Equivalence of Primal and Dual Simplex Algorithms for the Maximum Flow Problem

by
Ravindra K. Ahuja
James B. Orlin

WP #3884-96 March 1996

Equivalence of Primal and Dual Simplex Algorithms
for the
Maximum Flow Problem

Ravindra K. Ahuja
Dept. of Industrial & Management Engg.
Indian Institute of Technology
Kanpur 208 016, INDIA
e-mail: ahuja@iitk.ernet.in

James B. Orlin
Sloan School of Management
Massachusetts Institute of Technology
Cambridge, MA 02139, USA
e-mail: jorlin@mit.edu

Equivalence of the Primal and Dual Simplex Algorithms for the Maximum Flow Problem

Ravindra K. Ahuja[1] and James B. Orlin[2]

ABSTRACT

In this paper, we study the primal and dual simplex algorithms for the maximum flow problem. We show that *any* primal simplex algorithm for the maximum flow problem can be converted into a dual simplex algorithm that performs the same number of pivots and runs in the same time. T he converse result is also true though in a somewhat weaker form.

1 Dept. of Industrial & Management Engg., Indian Institute of Technology, Kanpur - 208 016, INDIA

2 Sloan School of Management, Massachusetts Institute of Technology, Cambridge, MA 02139, USA

1. INTRODUCTION

The maximum flow problem is the problem of determining the maximum amount of flow that can be sent from a source node s to a sink node t through a capacitated network. The maximum flow problem arises in a wide variety of situations and in several forms (see, for example, Ahuja, Magnanti and Orlin [1993]). The maximum flow problem is a special case of the linear programming problem. Consequently, the primal simplex algorithm and the dual simplex algorithm for linear programming can be adapted for this problem. Goldfarb and Hao [1990, 1991] developed the first polynomial-time primal simplex algorithms for the maximum flow problem. Their algorithms run in $O(n^2m)$ time. Goldberg, Grigoriadis and Tarjan [1991] showed how to implement some of these algorithms in O(nm log n) time using a variant of the dynamic trees data structure. Armstrong and Jim [1992], Goldfarb and Chen [1992], and Armstrong et al. [1994] have developed dual simplex algorithms for the maximum flow problem. These algorithms performs O(nm) pivots and run in $O(n^2m)$ time if implemented in a straightforward manner. The algorithm by Armstrong et al. [1994] runs in $O(n^3)$ time if implemented in an appropriately clever way as a preflow-push algorithm.

In this paper, we show that *any* primal simplex algorithm for the maximum flow problem can be converted into a dual simplex algorithm. The converse result is also true, though in a somewhat weaker form. This paper unifies some results from the literature on primal and dual simplex algorithms for the maximum flow problems, and gives an efficient mechanism to go from a primal simplex algorithm to a dual simplex algorithm and vice-versa. Applications of our results to the primal simplex algorithms by Goldfarb and Hao [1990, 1991] yield the dual simplex algorithms presented in the papers by Goldfarb and Chen [1992], and Armstrong and Jim [1992]. Further, when our results are applied to the algorithm of Goldberg, Grigoriadis and Tarjan [1991], it shows the existence of an O(nm log n) time dual simplex algorithm, which is currently the fastest dual simplex algorithm for most classes of maximum flow problem.

This paper is organized as follows. Section 2 presents some background material. Section 3 and 4, respectively, present brief descriptions of primal and dual simplex algorithms for the maximum flow problem. Section 5 establishes equivalences between the primal and dual simplex algorithms.

2. PRELIMINARIES

We consider a directed network G = (N, A) with node set N, arc set A, a specified *source* node s, and a specified *sink* node t. Let n = |N| and m = |A|. Each arc $(i, j) \in A$ has a nonnegative capacity u_{ij}. We assume that the network contains an arc (t, s) with capacity u_{ts} = M, where M is a strict upper bound on the maximum flow that can be sent from node s to node t. The maximum flow problem can be stated as the following linear program:

$$\text{Minimize} \quad -x_{ts} \tag{1a}$$

subject to

$$\sum_{\{j:\ (i,\ j)\in A\}} x_{ij} - \sum_{\{j:\ (j,\ i)\in A\}} x_{ji} = 0 \qquad \text{for all } i \in N, \tag{1b}$$

$$0 \le x_{ij} \le u_{ij} \qquad \text{for all } (i,\ j) \in A, \tag{1c}$$

We use standard network flows terminology as defined, for example, in Ahuja et al. [1993]. Terms such as paths, cycles, fundamental cycles, and cuts are consistent with that reference. We use the notation S \ Q to denote the set theoretic difference of S and Q, that is those elements of S that are not in Q.

