On Multicommodity Flows in Planar Graphs

Refael Hassin

Department of Statistics, Tel-Aviv University, Tel-Aviv 69978, Israel

Okamura and Seymour recently proved two properties of multicommodity flows in undirected planar networks where all the sources and the sinks are on a common face of the underlying graph. One is that a feasible solution is guaranteed whenever each cut's capacity is at least as large as the cut's demand. The second is that if all demands and capacities are integers then the flow values may be chosen half-integer-valued. In this paper we use the first property to construct two computational procedures; one examines the existence of a feasible flow, and the other constructs such a flow if one exists. We also show that the construction procedure can be used as an alternative proof to the above properties. Finally we show, by presenting counterexamples, that the half-integrality property does not necessarily hold when either the graph cannot be drawn in the plane with all sources and sinks on a common face, or the graph is directed.

I. INTRODUCTION

Okamura and Seymour [7] recently proved two properties of multicommodity flows in undirected planar graphs where all the sources and the sinks are on a common face. One is that a feasible solution is guaranteed whenever each cut's capacity is at least as large as the cut's demand. The second is that, if all demands and capacities are integers, the flow values may be chosen half-integer-valued. Similar results were discovered long ago for two-commodity flows in general graphs [2, 9, 11] and used to develop an algorithm that constructs two-commodity flows [2]. In this paper we use the results of Okamura and Seymour to develop two computational procedures concerning networks of the type considered in their paper. One procedure checks the existence of a feasible solution in $O(n^2 \log n)$ time, the other constructs such a solution in $O(n^4)$ time where n is the number of vertices in the graph. Then we show how the construction procedure supplies an alternative proof to the feasibility theorem. Finally, we show that the half-integrality result cannot be extended to the cases where either the graph cannot be drawn in the plane with all sources and sinks on a common face or the graph is directed.

Several other authors used feasibility and integrality theorems to develop network flow algorithms. For example, Rothfarb and Frisch [8] presented a theorem and an algorithm for three-commodity graphs with no internal vertices, and Sakarovitch [10] treated "completely planar" graphs (see also Kennington [4]). Further discussion on this subject can be found in the survey of multicommodity network flows [4].

NETWORKS, Vol. 14 (1984) 225–235

CCC 0028-3045/84/020225-11$04.00

II. MATHEMATICAL FORMULATION OF THE PROBLEM

Let G be a finite directed graph without loops, and let $(s_1, t_1), \ldots, (s_K, t_K)$ be pairs of vertices of G. Suppose that each edge $e \in E$ has a real-valued capacity $w(e) \geqslant 0$ and that q_k, $k = 1, \ldots, K$ are real-valued demands. Suppose also that G is planar and can be drawn in the plane so that $s_1, \ldots, s_K, t_1, \ldots, t_K$ are on a common face of G. Without loss of generality, we assume that this face is the exterior face of G, and denote the set of edges and vertices of this face as the *boundary* of G. For a set $X \subseteq V$, let $\partial(X) \subset E$ be the set of edges with one end in X and the other in V–X, let $D(X) = \{k | 1 \leqslant k \leqslant K, |\{s_k, t_k\} \cap X| = 1\}$, and let $\Delta(X) = \Sigma_{e \in \partial(X)} w(e) - \Sigma_{k \in D(X)} q_k$. The following theorem was proved in [7]:

Theorem 1. Statements (1) and (2) are equivalent:

(1) For $k = 1, \ldots, K$ there is a flow F_k from s_k to t_k of value q_k, such that, for each edge $e \in E$, $\Sigma_k |F_k(e)| \leqslant w(e)$.
(2) For every $X \subseteq V$, $\Delta(X) \geqslant 0$.

Moreover, if $q_1, \ldots, q_K$ and $w(e)$, $e \in E$, are integers then the flow values $F_k(e)$ may be chosen half-integer-valued.

Examination of (ii) is practically impossible since $2^{|V|}$ sets must be considered. Lemma 2.1 of [7] reduces the number of sets by stating that (3) is also equivalent to (1):

(3) Assume (without loss of generality) that G is connected. For each $X \subseteq V$ such that the subgraphs $G|X$ and $G|V - X$ are both connected, $\Delta(X) \geqslant 0$. ($G|X$ denotes the result of deleting all vertices in $V - X$ and their incident edges.)

However, the computational work involved in checking this condition is still prohibitive. In this paper we describe a procedure that efficiently examines (1) by checking only $0(|V|^2)$ sets. Then we propose a polynomial-time algorithm that constructs the flow function, if one exists. We note that the proof of Theorem 1 in [7] is constructive in the sense that it includes a procedure that constructs the flow. In each iteration of this procedure one unit of flow is sent through one edge and two new commodities are created. Our method uses this idea but in a more efficient way.

