AD A097623

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered) UNCLASSIFIED

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO. AD-A097623	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) (16) SECOND DERIVATIVE ALGORITHMS FOR MINIMUM DELAY DISTRIBUTED ROUTING IN NETWORKS		5. TYPE OF REPORT & PERIOD COVERED Paper
		6. PERFORMING ORG. REPORT NUMBER (14) LIDS-P-1082
7. AUTHOR(s) (10) Dimitri P. Bertsekas, Eli M. Gafni, Robert G. Gallager (13)		8. CONTRACT OR GRANT NUMBER(s) ~~ARPA Order No. 3045/5-7-75~~ (15) [redacted]N00014-75-C-1183, ARPA Order-3045
9. PERFORMING ORGANIZATION NAME AND ADDRESS Massachusetts Institute of Technology, Laboratory for Information and Decision Systems, Cambridge, Massachusetts 02139		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS Program Code No. 5T10, ONR Identifying No. 049-383
11. CONTROLLING OFFICE NAME AND ADDRESS Defense Advanced Research Projects Agency, 1400 Wilson Boulevard, Arlington, Virginia 22209 (11)		12. REPORT DATE March 1981
		13. NUMBER OF PAGES 47 (12) 50
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) Office of Naval Research, Information Systems Program, Code 437, Arlington, Virginia 22217		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

LEVEL II

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

DTIC ELECTE APR 10 1981 C

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

We propose a class of algorithms for finding an optimal quasistatic routing in a communication network. The algoritms are based on Gallager's method [1]. Their main feature is that they utilize second derivatives of the objective function and may be viewed as approximations to a constrained version of Newton's method. The use of second derivatives results in improved speed of convergence and automatic stepsize scaling with respect to level of traffic input. These advantages are of crucial importance for the practical implementation of the algorithm using distributed computation.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

410950

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

81 4 10 035

March 1981

LIDS-P-1082

SECOND DERIVATIVE ALGORITHMS FOR MINIMUM DELAY DISTRIBUTED ROUTING IN NETWORKS†

by

Dimitri P. Bertsekas*
Eli M. Gafni**
Robert G. Gallager***

ABSTRACT

We propose a class of algorithms for finding an optimal quasistatic routing in a communication network. The algorithms are based on Gallager's method [1]. Their main feature is that they utilize second derivatives of the objective function and may be viewed as approximations to a constrained version of Newton's method. The use of second derivatives results in improved speed of convergence and automatic stepsize scaling with respect to level of traffic input. These advantages are of crucial importance for the practical implementation of the algorithm using distributed computation.

†This research was conducted at the M.I.T. Laboratory for Information and Decision Systems with partial support provided by ARPA Grant No. ONR-N00014-75-C-1183 and National Science Foundation Grant NSF/ECS 79-19880.

*Room No. 35-210, Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, Mass. 02139

**Room No. 35-419, same as *.

***Room No. 35-206, same as *.

1. Introduction

We consider the problem of optimal routing of messages in a communication network so as to minimize average delay per message. We primarily have in mind a situation where the statistics of external traffic inputs change slowly with time as described in the paper by Gallager [1]. While algorithms of the type to be described can also be used for centralized computation, we place primary emphasis on algorithms that are well suited for distributed computation

Two critical requirements for the success of a distributed routing algorithm are speed of convergence and relative insensitivity of performance to variations in the statistics of external traffic inputs. Unfortunately the algorithm of [1] is not entirely satisfactory in these respects. In particular it is impossible to select in this algorithm a stepsize that will guarantee convergence and good rate of convergence for a broad range of external traffic inputs. The work described in this paper was motivated primarily by this consideration.

A standard approach for improving the rate of convergence and facilitating stepsize selection in optimization algorithms is to scale the descent direction using second derivatives of the objective function as for example in Newton's method. This is also the approach taken here. On the other hand the straightforward use of Newton's method is inappropriate for our problem primarily because of large dimensionality. We have thus introduced various approximations to Newton's method which exploit the network structure of the problem and facilitate distributed computation.

In Section 2 we describe a broad class of algorithms for minimum delay routing. This class is patterned after a gradient projection method for nonlinear programming [2],[3] as explained in [4], and contains as a special case Gallager's original algorithm except for a variation in the definition of a blocked node [compare with equation (15) of [1]]. This variation is essential in order to avoid unnecessary complications in the statement and operation of our algorithms and despite its seemingly minor significance it has necessitated a major divergence in the proof of convergence from the corresponding proof of [1].

Section 3 describes in more detail a particular algorithm from the class of Section 2. This algorithm employs second derivatives in a manner which approximates a constrained version of Newton's method [3] and is well suited for distributed computation.

The algorithm of Section 3 seems to work well for most quasistatic routing problems likely to appear in practice as extensive computational experience has shown [5]. However there are situations where the unity stepsize employed by this algorithm may be inappropriate. In Section 4 we present another distributed algorithm which automatically corrects this potential difficulty whenever it arises at the expense of additional computation per iteration. This algorithm also employs second derivatives, and is based on minimizing at each iteration a suitable upper bound to a quadratic approximation of the objective function.

Proofs of convergence have been relegated to Appendices. Both algorithms of Sections 3 and 4 have been tested extensively and computational results have been documented in [5] and [6]. These results substantiate the as- .ions made here regarding the practical properties of the algorithms. There are also other related second derivative algorithms [7],[8] that operate in the space of path flows and exhibit similar behavior as the ones of this paper. These algorithms are well suited for centralized computation and virtual circuit networks but, in contrast with the ones of the present paper, require global information at each node regarding the network topology and the total flow on each link.

We finally mention that while we have restricted attention to the problem of routing, the algorithms of this paper can be applied to other problems of interest in communication networks. For example problems of optimal adaptive flow control or combined routing and flow control have been formulated in [9],[10] as nonlinear multicommodity flow problems of the type considered here, and the algorithms of this paper are suitable for their solution.

2. A Class of Routing Algorithms

Consider a network consisting of N nodes denoted by 1,2,...,N and L directed links. The set of links is denoted by L. We denote by (i,ℓ) the link from node i to node ℓ and assume that the network is connected in the sense that for any two nodes m,n there is a directed path from m to n. The flow on each link (i,ℓ) for any destination j is denoted by $f_{i\ell}(j)$. The total flow on each link (i,ℓ) is denoted by $F_{i\ell}$, i.e.

$$F_{i\ell} = \sum_{j=1}^{N} f_{i\ell}(j).$$

The vector of all flows $f_{i\ell}(j)$, $(i,\ell)\varepsilon L$, $j = 1,\dots,N$ is denoted by f.

We are interested in numerical solution of the following multicommodity network flow problem:

$$\text{minimize} \qquad \sum_{(i,\ell)\varepsilon L} D_{i\ell}(F_{i\ell}) \qquad \text{(MFP)}$$

$$\text{subject to} \quad \sum_{\ell\varepsilon O(i)} f_{i\ell}(j) - \sum_{m\varepsilon I(i)} f_{mi}(j) = r_i(j),$$

$$\forall i = 1,\dots,N,\ i \neq j$$

$$f_{i\ell}(j) \geq 0, \quad \forall (i,\ell)\varepsilon L,\ i = 1,\dots,N,\ j = 1,\dots,N$$

$$f_{j\ell}(j) = 0, \quad \forall (j,\ell)\varepsilon L,\ j = 1,\dots,N,$$

where, for $i \neq j$, $r_i(j)$ is a known traffic input at node i destined for j, and O(i) and I(i) are the sets of nodes ℓ for which $(i,\ell)\varepsilon L$ and $(\ell,i)\varepsilon L$ respectively.

The standing assumptions throughout the paper are:

a) $r_i(j) \geq 0, \quad i,j = 1,\ldots,N,\ i \neq j$

b) Each function $D_{i\ell}$ is defined on an interval $[0,C_{i\ell})$ where $C_{i\ell}$ is either a positive number (the link capacity) or $+\infty$; $D_{i\ell}$ is twice continuously differentiable on $(0,C_{i\ell})$. The first and second derivatives of $D_{i\ell}$ at zero are defined by taking the limit from the right. Furthermore $D_{i\ell}$ is convex, continuous, and has strictly positive first and second derivatives on $[0,C_{i\ell})$.

c) (MFP) has at least one feasible solution f satisfying $F_{i\ell} < C_{i\ell}$ for all $(i,\ell)\varepsilon L$.

For notational convenience in describing various algorithms <u>we will suppress in what follows the destination index and concentrate on a single destination chosen for concreteness to be node N</u>. Our definitions, optimality conditions, and algorithms are essentially identical for each destination, so this notational simplification should not become a source of confusion. In the case where there are multiple destinations it is possible to implement our algorithms in at least two different ways. Either iterate simultaneously for all destinations (the "<u>all-at-once</u>" version), or iterate sequentially one destination at a time in a cyclic manner with intermediate readjustment of link flows (the "<u>one-at-a-time</u>" version). The remainder of our notation follows in large measure the one employed in [1]. In addition all vectors will be considered to be column vectors, transposition will be denoted by a superscript T, and the standard Euclidean norm of a vector will be denoted by $|\cdot|$, i.e. $x^T x = |x|^2$ for any vector x. Vector inequalities are meant to be componentwise, i.e. for $x = (x_1,\ldots,x_n)$ we write $x \geq 0$ if $x_i \geq 0$ for all $i = 1,\ldots,n$.