Simplex algorithms for the maximum flow problem maintain a *basic solution* at each stage. A basic solution of the minimum cost flow problem is denoted by the triple (**B**, **L**, **U**), where **B**, **L**, and **U** partition the arc set A. The set **B** denotes the set of *basic arcs* (that is, the arcs of a spanning tree), and **L** and **U** denote, respectively, the sets of *nonbasic arcs* at their lower and upper bounds. We refer to the triple (**B**, **L**, **U**) as a *basis structure*. A basis structure (**B**, **L**, **U**) is called *primal feasible* if by setting x_{ij} = 0 for each (i, j) ∈ **L**, and by setting $x_{ij} = u_{ij}$ for each (i, j) ∈ **U**, the problem has a *primal* solution x satisfying (1b) and (1c).

A *dual* solution of the minimum cost flow problem is a vector π of *node potentials*. For a given dual solution π, we define the reduced cost of an arc (i, j) as $c^{\pi}_{ij} = c_{ij} - \pi(i) + \pi(j)$. Recall that in the case of the maximum flow problem, $c_{ts} = -1$, and $c_{ij} = 0$ for $(i,j) \neq (t,s)$. A basis structure (**B**, **L**, **U**) is called dual *feasible* if there exists a set of node potentials π satisfying the following *optimality conditions* :

$$c^{\pi}_{ij} = 0 \text{ for each arc } (i,\ j) \in \mathbf{B}, \tag{2a}$$
$$c^{\pi}_{ij} \ge 0 \text{ for each arc } (i,\ j) \in \mathbf{L}, \tag{2b}$$
$$c^{\pi}_{ij} \le 0 \text{ for each arc } (i,\ j) \in \mathbf{U}. \tag{2c}$$

In the subsequent discussion, we will refer to a primal feasible basis structure (**B**, **L**, **U**) simply as a *primal basis structure*, and in it we refer to **B** as a *primal basis*. Similarly, we will refer to a dual feasible basis structure (**B**, **L**, **U**) simply as a *dual basis structure*, and in it we refer to **B** as a *dual basis*. In a primal or dual basis, there is a unique path consisting of basic arcs between any pair of nodes; we refer to this path as a *basis path*.

3. THE PRIMAL SIMPLEX ALGORITHM

We preset here a brief review of the primal simplex algorithm for the maximum flow problem. A detailed description of the algorithm can be found in Ahuja et al. [1993]. The primal (network) simplex algorithm maintains a basis structure (**B**, **L**, **U**), which is primal feasible but dual infeasible. The algorithm performs a sequence of *primal pivots* until the basis structure maintained by it also becomes dual feasible.

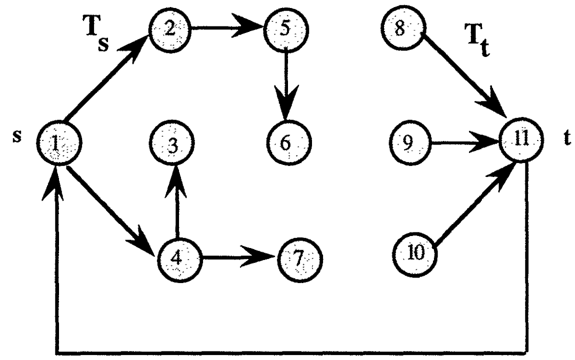


Figure 1. An example of a primal basis.

A primal basis **B** of the maximum flow problem consists of two subtrees T_s (containing the source node), and T_t (containing the sink node), and an arc (t, s) connecting these two subtrees. An example of a primal basis is shown in Figure 1. The node potentials associated with a primal basis are: $\pi(i) = 1$ for all $i \in T_s$ and $\pi(i) = 0$ for all $i \in T_t$. The set of arcs with one endpoint in T_s and another in T_t defines an s-t cut $[T_s, T_t]$. The nonbasic arcs, which violate the dual feasibility conditions stated in (2b) or (2c), are *eligible* to enter the basis; Table 1 lists such arcs.

Arc Type	Reduced Cost
(i) $i \in T_s$, $j \in T_t$ and $x_{ij} = 0$	$c^{\pi}_{ij} = -1$
(ii) $i \in T_t$, $j \in T_s$ and $x_{ij} = u_{ij}$	$c^{\pi}_{ij} = +1$

Table 1. Arcs eligible to enter a primal basis.

