders are in general not so well-structured as in the above examples.

It would be a step forward if Möbius Ladders could be characterized in a more "pleasant" way. Checking axiom (M4) is *NP*-hard for general digraphs D. However, this does surely not mean that the definition given above is useless, since we can easily exhibit very large classes of Möbius Ladders whose algorithmic exploitation can help in the solution of large instances of the ASP.

It is easy to see that the axioms (M1), ..., (M4) imply that no two different dipaths P_i and P_j have a common arc (they may however have a common node, see Figure II.16. Moreover, in view of Lemma II.12 we know that all dicycles C_i have length at least three. In case $k = 3$, all three dicycles C_i have to have length at least four, Figure II.19 shows the - up to isomorphism - unique Möbius Ladder defined on three dicycles with the minimum number of arcs, where the common arcs of adjacent dicycles have been emphasized.

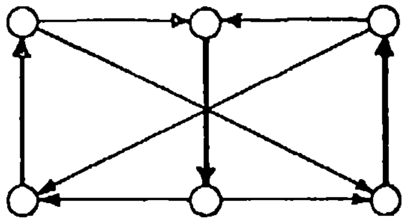

Figure II.19

Theorem II.20. (Möbius Ladders) *Let* M *be a Möbius Ladder defined by the dicycles* $C_1, C_2, \ldots, C_k$ *in* $D = (V,A)$. *Then the Möbius Ladder Inequality* $x(M) \leq |M| - \frac{k+1}{2}$ *defines a facet of* $P_{AC}(D)$. ▯

We have already given some examples for which the membership in the class of Möbius Ladder Inequalities is more or less straightforward to check. In many more cases the determination of this membership is not as easy.

As an example we consider a class of digraphs which can be directly derived from Möbius Ladders consisting only of 4-dicycles C_i, where every dipath P_i common to C_i and C_{i+1} consists of exactly one arc p_i and $p_i \cap p_j = \emptyset$ for all $i \neq j$, i. e. p_i and p_j have no common endnode.

Definition II.21. *Let* $D = (V,M)$ *be a Möbius Ladder consisting of* $k \geq 3$ *dicycles* $C_1, C_2, \ldots, C_k$ *of length four such that each pair of adjacent dicycles* C_i *and* C_{i+1}, $i \in \{1,2,\ldots,k-1\}$ *respectively* C_k *and* C_1 *intersects in exactly one arc* (a_i, b_i), $i \in \{1,2,\ldots,k\}$ *such that the arcs* (a_i, b_i) *are pairwise node disjoint and let* $v \notin V$. *Then we call the digraph* $D' = (V',W)$ *with* $V' := V \cup \{v\}$ *and*

$$W := M \cup \{(a_i, v), (v, b_i) \mid i \in \{1,2,\ldots,k\}\}$$

a <u>*simple k-Wheel.*</u>

Figure II.22 shows a simple 3-Wheel and a simple 5-Wheel.

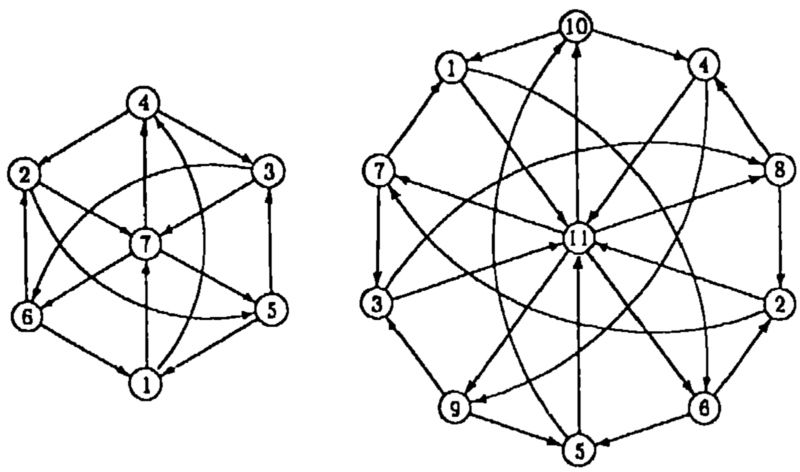

Figure II.22

A digraph $D = (V,A)$ is a *k-Wheel* if it can be obtained from a simple k-Wheel D' by repeated subdivision of arcs. With every k-Wheel $D = (V,A)$ we associate the *k-Wheel Inequality*

$$\text{(II.23)} \qquad x(A) \leq |A| - \frac{3k+1}{2} \ .$$

It can be shown that k-Wheels are Möbius Ladders defined on $k' = 3k$ dicycles. Thus we obtain

Theorem II.24. *Let $D = (V,A)$ be a digraph and $D' = (V',A')$ be a k-Wheel contained in D. Then the k-Wheel Inequality $x(A') \leq |A'| - \frac{3k+1}{2}$ defines a facet of $P_{AC}(D)$.*

□

For more details and more nonobvious examples of Möbius Ladders see JÜNGER (1984).

Considering again the Möbius Ladder with $k = 3$ depicted in Figure II.19 and drawing it in a different way, cf. II.25, yields a basis for a further class of facet defining inequalities.

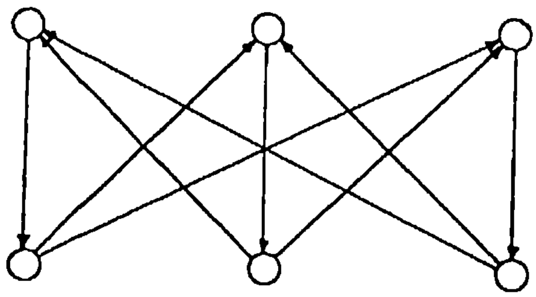

Figure II.25

Definition II.26. *For every integer $k \geq 3$ a digraph $D = (V,A)$ with $2k$ nodes is called a simple k-Fence if V consists of disjoint node sets $U = \{u_1, u_2, \ldots, u_k\}$ and $W = \{w_1, w_2, \ldots, w_k\}$ such that*

$$\text{(II.27)} \qquad A = \bigcup_{i=1}^{k} \left(\{(u_i,w_i)\} \cup (\{w_i\} : U \setminus \{u_i\})\right) \ .$$

The nodes in U are called the upper nodes, those in W the lower nodes and the arcs (u_i,w_i) are called pales and the arcs (w_i,u_j), $i \neq j$, are called pickets.

A simple k-Fence is a particular orientation of the complete bipartite graph $K_{k,k}$. (For $k = 2$, a simple 2-Fence would be a 4-dicycle.) See Figure II.28 for a simple 4-Fence.

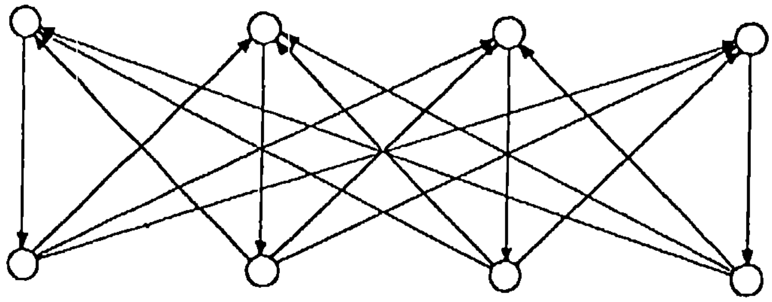

Figure II.28

A k-Fence is a digraph $D = (V,A)$ which can be obtained from a simple k-Fence $D' = (V',A')$ by repeated subdivision of arcs. If U' and W' are the upper and lower nodes, respectively, of D', then clearly these node sets correspond in a unique way to node sets U respectively W in D. We call the nodes in U and W the upper and lower nodes, respectively, of D.

Theorem II.29. (k-Fences) *Let $D = (V,A)$ be a digraph and $D' = (V',A')$ be a k-Fence contained in D. Then the k-Fence Inequality $x(A') \le |A'| - k+1$ defines a facet of $P_{AC}(D)$.*

□

Although all k-Fences contain Möbius Ladders as subdigraphs, no k-Fence is - unlike k-Wheels - equivalent to a Möbius Ladder except for the 3-Fence. For assume that a k-Fence, $k \ge 4$, is a Möbius Ladder M defined on k' dicycles. Comparing the righthand sides of the associated inequalities, we obtain $k-1 = \frac{k'+1}{2}$ and therefore $k' = 2k-3$. By definition (Axiom (M3) of II.13), M must contain $2k-3$ dipaths P_i such that the removal of certain $\frac{2k-3+1}{2} - 1 = k-2$ arcs belonging to distinct such dipaths leaves exactly one dicycle in M. None of these dipaths can be a picket or a dipath obtained by subdividing a picket, since it is easy to see that regardless of the choice of the remaining $k-3$ arcs to be removed, at least two different dicycles remain. (In fact, the best possible choice are $k-3$ arcs contained in pales not incident with the chosen picket.) So the only candidates for the $2k-3$ dipaths with the desired property are the (possibly subdivided) pales of the k-Fence. But there are only k such dipaths and $k < 2k-3$ for all $k \ge 4$. Thus we have proven

Remark II.30. *No k-Fence Inequality, $k \ge 4$, is equivalent to a Möbius Ladder Inequality.*

□

Summarizing the previous results of this section we obtain

Theorem II.31.

$P_{AC}(D) \subseteq P := \{x \in \mathbb{R}^A \mid x \geq 0,$

$x_{uv} \leq 1$ *for all* $(u,v) \in A$ *such that* $(v,u) \notin A$,

$x(C) \leq |C| - 1$ *for all dicycles* $C \subseteq A$,

$x(M) \leq |M| - \frac{k+1}{2}$ *for all Möbius Ladders* $M \subseteq A$,

$x(F) \leq |F| - k+1$ *for all k-Fences* $F \subseteq A,\ k \geq 4\}$

and the inequalities defining P *are a partial and nonredundant linear description of* $P_{AC}(D)$.