Let t_i be the total incoming traffic at node i

$$t_i = r_i + \sum_{\substack{m\varepsilon I(i)\\ m\neq N}} f_{mi}, \quad i = 1,\ldots,N-1, \tag{1}$$

and for $t_i \neq 0$ let $\phi_{i\ell}$ be the fraction of t_i that travels on link (i,ℓ)

$$\phi_{i\ell} = \frac{f_{i\ell}}{t_i}, \quad i = 1,\ldots,N-1 \ (i,\ell)\varepsilon L.$$

Then it is possible to reformulate the problem in terms of the variables $\phi_{i\ell}$ as follows [1].

For each node $i \neq N$ we fix an order of the outgoing links (i,ℓ), $\ell\varepsilon O(i)$. We identify with each collection $\{\phi_{i\ell} | (i,\ell)\varepsilon L,\ i = 1,\ldots,N-1\}$ a column vector $\phi = (\phi_1^T, \phi_2^T, \ldots, \phi_{N-1}^T)^T$, where ϕ_i is the column vector with coordinates $\phi_{i\ell}, \ell\varepsilon O(i)$. Let

$$\overline{\Phi} = \{\phi | \phi_{i\ell} \geq 0, \sum_{\ell\varepsilon O(i)} \phi_{i\ell} = 1, \quad (i,\ell)\varepsilon L,\ i = 1,\ldots,N-1\} \tag{2}$$

and let Φ be the subset of $\overline{\Phi}$ consisting of all ϕ for which there exists a directed path $(i,\ell),\ldots,(m,N)$ from every node $i = 1,\ldots,N-1$ to the destination N along which $\phi_{i\ell} > 0,\ldots,\phi_{mN} > 0$. Clearly Φ and $\overline{\Phi}$ are convex sets, and the closure of Φ is $\overline{\Phi}$. It is shown in [1] that for every $\phi\varepsilon\Phi$ and $r = (r_1, r_2, \ldots, r_{N-1})$ with $r_i \geq 0$, $i = 1,\ldots,N-1$ there exist unique vectors $t(\phi,r) = (t_1(\phi,r),\ldots,t_{N-1}(\phi,r))$ and $f(\phi,r)$ with coordinates $f_{i\ell}(\phi,r)$, $(i,\ell)\varepsilon L$, $i \neq N$ satisfying

$$t(\phi,r) \geq 0, \ f(\phi,r) \geq 0$$

$$t_i(\phi,r) = r_i + \sum_{\substack{m\varepsilon I(i) \\ m\neq N}} f_{mi}(\phi,r), \qquad i = 1,2,\ldots,N-1$$

$$\sum_{\ell\varepsilon O(i)} f_{i\ell}(\phi,r) - \sum_{\substack{m\varepsilon I(i) \\ m\neq N}} f_{mi}(\phi,r) = r_i, \qquad i = 1,\ldots,N-1$$

$$f_{i\ell}(\phi,r) = t_i(\phi,r)\phi_{i\ell}, \qquad i = 1,\ldots,N-1,\ (i,\ell)\varepsilon L.$$

Furthermore the functions $t(\phi,r)$, $f(\phi,r)$ are twice continuously differentiable in the relative interior of their domain of definition $\Phi x\{r|r \geq 0\}$. The derivatives at the relative boundary can also be defined by taking the limit through the relative interior. Furthermore for every $r \geq 0$ and every f which is feasible for (MFP) there exists a $\phi \varepsilon \Phi$ such that $f = f(\phi,r)$.

It follows from the above discussion that the problem can be written in terms of the variables $\phi_{i\ell}$ as

$$\text{minimize} \quad D(\phi,r) \triangleq \sum_{(i,\ell)\varepsilon L} D_{i\ell}[f_{i\ell}(\phi,r)] \tag{3}$$

subject to $\phi \varepsilon \Phi$,

where we write $D(\phi,r) = \infty$ if $f_{i\ell}(\phi,r) \geq C_{i\ell}$ for some $(i,\ell)\varepsilon L$.

Similarly as in [1], our algorithms generate sequences of loopfree routing variables ϕ and this allows efficient computation of various derivatives of D. Thus for a given $\phi \varepsilon \Phi$ we say that node k is <u>downstream</u> from node i if there is a directed path from i to k, and for every link (ℓ,m) on the path we have $\phi_{\ell m} > 0$. We say that node i is <u>upstream</u> from node k if k is downstream from i. We say that ϕ is <u>loopfree</u> if there is no pair of nodes i,k such that i is both upstream and downstream from k. For any $\phi \varepsilon \Phi$ and $r \geq 0$ for which

$D(\phi,r) < \infty$ the partial derivatives $\frac{\partial D(\phi,r)}{\partial \phi_{i\ell}}$ can be computed using the following equations [1]

$$\frac{\partial D}{\partial \phi_{i\ell}} = t_i(D'_{i\ell} + \frac{\partial D}{\partial r_\ell}), \quad (i,\ell)\varepsilon L, \ i = 1,\ldots,N-1 \tag{4}$$

$$\frac{\partial D}{\partial r_i} = \sum_{\ell\varepsilon 0(i)} \phi_{i\ell}(D'_{i\ell} + \frac{\partial D}{\partial r_\ell}), \quad i = 1,\ldots,N-1 \tag{5}$$

$$\frac{\partial D}{\partial r_N} = 0$$

where $D'_{i\ell}$ denotes the first derivative of $D_{i\ell}$ with respect to $f_{i\ell}$. The equations above uniquely determine $\frac{\partial D}{\partial \phi_{i\ell}}$ and $\frac{\partial D}{\partial r_i}$ and their computation is particularly simple if ϕ is loopfree. In a distributed setting each node i computes $\frac{\partial D}{\partial \phi_{i\ell}}$ and $\frac{\partial D}{\partial r_i}$ via (4),(5) after receiving the value of $\frac{\partial D}{\partial r_\ell}$ from all its immediate downstream neighbors. Because ϕ is loopfree the computation can be organized in a deadlock-free manner starting from the destination node N and proceeding upstream [1].

A necessary condition for optimality is given by (see [1])

$$\frac{\partial D}{\partial \phi_{i\ell}} = \min_{m\varepsilon 0(i)} \frac{\partial D}{\partial \phi_{im}} \qquad \text{if } \phi_{i\ell} > 0$$

$$\frac{\partial D}{\partial \phi_{i\ell}} \geq \min_{m\varepsilon 0(i)} \frac{\partial D}{\partial \phi_{im}} \qquad \text{if } \phi_{i\ell} = 0,$$

where all derivatives are evaluated at the optimum. In view of (4), this condition can be written for $t_i > 0$

$$D'_{i\ell} + \frac{\partial D}{\partial r_\ell} = \min_{m\epsilon O(i)} (D'_{im} + \frac{\partial D}{\partial r_m}) \qquad \text{if } \phi_{i\ell} > 0$$

$$D'_{i\ell} + \frac{\partial D}{\partial r_\ell} \geq \min_{m\epsilon O(i)} (D'_{im} + \frac{\partial D}{\partial r_m}) \qquad \text{if } \phi_{i\ell} = 0.$$

Combining these relations with (5) we have that if $t_i \neq 0$ then

$$\frac{\partial D}{\partial r_i} = \min_{m\epsilon O(i)} \delta_{im} \tag{6}$$

where

$$\delta_{im} = D'_{im} + \frac{\partial D}{\partial r_m}, \qquad \forall\, m\epsilon O(i) \tag{7}$$

In fact if (6) holds for all i (whether $t_i = 0$ or $t_i > 0$) then it is sufficient to guarantee optimality (see [1], Theorem 3).

We consider the class of algorithms

$$\phi_i^{k+1} = \phi_i^k + \Delta\phi_i^k \quad , \quad i = 1,\ldots,N-1 \tag{8}$$

where, for each i, the vector $\Delta\phi_i^k$ with components $\Delta\phi_{i\ell}^k$, $\ell\epsilon O(i)$ is any solution of the problem

$$\text{minimize } \delta_i^T \Delta\phi_i + \frac{t_i}{2\alpha} \Delta\phi_i^T M_i^k \Delta\phi_i \tag{9}$$

$$\text{subject to } \phi_i^k + \Delta\phi_i \geq 0, \quad \sum_\ell \Delta\phi_{i\ell} = 0,$$

$$\Delta\phi_{i\ell} = 0, \qquad \forall\, \ell\epsilon B(i;\phi^k).$$

The scalar α is a positive parameter. The vector δ_i has components [cf. (7)]

$$\delta_{i\ell} = D'_{i\ell} + \frac{\partial D}{\partial r_\ell} \quad , \qquad \forall\, \ell\epsilon O(i)$$

where all derivatives are evaluated at ϕ^k and $f(\phi^k,r)$, and δ_i^T (or $\Delta\phi_i^T$) denotes transpose of δ_i (or $\Delta\phi_i$). For each i for which $t_i(\phi^k,r) > 0$, the matrix M_i^k is some symmetric matrix which is positive definite on the subspace $\{v_i | \sum_{\ell\varepsilon O(i)} v_i = 0\}$, i.e.

$$v_i^T M_i^k v_i > 0, \quad \forall\, v_i \neq 0, \quad \sum_{\ell\varepsilon O(i)} v_{i\ell} = 0.$$

This condition guarantees that the solution to problem (9) exists and is unique. For nodes i for which $t_i(\phi^k,r) = 0$ the definition of M_i^k is immaterial. The set of indices $B(i;\phi^k)$ is specified in the following definition:

<u>Definition</u>: For any $\phi\varepsilon\Phi$ and $i=1,\ldots,N-1$ the set $B(i;\phi)$, referred to as <u>the set of blocked nodes for ϕ at i</u>, is the set of all $\ell\varepsilon O(i)$ such that $\phi_{i\ell} = 0$, and either $\frac{\partial D(\phi,r)}{\partial r_i} \leq \frac{\partial D(\phi,r)}{\partial r_\ell}$, or there exists a link (m,n) referred to as an <u>improper link</u> such that $m=\ell$ or m is downstream of ℓ and we have $\phi_{mn} > 0$, $\frac{\partial D(\phi,r)}{\partial r_m} \leq \frac{\partial D(\phi,r)}{\partial r_n}$.