III. FEASIBILITY EXAMINATION

Suppose that G is drawn in the plane so that $s_1, \ldots, s_K, t_1, \ldots, t_K$ are on its boundary. Denote by $B \subseteq E$ and $V_B \subseteq V$ the sets of edges and vertices of this boundary, respectively.

Figure 1 describes a ***multiple-source dual graph*** G^D of a graph G. It is defined similarly to a regular dual graph (see [1, 3, 6]) with the exception that a distinct source vertex is defined in the exterior face for each edge in B. The set of sources of G^D will be denoted by S^D. In Figure 1, G and G^D are described by the solid and broken lines, respectively.

Consider a pair of sources $i, j \in S^D$. There are many different paths connecting i and j on G^D, and each of them partitions V into different subsets $(X, V - X)$. However,

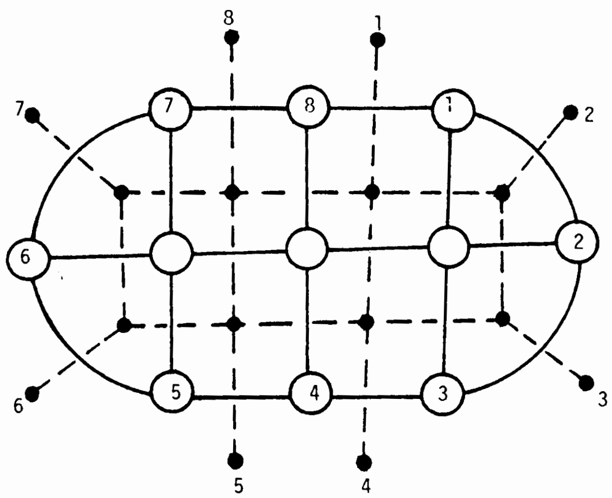

FIG. 1.

the value of $\Sigma_{k \in D(X)} q_K$ is identical for all the sets X formed this way. Therefore, to find the minimum value of $\Delta(X)$ among all these sets, it suffices to find the minimum value of $\Sigma_{e \in \partial(X)} w(e)$ among them. This is simply the length of the shortest i-j path on G^D (as described in [3], the length of an edge in G^D equals the capacity of the corresponding edge in G). Denote the minimal value of $\Delta(X)$ among the above sets by Δ_{ij}. Noting that every set in (3) defines an i-j path on G^D for some $i, j \in S^D$, we have established the equivalence of (1) to the following statement:

(4) For every $i, j \in S^D$, $\Delta_{ij} \geqslant 0$.

Condition (4) can be checked as follows: Let the vertices of V_B and S^D be indexed in clockwise direction (mod $|B|$) as in Figure 1. For every u and v in V_B let q_{uv} be equal to the demand q_k of the commodity k with $\{s_k, t_k\} = \{u, v\}$ if such a commodity exists (without loss of generality we assume that if such k exists then it is unique). Otherwise, let $q_{uv} = 0$. Let $A_{uv} = \Sigma_{r=u+1}^{v-1} q_{ur} - \Sigma_{r=v}^{u-1} q_{ur}$, then $A_{u,v+1} = A_{uv} + 2q_{uv}$ so that A_{uv} can be computed in $O(|B|^2)$ time for all u and v in V_B. For every i and j in S^D let $Q_{ij} = \Sigma_{u=i}^{j-1} \Sigma_{v=j}^{i-1} q_{uv}$, then $Q_{i+1,j} = Q_{ij} + A_{ij}$ so that Q_{ij} can be computed in $O(|B|^2)$ time for all $i, j \in S^D$. The lengths d_{ij} of the shortest i-j paths can be computed for all i-j pairs, by applying Dijkstra's algorithm $|B|$ times, in $O(|B||V| \log |V|)$ time. Finally, the values of Δ_{ij} can be computed by subtracting Q_{ij} from d_{ij}. Therefore the overall computational effort needed is of order $O(|B||V| \log |V|)$. If $K < |B|$ this bound can be replaced by $O(K|V| \log |V|)$, since an equivalent graph whose set of boundary vertices is $\{s_1, \ldots, s_K, t_1, \ldots, t_K\}$ can be formed by suitable additional of edges with zero capacity.

IV. CONSTRUCTION OF A FEASIBLE MULTICOMMODITY FLOW

In this section we describe an algorithm that constructs a feasible flow, if one exists. Before presenting the algorithm, we outline the main ideas underlying it and prove two lemmas that are necessary to compute flow values while preserving the feasibility conditions.

In each iteration of the algorithm a commodity $k \in \{1, \cdots, K\}$ and a boundary edge $e \in B$ are selected. Then δ units of k-flow are sent along e. To attain feasibility it is necessary that δ units of k-flow be sent from s_k to one end of e and from the other end of e to t_k. Suppose $e = \{a, b\}$ and the order of the nodes on the boundary is s_k, a, b, t_k. Lemma 1 claims that we can restrict ourselves to solutions in which these δ units will be sent from s_k to a and from b to t_k. Therefore the following transformation is made: $w(e)$ and q_k are reduced by δ, and two ***artificial commodities*** with $q = \delta$ are introduced, one with its source at s_k and its sink at a, the other has its source at b and its sink at t_k. The flows of these commodities will be considered at the end of the computations as k-flows.