□

All inequalities in II.31 have in common that their coefficients are either 0 or 1. We shall now state a lemma which will enable us to derive facet defining inequalities with arbitrary coefficients using the nontrivial inequalities of II.31 as a basis.

Let $D = (V,A)$ be a digraph and $d \in \mathbb{N}_0^A$. For any $v \in V$ we call

$$\deg_d^-(v) := \sum_{(u,v) \in A} d_{uv}$$

the *d-weighted indegree* of v and

$$\deg_d^+(v) := \sum_{(v,u) \in A} d_{uv}$$

the *d-weighted outdegree* of v. By $\omega^+(v)$ resp. $\omega^-(v)$ we denote the set of arcs in A leaving resp. entering v.

Lemma II.32. (The Node Splitting Lemma) *Let* $D = (V,A)$ *be a digraph and assume that* $d^T x \leq d_0$ *defines a nontrivial facet of* $P_{AC}(D)$. *Suppose* $v \in V$ *and there is an acyclic arc set* $B \subseteq A$ *such that* $d(B) = d_0$ *and either* $B \cap \omega^+(v) = \emptyset$ *or* $B \cap \omega^-(v) = \emptyset$. *Define* $D' = (V',A')$ *such that* $V' = (V \setminus \{v\}) \cup \{u,w\}$ *and* $A' = (A \setminus \omega(v)) \cup \{(p,u) \mid (p,v) \in A\} \cup \{(w,q) \mid (v,q) \in A\} \cup \{(u,w)\}$. *Let* $\delta = \min\{\deg_d^+(v), \deg_d^-(v)\}$ *and*

$d'_{pq} = d_{pq}$ *for all* $(p,q) \in A$ *with* $v \notin \{p,q\}$

$d'_{pu} = d_{pv}$ *for all* $(p,u) \in A'$

$d'_{wq} = d_{vq}$ *for all* $(w,q) \in A'$

$d'_{uw} = \delta$

$d'_0 = d_0 + \delta$.

Then the inequality $d'^T x \leq d'_0$ *defines a facet of* $P_{AC}(D)$. ▯

The Node Splitting Lemma is a helpful tool for deriving new facets of $P_{AC}(D)$.

If we consider the Dicycle Inequalities, we cannot obtain any new facets since in this case each application of the Node Splitting Lemma to a k-Dicycle Inequality yields nothing but a (k+1)-Dicycle Inequality. In fact, node splitting is equivalent to subdivision of arcs in this case. Similarly, the node splitting procedure does not yield any new facets obtainable from Fence Inequalities, since each possible application is equivalent to the subdivision of a pale.

The definition of Möbius Ladders seems to be inappropriate to allow general statements about the new facets obtainable from Möbius Ladders by node splitting. We confine ourselves to some special cases. As a first example, we apply the Node Splitting Lemma to the Möbius Ladder depicted in Figure II.16 where no node splitting operation is equivalent to subdivision of arcs, see Figure II.33, in which the arcs are labelled by their respective coefficients and unlabelled arcs have coefficient one.

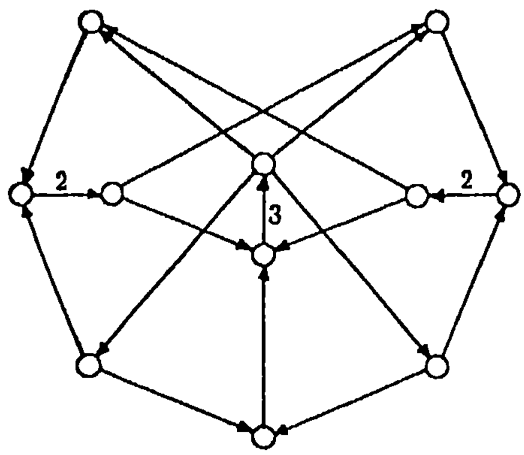

Figure II.33

For Wheels $D = (V,A)$ it is not hard to see that each node $v \in V$ satisfies the hypothesis of the Node Splitting Lemma. Figure II.34 shows an application with respect to the second digraph depicted in Figure II.22.

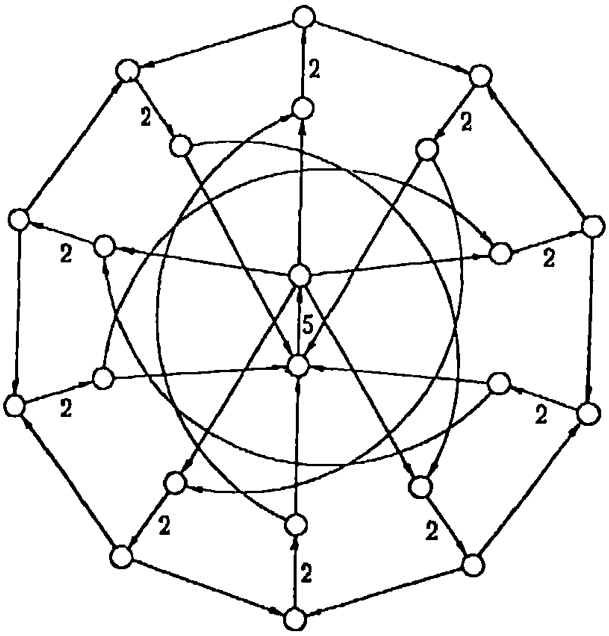

Figure II.34

III. Weakly Acyclic Digraphs

Let us recall the definition of $P_C(D)$:

(III.1) $P_C(D) := \{x \in \mathbb{R}^A \mid 0 \le x_{ij} \le 1$ for all $(i,j) \in A$,
$x(C) \le |C| - 1$ for all dicycles
$C \subseteq A$ in $D = (V,A)\}$.

We know that $P_{AC}(D) \subseteq P_C(D)$ and that all facets of $P_C(D)$ are also facets of $P_{AC}(D)$. In this section we want to show that linear programs over $P_C(D)$ are solvable in polynomial time. By a result of GRÖTSCHEL, LOVÁSZ & SCHRIJVER (1981) a linear objective function can be maximized over $P_C(D)$ in polynomial time if and only if the separation problem for $P_C(D)$ can be solved in polynomial time. The separation problem for $P_C(D)$ is the following

(III.2) *"Given a vector $y \in \mathbb{Q}^A$, determine whether $y \in P_C(D)$ and* *if not, find a vector* $d \in \mathbb{Q}^A$ *such that* $d^T y > d^T x$ *for all* $x \in P_C(D)$ *(a separating hyperplane)."*

We shall now show that problem (III.2) is solvable in polynomial time. Note that this cannot be done by checking all inequalities defining $P_C(D)$ one by one, since $P_C(D)$ may have a number of nonequivalent facets which is exponential in $|V|$. For instance, $P_C(D_n)$ has exactly $\binom{n}{k}(k-1)!$ facets arising from k-dicycles, $2 \leq k \leq n$, so there are together

$$\text{(III.3)} \qquad \gamma_n := \sum_{k=2}^{n} \binom{n}{k}(k-1)! \geq (n-1)!$$

nonequivalent nontrivial facets of $P_C(D_n)$, e. g. $P_C(D_{50})$ has more than 1.69×10^{63} nontrivial facets.

Suppose a vector $y \in \mathbb{Q}^A$ is given, and we want to solve (III.2). We can easily check by substitution whether y satisfies the trivial inequalities $0 \leq y_{ik} \leq 1$. Hence, if one of these is violated we have found a separating hyperplane. If $y_{ik} < 0$, then $x_{ik} > 0$ defines a separating facet and if $y_{ik} > 1$ then $x_{ik} \leq 1$ is an inequality separating y from $P_C(D)$. For the following we may therefore assume that the given $y \in \mathbb{Q}^A$ satisfies $0 \leq y_{ik} \leq 1$ for all $(i,k) \in A$.

For every arc in A we define a "weight" $w_{ik} = 1-y_{ik}$. If C is any dicycle in D, then clearly $y(C) \leq |C| - 1$ if and only if $w(C) \geq 1$. This implies that we can check whether y violates a dicycle inequality by finding a dicycle C^* whose weight $w(C^*)$ is minimum, i. e. a shortest dicycle in D under w. Namely, if the minimum weight $w(C^*)$ satisfies $w(C^*) \geq 1$ then all Dicycle Inequalities $x(C) \leq |C| - 1$ are satisfied by y; if $w(C^*) < 1$, then $y(C^*) > |C^*| - 1$ and hence a separating hyperplane is found, which is a facet of $P_C(D)$ by Theorem II.11.

What remains to be shown is that, given a digraph $D = (V,A)$ with arc weights $w_a \geq 0$ for all $a \in A$, a shortest dicycle under w can be found in polynomial time. But this is easy by making appropriate modifications of any polynomial time shortest dipath algorithm (like the Dijkstra or Floyd-Warshall-method).

In fact, the separation algorithm for $P_C(D)$ outlined above can be implemented so that its running time is $O(|V|^3)$. From this we can conclude that for any digraph $D = (V,A)$ and any $c \in \mathbb{R}^A$ the linear program

(III.4) $\max c^T x,\ x \in P_C(D)$

can be solved in polynomial time.

Definition III.5. *A digraph $D = (V,A)$ is called* <u>*weakly acyclic*</u> *if the acyclic subdigraph polytope $P_{AC}(D)$ equals $P_C(D)$. Digraphs which are not weakly acyclic are called* <u>*strongly cyclic.*</u>

▯

The version of the ellipsoid method described in GRÖTSCHEL, LOVÁSZ and SCHRIJVER (1981) finds an optimum vertex solution when applied to the problem $\max\{c^T x \mid x \in P_C(D)\}$. So in case $P_{AC}(D) = P_C(D)$ the incicence vector of an acyclic arc set is found, and so we get

Theorem III.6. *The Acyclic Subdigraph Problem for weakly acyclic digraphs can be solved in polynomial time.*

▯

Weakly acyclic digraphs are not too well understood yet. Below we have collected what is known.