At each iteration, the primal simplex algorithm selects an eligible arc to enter the basis. In principle, the primal simplex algorithm can select *any* eligible arc to enter the basis. Different specific implementations can be obtained by specifying different rules for the selection of entering arcs. Suppose that the algorithm selects an arc (k, *l*) to enter the basis **B**. Adding arc (k, *l*) to **B** forms a fundamental cycle $\mathbf{W} = \mathbf{P} \cup \{(t, s)\}$, where **P** is the path from node s to node t in $\mathbf{B} \cup \{(k,l)\} \setminus \{t,s\}$. The algorithm then augments the maximum possible flow along W. The maximum increase Δ_{ij} permitted by an arc (i, j) is given by

$$\Delta_{ij} = \begin{cases} u_{ij} - x_{ij} & \text{if (i, j) is a forward arc in } \mathbf{W} \\ x_{ij} & \text{if (i, j) is a backward arc in } \mathbf{W} . \end{cases} \tag{3}$$

The algorithm augments $\Delta = \min \{\Delta_{ij} : (i, j) \in \mathbf{W}\}$ units of flow along **W**. An arc (p, q) satisfying $\Delta_{pq} = \Delta$ is the leaving arc. A new basis structure is obtained by replacing arc (p, q) by (k, *l*) in **B**, and updating **L** and **U**. The process of moving from one primal basis structure to another primal basis structure is called a *primal pivot* operation. Thus the primal simplex algorithm performs a sequence of primal pivot operations until the set of eligible arcs is empty. At this point, the primal basis structure is also dual feasible, and its associated flow is a maximum flow.

4. THE DUAL SIMPLEX ALGORITHM

In this section, we describe the dual simplex algorithm for the maximum flow problem. This algorithm is a special case of the dual simplex algorithm for the minimum cost flow problem, described, for example, in Ahuja et al. [1993]. The dual (network) simplex algorithm maintains a basis structure (**B**, **L**, **U**), which is dual feasible but primal infeasible. The algorithm performs a sequence of dual pivots until the basis structure maintained by it also becomes primal feasible.

The flow x associated with the dual basis structure (**B**, **L**, **U**) maintained by the dual simplex algorithm satisfies the mass balance constraint (1b) at all nodes, but basic arcs might violate their flow bound constraints (1c). We refer to an arc **B** violating its flow bounds as an *infeasible arc*, and the amount by which it violates one of its bounds as its *infeasibility*. The dual basis structure (**B**, **L**, **U**) maintained by our dual simplex algorithm satisfies the following invariant properties:

Invariant 1. *(t, s)* $\in$ ***U***.

Invariant 2. *All infeasible arcs lie on the basis path **P** from node s to node t. Moreover, it is possible to decrease flow in the cycle* $(P \cup \{(t,s)\}$ *so that the resulting flow is feasible.*

A consequence of Invariant 1 is that the dual basis **B** is a spanning tree of A/{(t, s)}. Figure 2 shows an example of a dual basis **B**; in this basis, only the arcs in the path 1-4-7-9-11 are allowed to be infeasible.

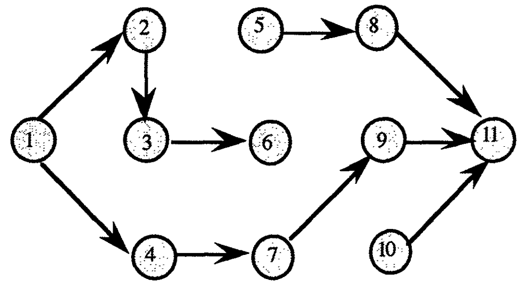


Figure 2. An example of a dual basis.

Our dual simplex algorithm obtains an initial dual basis structure (**B**, **L**, **U**) satisfying Invariants 1 and 2 in the following manner. Let **T** be any spanning tree of A/{(t, s)}. We set **B** = **T**, **L** = A/(**B** $\cup$ {(t, s)}), and **U** = {(t, s)}. We obtain the flow x corresponding this basis structure by first setting x = 0 and then augmenting M units of flow in the cycle **P** $\cup$ {(t, s)}, where **P** is the basis path from node s to node t. This basis structure is dual feasible, because $\pi(i) = 0$ for all i $\in$ N satisfying the conditions in (2), but is not primal feasible because at least one arc in the basis path **P** violates its lower or upper bound. Observe that there is no feasible circulation with x_{ts} = M, since M is a strict upper bound on the maximum flow from node s to node t.

We now explain how to perform a dual pivot. We define δ_{ij} for an arc (i, j) in the basis path **P** in the following manner:

$$\delta_{ij} = \begin{cases} x_{ij} - u_{ij} & \text{if (i, j) is a forward arc in } \mathbf{P} \\ -x_{ij} & \text{if (i, j) is a backward arc in } \mathbf{P}. \end{cases} \tag{4}$$

Observe that if $\delta_{ij} \geq 0$, then it denotes the infeasibility of the arc (i, j). Let $\delta = \max \{\delta_{ij} : (i, j) \in \mathbf{P}\}$. Then, $\delta > 0$, and δ denotes the maximum infeasibility of an arc. In the generic version of the dual simplex algorithm, any arc with positive infeasibility δ_{ij} can be selected as the leaving arc. Our more restrictive dual simplex algorithm uses the following rule to select the leaving arc.

