SIAM J. CONTROL AND OPTIMIZATION
Vol. 25, No. 5, September 1987

007

RELAXATION METHODS FOR NETWORK FLOW PROBLEMS WITH CONVEX ARC COSTS*

DIMITRI P. BERTSEKAS†, PATRICK A. HOSEIN† AND PAUL TSENG†

Abstract. We consider the standard single commodity network flow problem with both linear and strictly convex possibly nondifferentiable arc costs. For the case where all arc costs are strictly convex we study the convergence of a dual Gauss-Seidel type relaxation method that is well suited for parallel computation. We then extend this method to the case where some of the arc costs are linear. As a special case we recover a relaxation method for the linear minimum cost network flow problem proposed in Bertsekas [1] and Bertsekas and Tseng [2].

Key words. network flow, relaxation methods, parallel computation

1. Introduction. Consider a directed graph with set of nodes N and set of arcs A. We will write $j \sim (i, k)$ to denote that the start and end nodes of arc j are i and k, respectively. The network incidence matrix is denoted by E and has elements e_{ij} given by

$$e_{ij} = \begin{cases} 1 & \text{if } i \text{ is the start node of arc } j, \\ -1 & \text{if } i \text{ is the end node of arc } j, \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

We denote by x_j the *flow* of arc j, and by d_i the *deficit* of node i which is defined by

$$d_i = \sum_{j \in A} e_{ij} x_j \quad \forall i \in N. \tag{2}$$

In words d_i is the balance of flow outgoing from i and flow coming into i. The vectors with coordinates x_j and d_i are denoted x and d respectively. Thus (2) is written as

$$d = Ex. \tag{3}$$

In what follows the association of particular deficit vectors and flow vectors via (3) should be clear from the context.

Each arc j has associated with it a cost function $f_j : R \to (-\infty, +\infty]$. We consider the problem of minimizing total cost subject to a conservation of flow constraint at each node:

$$\begin{aligned} &\text{minimize} \quad f(x) = \sum_{j \in A} f_j(x_j) \\ &\text{subject to} \quad x \in C \end{aligned} \tag{4}$$

where C is the *circulation subspace*:

$$C = \{x | d_i = 0, i \in N\} = \{x | Ex = 0\}. \tag{5}$$

We make the following assumptions on f_j.

Assumption A. Each function f_j is convex, lower semicontinuous, and there exists at least one feasible solution for problem (4), i.e., the effective domain of f

$$\operatorname{dom}(f) = \{x | f(x) < +\infty\}$$

and the circulation subspace C have a nonempty intersection.

* Received by the editors January 6, 1986; accepted for publication (in revised form) September 3, 1986. This work was supported by the National Science Foundation under grant NSF-ECS-8217668.

† Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139.

Assumption B. The conjugate convex function of each f_j defined by

$$g_j(t_j) = \sup_{x_j} \{t_j x_j - f_j(x_j)\} \tag{6}$$

is real valued, i.e., $-\infty < g_j(t_j) < +\infty$ for all $t_j \in R$.

Assumption B implies that $f_j(x_j) > -\infty$ for all x_j and j. It follows that the set of points where f_j is real valued, denoted dom (f_j), is a nonempty interval the right and left endpoints of which (possibly $+\infty$ or $-\infty$) we denote by c_j and l_j, respectively, i.e.,

$$c_j = \sup \{x_j | f_j(x_j) < +\infty\}, \qquad l_j = \inf \{x_j | f_j(x_j) < +\infty\}.$$

We call c_j and l_j the *upper and lower capacity bounds* of f_j respectively. It is easily seen that Assumptions A and B imply that for every t_j there is some $x_j \in \text{dom}\,(f_j)$ attaining the supremum in (6), and furthermore

$$\lim_{|x_j| \to +\infty} f_j(x_j) = +\infty.$$

It follows that the cost function of (4) has bounded level sets, and therefore (using also the lower semicontinuity of f) *there exists at least one optimal flow vector.*

Assumptions A and B are satisfied, if, for example, f_j is of the form

$$f_j(x_j) = \begin{cases} \hat{f}_j(x_j, & \text{if } x_j \in [l_j, c_j], \\ \infty & \text{otherwise} \end{cases} \tag{7}$$

where l_j, c_j are given upper and lower bounds on the flow of arc j, and $\hat{f}_j$ is a real valued convex function on the real line R. In this case $g_j(t_j)$ is linear for $|t_j|$ large enough with slopes l_j and c_j as t_j approaches $-\infty$ and $+\infty$, respectively (see Fig. 1.1).

Problem (4) is called the *optimal distribution problem* in Rockafellar [3]. The same reference develops in detail a duality theory (a refinement of what can be obtained from Fenchel's duality theorem) involving the dual problem

$$\begin{aligned} &\text{minimize} \quad g(t) \triangleq \sum_{j \in A} g_j(t_j) \\ &\text{subject to} \quad t \in C^\perp \end{aligned} \tag{8}$$

where t is the vector with coordinates t_j, $j \in A$, and $C^\perp$ is the orthogonal complement of C. We call t_j the *tension* of the arc j and $C^\perp$ the *tension subspace*. From (1)-(3) and

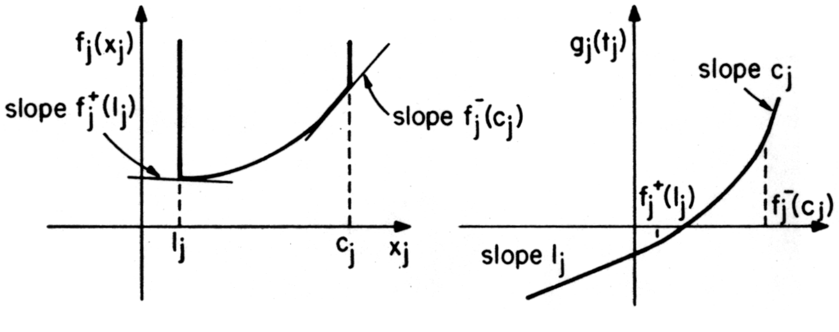

FIG. 1.1. *Primal cost function of the form* (7) *and its dual.*

(5) we have that $t \in C^{\perp}$ if and only if there exist scalars p_i, $i \in N$, called *prices*, such that

$$t_j = p_i - p_k \quad \forall j \in A \text{ with } j \sim (i, k), \tag{9}$$

or equivalently

$$t = E^T p \tag{10}$$

where E^T is the transpose of the network incidence matrix E and p is the vector with coordinates p_i, $i \in N$. Therefore the dual problem (8) can also be written as

$$\begin{aligned} &\text{minimize} \quad q(p) \\ &\text{subject to no constraints on } p \end{aligned} \tag{11}$$

where q is the *dual functional*

$$q(p) = \sum_{\substack{j \in A \\ j \sim (i,k)}} g_j(p_i - p_k). \tag{12}$$

As shown in [3, p. 349], Assumption A guarantees that there is no duality gap in the sense that the primal and dual optimal costs are opposites of each other.

An important fact for the purposes of the present paper is that (in view of Assumption B above) *the dual problem* (11) *is an unconstrained optimization problem.* If each function f_j is strictly convex, the dual functional is also differentiable ([4, p. 253]) and as a result unconstrained smooth optimization methods can be applied for solution. This is particularly so since the gradient of the dual cost can be easily calculated. Indeed, when f_j is strictly convex, for every tension vector t there exists a unique flow vector x such that

$$x_j = \arg\max_{z_j} \{t_j z_j - f_j(z_j)\} \quad \forall j \in A \tag{13}$$

and it can be shown [4, p. 218] that x_j is the gradient of g_j at t_j

$$x_j = \nabla g_j(t_j) \quad \forall j \in A. \tag{14}$$

From (1) and (12) we see that for a given price vector p the partial derivatives of the dual functional q are given by

$$\frac{\partial q(p)}{\partial p_i} = \sum_{j \in A} e_{ij} \nabla g_j(t_j \quad \forall i \in N. \tag{15}$$

Equivalently (cf. (2)), *the partial derivative $\partial q(p)/\partial p_i$ equals the deficit of node i when the arc flows x_j are the unique scalars defined by* (13).

The differentiability of the dual cost when the primal cost is strictly convex motivates a Gauss-Seidel type of algorithm whereby, given a price vector p, one calculates the corresponding flows $x_j = \nabla g_j(t_j)$, $j \in A$, chooses a node i with positive (negative) deficit and decreases (increases) p_i up to the point where the corresponding partial derivative $\partial q/\partial p_i$ becomes zero. (This amounts to minimizing the dual functional q along the coordinate p_i.) One then repeats the procedure iteratively. The algorithm above is attractive not only because of its simplicity but also because *it lends itself naturally to distributed computation,* whereby minimization along different price coordinates is carried out simultaneously by several processors. Indeed, this can be done in an asynchronous format as described and analyzed in Bertsekas and El Baz [5]. Simulations of a synchronous parallel method of this type [19] have shown remarkable speedup in computation time.

Gauss-Seidel relaxation methods for unconstrained optimization have been studied extensively [6]-[10]. However they typically require for convergence something like a strict convexity assumption on the cost minimized as well as boundedness of its level sets (see [10] for a counterexample). Unfortunately the dual cost (12) always has unbounded level sets since adding the same constant to all node prices leaves the cost unchanged. Even if we remove this degree of freedom by restricting the price of some special node to be zero (i.e. passing to a quotient space), the dual cost may still have unbounded levels sets and is not strictly convex when the functions f_j are nondifferentiable as in the important special case (7) where they imply capacity constraints. One contribution of the present paper (§ 2) is to show convergence of a flow sequence generated by the Gauss-Seidel method to the unique optimal solution of the primal problem (4). Convergence of the corresponding price vector sequence to some optimal solution of the dual problem (11) is also shown assuming the dual has an optimal solution. For this we actually require that the minimization along coordinates be done only approximately. Furthermore nodes can be relaxed in arbitrary order. The only requirement is that each node is relaxed infinitely often. This result is new and is remarkable in that it requires a rather unconventional method of proof. It improves on a result by Pang [11] (see also an earlier paper by Cottle and Pang [12]) which asserts convergence of the flow vector sequence under the assumption that g_j is of the form (7) with $\hat{f}_j$ differentiable, and strongly convex (rather than just strictly convex as we assume). Pang's result requires exact minimization along each coordinate and contains no assertion on convergence of the price vector sequence; however it applies to a more general problem where the primal cost function need not be separable and the linear constraints need not have a network structure. The paper by Cottle and Pang [12] asserts subsequence convergence to a dual optimal solution for a transportation problem with quadratic arc costs but also uses a nondegeneracy assumption and places a restriction in the way relaxation is carried out. This result is strengthened in our analysis as described above.