It is clear that acyclic digraphs are weakly acyclic, since a digraph is acyclic if and only if $P_{AC}(D)$ is the unit hypercube. On the other hand, every digraph containing the support of any facet defining inequality presented in Section II (except for trivial inequalities and Dicycle Inequalities) must be strongly cyclic.

On the other hand, if D is weakly acyclic then any digraph obtained from D by adding a source or a sink is weakly acyclic, more generally, if D' and D'' are two node-disjoint weakly acyclic digraphs and we create a new digraph D from D' and D'' by adding some arcs going from a node in D' to a node in D'', then D is weakly acyclic. Similarly, if D' and D'' are node disjoint and weakly acyclic then the digraph obtained by identifying a node in D' and a node in D'' is weakly acyclic.

Moreover, it is easy to verify that every subdigraph of a weakly acyclic digraph is weakly acyclic.

By the Subdivision Lemma and the Contraction Lemma it is clear that the class of strongly cyclic digraphs is closed under subdivision of arcs and contraction of arcs one of whose endnodes have indegree and outdegree one. It turns out that this is also true for weakly acyclic digraphs.

A particulary interesting class of weakly acyclic digraphs is given by the following observation.

Remark III.7. *Planar digraphs are weakly acyclic.*

Proof. This follows immediately from the planar version of the theorem of Lucchesi and Younger (1978).

□

A direct consequence of III.7 is the following observation.

Corollary III.8. *Let* $a^T x \leq a_0$ *define a nontrivial facet of* $P_{AC}(D)$ *for some digraph* $D = (V,A)$. *Let* $D' = (V',A')$ *be the subdigraph of* D *defined by* $A' := \{(i,j) \in A \mid a_{ij} > 0\}$. *Then* D' *is nonplanar or a dicycle.*

□

Moreover, we can show

Remark III.9. *A digraph containing at most four different dicycles is weakly acyclic.*

□

Note that Remark III.9 is sharp in the sense that there is a strongly cyclic digraph containing exactly five dicycles, namely the Möbius Ladder depicted in Figure II.19.

A graph G (digraph D) is *contractible* to a graph G' (digraph D') if G' (D') can be obtained from G(D) by repeated deletion of nodes and deletion or contraction of edges (arcs). Kuratowski (1930) has characterized planar graphs (digraphs) as those graphs (digraphs) which are not contractible to (any orientation of) the nonplanar so-called Kuratowski Graphs $K_{3,3}$ or K_5. As an application of Remark III.9, Figure III.10 shows two weakly acyclic orientations of the Kuratowski Graphs. (The first one is a 3-Fence with one reversed picket.)

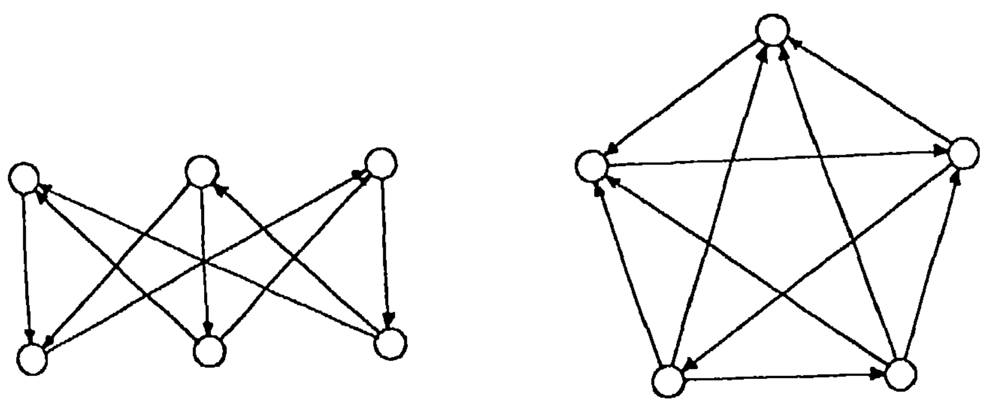

Figure III.10

For open problems with respect to weakly acyclic digraphs see Section VII.

IV. The Linear Ordering Polytope P^n_{LO}

We now turn to the discussion of the linear ordering problem. Let $D_n = (V,A_n)$ be the complete digraph on n nodes. We shall define a polytope whose vertices correspond to the spanning acyclic tournaments in A_n and vice versa. Similar to the acyclic subdigraph polytope we shall then study the properties of this polytope and try to describe it (at least partially) by facet defining inequalities.

Let $D_n = (V,A_n)$ be the complete digraph on n nodes and $m = n(n-1)$. Then the linear ordering polytope P^n_{LO} is defined as the convex hull of the incidence vectors of spanning acyclic tournaments in D_n, i. e.

(IV.1) $$P^n_{LO} = \text{conv}\{x^A \in \{0,1\}^m \quad A \subseteq A_n \text{ is a spanning acyclic tournament}\}.$$

A first observation shows that this polytope is not full dimensional since every point $x \in P^n_{LO}$ satisfies $x_{ij} + x_{ji} = 1$, for all $(i,j) \in A_n$. Therefore a complete linear description of P^n_{LO} will require inequalities and equations.

A minimal equation system for P^n_{LO} can easily be given.

Theorem IV.2. *Let $n \geq 2$. Then the system*

$$x_{ij} + x_{ji} = 1\ , \quad \text{for all } i,j \in V\ , \quad i \neq j\ , \quad i < j$$

is a minimal equation system for P^n_{LO}.

□

As an immediate corollary we get

Corollary IV.3.

$$\dim P^n_{LO} = \frac{1}{2}\, n(n-1) = \binom{n}{2}.$$

□

Hence P^n_{LO} is a $\binom{n}{2}$-dimensional face of $P_{AC}(D_n)$ (the acyclic subdigraph polytope for the complete digraph D_n). It is therefore reasonable to check which of the facets of $P_{AC}(D_n)$ (derived in Section II) are also facets of P^n_{LO} .

We first give some general properties of inequalities defining facets of P^n_{LO} . From the simple structure of the minimal equation system we get

Corollary IV.4. *For every facet of* P^n_{LO} *there exists an inequality* $a^T x \le a_0$ *defining it with nonnegative coefficients and the property that for every pair of nodes* $i,j \in V$ *at least one of the coefficients* a_{ij} *or* a_{ji} *is equal to zero.*

▯

We call inequalities having the property that at least one of the coefficients a_{ij} or a_{ji} is zero to be *support reduced*. If besides this, all coefficients a_{ij} are nonnegative we say that the inequality $a^T x \le a_0$ is *nonnegative support reduced*. The notion of nonnegative support reduced inequalities defines a kind of a normal form for inequalities valid for P^n_{LO} . We can show the following.

Lemma IV.5. *Let* $a^T x \le a_0$ *and* $b^T x \le b_0$ *be facet defining inequalities for* P^n_{LO} *which are support reduced and have nonnegative coefficients. If there exists an arc* $(i,j) \in A_n$ *with* $a_{ij} > 0$ *and* $b_{ij} = 0$ *(or* $b_{ij} > 0$ *and* $a_{ij} = 0$*) then the inequalities define different facets.*

▯

It is an open problem whether two nonnegative support reduced facet defining inequalities $a^T x \le a_0$ and $b^T x \le b_0$ satisfying $a_{ij} > 0$ if and only if $b_{ij} > 0$ (i. e. having the same support) can be non-equivalent.

Lemma IV.5 clearly gives an easy to verify condition for proving that two facet defining inequalities are not equivalent with respect to P^n_{LO} .

We shall now state a lemma describing some useful general properties of facet defining inequalities.

Lemma IV.6. *Let* $a^Tx \leq a_0$ *be a facet defining inequality for* P^n_{LO}, $n \geq 2$.

(a) *(Trivial lifting lemma)*

Define $\bar{a} \in \mathbb{R}^{(n+1)n}$ *by setting* $\bar{a}_{ij} = a_{ij}$, *for all* $(i,j) \in A_n$ *and* $\bar{a}_{i,n+1} = \bar{a}_{n+1,i} = 0$, *for* $i = 1,\dots,n$. *Then* $\bar{a}^Tx \leq a_0$ *defines a facet of* P^{n+1}_{LO}.

(b) *(Reversion lemma)*

If $b \in \mathbb{R}^{n(n+1)}$ *is defined by* $b_{ij} = a_{ji}$ *for all* $(i,j) \in A_n$ *then* $b^Tx \leq a_0$ *is facet defining for* P^n_{LO}.

▯

Contrary to the acyclic subdigraph polytope where the trivial inequalities are not always facet defining, we can show for the linear ordering polytope

Theorem IV.7. *Let* $n \geq 2$. *The inequalities* $x_{ij} \leq 1$ *and* $-x_{ij} \leq 0$ *define facets of* P^n_{LO} *for all* $(i,j) \in A_n$. *No two of the inequalities* $x_{ij} \leq 1$ *and no two of the inequalities* $-x_{ij} \leq 0$ *are equivalent. An inequality* $x_{ij} \leq 1$ *defines the same facet as* $-x_{uv} \leq 0$ *if and only if* $i = v$ *and* $j = u$.

▯

We now discuss the question of whether k-Dicycles induce facets of P^n_{LO} According to Theorem IV.2 the 2-Dicycle Inequalities are always satisfied with equality by every point in P^n_{LO}. It is easy to show the following result.

Theorem IV.8. *Let* C *be a k-Dicycle in* D_n, $n \geq k > 3$.

(a) *The k-Dicycle Inequality* $x(C) \leq |C| - 1$ *defines a face of* P^n_{LO} *of dimension* $\binom{n}{2} - \frac{k(k-3)}{2}$.

(b) The k-Dicycle Inequality $x(C) \leq |C| - 1$ *defines a facet of* P^n_{LO} *if and only if* $k = 3$.

∎

Based on the above considerations we can derive a result about the structure of facet defining inequalities which is similar to Lemma II.12.

