JOURNAL OF COMBINATORIAL THEORY, Series B 59, 163–171 (1993)

Integer Plane Multiflows with a Fixed Number of Demands

ANDRÁS SEBŐ*

CNRS, ARTEMIS IMAG, Université Joseph Fourier, BP 53x, 38041 Grenoble Cedex, France

Received March 28, 1989; revised April 14, 1992

We give a polynomial algorithm which decides the integer solvability of multicommodity flow problems where the union of "capacity-" and "demand-edges" forms a planar graph, and the number of demand edges is bounded by a prefixed integer k. This problem was solved earlier for $k=2$ by Seymour and for $k=3$ by Korach. For $k=4$ much work has been done by Korach and Newmann. The main result of the present note is a polynomial algorithm that finds such a multiflow or proves that it does not exist, for arbitrary fixed k. Middendorf and Pfeiffer have recently proved that this problem is NP-complete in general (without fixing k). We actually give a more general polynomial algorithm, namely to decide whether the relation $\nu(G, T) = \tau(G, T)$ or its weighted generalization holds for the pair (G, T) (where G is not necessarily planar), provided $|T|$ is fixed, thus extending Seymour's method and result for $|T| = 4$.

1. INTRODUCTION

We are going to study multicommodity flow feasibility in the case when the graph defined by the union of the demand- and capacity-edges forms a planar graph. Let us formulate this problem precisely:

Suppose that G is a planar graph, $R \subseteq E(G)$, $r: R \to \mathbb{N}$ and $c: E(G)\backslash R \to \mathbb{N}$ ($\mathbb{N}$ is the set of positive integers). A set $\mathscr{C}$ of circuits and a function $f: \mathscr{C} \to \mathbb{N}$ has to be found so that

(i) For all $C \in \mathscr{C}: |C \cap R| = 1$

(ii) For all $e \in R: \sum_{e \in C \in \mathscr{C}} f(C) = r(e)$,

(iii) For all $e \in E(G)\backslash R: \sum_{e \in C \in \mathscr{C}} f(C) \leqslant c(e)$.

(Think of f as a function telling the *multiplicities* of elements of $\mathscr{C}$.) We shall say that (G, R, r, c) is a *network*. $r(xy)$ is called the *demand* of the pair

* Research partially supported by the Alexander von Humboldt foundation, while the author visited the Institut für Diskrete Mathematik, Bonn, and by the European project SCIENCE.

0095-8956/93 $5.00

x, y, $xy \in R$, *and* $c(e)$, $e \in E(G) \setminus R$ is the capacity of e. f, or more precisely $(\mathscr{C}, f)$ will be called a *flow*, if it satisfies (i), (ii), and (iii). (A flow is always *integer* in this paper.)

The problem of finding a flow in a planar network has been investigated a lot, partly due to the underlying nice combinatorial structure and its appealing relation to matching theory. Seymour [23] discovered its relation to the Chinese postman problem and used it to solve the plane multicommodity flow problem for Eulerian graphs and to settle the case of two demand-edges. This relation has become the alpha and the omega of results about planar multiflows (see, for example, Barahona [1], Korach [12], etc.) In Section 3 we shall explain this relation, which will be the starting point in the proof of our main result. It will enable us to use the results of Section 2 which concern the Chinese postman problem.

It has been a long-standing unsolved problem whether the plane multicommodity flow problem is polynomially solvable, until Middendorf and Pfeiffer [19] proved that it is NP-complete. *The result of this paper is that this problem is polynomially solvable if the number of demand-edges is bounded by a fixed integer k. In other words, for any integer k we shall give a polynomial algorithm that decides the existence of a multicommodity flow for an arbitrary network* (G, R, r, c) *for which G is planar and* $|R| \leq k$. For $k=2$ this problem has been settled earlier by Seymour [23] (cf. also Lomonosov [16] and Sebő [22], for a generalization cf. Frank [8], [9]) and for $k=3$ by Korach [12]. For $k=4$ many results were achieved by Korach and Newmann [13]. For recent surveys consult [9] or [7].

In the special case where all capacities are 1, the multicommodity flow problem specializes of course to the problem of finding edge-disjoint paths between a given set of pairs of vertices. Note that even this problem is NP-complete (Even, Itai, and Shamir [6], cf. Garey and Johnson [10]), but if the number of demand-edges is fixed and r is 1 everywhere on R, it is polynomially solvable according to the celebrated result of Robertson and Seymour [20]. Our problem is independent of this latter one: we allow *arbitrary demands and capacities* but we suppose *planarity*. Without supposing planarity the same problem is NP-complete even for $k=2$ (see [6] again).