It is shown below that if ϕ^k is loopfree, then ϕ^{k+1} generated by the algorithm is also loopfree. Thus the algorithm generates a sequence of loopfree routings if the starting ϕ^0 is loopfree. We refer to [1] for a description of the method for generating the sets $B(i;\phi^k)$ in a manner suitable for distributed computation. Our definition of $B(i;\phi^k)$ differs from the one of [1] primarily in that a special device that facilitated the proof of convergence given in [1] is not employed (compare with equ. (15) of [1]).

We now demonstrate some of the properties of the algorithm in the following proposition.

<u>Proposition 1</u>: a) If ϕ^k is loopfree then ϕ^{k+1} is loopfree.

b) If ϕ^k is loopfree and $\Delta\phi^k = 0$ solves problem (9) then ϕ^k is optimal.

c) If ϕ^k is optimal then ϕ^{k+1} is also optimal.

d) If $\Delta\phi_i^k \neq 0$ for some i for which $t_i(\phi^k,r) > 0$ then there exists a positive scalar η_k such that

$$D(\phi^k + \eta\Delta\phi^k,r) < D(\phi^k,r), \quad \forall \eta\varepsilon(0,\eta_k].$$

<u>Proof:</u> a) Assume that ϕ^{k+1} is not loopfree and there exists a sequence of links forming a directed loop al[illegible]ich ϕ^{k+1} is positive. Then there must exist a link (m,n) on the loop for which $\frac{\partial D(\phi^k,r)}{\partial r_m} \leq \frac{\partial D(\phi^k,r)}{\partial r_n}$. From the definition of $B(m;\phi^k)$ we must have $\phi_{mn}^k > 0$ and hence (m,n) is an improper link. Now move backwards around the loop to the first link (i,ℓ) for which $\phi_{i\ell}^k = 0$. Such a link must exist since ϕ^k is loopfree. Since ℓ is upstream of m and (m,n) is improper, we have $\ell\varepsilon B(i;\phi^k)$ which contradicts the hypothesis $\phi_{i\ell}^{k+1} > 0$.

b) If $\Delta\phi^k = 0$ solves problem (9) then we must have $\delta_i^T\Delta\phi_i \geq 0$ for each i and $\Delta\phi_i$ satisfying the constraints of (9)

$$\Delta\phi_i \geq -\phi_i^k, \quad \sum_\ell \Delta\phi_{i\ell} = 0, \quad \Delta\phi_{i\ell} = 0, \qquad \forall \ell\varepsilon B(i;\phi^k). \tag{10}$$

By writing $\Delta\phi_i = \phi_i - \phi_i^k$ and using (5), (7) we have

$$\delta_i^T(\phi_i-\phi_i^k) = \sum_\ell \delta_{i\ell}\phi_{i\ell} - \sum_\ell \delta_{i\ell}\phi_{i\ell}^k$$

$$= \sum_\ell \delta_{i\ell}\phi_{i\ell} - \frac{\partial D}{\partial r_i} \geq 0.$$

By considering $\phi_{i\ell} = 1$ individually for each $\ell \notin B(i;\phi^k)$, we obtain

$$\frac{\partial D}{\partial r_i} \leq \delta_{i\ell}, \quad \forall \ell \notin B(i;\phi^k).$$

From (5) and (7) then

$$\frac{\partial D}{\partial r_i} = \delta_{i\ell}, \quad \forall \ell \notin B(i;\phi^k), \text{with } \phi^k_{i\ell} > 0.$$

Since $D'_{i\ell} > 0$ for all $(i,\ell)\varepsilon L$ it follows from (5),(7) and the relation above that there are no improper links, and using the definition of $B(i;\phi^k)$ we obtain

$$\frac{\partial D}{\partial r_i} = \min_{\ell \varepsilon O(i)} \delta_{i\ell}$$

which is a sufficient condition for optimality of ϕ^k [cf. (6)].

c) If ϕ^k is optimal then from the necessary condition for optimality (6) we have that for all i with $t_i > 0$

$$\frac{\partial D}{\partial r_i} = \min_{m \varepsilon O(i)} \delta_{im}$$

It follows using a reverse argument to the one in b) above that

$$\Delta\phi^k_i = 0 \qquad \text{if } t_i > 0.$$

Since changing only routing variables of nodes i for which $t_i = 0$ does not affect the flow through each link we have $D(\phi^k,r) = D(\phi^{k+1},r)$ and ϕ^{k+1} is optimal.

d) Since M^k_i is positive semidefinite for all i with $t_i > 0$ and $\Delta\phi^k_i$ is a solution of problem (9) we have

$$\delta_i^T \Delta\phi^k_i \leq 0$$

If $t_i > 0$ then M^k_i is positive definite on the appropriate subspace and the solution of problem (9) is unique, so if in addition $\Delta\phi^k \neq 0$ then we have

$$\delta_i^T \Delta\phi_i^k < 0.$$

Using the fact [cf. (4),(7)]

$$\frac{\partial D}{\partial \phi_i} = t_i \delta_i,$$

we obtain that

$$\frac{\partial D}{\partial \phi_i}^T \Delta\phi_i^k < 0$$

Hence $\Delta\phi^k$ is a direction of descent at ϕ^k and the result follows. Q.E.D.

The following proposition is the main convergence result regarding algorithm (8),(9). Its proof is quite complex and is given in Appendix A. The proposition applies to the multiple destination case in the "all-at-once" and the "one-at-a-time" version.

<u>Proposition 2</u>: Let the initial routing ϕ^o be loopfree and satisfy $D(\phi^o,r) \leq D_o$ where D_o is some scalar. Assume also that there exist two positive scalars λ,Λ such that the sequences of matrices $\{M_i^k\}$ satisfy the following two conditions:

a) The absolute value of each element of M_i^k is bounded above by Λ.

b) There holds

$$\lambda|v_i|^2 \leq v_i^T M_i^k v_i$$

for all v_i in the subspace $\{v_i | \sum_{\ell \notin B(i;\phi^k)} v_{i\ell} = 0\}$.

Then there exists a positive scalar $\bar{\alpha}$ (depending on D_o, λ, and Λ) such that for all

$\alpha\varepsilon(0,\bar{\alpha}]$ and k=0,1,... the sequence $\{\phi^k\}$ generated by algorithm (8),(9) satisfies

$$D(\phi^{k+1},r) \leq D(\phi^k,r) \ , \quad \lim_{k\to\infty} D(\phi^k,r) = \min_{\phi\varepsilon\Phi} D(\phi,r).$$

Furthermore every limit point of $\{\phi^k\}$ is an optimal solution of problem (3).

Another interesting result which will not be given here but can be found in [11] states that, after a finite number of iterations, improper links do not appear further in the algorithm so that for rate of convergence analysis purposes the potential presence of improper links can be ignored. Based on this fact it can be shown under a mild assumption that for the single destination case the rate of convergence of the algorithm is linear [11].

The class of algorithms (8),(9) is quite broad since different choices of matrices M_i^k yield different algorithms. A specific choice of M_i^k yields Gallager's algorithm [1] [except for the difference in the definition of $B(i;\phi^k)$ mentioned earlier]. This choice is the one for which M_i^k is diagonal with all elements along the diagonal being unity except the $(\bar{\ell},\bar{\ell})$th element which is zero where $\bar{\ell}$ is a node for which

$$\delta_{i\bar{\ell}} = \min_{\ell\varepsilon O(i)} \delta_{i\ell}.$$

We leave the verification of this fact to the reader. In the next section we describe a specific algorithm involving a choice of M_i^k based on second derivatives of $D_{i\ell}$. The convergence result of Proposition 2 is applicable to this algorithm.

3. An Algorithm Based on Second Derivatives

A drawback of the algorithm of [1] is that a proper range of the stepsize parameter α is hard to determine. In order for the algorithm to have guaranteed convergence for a broad range of inputs r, one must take α quite small but this will lead to a poor speed of convergence for most of these inputs. It appears that in this respect a better choice of the matrices M_i^k can be based on second derivatives. This tends to make the algorithm to a large extent scale free, and for most problems likely to appear in practice, a choice of the stepsize α near unity results in both convergence and reasonably good speed of convergence for a broad range of inputs r. This is supported by extensive computational experience some of which is reported in [5] and [6].

We use the notation

$$D''_{i\ell} = \frac{\partial^2 D_{i\ell}}{[\partial f_{i\ell}]^2}$$

We have already assumed that $D''_{i\ell}$ is positive in the set $[0,C_{i\ell})$. We would like to choose the matrices M_i^k to be diagonal with $t_i^{-2} \frac{\partial^2 D(\phi^k,r)}{[\partial\phi_{i\ell}]^2}$ along the diagonal. This corresponds to an approximation of a constrained version of Newton's method (see [3]), where the off-diagonal terms of the Hessian matrix of D are set to zero. This type of approximated version of Newton's method is often employed in solving large scale unconstrained optimization problems. Unfortunately the second derivatives $\frac{\partial^2 D}{[\partial\phi_{i\ell}]^2}$ are difficult to compute. However, it is possible to compute easily upper and lower bounds to them which, as shown by computational experiments, are sufficiently accurate for practical purposes.