Invariant 3. *Select an arc with the maximum infeasibility in the basis path* ***P*** *as the leaving arc.*

We point out that there may be several arcs in the basis path **P** with infeasibility equal to δ; the dual simplex algorithm can select any one of these arcs as the leaving arc. Suppose that the algorithm selects arc (p, q) as the leaving arc. Dropping arc (p, q) from the basis **B** forms two subtrees: T_s (containing node s) and T_t (containing node t). The arcs eligible to enter the dual basis are given in Table 2.

Arc Type	Reduced Cost
(i) $i \in T_s$, $j \in T_t$ and $x_{ij} = 0$	$c^{\pi}_{ij} = 0$
(ii) $i \in T_t$, $j \in T_s$ and $x_{ij} = u_{ij}$	$c^{\pi}_{ij} = 0$
(iii) (t, s)	$c^{\pi}_{ts} = -1$

Table 2. Arcs eligible to enter the dual basis.

In the dual simplex algorithm, the entering arc is selected by using the minimum ratio pivot rule. According to this rule, any arc of type (i) or type (ii) can be selected to enter the basis. If no arc of type (i) or type (ii) exists, then arc (t, s) will enter the basis. Let arc (k, *l*) be the entering arc. The dual simplex algorithm performs the dual pivot operation according to the following two cases.

Case 1. *(k, l) ≠ (t, s).* Replacing the leaving arc (p, q) by the entering arc (k, *l*) gives us a new dual basis **B'** with the corresponding basis path **P'**. The pivot consists of (i) decreasing the flow in the path **P** by δ units (after which no arc in **P** will be infeasible); and (ii) increasing the flow in the path **P'** by δ units (after which some arcs in **P'** will become infeasible).

Case 2. *(k, l) = (t, s).* Replacing arc (p, q) by the arc (t, s) gives us the new dual basis **B'** with the corresponding basis path **P'**. The basis path **P'** contains the arc (t, s) as a backward arc. The dual operation consists of decreasing the flow in the cycle $\mathbf{P'} \cup \{(k, l)\}$ by δ units, after which all arcs become feasible.

Thus the dual simplex algorithm repeats the above process performing dual pivots according to Case 1, until finally arc (t, s) is selected as an entering arc, a dual pivot according to Case 2 is performed, and the algorithm terminates with an optimal flow.

We would like to point out that in Case 1 if the leaving arc is not an arc with the maximum infeasibility, then the next dual basis may not satisfy Invariant 2. The purpose of selecting an arc with the maximum infeasibility is to ensure that the next dual basis satisfies Invariant 2.

5. EQUIVALENCE OF PRIMAL AND DUAL BASIS STRUCTURES

In this section, we prove the main result of the paper, which is to show the equivalence between the primal and dual simplex algorithms for the maximum flow problem. To do so, we need to define the dual basis structures induced by the primal basis structures and, conversely, the primal basis structures induced by the dual basis structures.

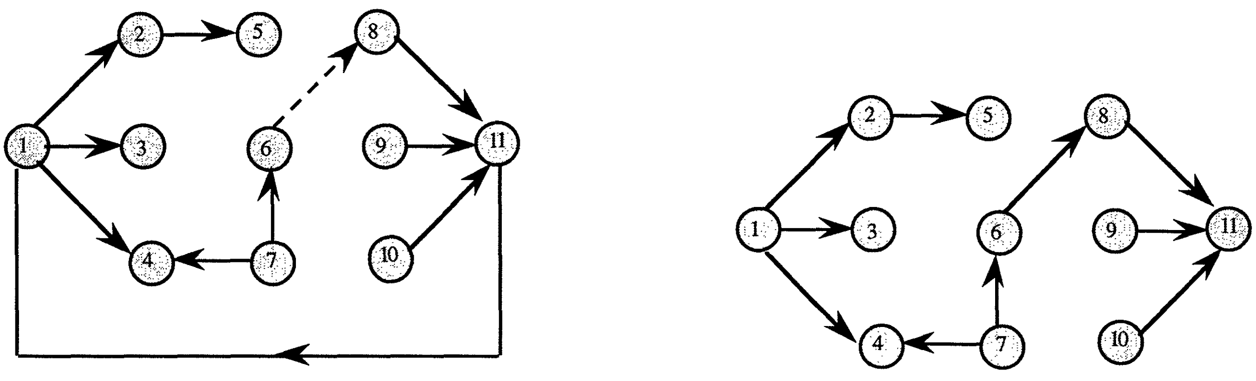
Induced Dual Basis Structure

Let $\mathcal{B}$ = (**B**, **L**, **U**) be any primal basis structure of the maximum flow problem with an associated flow x. Then, the primal basis structure $\mathcal{B}$ together with an arc (k, *l*), which is eligible to enter the basis, induces a dual basis structure $\mathcal{B}'$ = (**B'**, **L'**, **U'**) according to the following two cases:

Case 1. (k, *l*) ∈ **L**. Augment (M - x_{ts}) units of flow along the fundamental cycle **W** induced by the arc (k, *l*), and set **B'** = **B**∪{(k, *l*)}/{(t, s)}, **U'** = **U**∪{(t, s)}, and **L'** = **L**/{(k, *l*)}.