When some of the arc cost functions f_j are not strictly convex, the dual cost is not differentiable, and the Gauss-Seidel method breaks down. However Bertsekas [1] and Bertsekas and Tseng [2] have proposed methods that are conceptually related to Gauss-Seidel and work with linear arc costs. They allow line minimization along directions involving several coordinates to cope with situations where minimizing along a single coordinate is not possible. Computational experimentation with standard benchmark problems and a code named RELAX [1], [2] shows that these methods are very promising and outperform, in terms of computation time, some of the best primal simplex and primal dual codes currently available. The second objective of this paper is to propose in § 3 a new relaxation method that in some sense bridges the gap between the strictly convex arc cost Gauss-Seidel method described earlier and the Bertsekas-Tseng linear arc cost version. We show that this method works with both linear and nonlinear (convex) arc costs and contains as special cases both relaxation methods described above. To our knowledge the only other known algorithm for network problems with both linear and nonlinear, possibly nondifferentiable, arc costs is Rockafellar's fortified descent method [3, Chap. 9]. Our algorithm relates in roughly the same way to the Bertsekas-Tseng relaxation method, as Rockafellar's relates to the classical primal-dual method. We note that the methods considered here for linear costs and, more generally, not strictly convex costs are not easily parallelizable. Related synchronous and asynchronous relaxation methods that admit massive parallelization have been proposed recently in [20], [21].

The last section of the paper provides results of computational experimentation with codes implementing both of the relaxation algorithms proposed.

2. The relaxation method for strictly convex arc costs. In this section in addition to Assumptions A and B, there will be a standing assumption that each f_j is *strictly convex*. Two important consequences of this assumption are that *the optimal flow vector is unique* and *the conjugate functions g_j are differentiable* (in addition to being real valued by Assumption B). Indeed it is easily verified (see also [3] and [4, p. 218]) that we have for all t_j

$$\nabla g_j(t_j)=\arg\max_{x_j}\{t_jx_j-f_j(x_j)\}. \tag{16}$$

Furthermore $\nabla g_j(t_j)$ is the unique scalar x_j satisfying together with t_j the *Complementary Slackness* (CS) *condition*

$$f_j^-(x_j)\leqq t_j\leqq f_j^+(x_j) \tag{17}$$

where $f_j^-(x_j)$ and $f_j^+(x_j)$ denote the left and right derivatives of f_j at x_j (see Fig. 2.1). These derivatives are defined in the usual way for x_j in the interior of dom (f_j). When $-\infty<l_j<c_j$ we define

$$f_j^+(l_j)=\lim_{\xi\downarrow l_j}\ {}_j^+(\xi),\qquad f_j^-(l_j)=-\infty.$$

When $l_j<c_j<+\infty$ we define

$$f_j^-(c_j)=\lim_{\xi\uparrow c_j}f_j^-(\xi),\qquad f_j^+(c_j)=+\infty.$$

Finally when $l_j=c_j$ we define $f_j^-(l_j)=-\infty, f_j^+(c_j)=+\infty$. Note that $\nabla g_j(t_j)$ is continuous and monotonically nondecreasing. We define the *deficit functions* d_i by

$$d_i(p)=\sum_{j\in A}e_{ij}\nabla g_j(t_j)\quad\forall i\in N$$

where $t=E^Tp$, and denote by $d(p)$ the vector with coordinates $d_i(p)$. Note that the definition of d is identical to that given in (2), except that here we have used the strict convexity of f_j to express flow and deficit as functions of the dual price vector. In view of the form of the dual functional, the relation above yields

$$d_i(p)=\frac{\partial q(p)}{\partial p_i}\quad\forall i\in N.$$

Since $d_i(p)$ is a partial derivative of a differentiable convex function, we have that *$d_i(p)$ is continuous and monotonically nondecreasing in the coordinate p_i*.

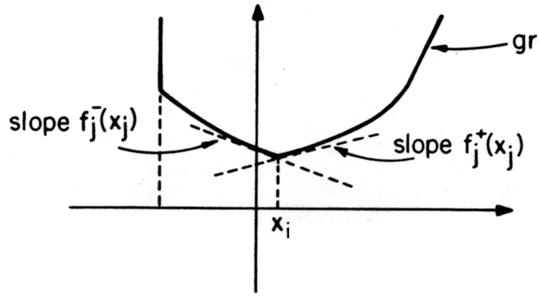

FIG. 2.1. *The left and right derivatives of f_j.*

Proof. Fix an index $r \geqq 0$. Denote $s = s^r$ and $\Delta = p_s^{r+1} - p_s^r$. From (6), (12) and (16) we have

$$q(p^r) = \sum_{j \in A} [t_j^r x_j^r - f_j(x_j^r)] \quad \forall r \geqq 0.$$

Therefore

$$\begin{aligned} q(p^r) - q(p^{r+1}) &= \sum_{j \in A} [t_j^r x_j^r - f_j(x_j^r)] - \sum_{j \in A} [t_j^{r+1} x_j^{r+1} - f_j(x_j^{r+1})] \\ &= \sum_{j \in A} [t_j^r x_j^r - f_j(x_j^r)] - \sum_{j \in A} [(t_j^r + e_{sj}\Delta) x_j^{r+1} - f_j(x_j^{r+1})] \\ &= \sum_{j \in A} [f_j(x_j^{r+1}) - f_j(x_j^r) - (x_j^{r+1} - x_j^r) t_j^r - e_{sj}\Delta x_j^{r+1}] \\ &= \sum_{j \in A} [f_j(x_j^{r+1}) - f_j(x_j^r) - (x_j^{r+1} - x_j^r) t_j^r] - \Delta \sum_{j \in A} e_{sj} x_j^{r+1} \\ &= \sum_{j \in A} [f_j(x_j^{r+1}) - f_j(x_j^r) - (x_j^{r+1} - x_j^r) t_j^r] - \Delta d_s(p^{r+1}). \end{aligned}$$

Since $\Delta d_s(p^{r+1}) \leqq 0$ (and $d_s(p^{r+1}) = 0$ if we use line minimization) the left side of (18) follows. The right side of (18) follows from the strict convexity of f_j and the fact $x^{r+1} \neq x^r$. QED

PROPOSITION 2.2. *The sequence* $\{x^r\}$ *is bounded.*

Proof. We first note that at every iteration the total deficit does not increase, i.e.,

$$\sum_{i \in N} |d_i(p^{r+1})| \leqq \sum_{i \in N} |d_i(p^r)|.$$

(This follows from the fact that a flow change on an arc reflects itself in a change of the deficit of its start node and an opposite change in the deficit of its end node. Furthermore the deficit of node s^r chosen for relaxation at the rth iteration cannot increase in absolute value or change sign during that iteration.) It follows that $\{d(p^r)\}$ is bounded. We now argue by contradiction. Suppose $\{x^r\}$ is unbounded. Then there must exist an arc j and a subsequence R such that $|x_j^r| \to +\infty$ as $r \to \infty$, $r \in R$. Since $\{d(p^r)\}$ is bounded it follows (passing into another subsequence if necessary) that there exists a directed cycle Y such that $x_j^r \to +\infty$ for all $j \in Y^+$, and $x_j^r \to -\infty$ for all $j \in Y^-$ as $r \to \infty$, $r \in R$. Since by the CS condition (17)

$$f_j^-(x_j^r) \leqq t_j^r \leqq f_j^+(x_j^r),$$

and also

$$\sum_{j \in Y^+} t_j^r - \sum_{j \in Y^-} t_j^r = 0,$$

we have for all r

$$\sum_{j \in Y^+} f_j^-(x_j^r) - \sum_{j \in Y^-} f_j^+(x_j^r) \leqq 0.$$

This is a contradiction since $x_j^r \to +\infty$ implies $f_j^-(x_j^r) \to +\infty$ while $x_j^r \to -\infty$ implies $f_j^+(x_j^r) \to -\infty$. QED

The next result is remarkable in that it shows that under a mild restriction on the way the relaxation iteration is carried out (which is typically very easy to satisfy in practice), the sequence of price vectors approaches the dual optimal set in an unusual manner. The result depends on the monotonicity of the functions ∇g_j.

We now define a Gauss-Seidel type of algorithm similar to the one sketched in § 1 whereby at each iteration a node s with positive (negative) deficit $d_s(p)$ is chosen and p_s is decreased (increased) with the aim of decreasing the dual cost $q(p)$. More formally, we initially choose a price vector p and a fixed scalar δ in the interval (0, 1). Then we execute repeatedly the relaxation iteration described below.

```
*Relaxation iteration for strictly convex arc costs.*
If $d_i(p)=0\ \forall i\in N$ then STOP.
Else
    Choose any node $s$. Set $\beta=d_s(p)$.
    If $\beta=0$, do nothing.
    If $\beta>0$, then decrease $p_s$ so that $0\leqq d_s(p)\leqq\delta\beta$.
    If $\beta<0$, then increase $p_s$ so that $0\geqq d_s(p)\geqq\delta\beta$.
```

The only assumption we make regarding the order in which nodes are chosen for relaxation is the following.

Assumption C. Every node in N is chosen as the node s in the relaxation iteration an infinite number of times.

The relaxation iteration is well defined, in the sense that every step in the iteration is executable. To see this suppose that $\beta>0$ and there does not exist a $\Delta<0$ such that $d_s(p+\Delta e_s)\leqq\delta\beta$, where e_s denotes the sth coordinate vector. Then using the definition of d, l_j, and c_j, it is easily seen that

$$\lim_{\Delta\to-\infty} d_s(p+\Delta e_s)=\sum_{e_{sj}>0} e_{sj}l_j+\sum_{e_{sj}<0} e_{sj}c_j\geqq\delta\beta>0,$$

which implies that the flow deficit of node s is positive for any flow x within the upper and lower arc capacity bounds and contradicts the existence of a feasible flow (Assumption A). An analogous argument can be made for the case where $\beta<0$.

In order to obtain our convergence result we must show that the sequence of flow vectors generated by the relaxation algorithm approaches the circulation subspace C (given by (5)). The line of argument that we will use is as follows: We will lower bound the amount of improvement in the dual functional q per iteration by a positive quantity. We will then show that if the sequence of flow vectors do not approach the circulation subspace, the quantity itself can be lower bounded by a positive constant which implies that the optimal dual functional has a value of $-\infty$. This will contradict the finiteness of the optimal primal cost.

We will denote the price vector generated at the rth iteration by p^r, $r=0,1,2,\cdots$ and the node operated on at the rth iteration by s^r, $r=0,1,2,\cdots$. To simplify notation we will denote

$$t^r=E^Tp^r,\qquad x_j^r=\nabla g_j(t_j^r).$$

We denote by x^r the vector with coordinates x_j^r, $j\in A$. Note the symmetry following from the *CS* condition (16) or (17): *x_j^r is the gradient of the dual cost g_j at t_j^r, while t_j^r is a subgradient of the primal cost f_j at x_j^r.* For any directed cycle Y of the network we will use Y^+ to denote the set of arcs $\{j\in A|j$ is positively oriented in $Y\}$, and Y^- to denote $Y\backslash Y^+$. We first show three preliminary results.

PROPOSITION 2.1. *We have for all r such that $p^{r+1}\neq p^r$ [i.e. $d_{s^r}(p^r)\neq 0$]*

$$q(p^r)-q(p^{r+1})\geqq\sum_{j\in A}[f_j(x_j^{r+1})-f_j(x_j^r)-(x_j^{r+1}-x_j^r)t_j^r]>0,\tag{18}$$

with equality holding if line minimization is used $[d_{s^r}(p^{r+1})=0]$.