Lemma IV.9. *Let* $a^T x \leq a_0$ *be a nonnegative support reduced valid inequality for* P^n_{LO} , $n \geq 3$, *and* $D_a = (V(A),A)$ *be the support of* a *in* D_n . *Suppose there is a node* $j \in V$ *which is contained in exactly two arcs* (i,j), $(j,k) \in A$. *If* T *is an acyclic tournament whose incidence vector satisfies* $a^T x = a_0$ *then it also satisfies* $x_{ij} + x_{jk} + x_{ki} = 2$.

∎

Applying this lemma we obtain another proof of Theorem IV.8 and as a corollary we can conclude that there is no subdivision lemma for P^n_{LO} .

The minimal equation system and the facet defining inequalities derived so far define the following polytope, denoted by P^n_C .

(IV.10)

$$P^n_C = \{x \in \mathbb{R}^{n(n-1)} \mid \begin{array}{lll} x_{ij}+x_{ji} = 1 & & \text{for all } i,j \in \{1,\dots,n\},\ i<j \\ x_{ij}+x_{jk}+x_{ki} \leq 2 & & \text{for all 3-dicycles } \{(i,j),(j,k),(k,i)\} \text{ in } D_n \\ x_{ij} \geq 0 & & \text{for all } i,j \in \{1,\dots,n\},\ i \neq j\} \end{array}$$

This polytope will play an important role in the formulation of an algorithm for the solution of the linear ordering problem.

Running a vertex enumeration algorithm on a computer we could prove that the polytope P^n_{LO} is completely described by the trivial inequalities and by the 3-dicycle inequalities for $n \leq 5$. So the first candidate for having further facets is P^6_{LO} .

Since we cannot apply subdivision to produce new facet defining inequalities, k-Fences, Möbius Ladders, etc. which are not simple cannot induce facet defining inequalities for P^n_{LO} . In the case of k-Fences we get

Theorem IV.11. *Let $D = (V,A)$ be a simple k-Fence contained in D_n, $n \geq 2k$. Then the k-Fence Inequality $x(A) \leq k^2 - k + 1$ defines a facet of P^n_{LO}.*

□

According to Lemma IV.5 different k-Fences induce different facets of P^n_{LO}. Hence the number of facets of P^n_{LO}, $n \geq 6$, which are induced by k-Fences is

$$\sum_{k=3}^{\lfloor \frac{n}{2} \rfloor} \left(\binom{n}{2k} \cdot \binom{2k}{k} \cdot k! \right) = \sum_{k=3}^{\lfloor \frac{n}{2} \rfloor} \frac{n!}{(n-2k)!k!} .$$

Up to now we did not succeed in proving that all simple Möbius Ladders induce facets of the linear ordering polytope (but we conjecture this to be true). We were able to prove the facet inducing property for a large subclass of the simple Möbius Ladders. The result is stated in the following theorem.

Theorem IV.12. *Let M be the arc set of a simple Möbius Ladder in D_n consisting of $k \geq 3$ dicycles $C_1,\ldots,C_k$ having the following (additional) properties:*

a) The length of C_i is three or four, $i = 1,\ldots,k$.

b) Two adjacent dicycles have exactly one arc in common.

c) If two nonadjacent dicycles C_i and C_j, $i < j$, have a common node, say v, then v either belongs to all dicycles $C_i, C_{i+1},\ldots,C_j$ or to all dicycles $C_j, C_{j+1},\ldots,C_k, C_1, C_2,\ldots,C_i$.

Then the Möbius Ladder inequality

$$x(M) \leq |M| - \frac{k+1}{2}$$

defines a facet of P^n_{LO} for $n \geq |V(M)|$.

□

It can also be shown that all simple k-Wheel inequalities define facets of the linear ordering polytope.

V. A Cutting Plane Algorithm

Based on the theoretical investigations of the previous chapter we developed an algorithm for the solution of the linear ordering problem. We shall now describe this algorithm in more detail.

As has been stated before, the linear ordering problem for the complete digraph $D_n = (V, A_n)$ and a given vector $c \in \mathbb{R}^{n(n-1)}$ of arc weights can be formulated as the following linear program:

$$\text{(V.1)} \qquad \text{maximize} \quad c^T x \qquad \text{subject to} \quad x \in P^n_{LO} .$$

The key concept of our procedure is to solve a sequence of successively stronger relaxations of the above linear program. Moreover, our relaxations should consist only of inequalities which define facets of P^n_{LO} .

Of course we would like to use all 3-Dicycle Inequalities because any integral solution satisfying all 3-Dicycle Inequalities corresponds to an acyclic tournament. Since we do not know in general how to handle Möbius Ladder and k-Fence Inequalities efficiently we decided to develop heuristics to incorporate at least the inequalities associated with the smallest digraphs of these two classes into our algorithm. We have chosen the 3-Fence and the two Möbius Ladders containing six nodes (the first Möbius Ladder is depicted in Figure V.2 and the second one is obtained by reversing all arcs). For abbreviation we call these Möbius Ladders to be of type M_6 . Using these facet defining inequalities we can formulate the following linear programming relaxation.

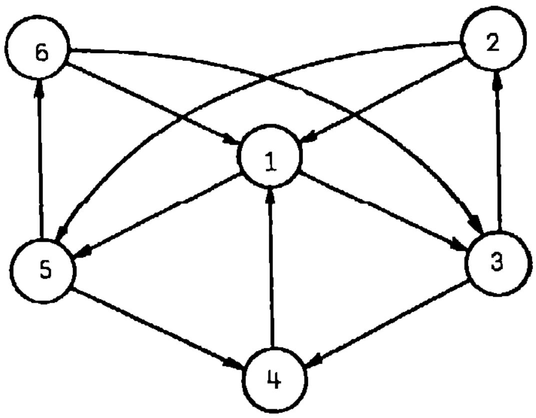

Figure V.2

$$\text{Maximize} \sum_{\substack{i,j \\ i \neq j}} c_{ij} x_{ij}$$

subject to

$$
\begin{aligned}
& x(C) \le 2 \quad \text{for all 3-dicycles } C \text{ in } A_n \\
& x(F) \le 7 \quad \text{for all 3-Fences } F \text{ in } A_n \\
\text{(V.3)} \qquad & x(M) \le 8 \quad \text{for all Möbius Ladders } M \text{ of type } M_6 \text{ in } A_n \\
& x_{ij} + x_{ji} = 1 \quad \text{for all } 1 \le i < j \le n \\
& x_{ij} \ge 0 \quad \text{for all } 1 \le i,j \le n, \; i \neq j \; .
\end{aligned}
$$

The central step of the algorithm is to first try to solve this linear program. If it has an integral optimum solution then it corresponds to an acyclic tournament of maximum weight since all 3-Dicycle Inequalities are satisfied; if not, we have to start an ordinary branch & bound routine to eventually get the integral optimum.

Observe that this linear program consists of $\binom{n}{2}$ equations, $n(n-1)$ nonnegativity conditions, $2\cdot\binom{n}{3}$ 3-Dicycle Inequalities, $120\cdot\binom{n}{6}$ 3-Fence Inequalities and $360\cdot\binom{n}{6}$ Möbius Ladder Inequalities. Due to this enormous number of constraints (e. g. 7, 627, 578, 875 for $n = 50$) it is unreasonable (and in practice even for small problems impossible) to list them all explicitly and solve the linear program using some commercial computer code. Therefore an approach via cutting planes is preferable. We now give a sketch of this procedure in a pseudo programming language.

(V.4) Procedure Cutting-Plane:

(*Solve LP using cutting planes*)

$$P := \{x \in \mathbb{R}^{n(n-1)} \mid x_{ij}+x_{ji} = 1, \text{ for all } 1 \le i < j \le n, \; x_{ij} \ge 0, \text{ for all } 1 \le i,j \le n\} \; ;$$

found:= true;
do while(found);

Solve $\max\{c^T x \mid x \in P\}$ and let x^* be the optimum solution; if there exists a facet defining inequality $a^T x \le a_0$ such that $a^T x^* > a_0$ then do;

```
            P:= P ∩ {x ∈ ℝ^{n(n-1)} | a^T x ≤ a_0} ;
            found:= true;
            end;
        else found:= false;
    end;
    if  x*  is integral
        then  x*  solves the linear ordering problem;
        else start branch & bound;
end cutting-plane.
```

This procedure just shows the principle of the cutting plane approach and several statements have to be made more precise. Especially the following problems have to be considered.

(1) How can the separation problem (i. e. the detection of violated inequalities) be solved ?

(2) Which violated inequalities should be added to the linear program if there are more than one available ?

(3) Should some classes of facet defining inequalities be preferred to other classes ?

We shall answer these questions in the sequel.
Before starting to solve the linear programming relaxation we can do some preprocessing to decrease the problem size.

By exploiting the structure of the minimal equation system we can eliminate one half of the variables. The variable x_{ij} , $j < i$ is substituted by $1-x_{ji}$ in all inequalities and in the objective function.

E. g. the 3-Dicycle Inequality

$$x_{ij} + x_{jk} + x_{ki} \leq 2$$

is transformed into

$$x_{ij} + x_{jk} - x_{ik} \leq 1$$

if $i < j < k$, or into the inequality

$$-x_{ji} - x_{kj} + x_{ki} \leq 0$$

if $i > j > k$. The trivial inequalities become $0 \leq x_{ij} \leq 1$ for all $1 \leq i < j \leq n$.

Also the objective function coefficients have to be updated accordingly and we get the new objective function $\bar{c}^T x$ with $\bar{c} \in \mathbb{R}^{\binom{n}{2}}$ and $\bar{c}_{ij} = (c_{ij} - c_{ji})$ for all $1 \leq i < j \leq n$. An easy calculation shows that the objective function values of corresponding solutions for the original and the transformed problem differ by the constant $\Sigma_{i>j}\, c_{ij}$.

Applying this transformation we do not loose too much insight into the original problem but have the advantage of decreasing the amount of storage needed and the computational effort in the computer implementation.