Case 2. (k, *l*) ∈ **U**. Augment (M - x_{ts}) units of flow opposite to the orientation of the fundamental cycle **W** induced by the arc (k, *l*) and set **B'** = **B**∪{(k, *l*)}/{(t, s)}, **U'** = **U**∪{(t, s)}/{(k, *l*)}, and **L'** = **L**.

We illustrate this process in Figure 3. Figure 3(a) shows a primal basis. Let arc (6, 8) be the entering arc at its lower bound. Figure 3(b) shows the induced dual basis.



(a) (b)

Figure 3. Illustrating induced dual basis structures.

Notice that the dual basis structure $\mathcal{B}'$ induced by the primal basis structure $\mathcal{B}$ and the entering arc (k, *l*) is dual feasible with respect to π = 0. Also notice that the flow x' corresponding to $\mathcal{B}'$ is x plus (M - x_{ts}) units of flow along the cycle W, which is the union of the basis path **P'** in $\mathcal{B}$ and arc (t, s). Consequently, x'_{ts} = M and all infeasible arcs lie on the basis path **P'**. Moreover, decreasing the flow on W by M-x_{ts} units results in the feasible flow x. Therefore, the dual basis structure $\mathcal{B}'$ satisfies Invariants 1 and 2. We denote the dual basis structure induced by the primal basis structure $\mathcal{B}$ and the eligible arc (k, *l*) by f($\mathcal{B}$, (k, *l*)), and summarize the preceding discussion by the following lemma.

Lemma 1. *Let $\mathcal{B}$ be any primal basis structure for the maximum flow problem, and (k, l) be any arc eligible to enter the basis. Then, the induced dual basis structure $\mathcal{B}' = f(\mathcal{B}, (k, l))$ satisfies Invariants 1 and 2.*

Induced Primal Basis Structures

Let $\mathcal{B}$ = (**B, L, U**) be any dual basis structure of the maximum flow problem satisfying Invariants 1 and 2, and which has an associated flow x. Let **P** denote the basis path in **B**. Assume that $\mathcal{B}$ is not primal feasible. Let δ denote the maximum infeasibility of any arc, and (p, q) be an arc with $\delta_{pq} = \delta$. Then the dual basis structure $\mathcal{B}$ together with the leaving arc (p, q) induces a primal basis structure $\mathcal{B}'$ = (**B, L, U**) according to the following two cases:

Case 1. *Arc (p, q) violates its lower bound.* Decrease the flow in the cycle **W** = **P**∪{(t, s)} by δ units, and set **B**' = **B**∪{(t, s)}/{(p, q)}, **L**' = **L**∪{(p, q)}, and **U**' = **U**/{(t, s)}. Let x' denote the modified flow.

Case 2. *Arc (p, q) violates its upper bound.* Decrease the flow in the cycle **W** = **P**∪{(t, s)} by δ units, and set **B**' = **B**∪{(t, s)}/{(p, q)}, **L**' = **L**, and **U**' = **U**∪((p, q)}/{(t, s)}. Let x' denote the modified flow.