Finally, let us formulate our problem more precisely. Suppose k is a positive integer, and define the following problem:

UNDIRECTED PLANE k-COMMODITY INTEGRAL FLOW.

Instance: Network (G, R, r, c), where G is a planar graph and $R \subseteq E(G)$, $|R| \leq k$.

Question: Does there exist a flow in this network?

I think it is really surprising how straightforward the solution of this problem turns out to be. It is also somewhat disappointing that the answer does not lie as deep as one could have expected.

2. Odd cut Packings

Let G be a graph, and $T \subseteq V(G)$, $|T|$ even. A *T-join* is a set of edges $F \subseteq E(G)$ such that $d_F(x)$ is odd if and only if $x \in T$. A *T-cut* is a cut $\delta(X)$ such that X is *T-odd*, that is, $|X \cap T|$ is odd. If $T = V(G)$, we just say that $\delta(X)$ is an *odd cut*. ($\delta(X)$ denotes the set of edges with exactly one endpoint in X. $d(X) := |\delta(X)|$. If we wish to emphasize that this set of edges is considered in the graph G, we write $\delta_G(X)$ and $d_G(X)$.)

Let now $w: E(G) \to \mathbb{Z}^+$. ($\mathbb{Z}^+$ is the set of non-negative integers). A w-packing of T-cuts is a family $\mathcal{T}$ of T-cuts with a function $g: \mathcal{T} \to \mathbb{N}$, which has the property that for all $e \in E(G)$, $\sum_{e \in C \in \mathcal{T}} g(C) \leqslant w(e)$. (Think of g as a function telling the *multiplicities* of the elements of $\mathcal{T}$; that is, $g(C)$ is the number of *copies* of C.) $\sum_{T \in \mathcal{T}} g(T)$ will be called the *value* of the w-packing.

The minimum weight of a T-join will be denoted by $\tau(G, T, w)$ and the maximum value of a w-packing of T-cuts will be denoted by $\nu(G, T, w)$. If F is a T-join and C is a T-cut, then obviously $|F \cap C|$ is odd. In particular, $|F \cap C| \geqslant 1$, and $\tau(G, T, w) \geqslant \nu(G, T, w)$ follows. We shall say that (G, T, w) has the *Seymour property* if $\tau(G, T, w) = \nu(G, T, w)$.

In this section we give an algorithm, which, for fixed T, determines a maximum w-packing of T-cuts in polynomial time. In Section 3 *we will see: this immediately implies that for fixed T the Seymour property can be tested in polynomial time, and this later problem, in turn, contains the plane k-commodity flow problem.*

Let us recall a well-known fact about packings of T-cuts: for any w-packing of T-cuts there exists another w-packing of T-cuts which is laminar, and whose value is equal to the value of the original packing. (A family of sets is called *laminar*, if for any two X_1, X_2 of its element, $X_1 \subseteq X_2$ or $X_2 \subseteq X_1$ or $X_1 \cap X_2 = \emptyset$. A packing of cuts is called laminar, if it is the set of coboundaries of a laminar family.)

Indeed, if $\mathcal{T}$ is not laminar, we can "uncross" it as follows. Take $\delta(X_1)$, $\delta(X_2)$ such that $X_1 \subsetneq X_2$, $X_2 \subsetneq X_1$, and $X_1 \cap X_2 \neq \emptyset$. We can suppose in addition that $X_1 \cap X_2$ is T-odd, because if not, we replace X_2 by $V(G) \backslash X_2$. Then $X_1 \cup X_2$ is also T-odd, and it is readily true that $\delta(X_1 \cup X_2) + \delta(X_1 \cap X_2) \leqslant \delta(X_1) + \delta(X_2)$, where the sum means the sum of the corresponding characteristic vectors. Thus, replacing $\varepsilon \in \mathbb{N}$ copies of each of $\delta(X_1)$ and $\delta(X_2)$ by ε copies of each of $\delta(X_1 \cap X_2)$ and $\delta(X_1 \cup X_2)$ we obtain a new w-packing of T-cuts. It is not difficult to see that after having applied this "uncrossing step" a finite number of times we arrive at a laminar packing. (We will not need a polynomial worst case bound on the complexity of this uncrossing procedure.)