Lemma 1: Let $e \in B$ have ends a and b, so that the order on B is s_k, a, b, t_k. Then a multicommodity flow exists if and only if one exists with no k-flow directed from b to a.

Proof: Consider a solution with k-flow directed from b to a; then there must exist a cycle of k-flow such as the cycle *c-b-a* shown in Figure 2. A feasible flow with no k-flow directed from b to a can be obtained by repeatedly reducing $F_k(\bar{e})$ by $\min_{t \in C} F_k(t)$ for every edge $\bar{e}$ of such a cycle C. ■

After all commodities and boundary edges are chosen, the boundary edges are deleted from the graph. The resulting graph is also planar and all source and demand vertices are on its boundary. To simplify the computations, boundary edges that become surrounded by the infinite region are deleted after flow is determined in them in the obvious way. This is done in Step 1 of the algorithm.

The main part of the algorithm is devoted to the computation of the maximum k-flow that can be sent through $e \in B$ without violating the feasibility requirements. The next lemma shows how to do it efficiently. Consider the set H of all subsets $X \subseteq V$ such that the subgraphs induced by X and $V - X$ are connected. For $e = \{a, b\} \in B$ and $k \in \{1, \ldots, K\}$ define $H(e, k)$ to be the set of all $X \subseteq H$ such that $X \cap \{a, b\} \neq \phi$ and $X \cap \{s_k, t_k\} = \phi$.

Lemma 2: Consider the following transformation: $w(e) \leftarrow w(e) - \delta$, $q_k \leftarrow q_k - \delta$, $q_{K+1} \leftarrow \delta$, $q_{K+2} \leftarrow \delta$ where $K + 1$ and $K + 2$ are new commodities with $s_{K+1} = s_k$. $t_{K+1} = a$, $s_{K+2} = b$, $t_{K+2} = t_k$. Then $\Delta(X)$ decreases by 2δ for $X \subseteq H(e, k)$ and remains unchanged for $X \subseteq H - H(e, k)$.

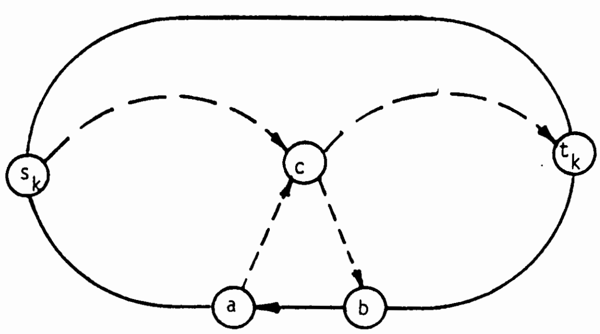

FIG. 2.

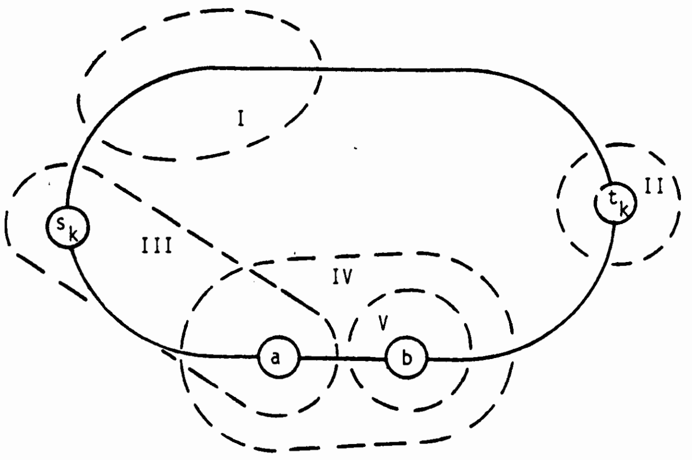

FIG. 3.

Proof. We consider five types of sets $X \subseteq H$ as illustrated in Figure 3 (all the other possible cases are equivalent to one of these five, as far as our lemma is concerned):

(I) Suppose that a, b, s_k, and t_k are all in $V - X$, then no q_k, $k \in D(X)$ or $w(e)$, $e \in \partial(X)$, is changed and so $\Delta(X)$ is unchanged.

(II) Suppose that X separates t_k from s_k, a, and b. In this case $k \in D(X)$ and $(K + 2) \in D(X)$ so that δ is subtracted and added to $\Sigma_{i \in D(X)} q_i$. Since also $e \notin \partial(X)$, $\Delta(X)$ is unchanged.

