Mathematical Programming 62 (1993) 215–222
North-Holland

The facets of the polyhedral set determined by the Gale–Hoffman inequalities

Stein W. Wallace

Division of Economics, Norwegian Institute of Technology, The University of Trondheim, Norway

Roger J.-B. Wets*

Department of Mathematics, University of California, Davis, CA, USA

Received 2 February 1992
Revised manuscript received 29 July 1992

This paper is dedicated to Phil Wolfe on the occasion of his 65th birthday.

The Gale–Hoffman inequalities characterize feasible external flow in a (capacitated) network. Among these inequalities, those that are redundant can be identified through a simple arc-connectedness criterion.

Key words: Facets, convex polyhedral set, feasible flow, Gale–Hoffman inequalities.

The Gale–Hoffman lemma characterizes feasible external flow (supply, demand) in a directed, capacitated network in terms of a number of linear inequalities. These inequalities are generated by considering all node subsets and requiring that the external flow associated with each subset be less or equal to the total capacity of the arcs exiting from it. For a particularly appealing statement of this lemma consult [11]. This paper is devoted to a refinement of this result. We are concerned with characterizing feasibility in terms of a *minimal* number of these inequalities. We are going to provide a simple criterion that identifies the linear inequalities that are redundant. One can also express this geometrically: the Gale–Hoffman inequalities determine a (convex) polyhedral set, and we are now interested in finding its facets. Thus, one can view this note as providing the description of the facets of one more polyhedral set encountered in combinatorial optimization, see e.g., [5], [10]; at a formal level, the criterion is similar in nature to the one obtained by Balas and Pulleyblank [1] for the perfectly matchable subgraph polytope of a bipartite graph.

The framework of this paper is somewhat more general than that in the Gale [3], Hoffman [6] papers. We shall be interested in the polyhedral set (which turns out to be a polyhedral convex cone) of feasible external flows *and* capacities. This requires a minor reformulation of Gale–Hoffman result, but no new argumentation is necessary.

Correspondence to: Roger J.-B. Wets, Department of Mathematics, University of California, Davis, CA 95616, USA.
Research supported in part by NATO Collaborative Research Grant 0785/87.
* Supported in part by a grant of the Air Force Office of Scientific Research.

Let $[\mathcal{N}, \mathcal{A}]$ denote the nodes and (directed) arcs of a network. For $Y \subset \mathcal{N}$, let

$$\vec{Y} := \{k \in \mathcal{A} \mid k = (i, j), i \in Y, j \in \mathcal{N} \setminus Y\}$$

be the (directed) arcs connecting Y to $\mathcal{N} \setminus Y$, and

$$F^+(Y) := \{j \in \mathcal{N} \mid (i, j) \in \vec{Y}\} \cup Y$$

consist of Y itself and all the nodes that can be reached (in 1 step) from Y.

The vector $b(Y) \in \mathbb{R}^{|\mathcal{N}|}$ is the node-membership vector associated with Y, i.e.,

$$b(Y)_i = \begin{cases} 1 & \text{if node } i \text{ belongs to } Y, \\ 0 & \text{otherwise.} \end{cases}$$

External flow is denoted by $f \in \mathbb{R}^{|\mathcal{N}|}$ and represents supply and demand: $f_i > 0$ means that node i is a supply node, $f_i < 0$ means that i is a demand node, and $f_i = 0$ corresponds to a transshipment node. The total external flow associated with a collection of nodes Y is thus $\langle b(Y), f\rangle$; this quantity could be positive (supply exceeds demand in Y), negative (demand exceeds supply in Y), or 0 (supply and demand are balanced in Y).

Similarly, $a(\vec{Y}) \in \mathbb{R}^{|\mathcal{A}|}$ records arc-membership in $\vec{Y}$, i.e.,

$$a(\vec{Y})_k = \begin{cases} 1 & \text{if arc } k \text{ is in } \vec{Y}, \\ 0 & \text{otherwise.} \end{cases}$$

Arc capacities will be denoted by a nonnegative vector $c \in \mathbb{R}^{|\mathcal{A}|}$. The total capacity of the flow that can exit a collection of nodes Y is thus limited by $\langle a(\vec{Y}), c\rangle$; this quantity is necessarily nonnegative.