PROPOSITION 2.3. *Given $p \in R^{|N|}$, let s be a node and let $\bar{p}$ denote a dual price vector obtained by applying the relaxation iteration to p using node s. Assume in addition that $\bar{p}$ is chosen so that*

$$\text{(19a)} \qquad \text{if } d_s(p) > 0 \text{ then } d_s[\bar{p} + \alpha(p - \bar{p})] > 0 \quad \forall \alpha > 0,$$

$$\text{(19b)} \qquad \text{if } d_s(p) < 0 \text{ then } d_s[\bar{p} + \alpha(p - \bar{p})] < 0 \quad \forall \alpha > 0.$$

Then for all $k \in N$, and all optimal dual price vectors p^* we have

$$\text{(20)} \qquad \min\{p_i - p_i^* | i \in N\} \leqq \bar{p}_k - p_k^* \leqq \max\{p_i - p_i^* | i \in N\}.$$

Note. Assumption (19) when $d_s(p) > 0$ $[d_s(p) < 0]$ is equivalent to assuming that $\bar{p}_s$ is chosen greater (less) or equal to the largest (smallest) minimizing point of the dual cost along the sth coordinate starting from p. It is automatically satisfied if the dual cost has a unique minimizing point along the line $\{p + \alpha e_s | \alpha \in R\}$.

Proof. Fix an optimal dual price vector p^* and consider an arbitrary price vector $\tilde{p}$. Let k be such that $\tilde{p}_k - p_k^* = \max\{\tilde{p}_i - p_i^* | i \in N\}$. We have

$$\tilde{p}_k - p_k^* \geqq \tilde{p}_i - p_i^* \quad \forall i \neq k$$

so that

$$\tilde{p}_k - \tilde{p}_i \geqq p_k^* - p_i^* \quad \forall j \sim (k, i), \qquad \tilde{p}_i - \tilde{p}_k \leqq p_i^* - p_k^* \quad \forall j \sim (i, k).$$

Since ∇g_j is a nondecreasing function, we have that

$$\nabla g_j(\tilde{p}_k - \tilde{p}_i) \geqq \nabla g_j(p_k^* - p_i^*) \quad \forall j \sim (k, i), \qquad \nabla g_j(\tilde{p}_i - \tilde{p}_k) \leqq \nabla g_j(p_i^* - p_k^*) \quad \forall j \sim (i, k).$$

Thus $d_k(\tilde{p}) \geqq d_k(p^*) = 0$.

The desired assertion (20) holds if $d_s(p) = 0$ since then we have $\bar{p} = p$. Assume that $d_s(p) < 0$. Consider the vector $\tilde{p}$ defined by

$$\tilde{p}_i = \begin{cases} p_i & \text{if } i \neq s, \\ p_s^* + \max\{p_j - p_j^* | j \in N\} & \text{if } i = s. \end{cases}$$

Then we have $\tilde{p}_s - p_s^* = \max\{\tilde{p}_i - p_i^* | i \in N\} = \max\{p_i - p_i^* | i \in N\}$ and by the preceding argument we have $d_s(\tilde{p}) \geqq 0$. Therefore, using assumption (19), we have $\bar{p}_s \leqq \tilde{p}_s$ while at the same time $p_s < \bar{p}_s$, and $p_i = \bar{p}_i$ for all $i \neq s$. The assertion (20) follows. The proof is similar when $d_s(p) > 0$. Q.E.D.

Note that Proposition 2.3 implies among other things that, if (19) is satisfied at all iterations, the sequence $\{p^r\}$ generated by the relaxation method is bounded. Furthermore if we can show that $\{p^r\}$ accumulates at an optimal price vector, the proposition implies that $\{p^r\}$ must converge to that vector. We are now ready to show our main result.

PROPOSITION 2.4. *Let $\{p^r, x^r\}$ be a sequence generated by the relaxation method for strictly convex arc costs. Then*

$$\text{(21)} \qquad \text{(a)} \quad \lim_{r \to \infty} d(p^r) = 0.$$

$$\text{(22)} \qquad \text{(b)} \quad \lim_{r \to \infty} x^r = x^*$$

where x^* is the unique optimal flow vector.

$$\text{(c)} \quad \lim_{r \to \infty} q(p^r) = -f(x^*) = \inf_p q(p).$$

(d) *If condition* (19) *is satisfied at each iteration, and the dual problem has an optimal solution, then*

$$\lim_{r\to\infty} p^r \to p^* \tag{23}$$

where p^* is some optimal price vector.

Proof. (a) We first show that

$$\lim_{r\to\infty} d_{s^r}(p^r) = 0. \tag{24}$$

Indeed, if this is not so there must exist an $\varepsilon > 0$ and a subsequence R such that $|d_{s^r}(p^r)| \geqq \varepsilon$ for all $r \in R$. Without loss of generality we assume that $d_{s^r}(p^r) \geqq \varepsilon$ for all $r \in R$. Since $\delta|d_{s^r}(p^r)| \geqq |d_{s^r}(p^{r+1})|$ we have that at the rth iteration some arc incident to node s^r must change its flow by at least Δ where $\Delta = (1-\delta)\varepsilon/|A|$. By passing to a subsequence if necessary we assume that this happens for the same arc j^* for all $r \in R$, and that $x_{j^*}^{r+1} - x_{j^*}^r \geqq \Delta$, for all $r \in R$. Using the boundedness of $\{x^r\}$ (Proposition 2.2) we may also assume that the subsequence $\{x_{j^*}^r\}_{r\in R}$ converges to some x_{j^*}. Using the convexity of f_j and Proposition 2.1 we have

$$\begin{aligned} q(p^r) - q(p^{r+1}) &\geqq f_{j^*}(x_{j^*}^{r+1}) - f_{j^*}(x_{j^*}^r) - (x_{j^*}^{r+1} - x_{j^*}^r)t_{j^*}^r \\ &\geqq f_{j^*}(x_{j^*}^r + \Delta) - f_{j^*}(x_{j^*}^r) - \Delta t_{j^*}^r \\ &\geqq f_{j^*}(x_{j^*}^r + \Delta) - f_{j^*}(x_{j^*}^r) - \Delta f_{j^*}^+(x_{j^*}^r) \end{aligned}$$

Taking the limit as $r \to \infty$, $r \in R$ and using the facts $x_{j^*}^r \to x_{j^*}$ and $\lim_{r\to\infty} f_{j^*}^+(x_{j^*}^r) \leqq f_{j^*}^+(x_{j^*})$ (in view of the upper semicontinuity of $f_{j^*}^+$) we obtain

$$\liminf_{\substack{r\to\infty \\ r\in R}} [q(p^r) - q(p^{r+1})] \geqq f_{j^*}(x_{j^*} + \Delta) - f_{j^*}(x_{j^*}) - \Delta f_{j^*}^+(x_{j^*}) > 0.$$

This implies that $\lim_{r\to\infty} q(p^r) = -\infty$. But this is not possible because from (6) and (12) we have $q(p) \geqq -\sum_{j\in A} f_j(x_j)$ for all p and $x \in C$. Therefore (24) is proved by contradiction.

We now show (21). Choose any $i \in N$. Take any $\varepsilon > 0$ and let R be the set of indices r such that $d_i(p^r) > 2\varepsilon$. Assume without loss of generality that $d_i(p^r) < \varepsilon$ for all r with $i = s^r$ (cf. (24)). For every $r \in R$ let r' be the first index with $r' > r$ such that $i = s^{r'}$. Then during iterations $r, r+1, \cdots, r'-1$ node i is not chosen for relaxation while its deficit decreases from greater than 2ε to lower than ε. We claim that during these iterations the total deficit $\sum_{k\in N} |d_k(p)|$ is decreased by an amount of more than 2ε. To see this, note that the total absolute deficit cannot increase at any iteration as noted earlier in the proof of Proposition 2.2. Next observe that for any of the iterations $r, r+1, \cdots, r'-1$, say $\bar{r}$, for which the deficit of i is decreased by an amount $\xi > 0$ from a positive value $d_i(p^{\bar{r}}) > 0$, it must be that the node s chosen for relaxation is a neighbor of i and has a negative deficit $d_s(p^r) < 0$. Since all increase in $d_s(p^r)$ during the iteration must be matched by decreases of the deficits of the neighbor nodes of s, and the deficit of s will remain nonpositive after the iteration, it follows that the total absolute deficit will be decreased by at least $2 \min \{\xi, d_i(p^r)\}$ during the iteration. This shows that during iterations $r, r+1, \cdots, r'-1$ the total absolute deficit must decrease by more than 2ε. It follows that the set R of indices r for which $d_i(p^r) > 2\varepsilon$ cannot be infinite. Since $\varepsilon > 0$ is arbitrary we obtain $\limsup_{r\to\infty} d_i(p^r) \leqq 0$. Similarly we can show that $\liminf_{r\to\infty} d_i(p^r) \geqq 0$ and therefore $d_i(p^r) \to 0$.

(b) For all r and arcs j we have the CS condition

$$f_j^-(x_j^r) \leqq t_j^r \leqq f_j^+(x_j^r).$$

If Y is any cycle we have

$$\sum_{j \in Y^+} t_j^r - \sum_{j \in Y^-} t_j^r = 0,$$

so from (25) we obtain

$$\sum_{j \in Y^+} f_j^-(x_j^r) - \sum_{j \in Y^-} f_j^+(x_j^r) \leqq 0 \leqq \sum_{j \in Y^+} f_j^+(x_j^r) - \sum_{j \in Y^-} f_j^-(x_j^r). \tag{26}$$

Let $\{x^r\}_{r \in R}$ be a subsequence converging to some $\bar{x}$ (cf. Proposition 2.2). Then from (26) and the lower (upper) semicontinuity of $f_j^-(f_j^+)$, we have for all cycles Y

$$\sum_{j \in Y^+} f_j^-(\bar{x}_j) - \sum_{j \in Y^-} f_j^+(\bar{x}_j) \leqq 0 \leqq \sum_{j \in Y^+} f_j^+(\bar{x}_j) - \sum_{j \in Y^-} f_j^-(\bar{x}_j),$$

while from part (a) we have $\bar{x} \in C$. This implies that $\bar{x}$ is an optimal flow ([3, Chap. 8]) and therefore must be equal to the unique optimal flow x^*. Since, by Proposition 2.2, $\{x^r\}$ is bounded we obtain $x^r \to x^*$.

(c) For every arc j for which $l_j < c_j$ there are three possibilities:

(1) $\{t_j^r\}$ is bounded.

(2) $x_j^* = c_j < +\infty$, $x_j^r \leqq x_j^*$ and $-\infty < \liminf_{r \to \infty} t_j^r \leqq \limsup_{r \to \infty} t_j^r = +\infty$.