From a geometrical point of view the transformation replaces the polytope P^n_{LO} by the polytope $\bar{P}^n_{LO}$, which is its projection into the real vector space $\mathbb{R}^{\binom{n}{2}}$ according to the minimal equation system.

For a drawing of the projected polytope $\bar{P}^3_{LO} \subseteq \mathbb{R}^3$ consider Figure V.5. This polytope has six vertices and 8 facets. Two facets are given by the 3-Dicycle Inequalities and all other facets are trivial.

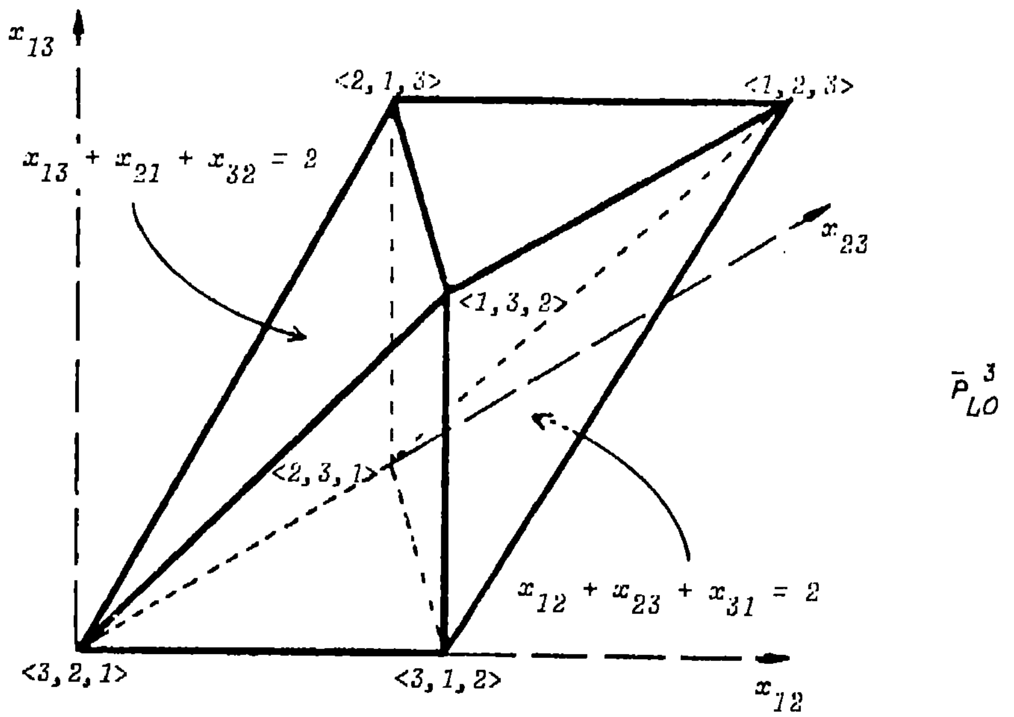

Figure V.5

We now focus attention on the central part of the cutting plane algorithm, i. e. the detection of violated inequalities. (Note that there is no separation problem for the trivial inequalities because the initial linear program consists of all the trivial inequalities and therefore they are satisfied by all subsequent solutions.)

(V.6) Detection of Violated 3-Dicycle Inequalities

We detect violated 3-Dicycle Ineqaulities simply by enumeration over all possible ones. This procedure has time complexity $O(n^3)$ since there exist $2\cdot\binom{n}{3}$ different 3-dicycles in A_n. It is not known whether there exists an algorithm of complexity less than $O(n^3)$ which finds a violated 3-Dicycle Inequality (if one exists). Because the 3-dicycles constraints are the most essential ones (at least from a practical viewpoint) it is valuable to enumerate all violated constraints of this class.

Moreover, a time complexity of $O(n^3)$ is tolerable in practical applications. In Section VI we shall discuss computational trade-offs observed when inserting all violated 3-Dicycle Inequalities or only certain subsets of them.

(V.7) Detection of Violated k-Fence Inequalities

In the case of facets induced by k-Fences enumeration of all possible inequalities is no longer feasible. Already for $k = 3$, enumeration has time complexity $O(n^6)$. Since we do not know of any efficient algorithm to solve the separation problem for k-Fences (such a procedure may not exist) we have to confine ourselves to heuristics. Since 3-Dicycle Inequalities are easily handled it is reasonable to only search for violated 3-Fence Inequalities if all 3-Dicycle Inequalities are satisfied and the current LP solution is fractional. Hence we are interested in the structure of vertices of P_C^n which violate a 3-Fence Inequality.

One such vertex can be determined in the following way. Let $F = (V_F, A_F)$ be a 3-Fence in the complete digraph D_6 and $c_F \in \mathbb{R}^{30}$ be its incidence vector. Maximizing the objective function $c_F^T x$ over P_C^6 we get an optimum solution y^* which can be depicted as follows

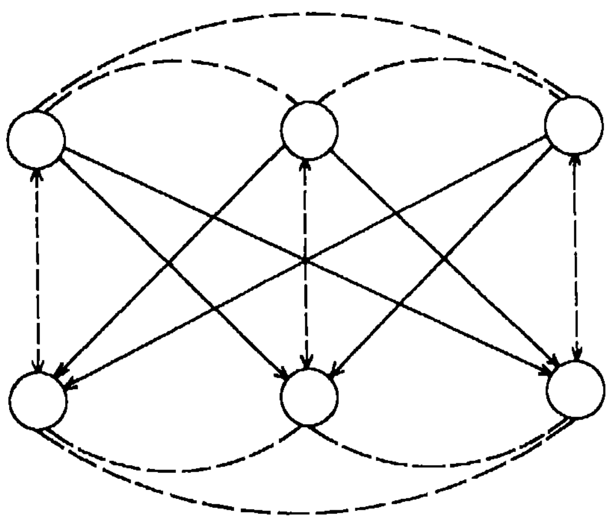

Figure V.8

In this figure broken lines correspond to fractional components with value $\frac{1}{2}$ (in both directions) and solid lines correspond to components having value 1. The components indexed by arcs antiparallel to the pickets have value 0.

This vertex solution clearly violates the 3-Fence Inequality $x(A_F) \leq 7$ since the sum of its F-components is equal to 7.5. Generalizing Figure V.8 we can construct for every $k > 3$ a vertex of P_C^n violating a k-Fence Inequality.

Our heuristic for finding violated 3-Fence Inequalities is based on the above observation but takes into consideration that the fractional components of vertices violating a 3-Fence Inequality may be different from $\frac{1}{2}$ whenever the current polytope is not the "pure" polytope P_C^n. It is assumed that a common property of vertices violating a 3-Fence Inequality is the nonintegrality of the pale components.

If y is the solution of the current linear program in the cutting plane algorithm then we define a corresponding (undirected) graph $G_y = (V(A_y), A_y)$ by setting $A_y = \{\{i,j\} \mid y_{ij}$ is fractional$\}$. The separation routine for 3-Fences can then be outlined as follows: Enumerate the triples of nonadjacent edges of G_y, i. e. edges having no common node. Take these edges as pales of a 3-Fence and test whether one of the eight 3-Fence Inequalities is violated which can be constructed from all possible orientations of these edges.

The worst case complexity of this procedure is still $O(n^6)$ but in practice the running time is moderate, especially when y has only few fractional components.

The following observation illustrates that it might not be worthwile to extend the heuristic to k-Fences with $k \geq 4$.

Theorem V.9. *Let* $F = (V_F, A_F)$ *be a k-Fence,* $k \geq 4$*. If the vector* y *is contained in* $P_C^n \cap \{x \in \mathbb{R}^{n(n-1)} \mid x$ *satisfies all l-Fence Inequalities for* $l < k\}$ *then* $y(A_F) \leq k^2 - k + 1 + \frac{1}{k-1}$. □

The theorem shows that a vector y violates a k-Fence Inequality by at most $\frac{1}{k-1}$ provided it satisfies all smaller l-Fence Inequalities. This indicates that the k-Fence Inequalities might be of less practical relevance for $k > 3$.

(V.10) Detection of Violated Möbius Ladder Inequalities

As the 3-Fence heuristic our Möbius Ladder heuristic is based on the examination of vertices of P_C^6 violating a certain Möbius Ladder Inequality.

Suppose $M = (V_M, A_M)$ is the Möbius Ladder of Figure V.11 and $c_M \in \mathbb{R}^{30}$ is its incidence vector. If the objective function $c_M^T x$ is maximized over P_C^6 we get the following optimum solution y^* with $c_M^T y^* = 8.5$.

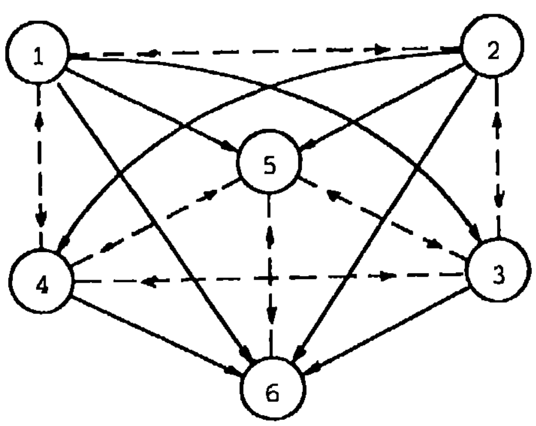

Figure V.11

Again broken lines correspond to components having value $\frac{1}{2}$ (in both directions), solid lines have value 1 and the components with value 0 are not shown. If we reverse all arcs of M and maximize as above we obtain the same figure except that all integral components of y^* are set to their opposite values.

The heuristic we developed exploits the noticeable fact that y^* has only seven (resp. fourteen when considered in both directions) fractional components which in addition have a nice structure. It works as follows: If y is the solution of the current linear program the "fractional" graph $G_y = (V(A_y), A_y)$ is constructed the same way as for the 3-Fence heuristic. By enumeration we look for 4-cycles without diagonals in this graph corresponding to the 4-cycle $\{\{1,2\}, \{2,3\}, \{3,4\}, \{4,1\}\}$ of Figure V.11. Then it is tried to find a further node w being adjacent to this cycle like node 5 of the figure. If this is successful we enumerate the nodes adjacent to w as possible candidates for the role of node 6. The six nodes determined by this method are treated as the nodes of a Möbius Ladder as shown in the figure and it is checked whether y violates one of the associated inequalities.