(III) Suppose that X separates t_k and b from s_k and a. Here $e \in \partial(X)$ and $\Sigma_{e \in \partial(X)} w(e)$ decreases by δ. However, also $\Sigma_{i \in D(X)} q_i$ decreases by δ since $\{k, K + 1, K + 2\} \subseteq D(X)$, so that $\Delta(X)$ is unchanged.

(IV) Suppose that X separates a and b from s_k and t_k, then $e \notin \partial(X)$, $k \notin D(X)$, $K + 1 \in D(X)$, and $K + 2 \in D(X)$ so that $\Delta(X)$ decreases by 2δ.

(V) Suppose that X separates b from a, s_k, and t_k, then $e \in \partial(X)$, $k \notin D(X)$, $K + 1 \notin D(X)$, and $K + 2 \in D(X)$. Hence both $\Sigma_{e \in \partial(X)} w(e)$ and $\Sigma_{i \in D(X)} q_i$ decrease by δ, so that $\Delta(X)$ decreases by 2δ.

Since the sets $X \in H(e, k)$ are of the types described in IV and V, the lemma is proved. ∎

Suppose that (4) holds, so that a feasible multicommodity flow exists. Denote the maximum value of k-flow which can be sent through $e \in B$ by $\delta(e, k)$. Then $\delta(e, k) = \min \{w(e), q_k, \{\min \frac{1}{2} \Delta(X) \mid X \in H(e, k)\}\}$. As a matter of fact, from the discussion in Section III it suffices to consider only the corresponding values of Δ_{ij}. Suppose that G is drawn as in Figure 3, and consider the vertices in S^D which correspond to the edges in B on the lower path connecting s_k and t_k. Let s_e be the vertex corresponding to e and let S_L (S_R) be the set of vertices which correspond to edges between s_k and a (b and t_k). Define $\bar{H}(e, k) = \{(i, j) \mid i, j \in S^D \text{ s.t. } i \in S_L \text{ and } j \in S_R, \text{ or } i = s_e \text{ and } j \in S_L \cap S_R\}$. Then

$$(5)\quad \delta(e, k) = \min \{w(e), q_k, \min \{\tfrac{1}{2} \Delta_{ij} \mid (i, j) \in \bar{H}(e, k)\}\}.$$

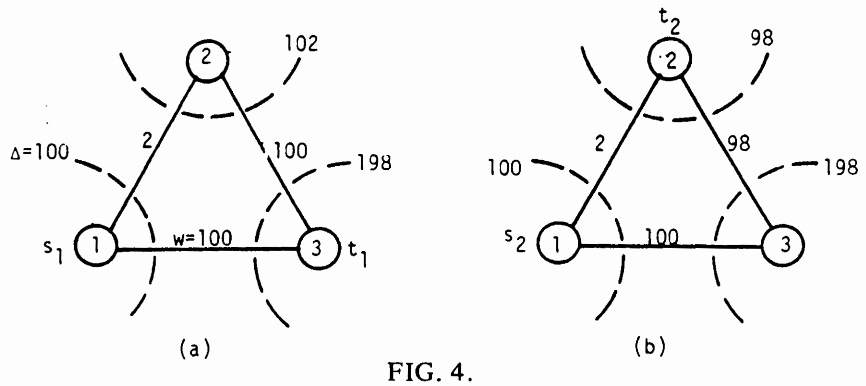

FIG. 4.

Consider the graph of Figure 4(a) where $q_1 = 2$. If edge $e = \{2, 3\}$ is chosen first then $\delta = 2$ and after the transformation we obtain Figure 4(b) with $q_2 = 2$. For the same edge and $k = 2$ we obtain again $\delta = 2$, which means that this quantity is reshipped to node 3, and the process may be repeated until some $\Delta(X)$ such that $e \in \partial(X)$ becomes zero. To prevent repeated shipments of a commodity in opposite directions, Step 2 of the algorithm reduces each $w(e)$ $e \in B$ as much as possible, without violating the feasibility requirements.