A pair $(f, c) \in \mathbb{R}^{|\mathcal{N}|} \times \mathbb{R}^{|\mathcal{A}|}$ is said to be *feasible* if there exists an (internal) flow that satisfies the capacity constraints so that each node is flow-balanced (flow on out-arcs = flow on in-arcs + external flow).

In these terms, the Gale–Hoffman result takes on the following form:

Gale–Hoffman lemma. *Given a network* $[\mathcal{N}, \mathcal{A}]$, *a pair of vectors* $(f, c) \in \mathbb{R}^{|\mathcal{N}|} \times \mathbb{R}^{|\mathcal{A}|}$ (*external flow, capacities*) *is feasible if and only if the external flow is balanced*

$$\langle e, f\rangle = 0$$

with $e = (1, 1, \ldots, 1)$, *and for all* $Y \subset \mathcal{N}$,

$$\langle b(Y), f\rangle \leqslant \langle a(\vec{Y}), c\rangle. \qquad \square$$

We refer to the preceding inequalities as the *Gale–Hoffman inequalities.*

Unfortunately, when trying to make use of this characterization of feasibility, one is usually somewhat overwhelmed by the size of this system of inequalities. For example, if $\mathcal{N}$ has 30 nodes, one has to cope with 1 073 741 825 inequalities $(2^{|\mathcal{N}|} + 1)$. Fortunately, in most applications, a significant number (sometimes more than 99%) of the Gale–Hoffman inequalities may be redundant; an inequality is *redundant* if it is implied by a combination of the other inequalities. This opens the door to a reduction in size by eliminating the redundant inequalities. Getting rid of redundant constraints is usually achieved by means of linear programming-based procedures, see [2, 4] and references therein. But this approach runs quickly out of steam as soon as $|\mathcal{N}|$ is mildly large (larger than 10, for example).

Our own motivation came from some questions in stochastic programming [9, 12] where one is interested in the probability that under a certain choice of capacities, there will be feasible flow, given that the external flow is only known in probability, i.e., external flow is a random quantity and one needs to check feasibility for *all* possible realizations. For example, given c and a probability distribution for f, we would like to compute

$$\text{Prob}\{\langle b(Y), f\rangle \leqslant \langle a(\vec{Y}), c\rangle \text{ for all } Y\subset\mathcal{N}\}.$$

This would be possible — by checking the inequalities for a few million samples, for example — provided the number of inequalities is not too large, suggesting the elimination of all redundant inequalities. This approach turns out to be much more efficient than checking feasibility by flow algorithms for a few million cases!

The following theorem provides a simple characterization of redundancy for the Gale–Hoffman inequalities, which can be checked without recourse to linear programming-based techniques. In fact, all that is required is a procedure that checks for arc-connectedness.

Recall that given a system of linear inequalities $\langle d_k, x\rangle \leqslant 0$ for $k\in\mathcal{K}$, a linear inequality $\langle d_{\hat{k}}, x\rangle \leqslant 0$ is *redundant* (i.e., implied by the other inequalities) if $d_{\hat{k}}$ can be obtained as a positive linear combination of the remaining d_k, $k\in(\mathcal{K}\setminus\hat{k})$.

A node-set $Y\subset\mathcal{N}$ is *arc-connected* if there are no sets V_1, V_2 that partition Y such that and $F^+(V_1)\cap F^+(V_2)\cap Y=\emptyset$; equivalently, if the arcs of $\mathcal{A}$ are taken to be undirected, the subgraph induced by the nodes of Y is connected.

Theorem (characterization of redundancy). *For all nonempty $Y\subset\mathcal{N}$, the associated inequality*

$$\langle b(Y), f\rangle \leqslant \langle a(\vec{Y}), c\rangle$$

is redundant if and only if Y or $\mathcal{N}\setminus Y$ are not arc-connected.