The worst case complexity of this method is of course still $O(n^6)$ but if there are not too many edges in G_y and if the search for the above mentioned subgraphs is implemented using data structures like adjacency lists one can hope for a tolerable running time in practical applications.

This finishes the discussion of the separation routines.

Having formulated the basic components we can now combine them to build the entire algorithm. The following design principles were considered to be reasonable.

(i) Avoid branch & bound as long as possible.

(ii) Call 3-Fence and Möbius Ladder separation heuristics only if there are provably no violated 3-Dicycle Inequalities.

The motivation for (i) is based on the practical experience in integer programming of many researchers who have shown that branch & bound algorithms tend to consume a large amount of computer time whenever the LP-relaxation is not tight enough. The 3-Dicycle Inequalities are preferred because they have to be present to exclude infeasible integral solutions and because they can be detected more efficiently than the other classes of inequalities. Moreover, their insertion keeps the sparsity of the matrix of constraints.

We now describe our computer implementation of the cutting plane algorithm which works in principle as the procedure outlined at the beginning of this chapter.

The initial LP can be solved trivially by setting a variable x_{ij} to 1 if its objective function coefficient $\bar{c}_{ij}$ is positive and to

0 otherwise. The augmented linear programs obtained after the insertion of violated inequalities are solved by using the IBM LP-package MPSX/370. Before solving a new LP we eliminate all constraints which are not binding at the previous solution in order to get smaller LPs. As long as violated 3-dicycles are detected we do not search for violated inequalities of the other two classes. For inserting violated 3-Dicycle Inequalities we tested three different strategies.

(i) <u>all violated</u>
All violated inequalities are inserted.

(ii) <u>k most violated</u>
This strategy adds the k most violated 3-Dicycle Inequalities to the linear program since they may have the greatest influence on the optimization process. To avoid expensive sorting we actually do not insert exactly the k most violated inequalities but determine the k cuts to be added by the following bucket sort procedure: We partition the interval $[0,1]$ (these are the possible violations) into small intervals $I_1, I_2, \ldots, I_l$; for each violated inequality

$a^T x \leq a_0$ its violation $s = a^T x^* - a_0$ is determined and the

inequality is sorted into the interval I_j with $s \in I_j$.

Then the k cutting planes are chosen from the "highest" intervals in the obvious way. In our application we have set $l = 20$.

(iii) <u>arc disjoint</u>
In this modification a subset of violated 3-dicycles is added with the property that no two corresponding 3-dicycles have an arc in common. This strategy is based on the heuristic idea that one inequality might be sufficient to locally decrease the infeasibility of the current solution and that it is not necessary to have one component x_{ij} in many newly

inserted cuts. The violation of the single inequalities is not taken into account.

If all 3-Dicycle Inequalities are satisfied and x^* is nonintegral it is tried to find violated Möbius Ladder Inequalities which are all added to the linear program. In case the heuristic could not find cutting planes of this class the separation routine for 3-Fences is called. The preferential treatment of the Möbius Ladders will become clear from the discussion of the computational results. We shall also compare the performance of the above strategies in the next chapter.

If no cuts could be generated and x^* is integral we are done because x^* solves the given linear ordering problem, if not, we have to pass over to a branch & bound procedure.

Since the implementation of branch & bound algorithms is well known we do not elaborate on this but just describe some features specific to our algorithm. We have implemented a depth first search through the branch & bound tree and the branching is done by setting some variable x_{ij} to 0 or 1. In each node of the tree we solve the respective linear program using MPSX/370 and still add cutting planes whenever violated ones could be detected.

For various features of a branch & bound algorithm (temporary and permanent fixing of variables, stopping criteria) one needs good lower bounds for the value of an optimum solution. As is usually done we try to heuristically find such good solutions, i. e. linear orderings with a "high" objective function value. Our heuristic first tries to take the structure of the LP solution x^* into account. For every node $i \in V$ we calculate

$$\text{(V.12)} \qquad s(i) = \sum_{k<i} c_{ik}(1-x^*_{ki}) + \sum_{i<j} c_{ij} x^*_{ij}$$

and sort the nodes such that $s(i_1) \geq s(i_2) \geq \ldots \geq s(i_n)$. Starting with the linear ordering $\langle i_1, i_2, \ldots, i_n \rangle$, we then try to make improvements by successive shifting operations of the form $\sigma = \langle 1,2,\ldots,i-1,i,i+1,\ldots,j-1,j,j+1,\ldots,n \rangle \rightarrow \langle 1,2,\ldots,i-1,i,j,i+1,\ldots,j-1,j+1,\ldots,n \rangle$ respectively $\sigma \rightarrow \langle 1,2,\ldots,i-1,i+1,\ldots,j-1,j,i,j+1,\ldots,n \rangle$ until a local optimum is obtained.

Another important component of a branch & bound algorithm is the possibility to fix the values of some variables permanently (i. e. the values do not change throughout the rest of the calculations) or temporarily (i. e. the values are valid in the whole subtree rooted at the current node).

We apply two criteria for the fixing of variables

(i) reduced cost criterion
We keep in memory the reduced costs after the linear programming part. The x^* found there gives a true upper bound for the optimum value of the linear ordering problem. Having improved the lower bound L we may be able to fix further variables permanently using a reduced cost criterion. Let d_{ij} denote the reduced cost of a nonbasic variable x^*_{ij} of the optimum LP-solution x^* at level 0, then we can do the following

- if $x^*_{ij} = 0$ and $\bar{c}^T x^* - L \leq -d_{ij}$ then we fix $x_{ij} = 0$ permanently (since every LP-solution with $x_{ij} = 1$ has a value which is not larger than L, no linear ordering with i before j can be better than the present one).

- If $x^*_{ij} = 1$ and $\bar{c}^T x^* - L \leq d_{ij}$ then we fix $x_{ij} = 1$ permanently (since every LP-solution with $x_{ij} = 0$ has a value which is not larger than L).

  If we take the reduced costs at the current node into account we may use the reduced cost criterion to fix variables temporarily in the subtree rooted at the current node.

(ii) transitive implications
If $x_{ij} = 1$ and $x_{jk} = 1$ have been fixed temporarily (permanently) then we can fix $x_{ik} = 1$ temporarily (permanently) because of transitivity. Similarly, $x_{ij} = 0$ and $x_{jk} = 0$ implies $x_{ik} = 0$ by transitivity.

Fixing of variables is a very essential component of a branch & bound algorithm, especially if the gap between upper bound and lower bound is small, quite a large number of variables can be fixed after the LP part. In spite the fact that we could physically eliminate the permanently fixed variables from the problem, we actually do not do this, mainly because data structures and LP updates would become more complicated.

This relatively short presentation of the algorithm should be sufficient to get an impression of how theoretical results can influence the development of procedures for solving problems arising in practical applications. In the next chapter we shall report on the computational results we obtained when applying our algorithm to the solution of the triangulation problem for input-output tables.

VI. Computational Results

In this section we report about the computational experiences we have with the algorithm described in the previous section. All experiments were run on a SIEMENS 7.865 of the Deutsche Forschungs- und Versuchsanstalt für Luft- und Raumfahrt in Oberpfaffenhofen. Due to the limited space we shall only state the main results; for a more extensive discussion cf. GRÖTSCHEL, JÜNGER & REINELT (1983a).

The linear ordering problem (or one of its equivalent versions) has many real-world applications. Lists of such applications can be found for instance in KORTE & OBERHOFER (1968), LENSTRA (1973), LENSTRA (1977) or JÜNGER (1984). One example is the triangulation of input-output tables.

In input-output analysis the economy of a region (usually a state) is divided into n sectors, and an (n,n)-input-output matrix X is constructed where the entry x_{ij} denotes the amount of deliveries from sector i to sector j in a certain year. The problem to permute the rows and columns of X simultaneously such that the sum of the entries of the permuted matrix above the main diagonal is as large as possible is called *triangulation problem*. Considering an entry x_{ij} of an (n,n)-input-output matrix X as the weight of the arc (i,j) of D_n one obtains a linear ordering problem. Triangulated input-output matrices allow interesting interpretations of the structure of an economy and comparisons between different countries, cf. WESSELS (1981).

Already in 1964, W. Krelle proposed to solve the triangulation problem by means of integer programming techniques, see KRELLE (1964). Algorithms have been designed by DE CANI (1969), who solved 10-sector problems by hand (after two hours of computation) using a linear programming relaxation and improved related approaches have been implemented as computer codes by MARCOTORCHINO and MICHAUD (1979) and BOENCHENDORF (1982). Both codes essentially maximize over (IV.10) thus having no guarantee for a feasible (integral) solution. For this reason branch & bound approaches searching for an integral optimum appeared to be more promising and in fact, the most heavily used algorithms in practical applications were "pure" branch and bound procedures which never solved linear programming relaxations explicitly. The first approach of this kind was designed by KORTE and OBERHOFER (1968, 1969), who implemented their algorithm on a computer and were able to solve random instances of the triangulation problem with up to 13 sector and real-world instances with up to 18 sector. LENSTRA (1973) presents an improvement of Korte and Oberhofer's algorithm, and KAAS (1981) was able to solve random 25-sector problems and real-world 34-sector problems on a computer by using a heuristic to obtain suboptimal dual solutions providing the necessary upper bounds in his branch & bound procedure.

We were mainly interested in solving real-world triangulation problems for input-output tables. We had 30 (44,44)-matrices of European countries compiled by the Statistical Office of the European Communities Luxemburg, one (50,50)-matrix from the Belgian Ministry of Economics and 14 matrices of the West German Economy (11 (56,56)-matrices compiled by the Deutsches Institut für Wirtschaftsforschung, Berlin, and 3 (60,60)-matrices compiled by the Statistisches Bundesamt, Wiesbaden). The size range of these tables convers almost all input-output-matrices that have been compiled so far all over the world. Moreover, it should be annotated that optimum solutions of the corresponding triangulation problems were not known.