(3) $x_j^* = l_j > -\infty$, $x_j^r \geqq x_j^*$ and $-\infty = \liminf_{r \to \infty} t_j^r \leqq \limsup_{r \to \infty} t_j^r < +\infty$,

while for an arc j with $l_j = c_j$ we must have $x_j^* = x_j^r$ for all r. Using this fact we can easily see that we can construct a subsequence R such that

$$\sum_{j \in A} t_j^r(x_j^r - x_j^*) \leqq \sum_{j \in B} t_j^r(x_j^r - x_j^*) \quad \forall r \in R$$

where B is a set of arcs j such that $\{t_j^r\}_R$ is bounded. We have (since $t^r \in C^\perp$, $x^* \in C$, and therefore $\sum_{j \in A} t_j^r x_j^* = 0$)

$$f(x^r) + q(p^r) = \sum_{j \in A} t_j^r x_j^r = \sum_{j \in A} t_j^r(x_j^r - x_j^*) \leqq \sum_{j \in B} t_j^r(x_j^r - x_j^*).$$

Since $x_j^r \to x_j^*$ and $\{t_j^r\}_R$ is bounded for $j \in B$ we obtain by taking the limit above $f(x^*) + \lim_{r \to \infty} q(p^r) \leqq 0$. On the other hand we have for all p using (6) and (12) $f(x^*) + q(p) \geqq 0$. This together with the preceding relation show the desired result.

(d) By Proposition 2.3, $\{p^r\}$ is bounded. Let $\{p^r\}_{r \in R}$ be a subsequence converging to a vector p^* and let $t^* = E^T p^*$. We have for all $j \in A$

$$f_j^-(x_j^r) \leqq t_j^r \leqq f_j^+(x_j^r) \quad \forall r \in R.$$

It follows using part (b) and the lower (upper) semicontinuity of $f_j^-(f_j^+)$ that for all $j \in A$, $f_j^-(x_j^*) \leqq t_j^* \leqq f_j^+(x_j^*)$ where x^* is the optimal flow vector. Therefore t^* satisfies together with x^* the complementary slackness conditions and must be dual optimal. Proposition 2.3 shows that $\{p^r\}$ cannot have two different dual optimal price vectors as limit points and the conclusion follows. QED

3. The relaxation method for mixed linear and strictly convex arc costs. We first introduce some terminology. We will say that a point $b \in \operatorname{dom}(f_j)$ is a *breakpoint* of f_j if $f_j^-(b) < f_j^+(b)$. Note that the dual functional q, as given by (12), is separable and is piecewise either linear or strictly convex. Roughly speaking each linear piece (breakpoint) of the primal cost function f_j corresponds to a breakpoint (linear piece) of the dual cost function g_j (see Fig. 3.1).

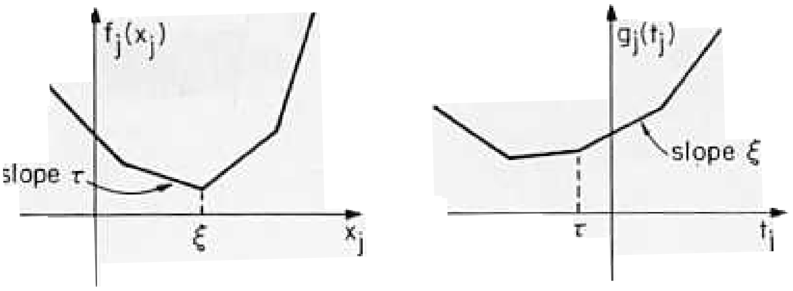

FIG. 3.1. *Correspondence between the breakpoints of f_j and the linear pieces of g_j (and vice versa).*

Assumption D. $f_j^+(x_j) > -\infty$ and $f_j^-(x_j) < +\infty$ for all $x_j \in \operatorname{dom}(f_j)$.

In the terminology of ([3, Chap. 8]), Assumption D implies that every feasible primal solution is regularly feasible and guarantees, together with Assumptions A and B, that the dual problem has an optimal solution ([3, p. 360]). For a given $\varepsilon > 0$, we say that $x \in R^{|A|}$ and $p \in R^{|N|}$ satisfy *ε-Complementary Slackness* (ε-CS for short) if

$$f_j^-(x_j) - \varepsilon \leqq t_j \leqq f_j^+(x_j) + \varepsilon \quad \forall j \in A \tag{27}$$

where $t = E^Tp$. For a given p, (27) defines upper and lower bounds, called *ε-bounds*, on the flow vector:

$$l_j^\varepsilon = \min\{\xi | f_j^+(\xi) \geqq t_j - \varepsilon\}, \qquad c_j^\varepsilon = \max\{\xi | f_j^-(\xi) \leqq t_j + \varepsilon\} \quad \forall j \in A. \tag{28}$$

Then x and p satisfying ε-CS is equivalent to

$$x_j \in [l_j^\varepsilon, c_j^\varepsilon] \quad \forall j \in A \tag{29}$$

where $t = E^Tp$. For a given t_j, we can obtain l_j^ε and c_j^ε from the graph of the subdifferential mapping of f_j as shown in Figs. 3.2–3.3. Intuition suggests that if x is in the

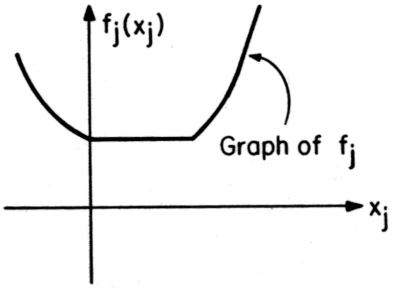

FIG. 3.2. *Graph of f_j.*

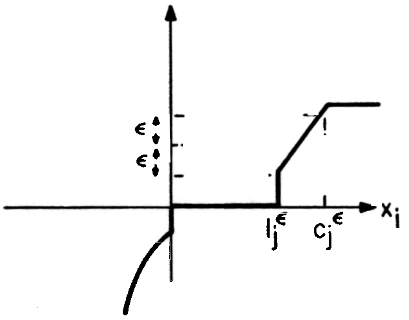

FIG. 3.3. *Graph of ∂f_j and ε-bounds corresponding to t_j.*

circulation subspace C, x and p satisfy ε-CS and ε is small, then both x and p should be near optimal. This idea will be made precise later when we explore the near optimality properties of the solution generated by a relaxation algorithm that uses the notion of ε-CS. The definition of ε-CS is related to the ε-subgradient idea introduced in nondifferentiable optimization in [13] as well as to the fortified descent method of Rockafellar [3]. The latter method, however, for a given p and $t = E^Tp$, uses different lower and upper bounds on x_j given by

$$\inf_{\Delta>0} \frac{g_j(t_j+\Delta)-g_j(t_j)+\varepsilon}{\Delta} \quad \text{and} \quad \sup_{\Delta>0} \frac{g_j(t_j)-g_j(t_j-\Delta)-\varepsilon}{\Delta}.$$

Our bounds of (28) seem simpler for implementation purposes particularly when some of the cost functions f_j are linear within their effective domain.

For a given x within the ε-bounds, we define the *deficit* of node i as in (2) and say that a sequence of nodes $\{n_1 \cdots n_k\}$ forms a *flow augmenting path* if

$$d_{n_1} < 0,\, d_{n_k} > 0 \quad \text{and} \quad \begin{cases} x_j < c_j^\varepsilon & \text{if } j \sim (n_m, n_{m+1}), \qquad m \in \{1, \cdots, k-1\}, \\ x_j > l_j^\varepsilon & \text{if } j \sim (n_{m+1}, n_m), \qquad m \in \{1, \cdots, k-1\}. \end{cases}$$

Let

$$\mu_m = \begin{cases} c_j^\varepsilon - x_j & \text{if } j \sim (n_m, n_{m+1}), \\ x_j - l_j^\varepsilon & \text{if } j \sim (n_{m+1}, n_m), \end{cases} \qquad m \in \{1, \quad , k \quad 1\}.$$

We will call

$$\mu = \min\{-d_{n_1}, d_{n_k}, \mu_1, \cdots, \mu_{k-1}\}$$

the *capacity* of the path. The relaxation algorithm of this section uses the labeling method of Ford and Fulkerson [14] for finding flow augmenting paths and for augmenting flow along them.

For a given tension vector $t \in C$ and any subset of nodes S, we define $C_\varepsilon(S, t)$ by

$$C_\varepsilon(S, t) = \sum_{j \in [S, N\backslash S]} l_j^\varepsilon - \sum_{j \in [N\backslash S, S]} c_j^\varepsilon \tag{30}$$

where we use the notation

$$[S, N\backslash S] = \{j \mid j \sim (i, k), i \in S, k \notin S\}, \qquad N\backslash S, S] = \{j \mid j \sim (i, k), i \notin S, k \in S\}.$$

We also define the $|N|$-vector $u(S)$ by

$$u_i(S) = \begin{cases} -1 & \text{if } i \in S, \\ 0 & \text{if } i \notin S. \end{cases}$$

The importance of these notions is due to the fact that for any $\varepsilon \geqq 0$, $C_\varepsilon(S, t) > 0$ implies that $u(S)$ is a dual descent direction at p, where p is any price vector satisfying $E^Tp = t$. This follows from the fact that the directional derivative of q at p in the direction $u(S)$ defined by

$$q'(p; u(S)) = \lim_{\Delta \downarrow 0} \frac{q(p + \Delta u(S)) - q(p)}{\Delta}$$

is easily verified to be

$$q'(p; u(S)) = \sum_{j \in [N\backslash S, S]} c_j^0 - \sum_{j \in [S, N\backslash S]} l_j^0 \leqq -C_\varepsilon(S, t)$$

where c_j^0, l_j^0 are the ε-bounds corresponding to $\varepsilon = 0$, and we are making use of the fact $c_j^0 \leqq c_j^\varepsilon$, $l_j^0 \geqq l_j^\varepsilon$, for all $\varepsilon \geqq 0$.

We now describe the relaxation algorithm. The algorithm is iterative and uses the ε-CS idea. The scalar ε is kept fixed throughout the algorithm. At the beginning of each iteration we have a dual price vector p and a flow vector x satisfying $l_j^\varepsilon \leqq x_j \leqq c_j^\varepsilon$ for all $j \in A$. If $x \in C$ then we terminate. Otherwise we use labeling to either find a flow augmenting path, in which case a flow augmentation is performed to bring x "closer" to C; or to find a dual descent direction, in which case a dual descent along this direction is performed. When each f_j is linear within its effective domain, $\varepsilon = 0$, and all problem data is integer, the algorithm coincides with the relaxation method of [1], [2]. When each f_j is strictly convex and $\varepsilon = 0$ the algorithm coincides with the exact line minimization version of the algorithm of the previous section ($\delta = 0$).

Relaxation iteration.

Step 0. Given p and x satisfying $l_j^\varepsilon \leqq x_j \leqq c_j^\varepsilon$ for all j, let t and d be the corresponding tension and deficit vectors.

Step 1. Pick a node s such that $d_s > 0$. If no such node exists terminate. Else set all nodes to be unlabeled and unscanned. Give the label 0 to node s. Set $S = \varnothing$ and go to Step 2.

Step 2. Choose a labeled but unscanned node k. Set $S \leftarrow S \cup \{k\}$ and go to Step 3.

Step 3. Scan the label of the node k as follows: Give the label k to all unlabeled nodes m such that $x_j < c_j^\varepsilon$ for $j \sim (m, k)$ and to all unlabeled nodes m such that $x_j > l_j^\varepsilon$ for $j \sim (k, m)$. If $C_\varepsilon(S, t) > 0$ then go to Step 5. Else if for any of the nodes m labeled from k we have $d_m < 0$ then go to Step 4. Else go to Step 2.