Recall that the fractional relaxation of a T-cut packing problem is just the dual of the Chinese postman problem. (see Edmonds and Johnson [5], or Lovász and Plummer [18]). Edmonds and Johnson also observed that the Chinese postman problem can be reduced to a weighted matching problem in a related weigthed auxiliary graph. Our algorithm will be based on the surprising fact that (integer) *optimal dual* solutions for the Chinese postman problem also correspond to (integer) maximum odd cut packings in this weigthed auxiliary graph. It follows in particular, that the well-known matching algorithms can be converted into algorithms for the Chinese postman problem, which have the same complexity. (For weighted matching algorithms see, for example, Edmonds [4], Cunningham and Marsh [3], Barahona and Cunningham [2]; the last finds an integral dual solution in the bipartite case.) However, here we will concentrate only on multicommodity flows, for more general results see the references in the remark below.

Suppose we are given the graph G, the set $T \subseteq V(G)$ ($|T|$ even), and the function $w: E(G) \to \mathbb{Z}_+$. Define the *distance function* $d(x, y) := d_{G,w}(x, y) := \min\{w(P): P$ is an (x, y) path in $G\}$. ($w(X)$ denotes the sum $\sum_{x \in X} w(x)$.)

Our key-result is the following:

For (G, T, w) with $|T| \leqslant k$, where $k \in \mathbb{N}$ is fixed in advance, a maximum w-packing of T-cuts can be determined in polynomial time. (1)

Proof. We formulate our problem as a linear integer program with a $\binom{k}{2} \times 2^{k-2}$ constraint matrix.

Let $\mathscr{P}$ be the set of partitions of T into two odd sets. Clearly, $|\mathscr{P}| = 2^{k-2}$. Define one variable x_P for each $P \in \mathscr{P}$, and put one constraint $\sum \{x_P: P \in \mathscr{P}, P$ separates t_1 and $t_2\} \leqslant d(t_1, t_2)$ for each pair $t_1, t_2 \in T$. Add to these constraints $x_P \geqslant 0$, and x_P integer, for all $P \in \mathscr{P}$. The size of this linear integer program is just what we claimed. Let us denote its solution set by $IP(w)$. (To determine the distances between pairs of vertices of T use any shortest path algorithm; see, for example, Lawler [14].) $\sum_{P \in \mathscr{P}} x_P$ will be called the *value* of $x := (x_P: P \in \mathscr{P})$; $\max_{x \in IP(w)} \sum_{P \in \mathscr{P}} x_P$ can be found in polynomial time, for by Lenstra's tesult [15] (cf. also in Schrijver [21]), integer programming problems in fixed dimension can be solved in polynomial time.

We show now a natural one-to-one correspondence between $IP(w)$ and w-packings of T-cuts (computable in linear time).

If $(\mathscr{T}, g)$ is a w-packing of T-cuts and $P \in \mathscr{P}$, let $a_P :=$

$\sum_{X:\{T\cap X, T\setminus X\}=P} g(\delta(X))$. It is easy to see that $a := (a_P: P\in\mathscr{P})\in IP(w)$, and the value of a is equal to the value of $(\mathscr{T}, g)$.

This correspondence can be reversed: let $a=(a_P: P\in\mathscr{P})\in IP(w)$. According to the preliminary remark we made on uncrossing we can suppose that $\mathscr{L} := \{P\in\mathscr{P}: a_P>0\}$ is laminar. (If not, we uncross $\{X: (X, T\setminus X)\in\mathscr{P}\}$ in the way dictated by the preliminary remark. Since $|T|$ is bounded by a constant, the uncrossing stops in constant time.) We now construct a w-packing of T-cuts which has the same value as a.

We use induction on $|V(G)| + \sum_{P\in\mathscr{P}} a_P$. If there are some edges $e\in E(G)$ with $w(e)=0$, then contract them and put the arising new vertex to T if and only if exactly one of the endpoints of e is in T. By induction, the statement is true for the arising graph, and then it obviously holds for G as well. Hence we can suppose $w(e)>0$ for all $e\in E(G)$. Then, for all $v\in T$, $m_T(v) := \min_{t\in T, t\neq v} d(v, t)\geqslant m_G(v) := \min_{x\in V(G), x\neq v} w(v, x)>0$.