Calculation of Upper and Lower Bounds to Second Derivatives

We compute $\frac{\partial^2 D}{[\partial\phi_{i\ell}]^2}$ evaluated at a loopfree $\phi\epsilon\Phi$, for all links $(i,\ell)\epsilon L$ fo ..ich $\ell \notin B(i;\phi)$. We have using (4)

$$\frac{\partial^2 D}{[\partial\phi_{i\ell}]^2} = \frac{\partial}{\partial\phi_{i\ell}} \{t_i (D'_{i\ell} + \frac{\partial D}{\partial r_\ell})\}.$$

Since $\ell \notin B(i;\phi)$ and ϕ is loopfree, the node ℓ is not upstream of i. It follows that $\frac{\partial t_i}{\partial\phi_{i\ell}} = 0$ and $\frac{\partial D'_{i\ell}}{\partial\phi_{i\ell}} = D''_{i\ell} t_i$. Using again the fact that ℓ is not upstream of i we have $\frac{\partial t_i}{\partial r_\ell} = 0$, $\frac{\partial D'_{i\ell}}{\partial r_\ell} = 0$ and it follows that

$$\frac{\partial^2 D}{\partial\phi_{i\ell}\partial r_\ell} = \frac{\partial}{\partial r_\ell} \frac{\partial D}{\partial\phi_{i\ell}} = \frac{\partial}{\partial r_\ell} \left\{ t_i (D'_{i\ell} + \frac{\partial D}{\partial r_\ell}) \right\} = t_i \frac{\partial^2 D}{[\partial r_\ell]^2}.$$

Thus we finally obtain

$$\frac{\partial^2 D}{[\partial\phi_{i\ell}]^2} = t_i^2 (D''_{i\ell} + \frac{\partial^2 D}{[\partial r_\ell]^2}). \tag{11}$$

A little thought shows that the second derivative $\frac{\partial^2 D}{[\partial r_\ell]^2}$ is given by the more general formula

$$\frac{\partial^2 D}{\partial r_\ell \partial r_m} = \sum_{(j,k)\epsilon L} q_{jk}(\ell) q_{jk}(m) D''_{ik}, \quad \forall\, \ell, m=1,\ldots,N-1 \tag{12}$$

where $q_{jk}(\ell)$ is the portion of a unit of flow originating at ℓ which goes through link (j,k). However calculation of $\frac{\partial^2 D}{[\partial r_\ell]^2}$ using this formula is complicated, and in fact there seems to be no easy way to compute this second derivative. However upper and lower bounds to it can be easily computed as we now show. By using (5) we obtain

$$\frac{\partial^2 D}{[\partial r_\ell]^2} = \frac{\partial}{\partial r_\ell} \{\sum_m \phi_{\ell m}(D'_{\ell m} + \frac{\partial D}{\partial r_m})\}.$$

Since ϕ is loopfree we have that if $\phi_{\ell m} > 0$ then m is not upstream of ℓ and therefore $\frac{\partial t_\ell}{\partial r_\ell} = 1$ and $\frac{\partial D'_{\ell m}}{\partial r_\ell} = D''_{\ell m}\ \phi_{\ell m}$. A similar reasoning shows that

$$\frac{\partial^2 D}{\partial r_\ell \partial r_m} = \frac{\partial}{\partial r_m} \{\sum_n \phi_{\ell n}(D'_{\ell n} + \frac{\partial D}{\partial r_n})\} = \sum_n \phi_{\ell n} \frac{\partial^2 D}{\partial r_m \partial r_n}$$

Combining the above relations we obtain

$$\frac{\partial^2 D}{[\partial r_\ell]^2} = \sum_m \phi_{\ell m}^2 D''_{\ell m} + \sum_m \sum_n \phi_{\ell m} \phi_{\ell n} \frac{\partial^2 D}{\partial r_m \partial r_n}. \tag{13}$$

Since $\frac{\partial^2 D}{\partial r_m \partial r_n} \geq 0$, by setting $\frac{\partial^2 D}{\partial r_m \partial r_n}$ to zero for $m \neq n$ we obtain the lower bound

$$\sum_m \phi_{\ell m}^2 (D''_{\ell m} + \frac{\partial^2 D}{[\partial r_m]^2}).$$

By applying the Cauchy-Schwartz inequality in conjunction with (12) we also obtain

$$\frac{\partial^2 D}{\partial r_m \partial r_n} \leq \sqrt{\frac{\partial^2 D}{[\partial r_m]^2} \frac{\partial^2 D}{[\partial r_n]^2}}.$$

Using this fact in (13) we obtain the upper bound

$$\sum_m \phi_{\ell m}^2 D''_{\ell m} + \left(\sum_m \phi_{\ell m} \sqrt{\frac{\partial^2 D}{[\partial r_m]^2}}\right)^2 .$$

It is now easy to see that we have for all k

$$\underline{R}_\ell \leq \frac{\partial^2 D}{[\partial r_\ell]^2} \leq \overline{R}_\ell$$

where $\underline{R}_\ell$ and $\overline{R}_\ell$ are generated by

$$\underline{R}_\ell = \sum_m \phi_{\ell m}^2 (D''_{\ell m} + \underline{R}_m) \tag{14}$$

$$\overline{R}_\ell = \sum_m \phi_{\ell m}^2 D''_{\ell m} + \left(\sum_m \phi_{\ell m} \sqrt{\overline{R}_m}\right)^2 \tag{15}$$

$$\underline{R}_N = \overline{R}_N = 0 \tag{16}$$

The computation is carried out by passing $\underline{R}_\ell$ and $\overline{R}_\ell$ upstream together with $\frac{\partial D}{\partial r_\ell}$ and this is well suited for a distributed algorithm. Upper and lower bounds $\underline{\Phi}_{i\ell}, \overline{\Phi}_{i\ell}$ for $\frac{\partial^2 D}{[\partial \phi_{i\ell}]^2}$, $\ell \notin B(i;\phi)$ are obtained simultaneoulsy by means of the equation [cf. (11)]

$$\underline{\Phi}_{i\ell} = t_i^2 (D''_{i\ell} + \underline{R}_\ell) \tag{17}$$

$$\overline{\Phi}_{i\ell} = t_i^2 (D''_{i\ell} + \overline{R}_\ell). \tag{18}$$

It is to be noted that in some situations occuring frequently in practice the upper and lower bounds $\underline{\Phi}_{i\ell}$ and $\overline{\Phi}_{i\ell}$ coincide and are equal to the true second derivative. This will occur if $\phi_{\ell_m}\phi_{\ell_n} \dfrac{\partial^2 D}{\partial r_m \partial r_n} = 0$ for $m \neq n$. For example if the routing pattern is as shown in Figure 1 (only links that carry flow are shown) then $\overline{\Phi}_{i\ell} = \underline{\Phi}_{i\ell} = \dfrac{\partial^2 D}{[\partial \phi_{i\ell}]^2}$ for all $(i,\ell)\in L$, $\ell \notin B(i;\phi)$.

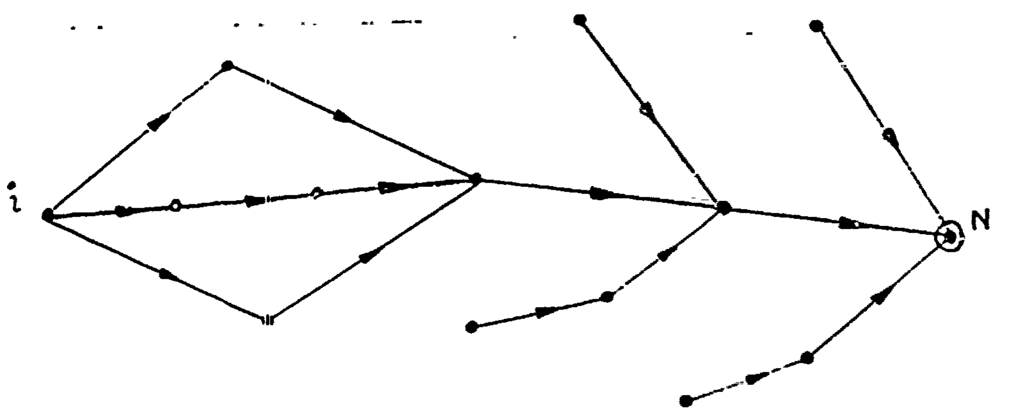

Figure 1

A typical case where $\overline{\Phi}_{i\ell} \neq \underline{\Phi}_{i\ell}$ and the discrepancy affects materially the algorithm to be presented is when flow originating at i splits and joins again twice on its way to N as shown in Figure 2.

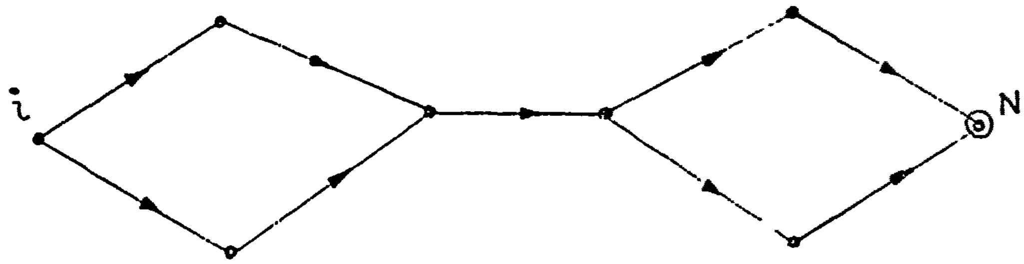

Figure 2

The Algorithm

The following algorithm seems to be a reasonable choice. If $t_i \neq 0$ we take M_i^k in (9) to be the diagonal matrix with $\frac{1}{t_i^2}\,\bar{\Phi}_{i\ell}$, $\ell\varepsilon O(i)$ along the diagonal where $\bar{\Phi}_{i\ell}$ is the upper bound computed from (18) and (14)-(16) and α is a positive scalar chosen experimentally. (In most cases $\alpha=1$ is satisfactory.)