Step 4. (Flow Augmentation Step). A flow augmenting path has been found which starts at the node m (with $d_m < 0$) identified in Step 3 and ends at the node s. The path can be constructed by tracing labels starting from m. Let μ be the capacity of the path. Increase by μ the flow of all arcs on the path oriented in the direction from m to s, decrease by μ the flow of all other arcs on the path. Update the deficit vector d and return.

Step 5. (Dual Descent Step). Determine λ^* such that

$$q(p + \lambda^* u(S)) = \min_{\lambda > 0} \{q(p + \lambda u(S))\}.$$

Set $p \leftarrow p + \lambda^* u(S)$ and update the bounds l_j^ε and c_j^ε. Update x to maintain the ε-CS condition $l_j^\varepsilon \leqq x_j \leqq c_j^\varepsilon$ and return.

Validity and finite termination of the relaxation iteration. We will show that, under Assumption D, all steps in the Relaxation Iteration are executable and that the iteration terminates in a finite number of operations.

Steps 0, 1, and 3 are trivially executable. Step 2 is certainly executable on its first pass since the node s is labeled but unscanned. To show that it remains executable on subsequent passes we only need to verify that each time we go to Step 2 from Step 3 there always exists a labeled but unscanned node. In Step 2, if all labeled nodes are also scanned we have

$$\sum_{i \in S} d_i = \sum_{j \in [S, N \setminus S]} x_j - \sum_{j \in [N \setminus S, S]} x_j$$

$$= \sum_{j \in [S, N \setminus S]} l_j^\varepsilon - \sum_{j \in [N \setminus S, S]} c_j^\varepsilon = C_\varepsilon(S, t).$$

Since node s has positive deficit and all other labeled nodes have nonnegative deficits we obtain that $C_\varepsilon(S, t) > 0$ and therefore in the previous pass through Step 3 we would have branched to Step 5 rather than to Step 2. Step 4 is executable since the rule for labeling ensures that a flow augmenting path exists from node m to node s, so a flow augmentation is possible. Step 5 is executable since $C_\varepsilon(S, t) > 0$ implies that $u(S)$ is a dual descent direction at p, and we can show that there exists a minimizing stepsize λ^*. To see this assume the contrary, i.e., that there does not exist a stepsize λ^* achieving the minimum along the direction $u(S)$. In that case the convexity of q implies that

$$q'(p+\lambda u(S); u(S)) < 0 \quad \forall \lambda > 0,$$

$$\lim_{\lambda \to +\infty} q'(p+\lambda u(S); u(S)) \leqq 0.$$

Then it can be easily seen that either

$$\sum_{j \in [S, N \backslash S]} l_j - \sum_{j \in [N \backslash S, S]}$$

in which case Assumption A is violated [dom $(f) \cap C$ is empty], or

$$\sum_{j \in [S, N \backslash S]} l_j - \sum_{j \in [N \backslash S, S]} c_j = 0$$

and either $f_j^+(l_j) = -\infty$ for some $j \in [S, N/S]$ or $f_j^-(c_j) = +\infty$ for some $j \in [N \backslash S, S]$, in which case Assumption D is violated. To complete the proof that the relaxation iteration terminates in a finite number of operations we note that we cannot loop between Step 2 and Step 3 infinitely often since the number of scanned nodes is increased by one each time we visit Step 3.

We next show that the relaxation algorithm, when applied in conjunction with an easily implementable labeling rule, terminates in a finite number of iterations. The proof may be divided into two separate parts. The first part involves showing that the number of dual descent steps is not infinite. This is done by arguing that the optimal dual cost is necessarily $-\infty$ if the number of dual descent steps is infinite. The second part involves showing that the number of flow augmentations between successive dual descent steps is finite. This is done by choosing an appropriate labeling scheme for the relaxation algorithm and showing that the number of flow augmentations is finite under the chosen scheme. For this purpose we will propose two schemes: *breadth-first search* and *arc discrimination.*

We first show that the stepsize in each dual descent step is bounded from below by ε. Indeed our definition of ε-CS was motivated primarily by this fact.

PROPOSITION 3.1. *The stepsize in each dual descent step is greater than ε.*

Proof. Under Assumption B, $q(p)$ is subdifferentiable everywhere. Let S denote the subset of nodes corresponding to the dual descent direction generated by the relaxation iteration. In other words, the dual descent direction u is given by

$$u_i = \begin{cases} -1 & \text{if } i \in S, \\ 0 & \text{if } i \notin S, \end{cases} \tag{31}$$

and S satisfies $C_\varepsilon(S, t) > 0$. Now consider p' given by $p' = p + \varepsilon u$ and $t' = E^T p'$. Then

$$t_j' = t_j - \varepsilon \quad \text{if } j \in [S, N \backslash S],$$

$$t_j' = t_j + \varepsilon \quad \text{if } j \in [N \backslash S, S],$$

$$t_j' = t_j \qquad \text{otherwise,}$$

so that

$$l_j^\varepsilon = \min\{\xi \mid f_j^+(\xi) \geqq t_j - \varepsilon\} = \min\{\xi \mid f_j^+(\xi) \geqq t_j'\} \quad \text{for all } j \in [S, N\backslash S],$$

$$c_j^\varepsilon = \max\{\xi \mid f_j^-(\xi) \leqq t_j + \varepsilon\} = \max\{\xi \mid f_j^-(\xi) \leqq t_j'\} \quad \text{for all } j \in [N\backslash S, S].$$

Therefore

$$q'(p'; u) = -\sum_{j \in [S, N\backslash S]} l_j^\varepsilon + \sum_{j \in [N\backslash S, S]} c_j^\varepsilon = -C_\varepsilon(S, t) < 0.$$

Since q is convex, $q'(p; u) < 0$ and $q'(p + \varepsilon u; u) < 0$ imply that $q'(p + \alpha u; u) < 0$ for all $\alpha \in [0, \varepsilon]$. Therefore the stepsize in a dual descent step is greater than ε. QED

We will now use Proposition 3.1 to prove that the number of dual descent steps is necessarily finite. The following result is a first step in this direction.

PROPOSITION 3.2. *Let p^r denote the price vector generated by the relaxation algorithm just before the rth dual descent step. Then for each $r \in \{0, 1, 2, \cdots\}$*

$$q(p^r) - q(p^{r+1}) > \sum_{\substack{j \in [S^r, N\backslash S^r] \text{ or} \\ j \in [N\backslash S^r, S^r]}} [f_j(\Psi_j^r) - f_j(X_j^r) - (\Psi_j^r - X_j^r)t_j^r] \geqq 0 \tag{32}$$

where we define

$$\Psi_j^r = \begin{cases} g_j^+(t_j^r - \varepsilon) & \text{if } j \in [S^r, N\backslash S^r], \\ g_j^-(t_j^r + \varepsilon) & \text{if } j \in [N\backslash S^r, S^r], \end{cases} \qquad X_j^r = \begin{cases} g_j^-(t_j^r) & \text{if } j \in [S^r, N\backslash S^r], \\ g_j^+(t_j^r) & \text{if } j \in [N\backslash S^r, S^r], \end{cases}$$

and S^r denotes the node subset corresponding to the descent direction at the rth dual descent step.

Proof. From the definition of Ψ_j^r and X_j^r we have that

$$g_j(t_j^r) = X_j^r t_j^r - f_j(X_j^r), \qquad g_j(t_j^r - \varepsilon) = \Psi_j^r(t_j^r - \varepsilon) - f_j(\Psi_j^r) \quad \forall j \in [S^r, N\backslash S^r],$$

$$g_j(t_j^r) = X_j^r t_j^r - f_j(X_j^r), \qquad g_j(t_j^r + \varepsilon) = \Psi_j^r(t_j^r + \varepsilon) - f_j(\Psi_j^r) \quad \forall j \in [N\backslash S^r, S^r].$$

From the definition of q, S^r and $u(S^r)$ we have that

$$q(p^r + \varepsilon u(S^r)) = q(p^r) + \sum_{j \in [S^r, N\backslash S^r]} [g_j(t_j^r - \varepsilon) - g_j(t_j^r)]$$
$$+ \sum_{j \in [N\backslash S^r, S^r]} [g_j(t_j^r + \varepsilon) - g_j(t_j^r)]$$

and from Proposition 3.1 we have that

$$q(p^r) - q(p^{r+1}) \geqq q(p^r) - q(p^r + \varepsilon u(S^r)).$$

Combining the above three sets of equalities and inequalities we obtain that

$$q(p^r) - q(p^{r+1}) \geqq \sum_{j \in [S^r, N\backslash S^r]} [[X_j^r t_j^r - f_j(X_j^r)] - [(t_j^r - \varepsilon)\Psi_j^r - f_j(\Psi_j^r)]]$$
$$+ \sum_{j \in [N\backslash S^r, S^r]} [[X_j^r t_j^r - f_j(X_j^r)] - [(t_j^r + \varepsilon)\Psi_j^r - f_j(\Psi_j^r)]]$$
$$= \sum_{\substack{j \in [S^r, N\backslash S^r] \text{ or} \\ j \in [N\backslash S^r, S^r]}} [f_j(\Psi_j^r) - f_j(X_j^r) - (\Psi_j^r - X_j^r)t_j^r]$$
$$+ \varepsilon\left[\sum_{j \in [S^r, N\backslash S^r]} \Psi_j^r - \sum_{j \in [N\backslash S^r, S^r]} \Psi_j^r\right]$$

Since

$$\sum_{j\in[S^r,N\setminus S^r]}\Psi_j^r-\sum_{j\in[N\setminus S^r,S^r]}\Psi_j^r=\sum_{j\in[S^r,N\setminus S^r]}g_j^+(t_j^r-\varepsilon)-\sum_{j\in[N\setminus S^r,S^r]}g_j^-(t_j^r+\varepsilon)$$
$$\geqq\sum_{j\in[S^r,N\setminus S^r]}g_j^-(t_j^r-\varepsilon)-\sum_{j\in[N\setminus S^r,S^r]}g_j^+(t_j^r+\varepsilon)$$
$$=-q'(p^r+\varepsilon u(S^r);\,u(S^r))>0$$

(where the last strict inequality is obtained from Proposition 3.1) the left side of (32) follows. The right side of (32) follows from the convexity of f_j. QED

PROPOSITION 3.3. *Under Assumption* D *the number of dual descent steps is finite.*

Proof. We will argue by contradiction. Suppose that the number of dual descent steps is infinite. We denote the price vector, the tension vector and the flow vector generated by the relaxation algorithm at the rth dual descent step by p^r, t^r, and x^r, respectively. First we show the following property of the sequence $\{t^r\}$:

For each j

(33) $\{t_j^r\}_R\to+\infty$ for some subsequence $R\Rightarrow c_j<+\infty,\quad f_j(c_j)<\infty,$

(34) $\{t_j^r\}_R\to-\infty$ for some subsequence $R\Rightarrow l_j>-\infty,\quad f_j(l_j)>-\infty.$

If $\{t^r\}$ is bounded then (33), (34) trivially hold. Consider a subsequence R such that $\{t^r\}_R$ is unbounded. Without loss of generality suppose that, for each arc $j\in A$, $\{t_j^r\}$ is either bounded, or tends to ∞, or tends to $-\infty$. We now partition N into a collection of nonempty subsets $N_0, N_1,\cdots,N_L$ $(L\geqq 1)$ such that

$$\{(p_i^r-p_k^r)\}\to\infty\quad\text{if } \alpha>\beta \text{ and } i\in N_\alpha,\ k\in N_\beta.$$

(One way to construct such a collection is to consider a graph identical to the original except that all arcs j such that $\{t_j^r\}_R$ is bounded are discarded, and all arcs j such that $\{t_j^r\}_R\to-\infty$ are reversed in their orientation. Since the sum of tensions along a directed cycle is zero we see that this graph is acyclic. The set N_0 is the set of nodes of this acyclic graph having no outgoing arcs. The set N_1 is obtained similarly after all arcs incident to N_0 in the acyclic graph have been discarded, etc.)