Multicommodity Flow Algorithm

Step 0. (Initialization)
For $k = 1, \ldots, K$:
Set $F_k(a, b) \leftarrow 0$ for every $a, b \in V$,
$q_{ijk} \leftarrow q_k$ if $i = s_k$ and $j = t_k$, and $q_{ijk} \leftarrow 0$ otherwise.
(The variable q_{ijk} denotes the amount of k-flow which must be sent from i to j.)
Step 1. (Deletion of edges surrounded by the exterior face)
(i) If $E = \phi$, stop (F is a feasible flow),
Set $B \leftarrow$ set of boundary edges of (E, V),
$V_B \leftarrow$ set of boundary vertices of (E, V).
(ii) Find $e = \{a, b\} \in B$ such that no other edge of B has one of its ends in a. If no such edge exists, go to 2.
Set $B \leftarrow B - e$, $E \leftarrow E - e$.
(iii) For every $v \in V_B$ and $k = 1, \ldots, K$:
Set $q_{bvk} \leftarrow q_{bvk} + q_{avk}$, $F_k(a, b) \leftarrow F_k(a, b) + q_{avk}$, $q_{vbk} \leftarrow q_{vbk} + q_{vak}$,
$F_k(b, a) \leftarrow F_k(b, a) + q_{vak}$, $w(e) \leftarrow w(e) - q_{avk} - q_{vak}$.
(iv) If $w(e) < 0$, stop (no feasible solution exists).
If $B = \phi$, go to (*i*). Else, go to (*ii*).
Step 2. (Capacity reduction in boundary edges)
(i) Set $G^D \leftarrow$ dual multisource graph of (E, V), $S^D \leftarrow$ dual source nodes, $E \leftarrow E - B$.
(ii) For every $n \in S^D$:
Compute Δ_{nj} for all $j \in S^D - n$.
Set $e \leftarrow$ edge in B that corresponds to n, $w(e) \leftarrow w(e) - \min_{j \in S^D} \Delta_{nj}$.
If $w(e) \leqslant 0$, set $B \leftarrow B - e$ and $w(e) \leftarrow 0$.
Set $\Delta_{ni} \leftarrow \Delta_{ni} - \min\{\min_{j \in S^D} \Delta_{nj}, w(e)\}$ for every $i \in S^D - n$.
(Note that $\Delta_{nj} = \Delta_{jn}$ and it is understood that only one value is stored and modified.)

Step 3. (Flow construction in boundary edges)
If $B = \phi$, go to 1.
Set $P \leftarrow \{(u, v) | u \neq v, u \in V_B, v \in V_B$, and $q_{uvk} > 0$ for some $k\}$.
(i) If $P = \phi$, go to 1.
Choose $(u, v) \in P$.
Set $P \leftarrow P - (u, v)$, $Q \leftarrow \{k | q_{uvk} > 0\}$, and $B' \leftarrow B$.
(ii) If $B' = \phi$, go to (i).
Choose $e \in B'$ and set $B' \leftarrow B' - e$.
Denote the ends of e by (a, b) so that the order on the boundary is (u, v, a, b).
Set $\bar{H} \leftarrow \bar{H}(e, r)$, where r is assumed to have $s_r = u$ and $t_r = v$.
Set $\delta \leftarrow \min \{w(e), \min \{\frac{1}{2} \Delta_{ij} (i, j) \in H\}\}$.
If $\delta = 0$, go to (ii).
Set $\Delta_{ij} \leftarrow \Delta_{ij} - 2 \min \{\delta, \Sigma_{k=1}^{K} q_{uvk}\}$ for every $(i, j) \in \bar{H}$.
[The quantity $\min \{\delta, \Sigma q_{uvk}\}$ is the total flow to be sent through e during the following executions of Step 3 (iii). Every $\Delta_{ij} (i, j) \in \bar{H}$ will be reduced by twice this quantity, as shown by Lemma 2.]
(iii) If $Q = \phi$, go to (i).
Choose $k \in Q$ and set $\delta' \leftarrow \min \{q_{uvk}, \delta\}$.
Set $q_{vak} \leftarrow q_{vak} + \delta'$, $q_{buk} \leftarrow q_{vuk} + \delta'$, $q_{uvk} \leftarrow q_{uvk} - \delta'$, $w(e) \leftarrow w(e) - \delta'$, $F_k(a, b) \leftarrow F_k(a, b) + \delta'$, and $\delta \leftarrow \delta - \delta'$.
If $q_{uvk} = 0$, set $Q \leftarrow Q - k$.
If $\delta = 0$, go to (ii). Else, go to (iii).

Theorem 2. If $q_1, \ldots, q_k$ and $w(e) \in E$ are all even integers, then the algorithm produces an integral solution.

Proof. Clearly all $\Delta(X)$ $X \subseteq V$ are initially even. We must show that δ is always integral, or equivalently that all $\Delta(X)$ remain even and all q and w remain integral.

The changes in q and w in Step 1 do not effect Δ and change some q values by integral amounts.

In Step 2, both Δ and w values are changed by $\min_{j \in S^D} \Delta_{nj}$, which is by our assumption even.