Let $v\in T$, $a_{\{v, T\setminus v\}}>0$. We can suppose that such a vertex exists, because if $L=\{L_1, L_2\}\in\mathscr{L}$ and here L_1 is minimal but not a vertex, then take any $v\in L_1$ and define $a'_{\{v, T\setminus v\}} := \min\{a_L, m_T(v)\}>0$, and $a'_L := a_L - a'_{\{v, T\setminus v\}}$. On $\mathscr{L}\setminus\{L\}$ define a' to be the same as a. Clearly, $a'\in IP(w)$ has the same value as a.

Let now $m := \min\{a_{\{v, T\setminus v\}}, m_G(v)\}$. Clearly, $m>0$. Define

$$w^v(e) := \begin{cases} w(e)-m & \text{if } e\in\delta(v)\\ w(e) & \text{otherwise,}\end{cases}$$

$$a^v_L := \begin{cases} a_L-m & \text{if } L=\{v, T\setminus v\}\\ a_L & \text{otherwise,}\end{cases}$$

and let $\mathscr{L}^v := \{L\subseteq T: a^v_L>0\}$; that is, $\mathscr{L}^v=\mathscr{L}\setminus\{L\}$. We prove for $a^v=(a^v_L: L\in\mathscr{L}^v)$,

$$a^v\in IP(w^v).$$

Then, by induction, there exists a w^v-packing in G which has the same value as a^v. Adding m copies of $\delta(v)$ to this w^v-packing, we obtain a w-packing in G which has the same value as a, and we are done. A polynomial algorithm for finding this w-packing can be obviously read out of this procedure.

So the only thing remaining to be proved is:

CLAIM. *Let $d^v := d_{G, w^v}$. Then $\sum\{x_L: L\in\mathscr{L}^v$ separates t_1 and $t_2\}\leqslant d^v(t_1, t_2)$ for each pair $t_1, t_2\in T$.*

Proof. If $t_1=v$, then the claim is easy: for arbitrary $t\in T$,

$$\sum\{a^v_L: L\in\mathscr{L}^v \text{ separates } t \text{ and } v\}\leqslant d(t, v)-m,$$

because $a \in IP(w)$, and for all terms on the left-hand side, $a^v_L = a_L$, except for $a^v_{\{v, T\setminus v\}} = a_{\{v, T\setminus v\}} - m$. On the other hand, $d(t, v) - m = d^v(t, v)$, because each path joining t and v contains exactly one edge incident to v.

Let now both $t_1, t_2 \in T$ be different from v. If $d^v(t_1, t_2) = d(t_1, t_2)$ we have nothing to prove, so suppose $d^v(t_1, t_2) \neq d(t_1, t_2)$; that is, $d^v(t_1, t_2) < d(t_1, t_2)$. This happens if and only if every w^v-minimum path of G joining t_1 and t_2 contains v. Let Q be such a path. Then v cuts Q into two parts; denote these by $Q_{t_1, v}$ and Q_{v, t_2}. Note that any $P \in \mathscr{P}$ separating t_1 and t_2 either separates t_1 and v, or t_2 and v. Thus, using the above inequality,

$$\begin{aligned}
&\sum \{a^v_L : L \in \mathscr{L}^v \text{ separates } t_1 \text{ and } t_2\}\\
&\quad \leqslant \sum \{a^v_L : L \in \mathscr{L}^v \text{ separates } t_1 \text{ and } v\}\\
&\qquad + \sum \{a^v_L : L \in \mathscr{L}^v \text{ separates } v \text{ and } t_2\}\\
&\quad \leqslant d(t_1, v) - m + d(v, t_2) - m = d^v(t_1, v) + d^v(v, t_2).
\end{aligned}$$

On the other hand,

$$d^v(t_1, v) + d^v(v, t_2) \leqslant w^v(Q_{t_1, v}) + w^v(Q_{v, t_2}) = w^v(Q) = d^v(t_1, t_2).$$

The claim, and thus the theorem is proved. ∎

Note that the algorithm generated by the above proof is doubly exponential in k. *A more practical algorithm would be still more interesting.*

Remark. In an earlier version of this note (Bonn Report 88534-OR) I stated a result about reducing the maximization of T-cut packings in general, to the maximization of odd cut packings in matching problems, which is now only implicit in the second half of the above proof. I have omitted them here, since the interested reader can find now these more general relationships in the literature:

1. Recently Alexandr Karzanov kindly pointed out to me that [11, Theorem 5.3] establishes a correspondence between some more general families of cuts and their restrictions to subsets.