For $a=1,2,\cdots,L$, we define the following arc sets:

$$A_a^+=\left\{j\sim(i,k)\,\middle|\, i\in\bigcup_{\tau\geqq a}N_\tau,\ k\in\bigcup_{\tau<a}N_\tau\right\},$$
$$A_a^-=\left\{j\sim(i,k)\,\middle|\, i\in\bigcup_{\tau<a}N_\tau,\ k\in\bigcup_{\tau\geqq a}N_\tau\right\}.$$

Then each set $A_a^+\cup A_a^-$ is a cut in the network and

$$t_j^r\to+\infty,\ r\in R\quad\text{if and only if } j \text{ belongs to some } A_a^+,$$
$$t_j^r\to-\infty,\ r\in R\quad\text{if and only if } j \text{ belongs to some } A_a^-.$$

Consider any fixed positive scalar Δ. Equation (35) implies that for all a

$$\lim_{r\to\infty,r\in R}g_j^-(t_j^r-\Delta)=c_j\quad\forall j\in A_a^+,\qquad \lim_{r\to\infty,r\in R}g_j^+(t_j^r+\Delta)=l_j\quad\forall j\in A_a^-.$$

$$q'(p^r+\Delta u;\,u)=-\sum_{j\in A_a^+}g_j^-(t_j^r-\Delta)+\sum_{j\in A_a^-}g_j^+(t_j^r+\Delta)$$

where u is given by

$$u_i = \begin{cases} & \text{if } i \in \bigcup_{\tau \geqq a} N_\tau, \\ 0 & \text{otherwise,} \end{cases}$$

it follows from (36) that

$$\lim_{r\to\infty, r\in R} q'(p^r + \Delta u; u) = -\sum_{j\in A_a^+} c_j + \sum_{j\in A_a^-} l_j. \tag{37}$$

Let Θ denote the right-hand side quantity in (37). We will argue that $\Theta = 0$. Clearly we cannot have $\Theta > 0$ since this would imply that there does not exist a primal feasible solution. We also cannot have $\Theta < 0$ since then (37) implies that for r sufficiently large

$$q(p^r + \Delta u) \leqq q(p^r) + \Delta\Theta.$$

This is not possible since Δ can be chosen arbitrarily large while $q(p^r)$ is nonincreasing with r. This leaves the only possibility that $\Theta = 0$ or that

$$\sum_{j\in A_a^+} c_j = \sum_{j\in A_a^-} l_j, \qquad a = 1, \cdots, L.$$

It follows that for every feasible flow vector we have

$$x_j = c_j \quad \forall j \in A_a^+, \quad x_j = l_j \quad \forall j \in A_a^-, \quad a = 1, \cdots, L.$$

This implies (33) and (34).

Now we will bound from below the amount of improvement in the dual cost per dual descent step by a positive constant. Proposition 3.1 assures us that at each dual descent step the step length is more than ε. Consider the interval $[\frac{1}{4}\varepsilon, \frac{3}{4}\varepsilon]$, which we denote by I. Also let u^r denote the dual descent direction at the rth dual descent step. We have that the dual cost is decreasing on the line segment connecting t^r and t^{r+1}. It follows from (33), (34) and Assumption D that there exists a subsequence R such that for r sufficiently large, $r \in R$, we have for all $\Delta \in I$

$$\begin{aligned} q'(p^r + \Delta u^r; u^r) = \sum_{j\in J^+} c_j v_j^r + \sum_{j\in J^-} l_j v_j^r + \sum_{\substack{j\in J^0 \\ v_j^r>0}} g_j^+(t_j^r + \Delta v_j^r) v_j^r \\ + \sum_{\substack{j\in J^0 \\ v_j^r<0}} g_j^-(t_j^r + \Delta v_j^r) v_j^r < 0 \end{aligned}$$

where we define

$$J^+ = \{j | \{t_j^r\}_{r\in R} \to \infty\}, \qquad J^- = \{j | \{t_j^r\}_{r\in R} \to -\infty\},$$
$$J^0 = \{j | \{t_j^r\}_{r\in R} \text{ is bounded}\}$$

and $v^r = E^T u^r$. Consider a fixed $r \in R$. Define $\Theta : R \to R$ by

$$\Theta(\Delta) = q(p^r + \Delta u^r).$$

We consider two cases. In case (i) the right derivative of $\Theta(\Delta)$ assumes at most $2|A|$ distinct values in the interval I. In case (ii) the right derivative of $\Theta(\Delta)$ assumes more than $2|A|$ distinct values in the interval I. In case (i) $q(p^r + \Delta u^r)$ is linear for Δ in some subinterval I^r of I of length at least $\varepsilon/4|A|$ and it follows that $q'(p^r + \Delta u^r; u^r)$ over I^r is linear of the form

$$q'(p^r + \Delta u^r; u^r) = \sum_{j\in J^+} c_j v_j^r + \sum_{j\in J^-} l_j v_j^r + \sum_{j\in J^0} b_j v_j^r \tag{38}$$

where $v^r = E^T u^r$ and b_j denotes some breakpoint of f_j. This implies that, for each $j \in J^0$ such that $v_j^r \neq 0$, the dual functional $g_j(t_j^r + \Delta v_j^r)$ is linear with slope b_j for Δ in I^r. For each $j \in J^0$, $\{t_j^r\}_{r \in R}$ is bounded, and therefore the number of distinct linear pieces of g_j of length $\geqq \varepsilon/4|A|$ encountered during the course of the algorithm is finite. This together with the fact that v^r is chosen from a finite set imply that $q'(p^r + \Delta u^r; u^r)$ (cf. (38)) can only assume one of a finite set of values over the subinterval I^r. It follows that in case (i) we can bound the amount of dual cost improvement from below by $\delta\varepsilon/4|A|$ where δ is some positive scalar. This implies that case (i) can occur for only a finite set of indexes r (for otherwise the dual cost tends to $-\infty$) and we need only to consider case (ii). In case (ii) for each $r \in R$ there must exist a $j \in J^0$ such that $v_j^r \neq 0$ and the right derivative of the function $h(\Delta)$ defined by $h(\Delta) = g_j(t_j^r + \Delta v_j^r)$ assumes at least three distinct values in the interval I. Since v_j^r equals either 1 or -1 it follows that either $t_j^{r+1} \geqq t_j^r + \varepsilon$ and $g_j^+(t_j^r + \Delta_1) < g_j^-(t_j^r + \Delta_2)$ for at least two points $\Delta_1 < \Delta_2$ in I or $t_j^{r+1} \leqq t_j^r - \varepsilon$ and $g_j^+(t_j^r - \Delta_2) < g_j^-(t_j^r - \Delta_1)$ for least two points $\Delta_1 < \Delta_2$ in I. Passing to a subsequence if necessary, we can assume that it is the same j and either $t_j^{r+1} \geqq t_j^r + \varepsilon$ or $t_j^{r+1} \leqq t_j^r - \varepsilon$ for all $r \in R$ that are sufficiently large. Without loss of generality we will assume that $t_j^{r+1} \geqq t_j^r + \varepsilon$ for all $r \in R$ that are sufficiently large. Since $j \in J^0$ the subsequence $\{t_j^r\}_{r \in R}$ is bounded and therefore has a limit point t_j^*. Passing to a subsequence if necessary, we assume that $\{t_j^r\}$ converges to t_j^*. Then it follows that there exists a fixed interval L such that

(39) $$L \subset [t_j^r, t_j^r + \varepsilon] \quad \forall r \in R,\ r \text{ sufficiently large}$$

$$\eta_1 < \eta_2 \quad \text{and} \quad g_j^+(\eta_1) < g_j^-(\eta_2)$$

for at least two distinct points η_1 and η_2 in L. We then define

$$\xi_1 = g_j^-(\eta_1), \qquad \xi_2 = g_j^+(\eta_2).$$

Then ξ_1 and ξ_2 belong to the interval

$$[g_j^-(a), g_j^+(b)]$$

where a, b are the left and the right endpoints of L, respectively, and they satisfy

(40) $$\xi_1 < \xi_2 \quad \text{and} \quad f_j^+(\xi_1) < f_j^-(\xi_2).$$

Then for r sufficiently large, $r \in R$, we obtain (cf. (39)) that

(41) $$g_j^+(t_j^r) \leqq \xi_1 < \xi_2 \leqq g_j^-(t_j^r + \varepsilon).$$

It follows from Proposition 3.2 that for all sufficiently large $r \in R$

(42) $$\begin{aligned} q(p^r) - q(p^{r+1}) &\geqq f_j(g_j^-(t_j^r + \varepsilon)) - f_j(g_j^+(t_j^r)) - f_j^+(g_j^+(t_j^r))(g_j^-(t_j^r + \varepsilon) - g_j^+(t_j^r)) \\ &\geqq f_j(\xi_2) - f_j(\xi_1) - f_j^+(\xi_1)(\xi_2 - \xi_1) \end{aligned}$$

where the second inequality follows from (41) and the convexity of f_j. From (40) and the convexity of f_j we obtain that the right-hand side of (42) is positive. Therefore the dual cost improvement per dual descent is bounded from below by a positive constant, and the dual cost tends to $-\infty$, contradicting Assumption A. QED

The second part of our finite termination proof involves showing that the number of flow augmentations between successive dual descent steps is finite. Since the ε-bounds remain unchanged between successive dual descent steps, the issue in effect is whether the labeling algorithm used will solve finitely the max flow problem with the given ε-bounds taken as capacity constraints. It was shown by Ford and Fulkerson

([14, p. 126]) that, when the data is irrational, an arbitrary choice of labeled nodes may result in an infinite number of flow augmentations, so a more specific scheme for labeling is necessary to deal with irrational data. Here we propose two such schemes: breadth-first search and arc discrimination. In practice, the data is always rational, being stored on a finite precision machine, and therefore finite convergence is assured even if labeling is done arbitrarily.

Breadth-first search is a well-known scheme used in labeling. It can be easily implemented using a FIFO queue. In [15, Chap. 9.3, 16] it was shown that, under breadth-first search, the number of flow augmentations is finite if *all* nodes with positive deficit are labeled initially. We now show that the same conclusion holds if a *single* node with positive deficit is labeled initially, as is the case for the relaxation iteration. This fact requires a nontrivial proof and, to our knowledge, is not reported in the literature.