In Step 3, w and q values are changed by δ and Δ values are decreased by 2δ. Therefore, if all q's and w's are integral and all Δ's even, δ is integral and the above properties of the q's, w's, and Δ's are preserved. ∎

Validity of the algorithm results from Theorem 1 through the computation of $\delta(e, k)$ by (5). An edge $e \in B$ is removed after either its capacity becomes zero or $\delta(e, k) = 0$ for $k = 1, \ldots, K$, since in both cases no further flow augmentation is possible in this edge. However, the algorithm can be used to give an alternative constructive *proof of Theorem 1:*

As in [7] we use induction $|E|$, noting that the theorem holds for $|E| = 0$. It suffices to prove, therefore, that if, for some $e \in B$, $\delta(e, k) = 0$, $k = 1, \ldots, K$, then e can be removed from E without violating (4). Let $n \in S^D$ correspond to e. Setting $w(e) \leftarrow [w(e) - \min_{j \in S^D} \Delta_{nj}]^+$ does not violate (4). We claim that now $w(e) = 0$ so that e can be removed from E. To prove this claim we show now that the following assumptions lead to a contradiction:

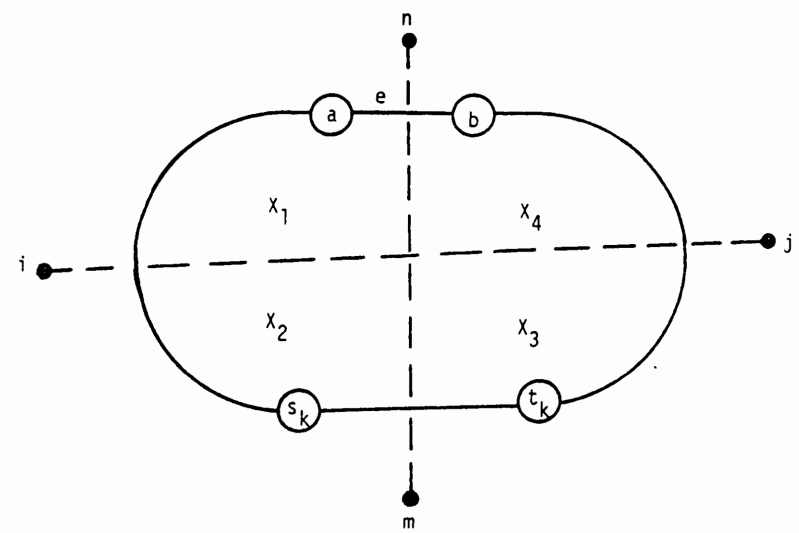

FIG. 5.

$$\begin{aligned} \Delta_{ij} &\geq 0 && \text{for every } i, j \in S^D, \\ \Delta_{nj} &= 0 && \text{for some } j \in S^D - n, \\ \delta(e, k) &= 0 && \text{for } k = 1, \ldots, K, \\ w(e) &> 0. \end{aligned}$$

Let us choose vertex m from among the vertices of S^D for which $\Delta_{nm} = 0$ to be the one that is closest (from the left) to n. Since $w(e) > 0$ and $\Delta_{nm} = 0$ there exists at least one commodity k with positive demand, such that its source and sink are on different sides of every path connecting n and m. Choose the one with the closest (from the right) source or sink to b. Since $\delta(e, k) = 0$, we conclude from (5) that there exists $(i, j) \in H(e, k)$ with $\Delta_{ij} = 0$. By our choice of m, $\Delta_{in} > 0$. The shortest i-j and n-m paths on G^D partition V into four subsets X_1, X_2, X_3, and X_4 as illustrated in Figure 5. Note that X_4 is not empty since $\Delta_{in} > 0$ and $\Delta_{ij} = 0$ so that $n \neq j$.

Let Q_{uv} equal the sum of q_r over all commodities r with $s_r \in X_u$ and $t_r \in X_v$ or $s_r \in X_v$ and $t_r \in X_u$. By our choice of k, $Q_{14} = Q_{24} = 0$. Since $\Delta_{nm} = 0$ then $Q_{13} + Q_{23} = \Sigma_{t \in \partial(X_1 \cup X_2)} w(t)$, and since $\Delta_{ij} = 0$ then $Q_{12} + Q_{13} + Q_{34} = \Sigma_{t \in \partial(X_1 \cup X_4)} w(t)$. On the other hand, $\Delta_{in} > 0$ implies $Q_{12} + Q_{13} < \Sigma_{t \in \partial(X_1)} w(t)$ and $\Delta_{jm} \geq 0$ implies $Q_{13} + Q_{23} + Q_{34} \leq \Sigma_{t \in \partial(x_3)} w(t)$. Since

$$\sum_{t \in \partial(X_1 \cup X_2)} w(t) + \sum_{t \in \partial(X_1 \cup X_4)} w(t) \geq \sum_{t \in \partial(X_1)} w(t) + \sum_{t \in \partial(X_3)} w(t)$$

a contradiction obtains.

V. COMPLEXITY ANALYSIS

Denote by $B_1, B_2, \ldots,$ the sets B formed in Step 1(i). Clearly $\Sigma |B_i| \leq |E|$.

Step 1(iii) is executed for every $v \in V_B$ and $k = 1, \ldots, K$ whenever an edge surrounded by the infinite region is located. Thus the complexity of this step is $O(K|B_i|)$ for each of these edges and altogether at most $O(|E|^4) = O(|V|^4)$.