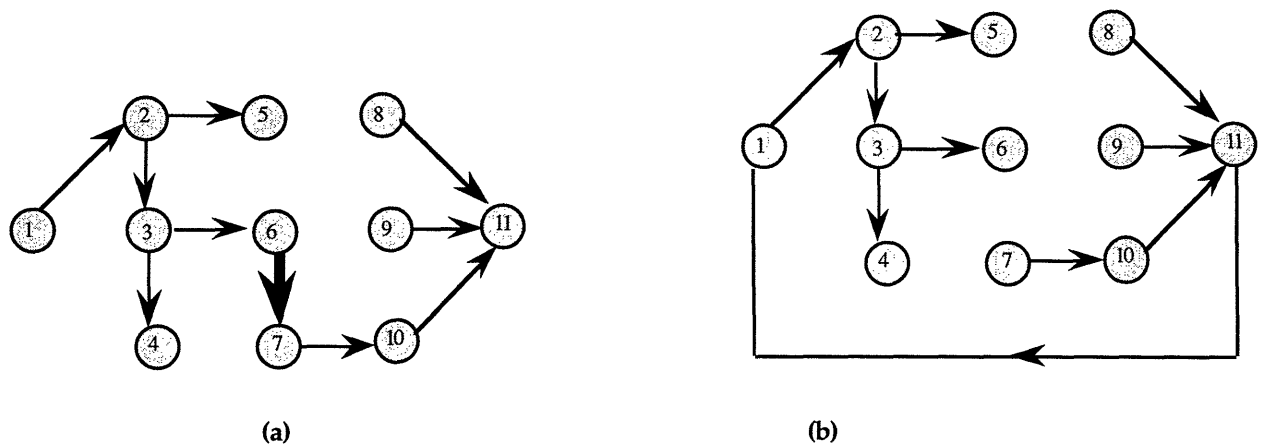


Figure 4. Illustrating induced primal basis structures.

We illustrate this process in Figure 4. Figure 4(a) shows a dual basis. Let arc (6, 7) be an arc with the maximum violation. Figure 4(b) shows the induced primal basis. By assumption, the dual basis structure $\mathcal{B}$ satisfies Invariants 1 and 2. Let x′ be obtained by decreasing the flow in the cycle **W** = **P**∪{(t, s)} by δ units, and let (p,q) be the exiting arc. By Invariants 1 and 2, the flow x′ is feasible, and

$x'_{pq} = 0$ or $x'_{pq} = u_{pq}$, and $x'_{ts} < u_{ts}$. It follows that $\mathcal{B}'$ is a primal basis structure. We denote the primal basis structure induced by the dual feasible structure $\mathcal{B}$ and the leaving arc (p, q) by $\mathcal{B}' = g(\mathcal{B}, (p, q))$, and summarize the preceding discussion by the following lemma.

Lemma 2. *Let $\mathcal{B}$ be any dual basis structure for the maximum flow problem, and let (p, q) be any arc eligible to leave the basis. Then, the induced basis structure $\mathcal{B}' = g(\mathcal{B}, (p, q))$ is primal feasible.*

Equivalence

We have shown in Lemma 1 that any primal basis structure plus an arc eligible to enter the basis induces a dual basis structure satisfying Invariants 1 and 2. We now extend this result to a sequence of primal basis structures.

Theorem 1. *Let $\mathcal{B}^1, \mathcal{B}^2, \ldots, \mathcal{B}^{k+1}$ be a sequence of primal basis structures for the maximum flow problem, terminating at an optimal primal basis structure. Let $(k^1, l^1), (k^2, l^2), \ldots, (k^K, l^K)$ be the sequence of entering arcs, and let $(p^1, q^1), (p^2, q^2), \ldots, (p^K, q^K)$ be the sequence of leaving arcs. Then the following results are true:*

1. *$f(\mathcal{B}^1, (k^1, l^1)), f(\mathcal{B}^2, (k^2, l^2)), \ldots, f(\mathcal{B}^k, (k^K, l^K))$ is a sequence of dual basis structures for the maximum flow problem satisfying Invariants 1 and 2, terminating at an optimal dual basis structure.*

2. *For each $1 \le i \le K$, the dual basis structure $(\mathcal{B}^{i+1}, (k^{i+1}, l^{i+1}))$ is obtained from the dual basis structure $(\mathcal{B}^i, (k^i, l^i))$ by performing a dual pivot which pivots out the arc (p^i, q^i) and pivots in the arc (k^{i+1}, l^{i+1}).*

Proof. The first result in Theorem 1 follows directly from Lemma 1. We will present a proof of the second result. To understand this proof, a reference to the example shown in Figure 5 will be helpful. Let Figures 5(a) and (b) depict two consecutive primal bases $\mathcal{B}^i$ and $\mathcal{B}^{i+1}$. Let Figure 5(c) and (d) depict the dual bases induced by $\mathcal{B}^i$ and $\mathcal{B}^{i+1}$, respectively.