Proof. If the sets V_1, V_2 partition Y, then $b(V_1)+b(V_2)=b(Y)$. Moreover, if they are not arc-connected there are no arcs from V_1 to V_2 and vice versa, i.e., $a(\vec{V}_1)+a(\vec{V}_2)=a(\vec{Y})$. Thus the inequality $\langle b(Y), f\rangle \leqslant \langle a(\vec{Y}, c\rangle$ is the sum of the two inequalities:

$$\langle b(V_1), f\rangle \leqslant \langle a(\vec{V}_1), c\rangle, \qquad \langle b(V_2), f\rangle \leqslant \langle a(\vec{V}_2), c\rangle.$$

Similarly, if U_1, U_2 partition $\mathcal{N}\setminus Y$, let $U_1' := U_1\cup Y$, $U_2' := U_2\cup Y$, then $b(U_1')+b(U_2')=b(Y)+e$. Also, $a(\vec{Y})=a(\vec{U}_1')+a(\vec{U}_2')$ since the condition

$F^+(U_1) \cap F^+(U_2) \cap (\mathcal{N} \setminus Y) = \emptyset$ implies that $\vec{U}_1'$ consists of the arcs from Y into U_2, and $\vec{U}_2'$ consists of the arcs from Y into U_1, i.e., $\vec{U}_1' \cup \vec{U}_2' = \vec{Y}$ and $\vec{U}_1' \cap \vec{U}_2' = \emptyset$.

To prove the "only if" direction, we need to show that the vector $(-a(Y), b(Y))$ cannot be obtained as a positive linear combination of all other such pairs and the vector $(0, -e)$ where this last vector comes from the inequality $\langle -e, f\rangle \leqslant 0$ associated with the condition that the external flow must be balanced; $\langle e, f\rangle \leqslant 0$ is the same as $\langle b(\mathcal{N}), f\rangle \leqslant 0 = \langle a(\vec{\mathcal{N}}), c\rangle$.

Let l, $\underline{l}$ be such that $Y_l := Y$ and $Y_{\underline{l}} = \mathcal{N} \setminus Y_l$. Let us also define $b^j := b(Y_j)$, $a^j = a(\vec{Y}_j)$ and $J_l := \{1, 2, \ldots, 2^{|\mathcal{N}|}\} \setminus \{l, \underline{l}\}$. What we have to show is that the following linear system has no solution:

$$a^l = \sum_{j \in J_l} x_j a^j + x_{\underline{l}} a^{\underline{l}}, \qquad b^l = \sum_{j \in J_l} x_j b^j + x_{\underline{l}} b^{\underline{l}} - x_0 e,$$

$$x_0 \geqslant 0, \qquad x_{\underline{l}} \geqslant 0, \qquad x_j \geqslant 0 \quad \forall j \in J_l.$$

To the contrary, let us suppose that $(\bar{x}_0, \bar{x}_{\underline{l}}, (\bar{x}_j, j \in J_l))$ is a (feasible) solution, and let us examine the properties that such a solution must possess. Let

$$J_l^+ := \{ j \in J_l \mid \bar{x}_j > 0\}.$$

Observe that if $k \notin \vec{Y}$, $a_k^l = 0$ and thus

$$0 = \sum_{j \in J_l^+} \bar{x}_j a_k^j + \bar{x}_{\underline{l}} a_k^{\underline{l}} \quad \forall k \notin \vec{Y}_l$$

which would mean that $\bar{x}_j = 0$ if $\vec{Y}_j \not\subset \vec{Y}_l$, i.e., $j \notin J_l^+$, or equivalently,

$$j \in J_l^+ \;\Rightarrow\; \vec{Y}_j \subset \vec{Y}_l.$$

For all $Y_j \subset \mathcal{N}$, we define

$$\tilde{Y}_j := Y_j \cap Y_l \quad \text{and} \quad \bar{Y}_j := Y_j \cap Y_{\underline{l}}.$$

We are going to argue that if Y_j is such that either $\tilde{Y}_j$ is a proper subset of Y_l $(\emptyset \neq \tilde{Y}_j \neq Y_l)$, or $\bar{Y}_j$ is a proper subset of $Y_{\underline{l}}$ $(\emptyset \neq \bar{Y}_j \neq Y_{\underline{l}})$ then $j \notin J_l^+$.