Convergence of this algorithm can be easily established by verifying that the assumption of Proposition 2 is satisfied. A variation of the method results if we use in place of the upper bound $\bar{\Phi}_{i\ell}$ the average of the upper and lower bounds $\frac{\bar{\Phi}_{i\ell}+\underline{\Phi}_{i\ell}}{2}$. This however requires additional computation and communication between modes.

Problem (9) can be written for $t_i \neq 0$ as

$$\begin{aligned}
&\text{minimize} && \sum_{\ell}\{\delta_{i\ell}\,\Delta\phi_{i\ell}+\frac{\bar{\Phi}_{i\ell}}{2\alpha t_i}(\Delta\phi_{i\ell})^2\} \\
&\text{subject to} && \Delta\phi_{i\ell}\ \geq\ -\phi_{i\ell}^k,\ \sum_{\ell}\Delta\phi_{i\ell}\ =\ 0,\ \Delta\phi_{i\ell}\ =\ 0\quad \forall\ell\varepsilon B(i;\phi^k)
\end{aligned}\tag{19}$$

and can be solved using a Lagrange multiplier technique. By introducing the expression (18) for $\bar{\Phi}_{i\ell}$ and carrying out the straightforward calculation we can write the corresponding iteration (8) as

$$\phi_{i\ell}^{k+1} = \max\{0,\phi_{i\ell}^{k} - \frac{\alpha(\delta_{i\ell}-\mu)}{t_i(D''_{i\ell}+\bar{R}_\ell)}\} \tag{20}$$

where μ is a Lagrange multiplier determined from the condition

$$\sum_{\ell\notin B(i;\phi^k)} \max\{0,\phi_{i\ell}^{k} - \frac{\alpha(\delta_{i\ell}-\mu)}{t_i(D''_{i\ell}+\bar{R}_\ell)}\} = 1. \tag{21}$$

The equation above is piecewise linear in the single variable μ and is nearly trivial computatinnally. Note from (20) that α plays the role of a stepsize parameter.

It can be seen that (20) is such that all routing variables $\phi_{i\ell}$ such that $\delta_{i\ell} < \mu$ will be increased or stay fixed at unity, while all routing variables $\phi_{i\ell}$ such that $\delta_{i\ell} > \mu$ will be decreased or stay fixed at zero. In particular the routing variable with smallest $\delta_{i\ell}$ will either be increased or stay fixed at unity, similarly as in Gallager's algorithm.

4. An Algorithm Based on an Upper Bound to Newton's Method

While the introduction of a diagonal scaling based on second derivatives alleviates substantially the problem of stepsize selection, it is still possible that in some iterations a unity stepsize will not lead to a reduction of the objective function and may even cause divergence of the algorithm of the previous section. This can be corrected by using a smaller stepsize as shown in Proposition 2 but the proper range of stepsize magnitude depends on the network topology and may not be easy to determine. This dependence stems from the replacement of the Hessian matrix of D by a diagonal approximation which in turn facilitates the computation of upper bounds to second derivatives in a distributed manner. Neglecting the off-diagonal terms of the Hessian means that while operating the algorithm for one destination we ignore changes which are caused by other destinations. The potential difficulties resulting from this can be alleviated (and for most practical problems eliminated) by operating the algorithm in a "one-at-a-time" version as discussed in Section 2. However the effect of neglecting the off-diagonal terms can still be detrimental in some situations such as the one depicted by Figure 3. Here $r_1 = r_2 = r_3 = r_4 > 0$, $r_5 = r_6 = 0$ and node 7 is the only destination. If the algorithm of the previous section is applied to this example with $\alpha=1$, then it can be verified that each of the nodes 1,2,3 and 4 will adjust its routing variables according to what would be Newton's method if all other variables remained unchanged. If we assume symmetric initial conditions and that the first and second derivatives D'_{57}, D''_{57} and D'_{67}, D''_{67} are much larger than the corresponding derivatives of all other links, then the algorithm would lead to a change of flow about four times larger than appropriate. Thus for example a value of $\alpha = 1/4$ is appropriate, while $\alpha=1$ can lead to divergence.

The algorithm proposed in this section bypasses these difficulties at the expense of additional computation per iteration. We show that if the initial flow vector is near optimal then the algorithm is guaranteed to reduce the value of the objective function at each iteration and to converge to the optimum with a unity stepsize. The algorithm "upper bounds" a quadratic approximation to the objective function D. This is done by first making a trial change in the routing variables using algorithm (8),(9). The link flows that would result from this change are then calculated going from the "most upstream" nodes downstream towards the destination. Based on the calculated flows the algorithm "senses" situations like the one in Figure 3 and automatically "reduces" the stepsize. We describe the algorithm for the case of a single destination (node N). The algorithm for the case of more than one destination consists of sequences of single destination iterations whereby all destinations are taken up cyclically (i.e. the one-at-a-time mode of operation).

The Algorithm

At the typical iteration of the algorithm we have a vector of loop-free routing variables ϕ and a corresponding flow vector f. We first carry out iteration (8),(9) with the choice of M_i^k described in Section 3 and a unity stepsize, and obtain a trial increment of routing variables denoted by $\Delta\phi^*$. Based on $\Delta\phi^*$ we calculate the new (and final) increment of routing variables $\Delta\tilde{\phi}$ and the new routing vector

$$\tilde{\phi} = \phi + \Delta\tilde{\phi} \tag{22}$$

by means of a procedure of the following type. Each node i computes the corresponding vector of routing variable increments $\Delta\tilde{\phi}_i$ by solving a problem of the form

$$\text{minimize} \quad Q_i(\Delta\phi_i) \tag{23}$$

subject to the constraints

$$\Delta\phi_{i\ell} \geq 0 \quad \text{if } \Delta\phi^*_{i\ell} > 0 \tag{24a}$$

$$\Delta\phi_{i\ell} \leq 0 \quad \text{if } \Delta\phi^*_{i\ell} < 0 \tag{24b}$$

$$\Delta\phi_{i\ell} = 0 \quad \text{if } \Delta\phi^*_{i\ell} = 0 \tag{24c}$$

$$\sum_{\ell} \Delta\phi_{i\ell} = 0 \tag{24d}$$

$$\phi_{i\ell} + \Delta\phi_{i\ell} \geq 0 \tag{24e}$$

where $Q_i(\Delta\phi_i)$ is a quadratic function of $\Delta\phi_i$ which depends on ϕ and $\Delta\phi^*$, and will be defined shortly. Notice that the constraint (24) guarantees that the new vector of routing variables $\tilde{\phi}$ is loopfree. In what follows we describe the procedure and rationale for obtaining the form of the quadratic function Q_i of (23), and show that all computations can be carried out in a distributed manner.

Let Δf denote an increment of flow such that $f + \Delta f$ is feasible. A constrained version of Newton's method [3] is obtained if Δf is chosen to minimize the quadratic objective function

$$N(\Delta f) = \sum_{i,\ell} D'_{i\ell}\Delta f_{i\ell} + \frac{1}{2}\sum_{i,\ell} D''_{i\ell}(\Delta f_{i\ell})^2 \tag{25}$$

subject to $f + \Delta f \in F$ where F is the set of all feasible flow vectors.

Let $\Delta\phi$ be the change in ϕ that corresponds to Δf. We write

$$\bar{\phi} = \phi + \Delta\phi \quad .$$

Finally let t and Δt be the vectors of total incoming traffic at the network nodes and corresponding changes [cf. (1)]. Then we have

$$\Delta t_i = \sum_{\ell} \Delta f_{\ell i}, \tag{26}$$

$$\Delta f_{i\ell} = \Delta t_i\bar{\phi}_{i\ell} + t_i\Delta\phi_{i\ell}. \tag{27}$$

Substituting (27) in (25) we obtain

$$N(\Delta f) = \sum_{i,\ell} D'_{i\ell}\Delta t_i\bar{\phi}_{i\ell} + \sum_{i,\ell} D'_{i\ell}t_i\Delta\phi_{i\ell} \tag{28}$$

$$+ \frac{1}{2}\sum_{i,\ell} D''_{i\ell}[(\Delta t_i\bar{\phi}_{i\ell})^2 + 2\Delta t_i\bar{\phi}_{i\ell}t_i\Delta\phi_{i\ell} + (t_i\Delta\phi_{i\ell})^2]$$

$$\bar{D}'_{i\ell} = D'_{i\ell} + D''_{i\ell} t_i \Delta\phi_{i\ell} \tag{29}$$

$$\bar{D}'_i = \sum_\ell \bar{\phi}_{i\ell}(\bar{D}'_{i\ell} + \bar{D}'_\ell) \;,\; \bar{D}'_N = 0 \tag{30}$$