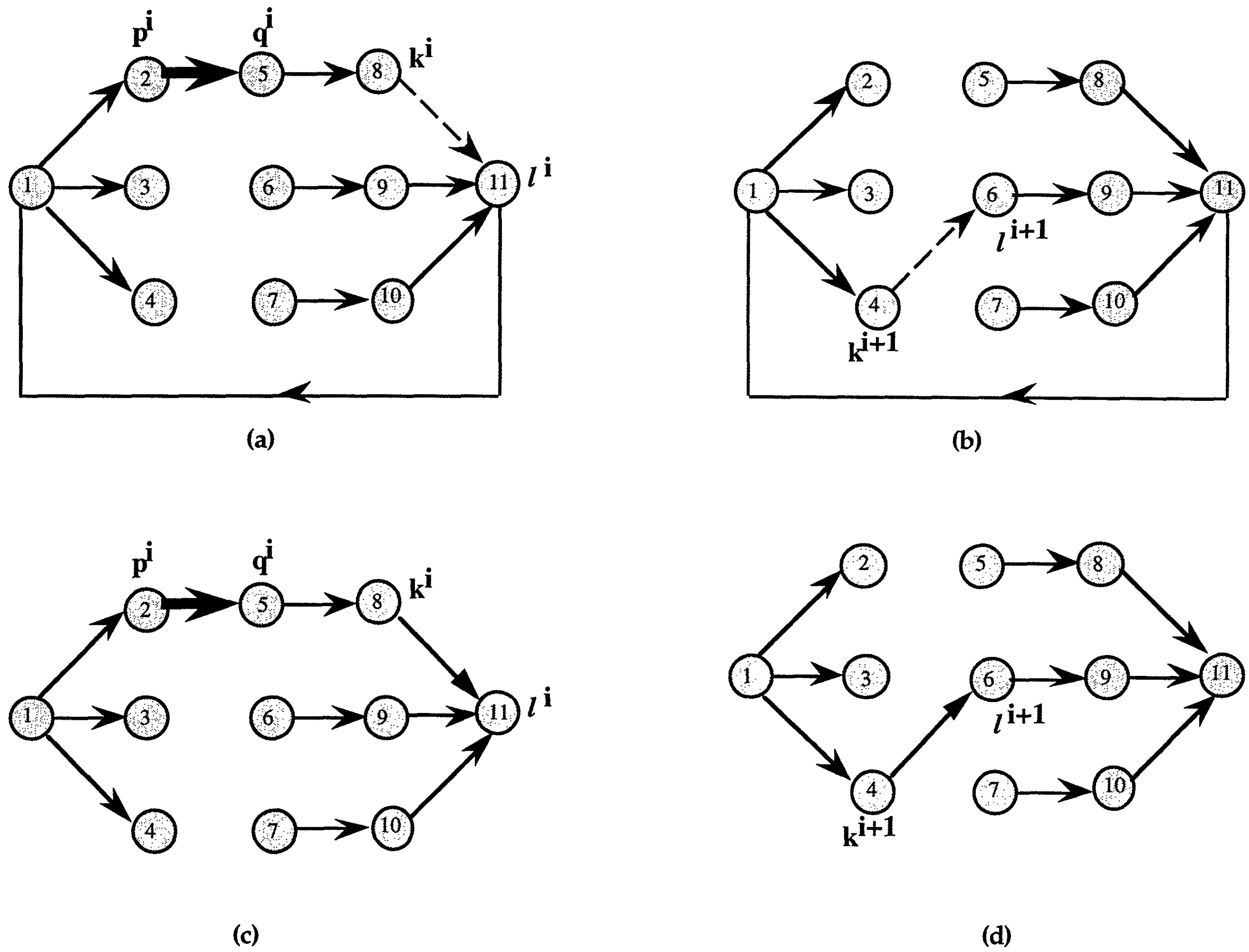


Figure 5. Illustrating equivalence of primal and dual basis structures.

Consider the primal basis $\mathcal{B}^i$ with (k^i, l^i) as the entering arc and x as the corresponding flow. Let **P** denote the path from node s to node t (not containing arc (t, s)) created by adding the arc (k^i, l^i) to $\mathcal{B}^i$. The following results are true:

1. **P** is a basis path from node s to node t in the dual basis induced by $\mathcal{B}^i$ and the entering arc (k^i, l^i).

2. The flow x' corresponding to the dual basis is x plus $\alpha = (M - x_{ts})$ units of flow augmented along the path **P**.

3. For each infeasible arc (u, v) in the path **P**, its residual capacity Δ_{uv} in the primal flow x (given by (3)), and its infeasibility δ_{uv} in the dual flow x' (given by (4)) satisfy the relation $\Delta_{uv} + \delta_{uv} = \alpha$.

It follows from the above facts that if the arc (p^i, q^i) is an arc of minimum residual capacity in **P** in the primal flow x, then the arc (p^i, q^i) is an arc of maximum infeasibility in the induced dual basis.