We first triangulated the 45 tables using the all-violated strategy. As a surprising fact, it turned out that all tables could be triangulated without ever entering the branch & bound stage, and moreover, that except for four cases already 3-Dicycle Inequalities were sufficient to solve the problems to optimality. In these four exceptions the insertion of a few additional Möbius Ladder Inequalities (which were readily found by our heuristic) lead to the determination of an optimum triangulation within the linear programming phase. In no case 3-Fence Inequalities were required. The computing times were approximately 1.5 minutes for the 44-sector tables, 2.5 minutes for the 50-sector table, 5 minutes for the 56-sector tables and 10 minutes for the 60-sector tables. These results show that the cutting plane approach compares favorably with the other approaches and can solve problems of sizes far beyond the scope of algorithms which are not LP-based.

In spite of the fact that the elimination of nonbinding constraints resulted in a considerable reduction, the sizes of the final linear programs varied from about 1,500 rows for the 44-sector problems to about 3,500 rows for the 60-sector problems. In case of the 60-sector tables we actually inserted at most 2,000 cutting planes in each step to prevent storage problems. We then applied the k-most violated and the arc-disjoint cutting plane generation strategy to some of the problems. Application of the arc-disjoint strategy could reduce the LP sizes by 70 % on the average compared to the all-violated strategy, requiring about six times the number of cutting plane generation steps. The CPU time was slightly increased. The k-most-violated strategy gives larger LPs and a smaller number of cutting plane phases with increasing k. The best CPU time was attained for a medium-sized k (e. g. $k = 300$ in case of the 44-sector problems). A possible explanation for this is the following: in the arc disjoint strategy or for a small k the cutting plane generation routine is called quite often and has a significant effect on the overall computing time; for large k the LPs involved contain many rows and the reoptimization is time consuming. Therefore, it seems to be the best compromise to limit the number of cutting planes inserted by choosing an appropriate k depending on the problem sizes. Whenever storage is a scarce resource one should insert arc disjoint cutting planes.

The previous discussion has shown that the real-world problems could be solved "quite easily". The success of the code with respect to these problems does not carry over to random problems. Particularly bad are uniform distributions. For instance we encountered one problem of size (50,50) which the code was unable to solve (within the time available to us). Of course, a uniform distribution means that almost every solution is nearly optimal, and these types of problems are usually extremely hard to solve. We also generated twenty random problems with 20 sectors, where the entries were uniformly distributed in the interval [0,100]. These problems were triangulated within about 25 seconds, but here 4 of the problems could only be solved using branch & bound (the corresponding branch & bound trees consisted of up to 11 nodes).

On the other hand, we successfully solved an (80,80)-problem which was randomly generated such that its "structure" roughly corresponds to a real input-output table. This indicates that 60 sectors are not the limit for our code when applied to real-world problems.

VII. Open Questions

In the previous sections we have tried to outline how the insight into the facet structure of polytopes related to the Acyclic Subdigraph Problem and the Linear Ordering Problem lead to new complexity results as well as to practically useful algorithms.

There are several interesting open problems associated with our polyhedral results and we would like to close our exposition with a short discussion of some of these problems.

The most interesting subject for further research appears to be a better characterization of Möbius Ladders. As we have seen in Section II, these digraphs give rise to a very rich class of facets of $P_{AC}(D)$ and we do not know of any nontrivial facet inducing inequality whose support does not contain a Möbius Ladder. (Also dicycles can be viewed as Möbius Ladders, as we have pointed out.) In fact, Definition II.13 is designed to postulate exactly those properties which are needed in our proof of Theorem II.20. However, the axioms given in Definition II.13 do certainly not give rise to an efficient procedure for deciding the membership of a digraph to the defined class. A better understanding of the nature of Möbius Ladders seems to be of great value, in particular, a better characterization is necessary to be able to attack the separation problem for $P_{AC}(D)$ associated with Möbius Ladders.

A less ambitious, but nevertheless nontrivial task is the polynomial time detection of violated Wheel or Fence Inequalities, the former class being an example of a better characterized subclass of Möbius Ladder Inequalities. Of course, the solution of the same problem is interesting for any well-characterized subclass of Möbius Ladder Inequalities, some of which have been outlined in Section II.

All theorems stating the facet defining property of certain inequalities $d^Tx \leq d_0$ for $P_{AC}(D)$ presented in Section II required only that the support D_d of these inequalities be contained in D. The general technique to prove these result was to show that $d^Tx \leq d_0$ defines a facet of $P_{AC}(D_d)$ and then derive the general result by lifting the inequality to a facet defining inequality for $P_{AC}(D)$ by adding zero-components to d for all $a \in A(D) \setminus A(D_d)$. This procedure is often called *trivial lifting* ("trivial" because all lifting coefficients are zero). While it was easy to show that trivial lifting is possible in all cases considered in Section II, we have unsuccessfully tried to obtain a general result of this kind, namely, under what conditions is it true that a facet defining inequality for $P_{AC}(D)$ can be trivially lifted to a facet defining inequality for $P_{AC}(D')$, where D' is any superdigraph of D ?

Remember that for P^n_{LO} we have a related result, namely the Trivial Lifting Lemma IV.6, which states that every facet of P^n_{LO} can be trivially lifted to a facet of P^{n+1}_{LO} . In the case of $P_{AC}(D)$ however, the facet defined by the inequality $x_a \leq 1$ for some $a \in A(D)$ cannot be trivially lifted to a facet of $P_{AC}(D')$ where D' arises from D by adding the reverse arc of a to A.

A number of further research problems are associated with weakly acyclic digraphs. Also here, a "nice" (nonpolyhedral) characterization would be desirable. A related problem is to investigate minimally strongly cyclic digraphs, i. e. strongly cyclic digraphs in which the deletion or contraction of any arc results in a weakly acyclic digraph. (The 3-Fence is a trivial example.) In Section III, we have given some constructions preserving weak acyclicity and provided sufficient conditions for a digraph to be weakly acyclic, but we feel that stronger results are possible. All nontrivial facet inducing digraphs we have discovered, except for the dicycles, are contractible to an orientation of $K_{3,3}$. Since all strongly cyclic digraphs must be nonplanar by Remark III.7 they must be contractible to an orientation of $K_{3,3}$ or K_5 by Kuratowski's characterization of nonplanar (di-)-graphs. We could not prove but we conjecture that D_5 is weakly acyclic. A partial enumeration of the vertices of $P_{AC}(D_5)$ on a computer supports our conjecture. Such a result would be a strong motivation to consider whether the class of all digraphs not contractible to an orientation of $K_{3,3}$ are weakly acyclic.

By the Lucchesi-Younger Theorem we know that the system of inequalities defining $\bar{P}_C(D) := \{y \in \mathbb{R}^A \mid 0 \leq y_{ij} \leq 1,\ y(C) \geq 1 \text{ for all dicycles } C \text{ in } D\}$ is totally dual integral (TDI) for planar digraphs.
It should be investigated whether the linear system defining $P_C(D)$ is totally dual integral for weakly acyclic digraphs. This would imply an interesting new min-max result relating the maximum weight of an acyclic arc set in a weakly acyclic digraph to the minimum weight of a certain dicycle covering in the digraph. However, our efforts to prove this have failed so far. The main reason appears to be the absence of an appropriate characterization of weakly acyclic digraphs and we feel that the solution to this problem will be a major contribution towards an answer to the TDI-ness question.

With respect to the linear ordering problem we do not know any inequality defining a facet of P_{LO}^n which - in its nonnegative support reduced form - has other coefficients than zero and one. We believe there are such facet defining inequlities. It would be interesting to have a good criterion to determine whether an inequality defining a facet of $P_{AC}(D_n)$ also defines a facet of P_{LO}^n.

VIII. Bibliography

(relevant for the subject, but not necessarily mentioned in the paper)

A. Bachem & M. Grötschel (1982) "New Aspects of Polyhedral Theory". in: B. Korte (ed.) "Modern Applied Mathematics". North-Holland, Amsterdam, 1982, 51 - 106.

K. Boenchendorf (1982) "Reihenfolgenprobleme / Mean-Flow-Time Sequencing". Mathematical Systems in Economics 74, Verlagsgruppe Athenäum/Hain/Scriptor/Hanstein, 1982.

V. J. Bowman (1972) "Permutation Polyhedra". SIAM J. Appl. Math., Vol. 22, No. 4 (1972), 580 - 589.

J. S. de Cani (1969) "Maximum Likelyhood Paired Comparison Ranking by Linear Programming". Biometrika 56 (1969), 537 - 545.

V. Chvátal (1973) "Edmonds Polytope and Weakly Hamiltonian Graphs". Mathematical Programming 5 (1973), 29 - 40.

H. Crowder, E. L. Johnson & M. W. Padberg (1981) "Solving Large Scale 0-1-Programming Problems". Research Report RC8888 (38963), IBM T. J. Watson Research Center, Yorktown Heights, New York, 1981.

H. Crowder & M. W. Padberg (1980) "Solving Large Scale Symmetric Travelling Salesman Problems". Management Science 26 (1980), 495 - 509.

G. B. Dantzig (1963) "Linear Programming and Extensions". Princeton University Press, Princeton, New Jersey, 1963.

G. B. Dantzig, D. R. Fulkerson & S. M. Johnson (1954) "Solution of a Large-Scale Travelling Salesman Problem". Operations Research 2 (1954), 393 - 410.

R. W. Floyd (1962) "Algorithm 97: Shortest Path". Comm. ACM, 5, No. 6 (1962), 345.

M. R. Garey & D. S. Johnson (1979) "Computers and Intractability: A Guide to the Theory of NP-Completeness". Freeman, San Francisco, 1979.