By multiplying (30) by Δt_i, summing over i and using (27) we obtain

$$\begin{aligned}
\sum_{i,\ell} \bar{D}'_{i\ell}\bar{\phi}_{i\ell}\Delta t_i &= \sum_i \bar{D}'_i\Delta t_i - \sum_{i,\ell} \bar{\phi}_{i\ell}\bar{D}'_\ell\Delta t_i \\
&= \sum_i \bar{D}'_i\Delta t_i - \sum_{i,\ell} \Delta f_{i\ell}\bar{D}'_\ell + \sum_{i,\ell} t_i\Delta\phi_{i\ell}\bar{D}'_\ell \\
&= \sum_i \bar{D}'_i\Delta t_i - \sum_\ell \Delta t_\ell\bar{D}'_\ell + \sum_{i,\ell} t_i\Delta\phi_{i\ell}\bar{D}'_\ell \\
&= \sum_{i,\ell} t_i\Delta\phi_{i\ell}\bar{D}'_\ell
\end{aligned}$$

By using this relation together with (29) we can write (28) as

$$\begin{aligned}
N(\Delta f) = \sum_i \{&\sum_\ell t_i\Delta\phi_{i\ell}(D'_{i\ell} + \bar{D}'_\ell) \\
&+ \frac{1}{2}\sum_\ell D''_{i\ell}[(\Delta t_i\bar{\phi}_{i\ell})^2 + (t_i\Delta\phi_{i\ell})^2]\}
\end{aligned} \tag{31}$$

Now if $(\Delta t_i)^2$ were available then we could conceive of a recursive scheme whereby node i would obtain the vector $\Delta\phi_i$ which minimizes the corresponding term in the right hand side of (31) after receiving the value of $\bar{D}_\ell$ from its downstream neighbors ℓ, and in fact it can be seen that such a computation can be carried out in distributed fashion starting from the destination and proceeding upstream similarly as for algorithm (8), (9). Unfortunately $(\Delta t_i)^2$ depends on the values of $\Delta\phi_m$ for nodes m that lie upstream of i. To bypass this difficulty we develop in what follows an upper bound for the troublesome term $\sum_{i,\ell} D''_{i\ell}(\Delta t_i\bar{\phi}_{i\ell})^2$ by making use of the increment $\Delta\phi^*$ obtained through an iteration of algorithm (8),(9). When this upperbound is substituted in (31) we will obtain an upper bound to

$N(\Delta f)$ of the form

$$N(\Delta f) \leq \sum_i t_i Q_i(\Delta\phi_i)$$

where $Q_i(\Delta\phi_i)$ is precisely the expression to be used in the algorithm [cf. (23)].

Let us define for all $i = 1,\ldots,N-1$, $(i,\ell)\varepsilon L$ and $\Delta\phi$ satisfying the constraint (24)

$$\Delta\phi^{*+}_{i\ell} = \max(0,\Delta\phi^*_{i\ell}) , \quad \Delta\phi^{*-}_{i\ell} = |\min(0,\Delta\phi^*_{i\ell})| \tag{32}$$

$$\Delta\phi^{+}_{i\ell} = \max(0,\Delta\phi_{i\ell}) , \quad \Delta\phi^{-}_{i\ell} = |\min(0,\Delta\phi_{i\ell})| \tag{33}$$

$$\Delta t^{*+}_{i} = \sum_\ell [t_\ell \Delta\phi^{*+}_{\ell i} + \Delta t^{*+}_\ell(\phi_{\ell i} + \Delta\phi^{*+}_{\ell i})] \tag{34}$$

$$\Delta t^{*-}_{i} = \sum_\ell [t_\ell \Delta\phi^{*-}_{\ell i} + \Delta t^{*-}_\ell(\phi_{\ell i} + \Delta\phi^{*-}_{\ell i})] \tag{35}$$

$$\Delta t^{+}_{i} = \sum_\ell [t_\ell \Delta\phi^{+}_{\ell i} + \Delta t^{+}_\ell(\phi_{\ell i} + \Delta\phi^{+}_{\ell i})] \tag{36}$$

$$\Delta t^{-}_{i} = \sum_\ell [t_\ell \Delta\phi^{-}_{\ell i} + \Delta t^{-}_\ell(\phi_{\ell i} + \Delta\phi^{-}_{\ell i})]. \tag{37}$$

The quantities Δt^{*+}_i, Δt^{*-}_i are well defined by virtue of the fact that the set of links

$$L^* = \{(i,\ell)\varepsilon L | \ \phi_{i\ell} > 0, \text{ or } \phi_{i\ell} + \Delta\phi^*_{i\ell} > 0\}$$

forms an acyclic network [in view of the manner that the sets of blocked nodes $B(\phi;i)$ are defined in algorithm (8),(9)]. As a result Δt^{*+}_i and Δt^{*-}_i are zero for all nodes i which are the "most upstream" in this acyclic network. Starting from these nodes and proceeding downstream the computation of Δt^{*+}_i and Δt^{*-}_i can be carried out in a distributed manner for all nodes i using (34) and (35). Similarly [in view of the constraint (24)] the quantities Δt_i^+, Δt_i^- are well defined. It can be easily seen that for all i we have

$$-\Delta t_i^- \leq \Delta t_i \leq \Delta t_i^+ .$$

As a result it follows that

$$(\Delta t_i)^2 \leq (\Delta t_i^+)^2 + (\Delta t_i^-)^2 . \tag{38}$$

We will develop upper bounds to the terms $(\Delta t_i^+)^2$ and $(\Delta t_i^-)^2$. To this end we need the following lemma the straightforward proof of which is left to the reader.

<u>Lemma 1</u>: Under the constraint (24)

$$\Delta t_i^{*+} = 0 \implies \Delta t_i^+ = 0 , \quad \forall i = 1,\ldots,N-1$$

$$\Delta t_i^{*-} = 0 \implies \Delta t_i^- = 0 , \quad \forall i = 1,\ldots,N-1.$$

By using (36),(34) and the Cauchy-Schwartz inequality we obtain for all i = 1,...,N-1 with $\Delta t_i^+ > 0$

$$\begin{aligned}
(\Delta t_i^+)^2 &= \Big[\sum_\ell (t_\ell \Delta\phi_{\ell i}^+ + \Delta t_\ell^+ \phi_{\ell i} + \Delta t_\ell^+ \Delta\phi_{\ell i}^+)\Big]^2 \\
&= \left[\sum_\ell \frac{t_\ell \Delta\phi_{\ell i}^+ (t_\ell \Delta\phi_{\ell i}^{*+})^{1/2}}{(t_\ell \Delta\phi_{\ell i}^{*+})^{1/2}} + \sum_\ell \frac{\Delta t_\ell^+ \phi_{\ell i} (\Delta t_\ell^{*+} \phi_{\ell i})^{1/2}}{(\Delta t_\ell^{*+} \phi_{\ell i})^{1/2}} + \sum_\ell \frac{\Delta t_\ell^+ \Delta\phi_{\ell i}^+ (\Delta t_\ell^{*+} \Delta\phi_{\ell i}^{*+})^{1/2}}{(\Delta t_\ell^{*+} \Delta\phi_{\ell i}^{*+})^{1/2}}\right]^2 \\
&\leq \left[\sum_\ell \frac{t_\ell (\Delta\phi_{\ell i}^+)^2}{\Delta\phi_{\ell i}^{*+}} + \sum_\ell \frac{(\Delta t_\ell^+)^2 \phi_{\ell i}}{\Delta t_\ell^{*+}} + \sum_\ell \frac{(\Delta t_\ell^+)^2 (\Delta\phi_{\ell i}^+)^2}{\Delta t_\ell^{*+} \Delta\phi_{\ell i}^{*+}}\right] \Delta t_i^{*+}
\end{aligned} \tag{39}$$

where from each summation above we exclude all nodes ℓ for which the corresponding denominator [and hence also the numerator by (24) and Lemma 1] is zero.

Similarly we obtain

$$(\Delta t_i^-)^2 \leq \left[\sum_\ell \frac{t_\ell(\Delta\phi_{\ell i}^-)^2}{\Delta\phi_{\ell i}^{*-}} + \sum_\ell \frac{(\Delta t_\ell^-)^2\phi_{\ell i}}{\Delta t_\ell^{*-}} + \sum_\ell \frac{(\Delta t_\ell^-)^2(\Delta\phi_{\ell i}^-)^2}{\Delta t_\ell^{*-}\Delta\phi_{\ell i}^{*-}}\right]\Delta t_i^{*-} . \tag{40}$$

Define now for all i=1,...N-1

$$D_i''^+ = \sum_\ell \left\{D_{i\ell}''(\overline{\Phi}_{i\ell})^2\Delta t_i^{*+} + D_\ell''^+\left[\phi_{i\ell} + \frac{(\Delta\phi_{i\ell}^+)^2}{\Delta\phi_{i\ell}^{*+}}\right]\right\} \tag{41}$$

$$D_i''^- = \sum_\ell \left\{D_{i\ell}''(\overline{\Phi}_{i\ell})^2\Delta t_i^{*-} + D_\ell''^-\left[\phi_{i\ell} + \frac{(\Delta\phi_{i\ell}^-)^2}{\Delta\phi_{i\ell}^{*-}}\right]\right\} \tag{42}$$

where the summation in (41) [(42)] is over all nodes ℓ such that $\Delta\phi_{i\ell}^{*+} \neq 0$ ($\Delta\phi_{i\ell}^{*-} \neq 0$). Define also

$$D_N''^+ = 0 \quad , \quad D_N''^- = 0. \tag{43}$$

Notice that given $\Delta\phi_{i\ell}$ and $D_\ell''^+$ for all downstream neighbors ℓ it is possible for node i to compute $D_i''^+$ and $D_i''^-$. Thus for each $\Delta\phi$ satisfying (24) the quantities $D_i''^+$, $D_i''^-$ are well defined and can be computed recursively starting from the destination N and proceeding upstream in a distributed manner.