In other words, in the dual basis structure $f(\mathcal{B}^i, (k^i, l^i))$, arc (p^i, q^i) is an eligible arc to leave the basis. When we pivot out arc (p^i, q^i), then the dual basis is partitioned into two subtrees T^s and T^t, which are exactly the same subtrees as in the primal basis $\mathcal{B}^{i+1}$ (see, for example, Figures 5(b) and (c)). Next observe from Table 1 and 2 that an arc $(k, l) \neq (t, s)$ is qualified to be an entering arc in the primal basis $\mathcal{B}^{i+1}$ if and only if it is qualified to be a entering arc in its induced dual basis. This implies that the arc (k^{i+1}, l^{i+1}) can be pivoted in to obtain a new dual basis (see for example, Figure 5(d)). To summarize, we have shown that for the dual basis structure $f(\mathcal{B}^i, (k^i, l^i))$, there exists a valid dual pivot which pivots out the arc (p^i, q^i) and pivots in the arc (k^{i+1}, l^{i+1}) and obtains the dual basis structure $f(\mathcal{B}^{i+1}, (k^{i+1}, l^{i+1}))$. This completes the proof of the theorem. ♦

We now state the converse of Theorem 1.

Theorem 2. *Let $(\mathcal{B}^1, \mathcal{B}^2, ..., \mathcal{B}^{K+1})$ be a sequence of dual basis structures for the maximum flow problem satisfying Invariants 1 and 2, and terminating at an optimal dual basis structure. Let (p^1, q^1), (p^2, q^2), ..., (p^K, q^K) be the sequence of leaving arcs, and let (k^1, l^1), (k^2, l^2), ..., (k^K, l^K) be the sequence of entering arcs. Then the following results are true:*

1. *$g(\mathcal{B}^1, (p^1, q^1))$, $g(\mathcal{B}^2, (p^2, q^2))$, ..., $g(\mathcal{B}^K, (p^K, q^K))$ is a sequence of primal basis structures for the maximum flow problem, terminating at an optimal primal basis structure.*

2. *For each $1 \le i \le K$, the primal basis structure $g(\mathcal{B}^{i+1}, (p^{i+1}, q^{i+1}))$ is obtained from the primal basis structure $g(\mathcal{B}^i, (p^i, q^i))$ by performing a primal pivot which pivots in the arc (k^i, l^i) and pivots out the arc (p^{i+1}, q^{i+1}).*

Proof. The first result in the theorem is a byproduct of Lemma 2. The proof of the second result is analogous to the proof of the second result in Theorem 1, and we only outline the proof here. First, one can establish that arc (k^i, l^i) is an eligible entering arc of $g(\mathcal{B}^i, (p^i, q^i))$. Next, as in the proof of Theorem 1, one can establish the equivalence of the minimum residual capacity arc on the path P from s to t in $g(\mathcal{B}^i, (p^i, q^i)) \cup \{ (k^i, l^i)\}$ and the maximum violating arc on the path P from s to t in $\mathcal{B}^{i+1}$. The equivalence establishes that an eligible leaving arc for $\mathcal{B}^{i+1}$ is also an eligible leaving arc for $g(\mathcal{B}^i, (p^i, q^i)) \cup \{ (k^i, l^i)\}$. ♦

An immediate consequence of Theorems 1 and 2 is that for every primal simplex algorithm for the maximum flow problem, there is a corresponding specialization of the dual simplex algorithm which performs the same number of pivots and runs in the same time. The converse result is true with some added restrictions. If a dual simplex algorithm for the maximum flow problem satisfies Invariants 1 and 2, then there is a corresponding primal simplex algorithm performing the same number of pivots and running in the same time.

ACKNOWLEDGEMENTS

This research was partially supported by ONR Grant N00014-96-1-0099 and a grant from UPS Foundation.

REFERENCES

Ahuja, R. K., T. L. Magnanti, and J. B. Orlin. 1993. *Network Flows: Theory, Algorithms, and Applications*. Prentice Hall, Englewood Cliffs, NJ.

Armstrong, R.D., W. Chen, D. Goldfarb, and Z. Jin. 1994. Strongly polynomial dual simplex methods for the maximum flow problem. Research Report, Graduate School of Management, Rutgers University, Newark, NJ.

Armstrong, R.D., and Z. Jin. 1992. A strongly polynomial dual (simplex) method for the max flow problem. Research Report, Graduate School of Management, Rutgers University, Newark, NJ.

Goldberg, A.V., M.D. Grigoriadis, and R.E. Tarjan. 1991. Use of dynamic trees in a network simplex algorithm for the maximum flow problem. *Mathematical Programming* **50**, 277-290.

Goldfarb, D., and W. Chen. 1992. An $O(n^3)$ dual simplex method for the maximum flow problem. Research Report, Dept. of Industrial Engg. & Operations Research, Columbia University, New York, NY.

Goldfarb, D., and J. Hao 1990. A primal simplex algorithm that solves the maximum flow problem in at most nm pivots and $O(n^2m)$ time. *Mathematical Programming* **47**, 353-365.

Goldfarb, D., and J. Hao. 1991. On strongly polynomial variants of the network simplex algorithm for the maximum flow problem. *Operations Research Letters* **10**, 383-387.