R. Giles & W. R. Pulleyblank (1979) "Total Dual Integrality and Integer Polyhedra". Linear Algebra and its Applications 25 (1979), 191 - 196.

F. Glover, T. Klastorin & D. Klingman (1974) "Optimal Weighted Ancestry Relationships". Management Science 20 (1974), 1190 - 1193.

M. Grötschel (1977) "Polyedrische Charakterisierungen kombinatorischer Optimierungsprobleme". Verlag Anton Hain, Meisenheim am Glan, 1977.

M. Grötschel (1983) "Polyhedral Combinatorics: An Annotated Bibliography". Preprint No. 15, Mathematisches Institut, Universität Augsburg, 1983.

M. Grötschel, M. Jünger & G. Reinelt (1982a) "On the Acyclic Subgraph Polytope". Report No. 82215-OR, Institut für Operations Research, Universität Bonn, 1982.

M. Grötschel, M. Jünger & G. Reinelt (1982b) "Facets of the Linear Ordering Polytope". Report No. 82217-OR, Institut für Operations Research, Universität Bonn, 1982.

M. Grötschel, M. Jünger & G. Reinelt (1983a) "A Cutting Plane Algorithm for the Linear Ordering Problem". Preprint No. 8, Mathematisches Institut, Universität Augsburg, 1983, to appear in Operations Research.

M. Grötschel, M. Jünger & G. Reinelt (1983b) "Optimal Triangulation of Large Real-World Input-Output Matrices". Preprint No. 9, Mathematisches Institut, Universität Augsburg, 1983, to appear in Statistische Jahreshefte.

M. Grötschel, L. Lovász & A. Schrijver (1981) "The Ellipsoid Method and its Consequences in Combinatorial Optimization". Combinatorica 1 (1981), 169 - 197.

M. Grötschel & M. W. Padberg (1979a) "On the Symmetric Travelling Salesman Problem I: Inequalities". Mathematical Programming 16 (1979), 265 - 280.

M. Grötschel & M. W. Padberg (1979b) "On the Symmetric Travelling Salesman Problem II: Lifting Theorems and Facets". Mathematical Programming 16 (1979), 281 - 302.

M. Grötschel & M. W. Padberg (1983a) "Polyhedral Aspects of the Travelling Salesman Problem I: Theory" in: E. L. Lawler, J. K. Lenstra & A. H. G. Rinnooy Kan (eds.) "The Travelling Salesman Problem". John Wiley and Sons, London, New York, 1983.

M. Grötschel & M. W. Padberg (1983b) "Polyhedral Aspects of the Travelling Salesman Problem II: Computation". in: E. L. Lawler, J. K. Lenstra & A. H. G. Rinnooy Kan (eds.) "The Travelling Salesman Problem". John Wiley and Sons, London, New York, 1983.

M. Grötschel & W. R. Pulleyblank (1981) "Clique Tree Inequalities and the Symmetric Travelling Salesman Problem". Report No. 81196-OR, Institut für Operations Research, Universität Bonn, 1981.

P. L. Hammer, E. L. Johnson & U. N. Peled (1975) "Facets of Regular 0-1-Polytopes". Mathematical Programming 8 (1975), 179 - 206.

K. Helbig Hansen & J. Krarup (1974) "Improvements of the Held-Karp Algorithm for the Symmetric Travelling Salesman Problem". Mathematical Programming 7 (1974), 87 - 96.

M. Held & R. M. Karp (1970) "The Travelling Salesman Problem and Minimum Spanning Trees". Operations Research 18 (1970), 1138 - 1182.

M. Held & R. M. Karp (1971) "The Travelling Salesman Problem and Minimum Spanning Trees: Part II". Mathematical Programming 1 (1971), 6 - 26.

A. J. Hoffman & J. B. Kruskal (1956) "Integral Boundary Points of Convex Polyhedra". in: H. W. Kuhn & A. W. Tucker (eds.) "Linear Inequalities and Related Systems". Princeton, University Press, Princeton, New Jersey (1956), 223 - 246.

E. L. Johnson & U. Suhl (1980) "Experiments in Integral Programming". Discrete Applied Mathematics 2 (1980), 39 - 55.

M. Jünger (1984) "Polyhedral Combinatorics and the Acyclic Subdigraph Problem", Mathematisches Institut, Universität Augsburg, to appear.

R. Kaas (1981) "A Branch and Bound Algorithm for the Acyclic Subgraph Problem". European Journal of Operational Research 8 (1981), 355 - 362.

R. M. Karp & C. H. Papadimitriou (1980) "On Linear Characterizations of Combinatorial Optimization Problems". Proc. of the 21st Annual Symp. on the Foundations of Computer Science, IEEE (1980), 1 - 9.

L. G. Khachian (1979) "A Polynomial Algorithm in Linear Programming". Doklady Academii Nauk SSSR 244 (1979) (English translation: Soviet Math. Dokl. 20 (1979), 191 - 194).

B. Korte & W. Oberhofer (1968) "Zwei Algorithmen zur Lösung eines komplexen Reihenfolgeproblems". Unternehmensforschung 12 (1969), 217 - 231.

B. Korte & W. Oberhofer (1969) "Zur Triangulation von Input-Output Matrizen". Jb. f. Nat. Ök. u. Stat. 182 (1969), 398 - 433.

W. Krelle (1964) in: F. Neumark (ed.) "Strukturwandlungen einer wachsenden Wirtschaft". Schriften des Vereins für Socialpolitik, NF Band 30/II (1964), 1059 - 1063.

K. Kuratowski (1930) "Sur le Probleme des Courbes Gauches en Topologie". Fund. Math. 15 (1930), 271 - 283.

A. H. Land & A. G. Doig (1960) "An Automatic Method for Solving Discrete Programming Problems". Econometrica 28 (1960), 497 - 520.

E. L. Lawler (1976) "Combinatorial Optimization: Networks and Matroids".

Holt, Rinehart and Winston, 1976.

E. L. Lawler, J. K. Lenstra & A. H. G. Rinnooy Kan (eds.) (1983) "The Travelling Salesman Problem". John Wiley and Sons, London, New York, 1983.

H. W. Lenstra Jr. (1973) "The Acyclic Subgraph Problem". Report BW26, Mathematisch Centrum, Amsterdam, 1973.

J. K. Lenstra (1977) "Sequencing by Enumerative Methods". Mathematical Centre Tracts 69, Mathematisch Centrum, Amsterdam, 1977.

L. Lovász (1976) "On Two Minimax Theorems in Graph Theory". J. Combinatorial Theory (B) 21 (1976), 96 - 103.

C. L. Lucchesi (1976) "A Minimax Equality for Directed Graphs". Doctoral Thesis, University of Waterloo, Waterloo, Ontario, 1976.

C. L. Lucchesi & D. H. Younger (1978) "A Minimax Relation for Directed Graphs". J. London Math. Soc. (2) 17 (1978), 369 - 374.

J. F. Marcotorchino & P. Michaud (1979) "Optimisation en Analyse Ordinale des Données". Masson, Paris, 1979.

M. W. Padberg (1979) "Covering, Packing and Knapsack Problems". Annals of Discrete Mathematics 4 (1979), 265 - 287.

M. W. Padberg & S. Hong (1980) "On the Symmetric Travelling Salesman Problem: A Computational Study". Mathematical Programming Studies 12 (1980), 78 - 107.

M. W. Padberg & M. R. Rao (1981) "The Russian Method for Linear Inequalities III: Bounded Integer Programming". INRIA, Rapport de Recherche, Rocquencourt, revised May 1981, to appear in Mathematical Programming Studies.

M. W. Padberg & M. R. Rao (1982) "Odd Minimum Cut Sets and b-Matchings". Mathematics of Operations Research 7 (1982), 67 - 80.

C. H. Papadimitriou & K. Steiglitz (1982) "Combinatorial Optimizations: Algorithms and Complexity". Prentice Hall, Englewood Cliffs, New Jersey, 1982.

W. R. Pulleyblank (1973) "Facets of Matching Polyhedra". Doctoral Thesis, University of Waterloo, Waterloo, Ontario, 1973.

W. R. Pulleyblank (1983) "Polyhedral Combinatorics". in: A. Bachem, M. Grötschel & B. Korte (eds.) "Mathematical Programming, Bonn 1982 - The State of the Art". Springer Verlag, Berlin (1983), 312 - 345.

W. R. Pulleyblank & J. Edmonds (1974) "Facets of 1-Matching Polyhedra". in: C. Berge & D. Ray-Chaudhuri (eds.) "Hypergraph Seminar". Springer Verlag, Berlin (1974), 214 - 242.

G. Reinelt (1984) "The Linear Ordering Problem: Algorithms and Applications". Mathematisches Institut, Universität Augsburg, to appear.

A. Schrijver (1981) "On Total Dual Integrality". Linear Algebra and its Applications 38 (1981), 27 - 32.

A. Schrijver (1983) "Min-Max Results in Combinatorial Optimization". in: A. Bachem, M. Grötschel & B. Korte (eds.) "Mathematical Programming, Bonn 1982 - The State of the Art". Springer Verlag, Berlin (1983), 439 - 500.

A. Schrijver (1984) "Polyhedral Combinatorics". John Wiley and Sons, New York, to appear.

P. Slater (1961) "Inconsistencies in a Schedule of Paired Comparisons". Biometrika 48 (1961), 303 - 312.

S. Warshall (1962) "A Theorem on Boolean Matrices". J. ACM, 9 No. 1 (1962), 11 - 12.

H. Wessels (1981) "Triangulation und Blocktriangulation von Input-Output Tabellen. Deutsches Institut für Wirtschaftsforschung, Beiträge zur Strukturforschung, Heft 63, Berlin, 1981.

H. Weyl (1935) "Elementare Theorie der konvexen Polyeder". Commentarii Mathematici Helvetici 7 (1935), 290 - 306.

H. P. Young (1978) "On Permutations and Permutation Polytopes". Mathematical Programming Study 8 (1978), 128 - 140.