The following proposition yields the desired upper bound.

<u>Proposition 3</u>: Under the constraint (24) we have

$$N(\Delta f) \leq \sum_i t_i Q_i(\Delta\phi_i) \tag{44}$$

where

$$Q_i(\Delta\phi_i) = \sum_\ell [(D_{i\ell}' + \overline{D}_\ell')\Delta\phi_{i\ell} + \tfrac{1}{2}(t_i D_{i\ell}'' + \beta_{i\ell})(\Delta\phi_{i\ell})^2] \tag{45}$$

and

$$\beta_{i\ell} = \begin{cases} \dfrac{D''^{+}_{\ell}}{\Delta\phi^{*+}_{i\ell}} & \text{if } \Delta\phi^{*}_{i\ell} > 0 \\[2ex] \dfrac{D''^{-}_{\ell}}{\Delta\phi^{*-}_{i\ell}} & \text{if } \Delta\phi^{*}_{i\ell} < 0 \\[2ex] 0 & \text{if } \Delta\phi^{*}_{i\ell} = 0 \end{cases} \tag{46}$$

<u>Proof</u>: In view of (31) it will suffice to show that

$$\sum_{i,\ell} D''_{i\ell}(\Delta t_i \bar{\phi}_{i\ell})^2 \le \sum_{i,\ell} t_i \beta_{i\ell}(\Delta\phi_{i\ell})^2. \tag{47}$$

From (38) we have

$$\sum_{i,\ell} D''_{i\ell}(\Delta t_i \bar{\phi}_{i\ell})^2 = \sum_i (\Delta t_i)^2 \sum_\ell D''_{i\ell}(\bar{\phi}_{i\ell})^2 \tag{48}$$

$$\le \sum_i (\Delta t_i^+)^2 \sum_\ell D''_{i\ell}(\bar{\phi}_{i\ell})^2 + \sum_i (\Delta t_i^-)^2 \sum_\ell D''_{i\ell}(\bar{\phi}_{i\ell})^2.$$

For all i with $\Delta t_i^+ > 0$ we have, using Lemma 1, $\Delta t_i^{*+} > 0$ so by dividing (41) by Δt_i^{*+} we obtain

$$\sum_i (\Delta t_i^+)^2 \sum_\ell D''_{i\ell}(\bar{\phi}_{i\ell})^2 = \sum_i (\Delta t_i^+)^2 \left[\frac{D''^{+}_{i}}{\Delta t_i^{*+}} - \sum_\ell \frac{D''^{+}_{\ell}}{\Delta t_i^{*+}}\left[\phi_{i\ell} + \frac{(\Delta\phi^{+}_{i\ell})^2}{\Delta\phi^{*+}_{i\ell}}\right]\right]$$

$$= \sum_i \frac{(\Delta t_i^+)^2 D''^{+}_{i}}{\Delta t_i^{*+}} - \sum_{i,\ell} D''^{+}_{\ell}\left[\frac{(\Delta t_i^+)^2\phi_{i\ell}}{\Delta t_i^{*+}} + \frac{(\Delta t_i^+)^2(\Delta\phi^{+}_{i\ell})^2}{\Delta t_i^{*+}\,\Delta\phi^{*+}_{i\ell}}\right].$$

By using (39) we obtain

$$\sum_i (\Delta t_i^+)^2 \sum_\ell D''_{i\ell}(\bar{\phi}_{i\ell})^2 \le \sum_i \frac{(\Delta t_i^+)^2 D_i''^+}{\Delta t_i^{*+}} - \sum_\ell \frac{(\Delta t_\ell^+)^2 D_\ell''^+}{\Delta t_\ell^{*+}} + \sum_{i,\ell} \frac{D_\ell''^+ t_i (\Delta\phi_{i\ell}^+)^2}{\Delta\phi_{i\ell}^{*+}}$$

$$= \sum_{i,\ell} \frac{D_\ell''^+ t_i (\Delta\phi_{i\ell}^+)^2}{\Delta\phi_{i\ell}^{*+}} \tag{49}$$

Similarly we obtain

$$\sum_i (\Delta t_i^-)^2 \sum_\ell D''_{i\ell}(\bar{\phi}_{i\ell})^2 \le \sum_{i,\ell} \frac{D_\ell''^- t_i (\Delta\phi_{i\ell}^-)^2}{\Delta\phi_{i\ell}^{*-}} . \tag{50}$$

By combining (48)-(50) and using the constraint (24) we obtain the desired relation (47). Q.E.D.

The algorithm can now be completely defined. After the routing increment $\Delta\phi^*$ is calculated in a distributed manner by means of algorithm (8),(9), each node i computes the quantities Δt_i^{*+} and Δt_i^{*-}. This is done recursively and in a distributed manner by means of equations (34),(35) starting from the "most upstream" nodes and proceeding downstream towards the destination. When this downstream propagation of information reaches the destination indicating that all nodes have completed the computation of Δt_i^{*+} and Δt_i^{*-}, the destination gives the signal for initiation of the second phase of the iteration which consists of computation of the actual routing increments $\Delta\tilde{\phi}_i$. To do this each node i must receive the values of $\bar{D}'_\ell$, $D_\ell''^+$, and $D_\ell''^-$ from its downstream neighbors ℓ and then determine the increments $\Delta\tilde{\phi}_{i\ell}$ which minimize $Q_i(\Delta\phi_i)$ subject to the constraint (24) and the new routing variables

$$\bar{\phi}_{i\ell} = \phi_{i\ell} + \Delta\tilde{\phi}_{i\ell}.$$

Then node i proceeds to compute $\bar{D}'_i$, $D_i''^+$, and $D_i''^-$ via (30), (41), and (42) and broadcasts these values to all upstream neighbors. Thus proceeding recursively upstream from the destination each node computes the actual routing increments $\Delta\tilde{\phi}_i$ in much the same way as the trial routing increments $\Delta\phi_i^*$ were computed earlier.

It is shown in Appendix B that if the starting flow vector f^o is sufficiently close to being optimal then the algorithm just described reduces the value of the objective function at each iteration and converges to the optimal value. We cannot expect to be able to guarantee theoretical convergence when the starting routing variables are far from optimal since this is not a generic property of Newton's method which the algorithm attempts to approximate. However in a large number of computational experiments with objective functions typically arising in communication networks and starting flow vectors which were far from optimal [5] we have never observed divergence or an increase of the value of the objective function in a single iteration. In any case it is possible to prove a global convergence result for the version of the algorithm whereby the expression $Q_i(\Delta\phi_i)$ is replaced by

$$Q_i^\alpha(\Delta\phi_i) = \sum_\ell [(D'_{i\ell} + \overline{D}'_\ell)\Delta\phi_{i\ell} + \frac{1}{2\alpha}(t_i D''_{i\ell} + \beta_{i\ell})(\Delta\phi_{i\ell})^2] \tag{51}$$

where α is a sufficiently small positive scalar stepsize. In Appendix B we show that by choosing α sufficiently small it is possible to guarantee a reduction of the objective function at each iteration for any starting point $\phi^o \varepsilon \Phi$. This fact can be used to prove a convergence result similar to the one of Proposition 2.

Appendix A: Proof of Proposition 2

The proof of Proposition 2 to be given in this appendix applies to the "all-at-once" version of algorithm (8),(9), i.e. the one where at each iteration k every node i solves problem (9) for all destinations j and adjusts the corresponding routing variables according to (8). A nearly identical proof applies to the "one-at-a-time" version (see Gafni [11]). The destination of flows, routing variables, etc. will be denoted within parentheses. Thus for example $\phi_{i\ell}(j)$ denotes the routing variable of link (i,ℓ) for destination j.

The following lemma bears close similarity in both statement and proof as Lemma 5 of Gallager [1]. The proof will be omitted, but may be found in [11].

Lemma A.1: Let the assumptions of Proposition 2 hold. There exists a scalar $\bar{\alpha}\varepsilon(0,1]$ (depending on D_o, λ, and Λ) such that, for every $\alpha\varepsilon(0,\bar{\alpha}]$, the corresponding sequence $\{\phi^k\}$ generated by algorithm (8),(9) satisfies

$$D(\phi^{k+1},r)-D(\phi^k,r) \leq -\rho\sum_{i,j} [t_i^k(j)]^2|\Delta\phi_i^k(j)|^2, \quad k=0,1,\ldots \tag{A.1}$$

$$\lim_{k\to\infty} t_i^k(j)|\Delta\phi_i^k(j)| = 0, \quad \forall i,j=1,2,\ldots,N,\ i\neq j \tag{A.2}$$

$$\lim_{k\to\infty} |f_{i\ell}^{k+1}(j) - f_{i\ell}^k(j)| = 0, \quad \forall (i,\ell)\varepsilon L,\ i,j=1,2,\ldots,N,\ i\neq j \tag{A.3}$$

where ρ is some positive scalar (depending on α, D_o, λ, Λ), $t_i^k(j)$ denotes the total traffic arriving at node i which is destined for j when the routing is ϕ^k, $\Delta\phi_i^k(j) = \phi_i^{k+1}(j)-\phi_i^k(j)$, and $f_{i\ell}^k(j)$, $f_{i\ell}^{k+1}(j)$ are the flows on link (i,ℓ) destined for j and corresponding to ϕ^k, ϕ^{k+1} respectively.