PROPOSITION 3.4. *When labeling is done by breadth-first search, the number of flow augmentations between successive dual descent steps is finite.*

Proof. We will assume that the number of flow augmentations is infinite and obtain a contradiction. For simplicity, we call a node with negative deficit a *source* and a node with positive deficit a *sink.* Since the number of flow augmentations is infinite, after a while the set of sources and sinks must become fixed (since a source cannot become a sink or vice versa) and the set of flow augmenting paths must repeat (since all flow augmenting paths are simple and therefore there are only a finite number of them). Let P be the set of flow augmenting paths that repeat infinitely often. We say that an arc belonging to a path $p \in P$ is *saturated in the direction of* p if the flow of the arc is at the upper (lower) bound and the arc is oriented from the source (sink) of p to the sink (source) of p.

Consider a path $p \in P$. After a flow augmentation using p as the path, some arc of p will become saturated in the direction of p. Let A_p denote the set of arcs on p that become saturated in the direction of p infinitely often. A_p is clearly nonempty. We will show, by induction, that A_p is empty when breadth-first search is done and thus obtain a contradiction.

Initialization. For all $p \in P$, every $a \in A_p$ is at least one arc away from the sink t of p.

Proof. This is true since if the arc on p incident to t is saturated, it must remain saturated from then on.

kth inductive step. Suppose that, for all $p \in P$, every $a \in A_p$ is at least k arcs away from the sink of p. We will show that, for all $p \in P$, every $a \in A_p$, is $k+1$ arcs away from the sink of p. Suppose the contrary. Then there exists a $p \in P$, whose source and sink we denote by s and t respectively, and an arc a in A_p such that a is k arcs away from t. After a becomes saturated, there must be a flow augmenting path p' to unsaturate it (see Fig. 3.4). From the inductive hypothesis, the arcs on p between a and t are unsaturated in the direction of p. Since the labeling is done by breadth-first search,

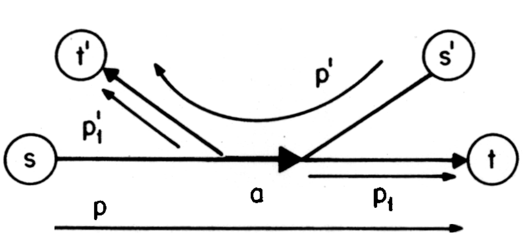

FIG. 3.4. *p' unsaturating the arc a.*

this implies that the number of arcs on the subpath p'_1 (see Fig. 3.4) must be *strictly* less than that of the subpath p_1 (otherwise during the iteration that generated p' as the flow augmenting path node t would have been labeled before node t'). It follows that, just before the iteration that generated path p, some arc of the subpath p'_1 must be saturated in the direction of p' (otherwise during the iteration that generated p node t' would be labeled before node t). This arc must then belong to $A_{p'}$. Since the number of arcs on p'_1 is strictly less than k, the inductive hypothesis is contradicted.

Since the inductive hypothesis holds for all k and the number of arcs on each flow augmenting path is at most $|N|-1$, it follows that A_p is empty for all p and the desired contradiction is obtained. QED

In the arc discrimination scheme, the order in which nodes are labeled and scanned is given by the following simple rule:

> Each labeled but unscanned node records whether it is connected to an unlabeled neighbor by an arc whose flow is strictly between the lower and upper bounds. A node with such a neighbor is scanned first.

The proof of finite convergence under this scheme is given in [17]. The implementation of the arc discrimination scheme requires more global information than breadth-first search. However, when the relaxation algorithm is extended to operate on both positive and negative deficit nodes between successive dual descent steps, which had been shown to be computationaly beneficial in the case of linear cost problems, arc discrimination can still be shown to yield finite convergence. It is not known if this is also true of breadth-first search.

Propositions 3.3 and 3.4 show that the relaxation algorithm of this section terminates after a finite number of iterations. Since the algorithm only terminates when all the node deficits have zero value, the final flow vector x must belong to C. Since ε-CS is maintained at all iterations of the algorithm, it follows that x and the final dual price vector must satisfy ε-CS also.

We next show that we can bring the cost of the solution generated by the relaxation algorithm arbitrarily close to the optimal cost by taking ε sufficiently small. The main part of the argument is embodied in the next proposition.

PROPOSITION 3.5. *Let x and p satisfy ε-CS, and let ξ and p satisfy CS. If $x \in C$ then*

$$0 \leqq f(x) + q(p) \leqq \varepsilon \sum_{j \in A} |x_j - \xi_j|.$$

Proof. Let $t = E^T p$. Since ξ and p satisfy CS we have

$$f_j(\xi_j) = \xi_j t_j - g_j(t_j) \quad \forall j \in A.$$

Take an arc j such that $x_j \geqq \xi_j$. Then by convexity of f_j

$$f_j(x_j) + (\xi_j - x_j) f_j^-(x_j) \leqq f_j(\xi_j) = \xi_j t_j - g_j(t_j).$$

Hence

$$\begin{aligned} f_j(x_j) + g_j(t_j) &\leqq (x_j - \xi_j)(f_j^-(x_j) - t_j) + x_j t_j \\ &\leqq |x_j - \xi_j| \varepsilon + x_j t_j \end{aligned}$$

where the second inequality follows from ε-CS. This inequality is similarly obtained when $x_j \leqq \xi_j$, so we have

$$f_j(x_j) + g_j(t_j) \leqq |x_j - \xi_j| \varepsilon + x_j t_j \quad \forall j \in A.$$

From the definition of g_j we also have

$$x_j t_j \leqq f_j(x_j) + g_j(t_j) \quad \forall j \in A.$$

By combining these two inequalities and adding over all $j \in A$, we obtain

$$\sum_{j\in A} x_j t_j \leqq \sum_{j\in A} [f_j(x_j)+g_j(t_j)] \leqq \varepsilon \sum_{j\in A} |x_j-\xi_j| + \sum_{j\in A} x_j t_j.$$

Since $x \in C$ we have $\sum_{j\in A} x_j t_j = 0$ and the result follows. QED

From Proposition 3.5 we can obtain a simple bound on the suboptimality of the solution in the special case where $l_j > -\infty$ and $c_j < +\infty$ for all $j \in A$.

COROLLARY 3.1. *Let x and p satisfy ε-CS. If $x \in C$, and $-\infty < l_j \leqq c_j < +\infty$ for all $j \in A$, then*

$$0 \leqq f(x) + q(p) \leqq \varepsilon \sum_{j\in A} (c_j - l_j).$$

For the general case we have the following.

PROPOSITION 3.6. *Let $x(\varepsilon)$ and $p(\varepsilon)$ denote any flow and price vector pair such that $x(\varepsilon)$ and $p(\varepsilon)$ satisfy ε-CS and $x(\varepsilon) \in C$. Then $f(x(\varepsilon)) + q(p(\varepsilon)) \to 0$ as $\varepsilon \to 0$.*

Proof. First we show that $x(\varepsilon)$ remains bounded as $\varepsilon \to 0$. If $x(\varepsilon)$ is not bounded as $\varepsilon \to 0$, then since $x(\varepsilon) \in C$ for all $\varepsilon > 0$ there exists a directed cycle Y and a sequence $\{\varepsilon_n\} \to 0$ such that $c_j = +\infty$, $x_j(\varepsilon_n) \to +\infty$ for all $j \in Y^+$ and $l_j = -\infty$, $x_j(\varepsilon_n) \to -\infty$ for all $j \in Y^-$. By Assumption B

$$\lim_{\xi\to+\infty} f_j^-(\xi) = +\infty \quad \text{for all } j \in Y^+, \qquad \lim_{\xi\to-\infty} f_j^+(\xi) = -\infty \quad \text{for all } j \in Y^-.$$

This implies that for n sufficiently large,

$$t_j(\varepsilon_n) > t_j(\varepsilon_0) \quad \text{for all } j \in Y^+ \quad \text{and} \quad t_j(\varepsilon_n) < t_j(\varepsilon_0) \quad \text{for all } j \in Y^- \tag{43}$$

where $t(\varepsilon_n) = E^T p(\varepsilon_n)$. Since $t(\varepsilon_n) = E^T p(\varepsilon_n)$ we have

$$\sum_{j\in Y^+} t_j(\varepsilon_n) - \sum_{j\in Y^-} t_j(\varepsilon_n) = 0 \quad \text{for all } n,$$

which contradicts (43). Therefore $x(\varepsilon)$ is bounded as $\varepsilon \to 0$.

Now we will show that $\xi_j(\varepsilon) - x_j(\varepsilon)$ is bounded for all $j \in A$ as $\varepsilon \to 0$, where $\xi(\varepsilon)$ is some vector satisfying $f_j^-(\xi_j(\varepsilon)) \leqq t_j(\varepsilon) \leqq f_j^+(\xi_j(\varepsilon))$, for all $j \in A$. If $c_j < \infty$ then $\xi_j(\varepsilon)$ is trivially bounded from above. If $c_j = +\infty$ then by Assumption B we have $f_j^-(\xi) \to +\infty$ as $\xi \to +\infty$. Since $x_j(\varepsilon)$ is bounded we have that $t_j(\varepsilon)$ is bounded from above which in turn implies that $\xi_j(\varepsilon)$ is bounded from above. Similarly, we can argue that $\xi_j(\varepsilon)$ is bounded from below. Therefore $|\xi_j(\varepsilon) - x_j(\varepsilon)|$ is bounded for all $j \in A$ as $\varepsilon \to 0$. This then completes our proof in view of Proposition 3.5. QED

Unfortunately Proposition 3.6 does not tell us how small ε must be to achieve a certain degree of near optimality. We need to solve the problem first for some guess ε to obtain $x(\varepsilon)$ and $\xi(\varepsilon)$, evaluate the quality of the solution on the basis of the gap $f(x(\varepsilon)) + q(p(\varepsilon))$ between primal and dual solution, and then decide whether ε needs to be decreased. If however the bounds l_j and c_j are finite, we can, by Corollary 3.1, obtain an a priori estimate on ε.

4. Computational experimentation.

Two experimental codes implementing the methods of the paper were developed and tested on linear benchmark problems and nonlinear variations.

The first code, named NRELAX, implements the relaxation method for strictly convex problems of § 2. The second code, named MNRELAX, implements the method for mixed linear and strictly convex problems of § 3. Both codes were written in Fortran on a VAX 11-750 and were compiled and run under the VMS version 3.7 operating system.

TABLE 1

Times for NETGEN *benchmark problems.* MNRELAX *uses* $\varepsilon = 0$. *Times in this and subsequent tables are in secs on a* VAX 11-750. *All codes are written in Fortran and compiled under* VMS *version* 3.7.