2. An improved presentation and simpler proofs of this omitted result of my Bonn report can be found now in András Frank's survey paper [9]. Note, however, that for applications to the complexity of the general Chinese postman problem (without fixing $|T|$), our argument is not really useful; the same complexity bounds follow in an almost trivial way, as it was noticed after the publication of his paper by A. Frank.

3. Planar Multiflows

We explain first the relation between flows and packings of T-cuts establised by Seymour [23]. The details are described for the sake of completeness, and also because we are mainly interested in the connection between slightly different problems, the *undirected plane k-commodity integral flow problem* and the following problem:

Suppose $k \in \mathbb{N}$.

Seymour Property Test for k.

Instance: A graph G, $T \subseteq V(G)$, $|T|$ even, $|T| \leqslant 2k$, $w: E(G) \to \mathbb{N}$.

Question: Does (G, T, w) have the Seymour property?

We show first that the planar special case of this problem is equivalent to *undirected plane k-commodity integral flow*.

If G is a planar graph, we shall always suppose that it is actually embedded in the plane. The dual graph (with respect to the given embedding) will be denoted by G^*, and the dual of an edge $e \in E(G)$, by e^*. (The embedding will not have any significance; we shall speak only about cuts and circuits.) If $F \subseteq E(G)$, then $F^* := \{e^* : e \in F\}$. Clearly, $(F^*)^* = F$, and C is a circuit of G if and only if C^* is a cut of G^* and vice versa.

Let (G, R, r, c) be a network, where G is a planar graph, and define

$$w: E(G^*) \to \mathbb{N}, \qquad w(e^*) := \begin{cases} r(e) & \text{if } e \in R \\ c(e) & \text{if } e \in E(G) \setminus R, \end{cases}$$

and $T := \{x \in V(G^*): d_{R^*}(x) \text{ is odd}\}$.

Proposition. *With the above notation*

(a) *There exists a flow in* (G, R, r, c) *if and only if* $r(R) = \tau(G^*, T, w) = \nu(G^*, T, w)$.

(b) *The problem of finding a flow in* (G, R, r, c) *is polynomially equivalent to finding a* $\tau(G^*, T, w)$ *element packing of T-cuts.*

(c) *If there exists a polynomial algorithm that solves the* Seymour property test for k, *then there exists one that solves the* undirected plane k-commodity integral flow.

Proof. (a) If there exists a flow $(\mathscr{C}, f)$, then each element of $\mathscr{C}^* = \{C^* : C \in \mathscr{C}\}$ contains exactly one edge of R^*. Thus the elements of $\mathscr{C}^*$ are T-cuts. $(\mathscr{C}^*, f^*)$ is a w-packing of T-cuts $(f^*(C^*) := f(C))$ of value $r(R)$, and R^* is a T-join of weight $w(R^*) = r(R)$. Hence, only the if part of (a) is proved. Conversely, if $r(R) = \tau(G^*, T, w) = \nu(G^*, T, w)$, then obviously, any cut in a w-maximum packing of T-cuts $(\mathscr{C}^*, f^*)$ intersects every mimimum T-join, in particular R^*, in exactly one element, and covers all the elements of R^* (complementary slackness). This means exactly that $(\mathscr{C}, f)$ is a flow, and the proof of (a) is complete.

(b) is an immediate consequence of the above proof of (a), and (c) is also a consequence of (a), with the additional remark that $|R| \leqslant k \Rightarrow |T| \leqslant 2k$.

We have arrived at our main result:

"Seymour Property test for k," and "undirected plane k-commodity integer flow" can be solved in polynomial time. (2)

Proof. The polynomial solvability of the former problem is an immediate consequence of (1), because $\tau(G, T, w)$ is the minimum weigth of a matching of T with weights $d(t_1, t_2)$ (well known, see Edmonds and Johnson [5]).

The polynomial solvability of the latter problem is an immediate consequence of the first, and of (c) of the above proposition. ∎

Let us emphasize again that the worst-case performance of our algorithms is doubly exponential in k. For $k = 5$, say, we just have a linear integer program with 45 constraints and 256 variables and a very particular structure. It could be an intriguing problem to look more carefully into this integer program to obtain a better bound and an algorithm that is applicable in practice which does not use Lenstra's general integer programming algorithm.