Step 2(ii) is executed for every $n \in S^D$, i.e., $\Sigma |B_i|$ times. Each execution requires computation of the shortest path from a given $n \in S^D$ to all other vertices in S^D, which takes $O(|V| \log |V|)$. Hence the overall complexity is $O(|V|^2 \log |V|)$.

Step 3(ii) computes δ for every $e \in B_i$. Computation of δ requires $O(|B_i|)$ operations, so that for u and v fixed this step requires $O(|B_i|^2)$. Each execution of Step 3(iii) decreases either q_{uvk} for some k or δ for some $e \in B_i$ to zero. Therefore, for u and v fixed this step requires $O(\max\{K, |B_i|\})$. The overall time consumed by Step 3 is therefore $O(\max\{K\, \Sigma |B_i|^2, \Sigma |B_i|^4\}) = O(|V|^4)$.

We conclude from the above discussion that the complexity of the algorithm is $O(|V|^4)$.

VI. COUNTEREXAMPLES FOR POSSIBLE EXTENSIONS

In [7] it has been shown, by a counterexample, that Theorem 1 cannot be extended to general planar graphs. In this section we show, by presenting a counterexample, that also the integrality theorem for input consisting of even integers cannot be ex-

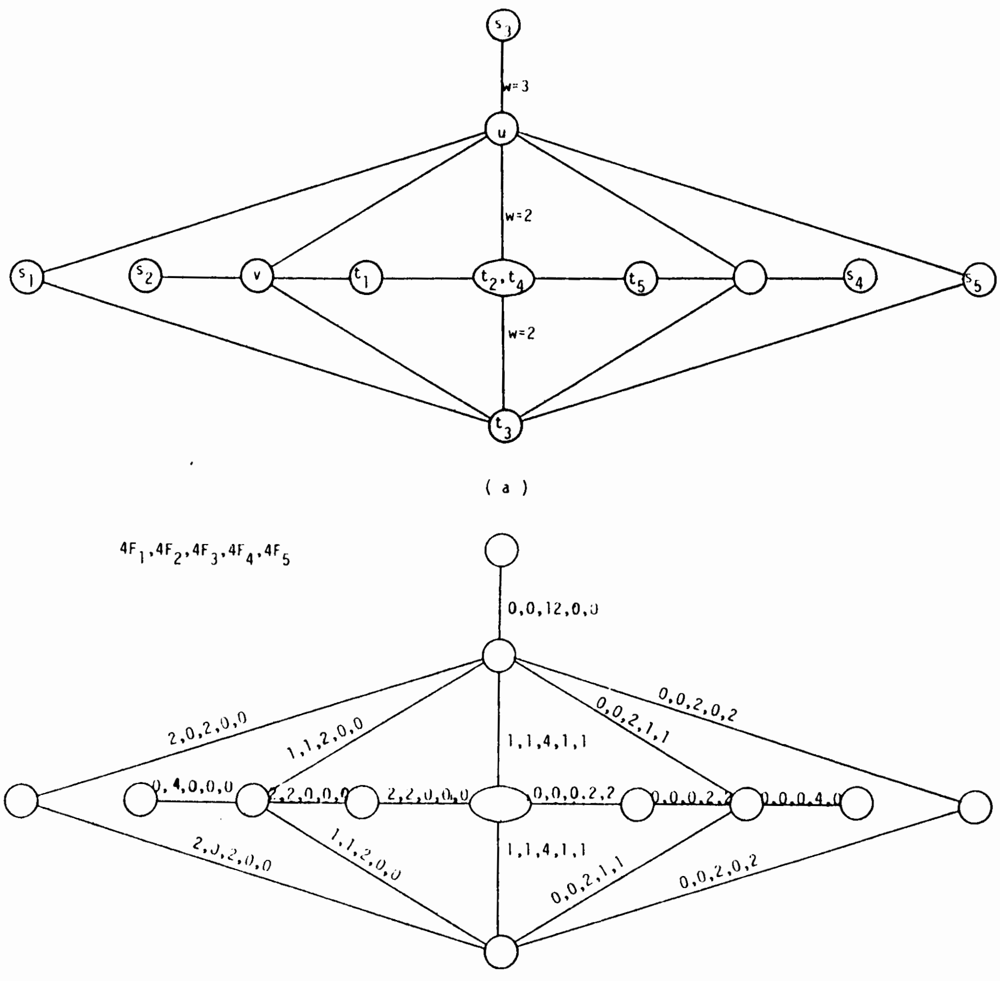

FIG. 6.

tended to general planar graphs. By a second counterexample we show that this integrality result does not apply to directed networks even when the graph is planar and all of the pairs (s_i, t_i) are on a common face of G.