Problem number	Number of nodes	Number of arcs	MNRELAX $\varepsilon = 0$	RELAX-II
	200	1300	5.13	2.07
	200	1500	6.33	2.12
	200	2000	4.86	1.92
	200	2200	7.74	2.52
	200	2900	6.83	2.97
	300	3150	12.85	4.37
7	300	4500	13.46	5.46
8	300	5155	14.54	5.39
9	300	6075	17.38	6.38
10	300	6300	14.39	4.12
	400	1500	4.71	1.23
12	400	2250	5.81	1.38
13	400	3000	6.27	1.68
14	400	3750	7.79	2.43
15	400	4500	9.64	2.79
16	400	1306	9.06	2.79
17	400	2443	8.87	2.67
18	400	1306	8.98	2.56
19	400	2443	8.81	2.73
20	400	1416	9.82	2.85
21	400	2836	10.36	3.80
22	400	1416	9.08	2.56
23	400	2836	13.80	4.91
24	400	1382	4.73	1.27
25	400	2676	7.15	2.01
26	400	1382	3.73	1.79
27	400	2676	6.41	2.15
28	1000	2900	20.17	4.90
29	1000	3400	19.15	5.57
30	1000	4400	25.62	7.31

The test problems were generated using the public domain code NETGEN [18]. There are 40 "standard" benchmark linear cost problems that can be obtained using this code. We tested our codes with some of these problems either in their standard (linear cost) form or in a modified form whereby a quadratic cost was added to the linear cost of some or all of the arcs as discussed below. In order to test coding efficiency we tested MNRELAX with $\varepsilon = 0$ against the very efficient linear cost code RELAX-II (see [1], [2]) under identical conditions on the first 30 NETGEN benchmark problems. The two codes are close to being mathematically equivalent on linear cost problems but MNRELAX uses floating point arithmetic. The results shown in Table 1 appear to indicate that MNRELAX is coded fairly efficiently.

There were two issues that we wanted to clarify through the experimentation:

(a) The effect of the parameter ε on the performance of MNRELAX;

(b) The relative efficiency of NRELAX versus MNRELAX with optimal choice of the parameter ε on strictly convex problems.

A large number of experiments some of which are presented in Tables 2 and 3 showed that for all except some very "difficult" problems it is best to operate

TABLE 2

Times for NETGEN *benchmark problems modified so that 50% of the arcs have an additional quadratic cost with coefficient from the range* [5, 10]. *Numbers in parentheses where present indicate significant digits of accuracy of the answer. In* MNRELAX *ε is kept constant during the solution of each problem.*

Problem number	Number of nodes	Number of arcs	MNRELAX $\varepsilon > 0$	MNRELAX $\varepsilon = 0$	Sign. digits of accuracy
1	200	1300	30.79	11.49	6
2	200	1500	40.66	11.73	6
3	200	2000	34.48	9.31	7
4	200	2200	32.05	11.60	6
5	200	2900	50.43	28.14	5
6	300	3150	74.10	26.01	6
7	300	4500	140.70	48.64	6
8	300	5155	116.06	76.35	6
9	300	6075	96.59	49.25	6
10	300	6300	94.71	36.43	6
11	400	1500	263.14	26.35	4
12	400	2250	180.93	31.86	4
13	400	3000	240.76	(5)36.25	4
14	400	3750	436.80	79.88	4
15	400	4500	146.69	(3)42.23	2
16	400	1306	144.08	(7)86.40	3
17	400	2443	261.91	(7)47.31	5
18	400	1306	294.88	53.71	4
19	400	2443	108.04	37.54	5
20	400	1416	214.99	(7)68.17	3
21	400	2836	37.24	(7)18.43	4
22	400	1416	366.85	(7)53.37	4
23	400	2836	34.56	(7)18.09	4
24	400	1382	66.58	(5)45.87	3
[illegible]	400	2676	167.53	(6)22.73	5

MNRELAX with $\varepsilon = 0$ and terminate the iterations when the deficit of all nodes becomes sufficiently close to zero. Indeed it appears that for such problems the time required for MNRELAX to terminate increases with ε. The reason is probably that with large ε the intervals defined by the ε-bounds become larger and, as a result, a large number of flow augmentations are needed before a descent direction can be found. Given that a large value of ε leads also to inaccurate solutions (see Proposition 3.5), it appears that for most problems the best way to operate MNRELAX is with $\varepsilon = 0$ or with ε very small.

When $\varepsilon = 0$ and all arc costs are strictly convex, MNRELAX and NRELAX are mathematically equivalent. However NRELAX is somewhat faster because of more efficient coding as shown in Table 3.

Finally in Table 4 we show results obtained on some "difficult" problems with strictly convex arc costs. These problems were constructed by choosing the quadratic cost efficients of some arcs to be very small relative to others as described in Table 4. This is similar to a situation in nonlinear unconstrained minimization where the Hessian matrix of the cost function has some eigenvalues that are very small relative to other eigenvalues. For this class of problems MNRELAX with nonzero ε can outperform both NRELAX and MNRELAX with $\varepsilon = 0$. This is not surprising in view of the coordinate descent interpretation of NRELAX. The version of MNRELAX that we found most efficient for these problems is one whereby we start with a moderate value

TABLE 3

Times for NETGEN *benchmark problems modified so that all arcs have an additional quadratic cost with coefficient from the range* [5, 10]. *Numbers in parentheses where present indicate significant digits of accuracy of the answer. In* MNRELAX ε *is kept constant during solution of each problem.*

Problem number	Number of nodes	Number of arcs	MNRELAX $\varepsilon > 0$	MNRELAX $\varepsilon = 0$	NRELAX	Sign. digits of accuracy
	200	1300	22.63	17.97	10.80	7
2	200	1500	23.37	19.12	11.51	7
3	200	2000	19.33	20.77	12.04	8
4	200	2200	37.87	25.38	17.22	7
5	200	2900	34.82	32.37	21.44	6
6	300	3150	113.75	57.95	40.23	7
7	300	4500	85.26	50.49	37.77	6
8	300	5155	95.11	70.08	49.38	8
9	300	6075	70.48	69.44	48.04	7
10	300	6300	(6)99.69	69.33	41.41	5
11	400	1500	(7)43.19	34.37	14.67	6
12	400	2250	(6)39.56	33.31	12.98	5
13	400	3000	(7)34.62	32.66	18.34	5
14	400	3750	(6)34.97	35.32	20.86	5
15	400	4500	64.90	42.53	24.95	5
16	400	1306	65.86	54.19	21.54	6
17	400	2443	60.62	46.20	21.89	6
18	400	1306	84.26	72.41	48.97	7
19	400	2443	60.56	46.18	20.80	6
20	400	1416	108.49	72.11	38.70	7
21	400	2836	62.78	38.79	38.69	
22	400	1416	95.91	55.25	42.03	
23	400	2836	43.41	33.21	20.70	
	400	1382	59.83	65.35	42.47	
25	400	2676	53.57	42.06	37.88	

of ε, operate MNRELAX to termination, then reduce ε by a factor of 10 and repeat the process up to the point where primal and dual values differ by a specified accuracy. Still, the proper starting value for ε was not easy to determine and it was necessary to do some initial experimentation with several of these difficult problems. The conclusion is that the methods of this paper may not be successful for such problems. We do not know, however, of a better alternative.

TABLE 4

Times for NETGEN *benchmark problems modified so that all arcs have an additional quadratic term. In* 50% *of the arcs the quadratic cost coefficient was small as indicated. In the other* 50% *of the arcs the quadratic cost coefficient was from the range* [5, 10]. *Numbers in parentheses where present indicate significant digits of accuracy of the answer. In* MNRELAX ε *is progressively decreased during solution of each problem.*

Problem number			Small quad coeff.	MNRELAX $\varepsilon > 0$		NRELAX
	200	1300	.0001	(4)52.98		(3)58.00
	200	2900	.001	(5)227.93	(2)27.16	(2)50.15
	300	4500	.0001	(3)301.03	(2)131.10	(2)218.24
	400	1500	.0001	(4)111.36		—
15	400	4500	.01	(3)361.71		(3)1957.70
16	400	1306	.001	(3)287.13		(2)350.97
18	400	1306	.001	(3)188.78		(2)321.76
24	400	1382	.001	(4)417.83	(2)46.21	(2)134.51

REFERENCES

[1] D. P. BERTSEKAS, *A unified framework for primal-dual methods in minimum cost network flow problems*, Math. Programming, 32 (1985), pp. 125-145.

[2] D. P. BERTSEKAS AND P. TSENG, *Relaxation methods for minimum cost ordinary and generalized network flow problems*, LIDS Report P-1462, Massachusetts Inst. of Technology, May 1985.

[3] R. T. ROCKAFELLAR, *Network Flows and Monotropic Programming*, Wiley-Interscience, New York, 1983.

[4] ———, *Convex Analysis*, Princeton Univ. Press, Princeton, NJ, 1970.

[5] D. P. BERTSEKAS AND D. EL BAZ, *Distributed asynchronous relaxation methods for convex network flow problems*, LIDS Report P-1417, Massachusetts Inst. of Technology, October 1984, this Journal, 25 (1987), pp. 74-85.

[6] W. I. ZANGWILL, *Nonlinear Programming*, Prentice-Hall, Englewood Cliffs, NJ, 1969.

[7] R. W. H. SARGENT AND D. J. SEBASTIAN, *On the convergence of sequential minimization algorithms*, J. Optim. Theory Appl., 12 (1973), pp. 567-575.

[8] E. POLAK, *Computational Methods in Optimization: A Unified Approach*, Academic Press, New York, 1971.

[9] D. G. LUENBERGER, *Linear and Nonlinear Programming*, Addison-Wesley, Reading, MA, 1984.

[10] M. J. D. POWELL, *On search directions for minimization algorithms*, Math. Programming, 4 (1973), pp. 193-201.

[11] J. S. PANG, *On the convergence of dual ascent methods for large-scale linearly constrained optimization problems*, Univ. of Texas at Dallas, unpublished manuscript, 1984.

[12] R. W. COTTLE AND J. S. PANG, *On the convergence of a block successive over-relaxation method for a class of linear complementarity problems*, Math. Programming Stud., 17 (1982), pp. 126-138.

[13] D. P. BERTSEKAS AND S. K. MITTER, *A descent numerical method for optimization problems with nondifferentiable cost functionals*, this Journal, 11 (1973), pp. 637-652.

[14] L. R. FORD, JR. AND D. R. FULKERSON, *Flows in Networks*, Princeton Univ. Press, Princeton, NJ, 1962.

[15] C. H. PAPADIMITRIOU AND K. STEIGLITZ, *Combinatorial Optimization: Algorithms and Complexity*, Prentice-Hall, Englewood Cliffs, NJ, 1982.

[16] J. EDMONDS AND R. KARP, *Theoretical improvements in algorithmic efficiency for network flow problems*, J. Assoc. Comput. Mach., 19 (1972), pp. 248-264.

[17] P. TSENG, *Relaxation methods for monotropic programming problems*, Ph.D. thesis, Operations Research Center, Massachusetts Inst. of Technology, 1986.

[18] D. KLINGMAN, A. NAPIER AND J. STUTZ, NETGEN—*A program for generation large scale (un)capacitated assignment, transportation and minimum cost network problems*, Management Sci., 20 (1974), pp. 814-822.

[19] S. ZENIOS AND J. M. MULVEY, *Simulating a distributed synchronous relaxation method for convex network problems*, Working Paper, Department of Civil Engineering, Princeton Univ., January 1985.

[20] D. P. BERTSEKAS, *Distributed relaxation methods for linear network flow problems*, Proc. of 25th Conf. on Decision and Control, Athens, Greece, December 1986.

[21] ———, *Distributed asynchronous relaxation methods for linear network flow problems*, LIDS Report P-1606, Massachusetts Inst. of Technology, Cambridge, MA, September 1986.