Acknowledgments

I am indebted to Bill Cook, András Frank, and Angelika Steger for useful discussions on the topic of the paper, and to Francisco Barahona and Lex Schrijver for the helpful reorganization of this note and other comments.

References

1. F. Barahona, Planar multicommodity flows, max cut, and the Chinese postman problem, *in* "Polyhedral Combinatorics, 1990, DIMACS 1, AMS, ACM" (W. Cook and P. D. Seymour, Eds.), pp. 189–202.
2. F. Barahona and W. Cunningham, On dual integrality in matching problems, *Oper. Res. Lett.* **8** (1989), 245–249.
3. W. H. Cunningham and A. B. Marsh, A primal algorithm for optimal matchings, *in* "Polyhedral Combinatorics" (M. Balinski and A. Hoffmann, Eds.), Math. Prog. Study, Vol. 8, pp. 50–72, North-Holland, 1978.

4. J. EDMONDS, Paths, trees, and flowers, *Canad. J. Math.* **17** (1965), 88–124.
5. J. EDMONDS AND E. L. JOHNSON, Matching, Euler tours, and the Chinese postman, *Math. Programming* **5** (1987), 88–124.
6. S. EVEN, A. ITAI, AND A. SHAMIR, On the complexity of timetable and multicommodity flow problems, *SIAM J. Comput.* **5** (1976), 88–124.
7. A. FRANK, "Graph Connectivity and Network Flows," Technical Report, Bonn, 1987, 87473-OR; *in* R. GRAHAM, M. GRÖTSCHEL, AND L. LOVÁSZ, "Handbook of Combinatorics," to appear.
8. A. FRANK, Packing paths in planar graphs, *Combinatorica*, to appear.
9. A. FRANK, Packing paths, circuits, and cuts—A survey, *in* "Paths, Flows, and VLSI-Layout" (Korte, Lovász, Prömel, and Schrijver, Eds.), pp. 47–100, Springer-Verlag, New York/Berlin, 1990.
10. M. R. GAREY AND D. S. JOHNSON, "Computers and Intractability," Freeman, San Francisco, 1978.
11. A. KARZANOV, A generalized MFMC-property and multicommodity cut problems, *in* "Infinite and Finite Sets, Proceedings of a Conference held in Eger, North Holland, 1984" (A. Hajnal and V. T. Sós, Eds.), pp. 443–486.
12. E. KORACH "Packings of *T*-cuts, and Other Aspects of Dual Integrality," Ph.D. thesis, Waterloo University, 1982.
13. E. KORACH AND I. NEWMANN, private communication, 1986.
14. E. LAWLER, "Combinatorial Optimization, Networks, and Matroids," Holt, Rinehart & Winston, New York, 1976.
15. H. W. LENSTRA, JR., Integer programming with a fixed number of variables, *Math. Oper. Res.* **8** (1983), 538–548.
16. M. W. LOMONOSOV, On the planar integer two-flow problem, *Combinatorica* **3**, No. 2, 1983, 207–218.
17. L. LOVÁSZ, 2-matchings and 2-covers of hypergraphs, *Acta Math. Acad. Sci. Hungar.* **21** (1975), 433–444.
18. L. LOVÁSZ AND M. PLUMMER, "Matching Theory," Akadémiai Kiadó, Budapest, 1986.
19. M. MIDDENDORF AND F. PFEIFFER, On the complexity of the disjoint paths problem, *in* "Polyhedral Combinatorics, 1990, DIMACS 1, AMS, ACM" (W. Cook and P. Seymour, Eds.), pp. 171–178.
20. N. ROBERTSON AND P. SEYMOUR, An outline of a disjoint paths algorithm, *in* "Paths, Flows, and VLSI-Layout" (Korte, Lovász, Prömel, and Schrijver, Eds.), pp. 47–100, Springer-Verlag, New York/Berlin, 1990.
21. A. SCHRIJVER, "Theory of Linear and Integer Programming," Wiley, Chichester, 1986.
22. A. SEBŐ, Dual Integrality and multicommodity flows, *in* "Infinite and Finite Sets, Proceeding of a Conference held in Eger, North Holland, July 1987" (A. Hajnal and V. T. Sós, Eds.), pp. 453–469.
23. P. D. SEYMOUR, On odd cuts and plane multicommodity flows, *Proc. London Math. Soc.* (3) **42** (1981), 178–192.