To see this we consider the system of equations involving b^l which we now write as follows:

$$(1 + \bar{x}_0) = \sum_{j \in J_l^+} \bar{x}_j b_i^j \quad \forall i \in Y_l, \qquad (\bar{x}_0 - \bar{x}_{\underline{l}}) = \sum_{j \in J_l^+} \bar{x}_j b_i^j \quad \forall i \in Y_{\underline{l}}.$$

If $q \in J_l^+$ and $\emptyset \neq \tilde{Y}_q \neq Y_l$, from the arc-connectedness assumption it follows that

$$\text{either} \quad F^+(\tilde{Y}_q) \cap Y_l \neq \emptyset \quad \text{or} \quad F^+(Y_l \setminus \tilde{Y}_q) \cap Y_l \neq \emptyset.$$

Since $q \in J_l^+$, $\vec{Y}_q \subset \vec{Y}_l$, we always have $F^+(\tilde{Y}_q) \cap Y_l = \emptyset$, hence

$$\exists (i \in Y_l \setminus \tilde{Y}_q,\, i' \in \tilde{Y}_q) \quad \text{such that } i' \in F^+(i).$$

Now observe that whenever this node $i \in Y_j$ for some $j \in J_l^+$, then i' necessarily belongs to Y_j. Indeed, $j \in J_l^+$ implies $\vec{Y}_j \subset \vec{Y}_l$ and thus the arc $(i, i') \notin \vec{Y}_j$, i.e., $i' \in Y_j$. For this pair i, i', we would have

$$(1+\bar{x}_0)=\sum_{j\in J_l^+}\bar{x}_j b_{i'}^j<\bar{x}_q+\sum_{j\in J_l^+}\bar{x}_j b_{i'}^j\leqslant\sum_{j\in J_l^+}\bar{x}_j b_i^j=(1+\bar{x}_0),$$

a clear contradiction. Thus, there are no $q\in J_l^+$ with $\tilde{Y}_q$ a proper subset of Y_l.

The same argument is used to prove that there are no $q\in J_l^+$ with $\bar{Y}_q$ a proper subset of $Y_{\underline{l}}$, using this time the system of equations

$$(\bar{x}_0-\bar{x}_{\underline{l}})=\sum_{j\in J_l^+}\bar{x}_j b_i^j\quad\forall i\in Y_{\underline{l}}.$$

We give the details for the sake of completeness. If $q\in J_l^+$ and $\emptyset\neq\bar{Y}_q\neq Y_{\underline{l}}$, from the arc-connectedness assumption it follows that

$$\text{either}\quad F^+(\bar{Y}_q)\cap Y_{\underline{l}}\neq\emptyset\quad\text{or}\quad F^+(Y_{\underline{l}}\setminus\bar{Y}_q)\cap Y_{\underline{l}}\neq\emptyset.$$

Since $q\in J_l^+$, $\bar{Y}_q\subset\bar{Y}_l$, we always have $F^+(\bar{Y}_q)\cap Y_{\underline{l}}=\emptyset$, hence

$$\exists(i\in Y_{\underline{l}}\setminus\bar{Y}_q,\, i'\in\bar{Y}_q)\quad\text{such that } i'\in F^+(i).$$

Now observe that whenever this $i\in Y_j$ for some $j\in J_l^+$, then i' necessarily belongs to Y_j. Indeed, $j\in J_l^+$ implies $\bar{Y}_j\subset\bar{Y}_l$ and thus the arc $(i, i')\notin\bar{Y}_j$, i.e., $i'\in Y_j$. For this pair, i, i', we would have

$$(\bar{x}_0-\bar{x}_l)=\sum_{j\in J_l^+}\bar{x}_j b_{i'}^j<\bar{z}_q+\sum_{j\in J_l^+}\bar{x}_j b_{i'}^j\leqslant\sum_{j\in J_l^+}\bar{x}_j b_i^j=(x_0-\bar{x}_l),$$

a contradiction. Thus, there are no $q\in J_l^+$ with $\bar{Y}_q$ a proper subset of $Y_{\underline{l}}$.

Thus the only index that could possibly belong to J_l^+ is the index corresponding to $\mathcal{N}$, say $Y_p=\mathcal{N}$. This would mean that $(\bar{x}_0, \bar{x}_{\underline{l}}, \bar{x}_p)$ satisfies the following relations:

$$1+\bar{x}_0=\bar{x}_p,\quad \bar{x}_0-\bar{x}_{\underline{l}}=\bar{x}_p,\qquad \bar{x}_0\geqslant 0,\ \bar{x}_{\underline{l}}\geqslant 0,\ \bar{x}_p\geqslant 0,$$

which is not possible. Thus J_l^+ must be empty. But that also is not possible, since then we would have $b^l=\bar{x}_{\underline{l}}b^{\underline{l}}-\bar{x}_0 e$ with $\bar{x}_0\geqslant 0$, $\bar{x}_l\geqslant 0$.