Example 1. The graph of Figure 6(a) is planar but cannot be drawn on the plane with s_i and t_i, $i = 1, 2, 3$ on a common face of the graph. It was inspired by an example of Hu [3] analyzed in [8]. We assume that $q_3 = 3$ and all edge capacities that are not explicitly shown, and all demands except for q_3, are of one unit. The sum of edge capacities is 21 and each s_i–t_i path includes at least 3 edges. Since the total demand is 7, if a feasible solution exists it uses only 3-edge paths and fully exploits each capacity. Therefore the unit demand of 1-flow must split into two halves before reaching t_1 and the other half-unit of capacity in the edges incident with t_1 must be filled by 2-flow. For this reason the other half-unit of 2-flow must split into quarters so that the edges $\{u, v\}$ and $\{v, t_3\}$ can be filled by 3-flow. Therefore a solution with no fractions other than halves does not exist. A feasible solution exists, however, as shown in Figure 6(b).

Example 2. The graph of Figure 7 is directed, planar, and drawn with s_1, t_1, s_2, and t_2 on the boundary of its infinite region. It was inspired by an example of Jewell [5]. We assume that $q_1 = 3$, $q_2 = 1$, and each edge has one unit of capacity. Suppose that $1 \geq r \geq 0$ units of 2-flow are shipped from s_2 to the right and $1 - r$ to the left. Considering only simple paths, the 2-flow solution is uniquely determined by r. It follows that no more than $3(1 - r)$ units of 1-flow can be shipped from s_1 to the right and no more than $3r$ to the left. However, it is clear that one unit of 1-flow must be sent

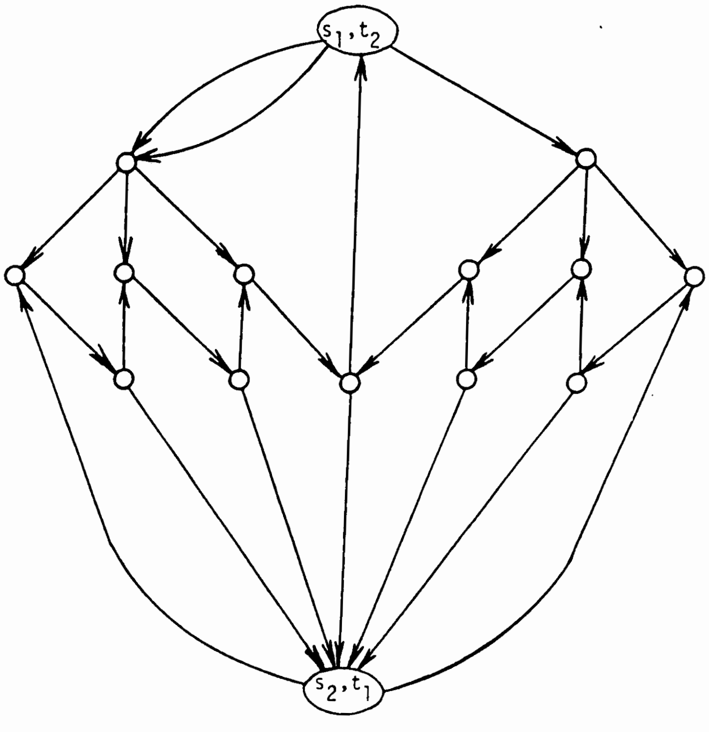

FIG. 7.

from s_1 to the right and two units to the left. Thus $2 \leqslant 3r$ and $1 \leqslant 3(1 - r)$ in every feasible solution, implying that a unique feasible solution exists, with $r = \frac{2}{3}$.

References

[1] R. Hassin, Maximum Flow in (s, t) Planar Networks. *Information Processing Lett.* 13 (1981) 107.

[2] T. C. Hu, On the feasibility of simultaneous flows in networks. *Operations Res.* 11 (1963) 344-360.

[3] T. C. Hu, *Integer Programming and Network Flows.* Reading, Addison-Wesley, Reading, MA (1969).

[4] J. L. Kennington, A Survey of linear cost multicommodity network flows. *Operations Res.* 26 (1978) 209–236.

[5] W. S. Jewell, Multi-commodity network solutions. ORC Report 66-23, Operations Research Center, University of California, Berkeley (1966).

[6] E. L. Lawler, *Combinatorial Optimization: Networks and Matroids.* Holt, Rinehart and Winston, New York (1976).

[7] H. Okamura and P. D. Seymour, Multicommodity Flows in planar graphs. *J. Combinatorial Theory Ser. B* 31 (1981) 75–81.

[8] B. Rothfarb and I. T. Frisch, On the 3-commodity flow problem. *SIAM J. Appl. Math.* 17 (1969) 46–58.

[9] B. Rothschild and A. Whinston, Feasibility of two commodity network flows. *Operations Res.* 14 (1966) 1121–1129.

[10] M. Sakarovitch, The multi-commodity maximal flow problem. ORC Report 66-25, Operations Research Center, University of California, Berkeley (1966).

[11] M. Sakarovitch, Two commodity network-flows and linear programming. *Math. Programming* 4 (1973) 1–20.