Thus, when Y and $\mathcal{N}\setminus Y$ are arc-connected,

$$\langle b(Y), f\rangle\leqslant\langle a(\tilde{Y}), c\rangle$$

is never redundant. □

If, capacities $\bar{c}$ are given, and one is interested in the facets of the polyhedron

$$P_f=\{f\mid\langle -e, f\rangle\leqslant 0,\ \langle b(Y), f\rangle\leqslant\langle a(\tilde{Y}), c\rangle\ \forall Y\subset\mathcal{N}\}$$

the frontal approach would again lead to linear programming problems of huge size (with $2^{|\mathcal{N}|}+1$ rows or columns depending on the formulation). Fortunately, we can first use the criterion provided by the preceding theorem to identify the facets of the polyhedral convex cone

$$P_{(f,c)}=\{(f, c)\mid\langle -e, f\rangle\leqslant 0,\ \langle b(Y), f\rangle-\langle a(\tilde{Y}), c\rangle\leqslant 0,\ \forall Y\subset\mathcal{N}\}$$

determined by the Gale–Hoffman inequalities, assign the value $\bar{c}$ to c, and only then have recourse to linear programming-based techniques.

Implementation. Given the arc-connectedness criterion, it is fairly easy to design a recursive procedure based on examination of the subsets of $\mathcal{N}$. Because of the symmetry of our criterion in Y and $\mathcal{N}\setminus Y$, it will really suffice to parse through only half of the subsets of $\mathcal{N}$, and for each $Y \in \mathcal{N}$ corresponding to a nonredundant inequality, we shall then also output the "complementary" inequality:

$$\langle b(\mathcal{N}\setminus Y), f\rangle \leqslant \langle a(\overrightarrow{\mathcal{N}\setminus Y}), c\rangle.$$

The following recursive algorithm, which represents a postorder, depth first, traversal of a binary tree, will do the job. It is initialized with $Y=\emptyset$ and $W=\mathcal{N}\setminus\{i\}$ for some node i.

```
procedure Facets(Y, W: set of nodes);
begin
  if W ≠ ∅ then begin
    PickNode(i, W);
    W := W\{i};
    Facets(Y, W);
    Y := Y ∪ {i};
    Facets(Y, W);
    if Connected(Y) and Connected(N\Y) then CreateIneq(Y);
  end;
end;
```

The procedure PickNode(i, W) simply picks a node i from the set W, and procedure CreateIneq(Y) outputs the two inequalities

$$\langle b(Y), f\rangle \leqslant \langle a(\check{Y}), c\rangle, \qquad \langle b(\mathcal{N}\setminus Y, f\rangle \leqslant \langle a(\overrightarrow{\mathcal{N}\setminus Y}), c\rangle.$$

The function Connected(Y) checks if the network defined by the nodes in Y is arc-connected. An example of such an algorithm appears in Section 6.2 in [7]. The preceding procedure will produce all the nonredundant inequalities except $\langle -e, f\rangle \leqslant 0$ associated with the requirement that the external flow must be balanced (total supply equals total demand).

As a small example, consider Figure 1. It shows a network with 5 nodes and 7 arcs, and all the cuts shown in Figure 1 give rise to two inequalities. In addition to the inequality that guarantees balanced (external) flow, there are $2^5=32$ Gale–Hoffman inequalities. Procedure Facets checks $2^{5-1}-1=15$ subsets of $\mathcal{N}$. It turns out that 12 out of the 15 sets give rise to nonredundant inequalities. We initialize procedure Facets with $W=\mathcal{N}\setminus\{5\}$, so that $5\in\mathcal{N}\setminus Y$ for all sets Y. The three sets giving rise to redundant inequalities are $Y=\{1,4\}$ and $Y=\{2,3\}$, both resulting in a disconnected Y and $Y=\{2,3,4\}$ making $\mathcal{N}\setminus Y$ disconnected. In total, the example therefore has $(12+1)\cdot 2=26$ nonredundant inequalities.

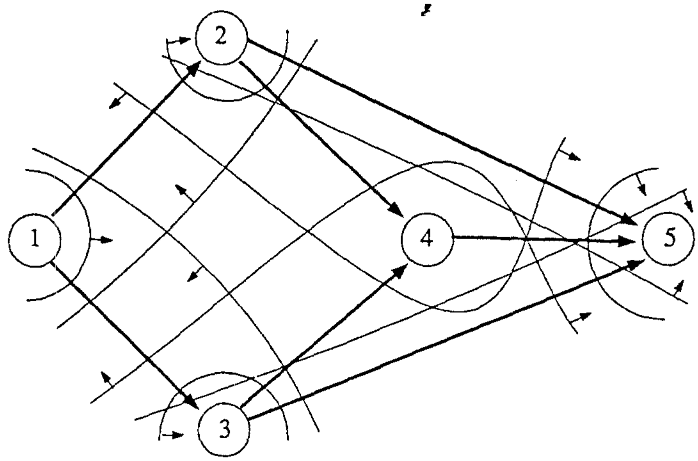

Fig. 1. All cuts drawn give rise to two nonredundant inequalities.

The procedure was also tested for some larger examples. Using NETGEN [8], we generated a number of test networks. As one would expect, the number of non-redundant inequalities (for a fixed number of nodes) is generally related to arc-density of the network. In a network with 20 nodes and 100 arcs starting with $2^{20} = 1\,048\,576$ Gale–Hoffman inequalities, we found that only 14.2% of the inequalities were redundant; excluding input/output times, it took about 50 minutes on a SUN 3/50 to identify all the redundant inequalities. In another example with 20 nodes and 31 arcs (in other words with much lower arc-density) we found that 99.65% of the inequalities were redundant, and the calculation took about 20 seconds on the same computer, using this time some performance improving (algorithmic) schemes outlined in [12]. This last example is very illustrative because it has a number of important features. Firstly, the number of Gale–Hoffman inequalities is so huge that one cannot hope to use them successfully for any purpose and certainly not to compute network reliability. Secondly, removing dominated inequalities based on linear programming-like algorithms is as good as out of the question because of the size of the linear programs involved. But the number of facets of the feasibility set is low enough (3632) that one can make use of them in different applications.

References

[1] E. Balas and W. Pulleyblank, "The perfectly matchable subgraph polytope of a bipartite graph," *Networks* 13 (1983) 495–516.

[2] J.W. Chinneck, "Localizing and diagnosing infeasibilities in networks," Working Paper SCE-90-14, Carleton University (Ottawa, Ont., 1990).

[3] D. Gale, "A theorem of flows in networks," *Pacific Journal of Mathematics* 7 (1957) 1073–1082.

[4] H.J. Greenberg, "Diagnosing infeasibility in min-cost network flow problems. Part I: Primal infeasibility," *IMA Journal of Mathematics in Management* 2 (1988) 39–51.

[5] M. Grötschel, *Polyedrische Charakterisierungen Kombinatorischer Optimierungsprobleme* (Hain, Meisenhein am Glan, 1977).

[6] A. Hoffman, "Some recent applications of the theory of linear inequalities to extremal combinatorial analysis," *Proceedings Symposium Applied Mathematics* 10 (1960) 113–128.

[7] E. Horowitz and S. Shani, *Fundamentals of Data Structures* (Pitman, London, 1976).

[8] D. Klingman, A. Napier and J. Stutz, "NETGEN – a program for generating large scale (un)capacitated assignment, transportation and minimum cost flow networks," *Management Science* 20 (1974) 814–822.

[9] A. Prékopa and E. Boros, "On the existence of a feasible flow in a stochastic transportation network," *Operations Research* 39 (1991) 119–129.

[10] W. Pulleyblank, "Polyhedral combinatorics," in: A. Bachem, M. Grötschel and B. Korte, eds., *Mathematical Programming: The State of the Art* (Springer, Berlin, 1983) pp. 312–345.

[11] R.T. Rockafellar, *Network Flows and Monotropic Optimization* (Wiley, New York, 1984).

[12] W. Wallace and R.J.-B. Wets, "Preprocessing in stochastic programming: the case of capacited networks," to appear in: *ORSA Journal on Computing* (1993).