The following lemma provides a key fact.

Lemma A.2: If $\alpha\varepsilon(0,\bar{\alpha}]$ where $\bar{\alpha}$ is as in Lemma A.1 and $\{\phi^k\}$ is a corresponding sequence generated by algorithm (8),(9) there holds

$$\lim_{k\to\infty} [\bar{\Delta}_i^k(j) - \underline{\Delta}_i^k(j)] = 0 , \quad \forall\, i,j = 1,\ldots,N,\ i \neq j, \tag{A.4}$$

where for all i,j,k

$$\bar{\Delta}_i^k(j) = \max_{\ell}\{\delta_{i\ell}^k(j) \,|\, \ell \ \varepsilon\ 0(i),\ \phi_{i\ell}^{k+1}(j) > 0\} \tag{A.5}$$

$$\underline{\Delta}_i^k(j) = \min_{\ell} \{\delta_{i\ell}^k(j) \quad | \ \ell \ \varepsilon\ 0(i),\ \ell \notin B(i,\phi^k)(j)\} \tag{A.6}$$

$$\delta_{i\ell}^k(j) = D_{i\ell}'(f_{i\ell}^k) + \frac{\partial D(\phi^k,r)}{\partial r_i(j)} . \tag{A.7}$$

Proof: From a necessary condition for optimality for problem (9) we obtain

$$[\delta_i^k(j) + \frac{t_i^k(j)}{\alpha} \quad M_i^k(j)\Delta\phi_i^k(j)]^T[\phi_i(j) - \phi_i^{k+1}(j)] \ \geq \ 0 \tag{A.8}$$

for all $\phi_i(j)$ which are feasible in problem (9). Let $\bar{\ell}$ and $\underline{\ell}$ be such that

$$\delta_{i\bar{\ell}}^k(j) = \bar{\Delta}_i(j), \quad \delta_{i\underline{\ell}}^k(j) = \underline{\Delta}_i^k(j).$$

If $\bar{\ell} \neq \underline{\ell}$ we define $\phi_i^*(j)$ to be the vector with components

$$\phi_{i\ell}^*(j) = \begin{cases} \phi_{i\ell}^{k+1}(j)-\varepsilon & \text{if } \ell = \bar{\ell} \\ \phi_{i\ell}^{k+1}(j)+\varepsilon & \text{if } \ell = \underline{\ell} \\ \phi_{i\ell}^{k+1}(j) & \text{otherwise} \end{cases}$$

where $\varepsilon>0$ is small enough so that $\phi^{k+1}_{i\bar{\ell}}(j)-\varepsilon>0$. By definition of $\bar{\Delta}^k_i(j)$ such an ε exists and by feasibility of $\phi^{k+1}_i(j)$ we have that $\phi^*_i(j)$ is also feasible. Substituting $\phi^*_i(j)$ in (A.8) in place of $\phi_i(j)$ we obtain

$$\varepsilon[\bar{\Delta}^k_i(j) - \underline{\Delta}^k_i(j)] \leq \frac{\varepsilon}{\alpha}[\mu^k_{i\underline{\ell}}(j) - \mu^k_{i\bar{\ell}}(j)]$$

where $\mu^k_{i\underline{\ell}}(j)$ and $\mu^k_{i\bar{\ell}}(j)$ are the $\underline{\ell}$ and $\bar{\ell}$ elements of the vector $t^k_i(j)M^k_i(j)\Delta\phi^k_i(j)$. Using the assumption that all elements of $M^k_i(j)$ are bounded above by Λ we obtain

$$0 \leq \bar{\Delta}^k_i(j) - \underline{\Delta}^k_i(j) \leq \frac{2\Lambda}{\alpha} t^k_i(j) \sum_{\ell} |\Delta\phi^k_{i\ell}(j)|.$$

This relation holds also if $\underline{\ell} = \bar{\ell}$ since then $\bar{\Delta}^k_i(j) = \underline{\Delta}^k_i(j)$. From (A.2) we see that the right hand side tends to zero. Equation (A.4) follows. Q.E.D.

Given any set of routing variables $\phi \in \Phi$ there is a unique corresponding set of flows $f_{i\ell}(j)$. If we view the first derivative $D'_{i\ell}(f_{i\ell})$ as the length of link (i,ℓ) then the corresponding shortest distance from any node i to any other node j is well defined and will be denoted by $S_{ij}(\phi)$. It is easily seen using equation (6) that a sufficient condition for optimality of a set of routing variables $\hat{\phi} \in \Phi$ is

$$S_{ij}(\hat{\phi}) = \frac{\partial D(\hat{\phi},r)}{\partial r_i(j)}, \quad \forall i,j = 1,\ldots,N,\ i \neq j. \tag{A.9}$$

Furthermore there holds

$$S_{ij}(\phi) \leq \frac{\partial D(\phi,r)}{\partial r_i(j)}, \ \forall \phi \in \Phi \text{ and } i,j = 1,\ldots,N,\ i \neq j. \tag{A.10}$$

We have the following lemma:

Lemma A.3: If $\alpha\varepsilon(0,\bar{\alpha}]$ where $\bar{\alpha}$ is as in Lemma A.1, $\{\phi^k\}$ is a corresponding sequence of algorithm (8),(9), $m \geq 1$ is an integer, and $\tilde{K}$ is an infinite index set such that the subsequences $\{\phi^k\}_{k\in\tilde{K}}$ and $\{\phi^{k-m}\}_{k\in\tilde{K}}$ converge to $\hat{\phi}$ and $\tilde{\phi}$ respectively then

$$f_{i\ell}(j)(\hat{\phi},r) = f_{i\ell}(j)(\tilde{\phi},r), \quad \forall (i,\ell)\in L,\ j = 1,\ldots,N \tag{A.11}$$

$$S_{ij}(\hat{\phi}) = S_{ij}(\tilde{\phi}) \quad , \quad \forall i,j = 1,2,\ldots,N \tag{A.12}$$

Proof: Equation (A.11) follows from (A.3), and equation (A.12) follows from the fact that $S_{ij}(\phi)$ depends on ϕ only through the flows $f_{i\ell}(j)(\phi,r)$.

Q.E.D.

We will use "two dimensional induction" to show that the limit of any convergent subsequence of $\{\phi^k\}$ satisfies the sufficient condition for optimality (A.9). Lemma A.4 that follows represents the basic step of the induction proof. We use repeatedly the fact that if some property 1 holds for all k with $k > k_1$ and some property 2 holds for all k with $k > k_2$ then both hold for all k with $k > \max(k_1, k_2)$. In what follows we will express this by writing "if 1 holds for all k large enough and 2 holds for all k large enough, then both hold for all k large enough".

Lemma A.4: Let $\alpha\varepsilon(0,\bar{\alpha}]$ where $\bar{\alpha}$ is as in Lemma A.1, let $\{\phi^k\}$ be a corresponding sequence generated by algorithm (8),(9) and let $\{\phi^{k-1}\}_{k\in\tilde{K}} \to \tilde{\phi}$ and $\{\phi^k\}_{k\in\tilde{K}} \to \hat{\phi}$ be two convergent subsequences of $\{\phi^k\}$. For each j let $S_j(\tilde{\phi})$ be the set of distances $\{S_{ij}(\tilde{\phi}) \mid i \in N\}$. Let $S_1(j),\ldots,S_p(j)$, $p \le N$ be the distinct elements of the set $S_j(\tilde{\phi})$ and assume without loss of generality that $0 = S_1(j) < S_2(j) <\ldots< S_p(j)$. Denote

$$I_q(j) = \{i \mid S_{ij}(\tilde{\phi}) \le S_q(j)\}, \quad q = 1,\ldots,p. \tag{A.13}$$

Assume that for some integer q we have:

a) $$\frac{\partial D(\hat{\phi},r)}{\partial r_i(j)} = \frac{\partial D(\tilde{\phi},r)}{\partial r_i(j)} = S_{ij}(\tilde{\phi}) \quad \forall\, i \in I_q(j), \quad j = 1,\ldots,N \tag{A.14}$$

b) For all k large enough, $k \in \tilde{K}$, and for any j, if $\phi_{mn}^{k-1}(j) > 0$ and $m \in I_q(j)$ then $\dfrac{\partial D(\phi^{k-1},r)}{\partial r_m(j)} > \dfrac{\partial D(\phi^{k-1},r)}{\partial r_n(j)}$.

Then:

a')

$$\frac{\partial D(\hat{\phi},r)}{\partial r_i(j)} = S_{ij}(\hat{\phi}) \qquad \forall\, i \in I_{q+1}(j),\ j = 1,\ldots,N \tag{A.15}$$

b') For all k large enough, $k \in \tilde{K}$, and for any j, if $\phi_{mn}^{k}(j) > 0$ and $m \in I_{q+1}(j)$ then $\dfrac{\partial D(\phi^{k},r)}{\partial r_m(j)} > \dfrac{\partial D(\phi^{k},r)}{\partial r_n(j)}$.

<u>Proof</u>: Let i be such that $i \in I_{q+1}(j)$

and denote

$$\ell_i(j) = \{\ell \,|\, S_{ij}(\tilde{\phi}) = D'_{i\ell}[f_{i\ell}(\tilde{\phi},r)] + S_{\ell j}(\tilde{\phi}),\ \ell \in O(i)\}\,.$$

By the definition of shortest distance we have

$$S_{ij}(\tilde{\phi}) < D'_{i\ell}[f_{i\ell}(\tilde{\phi},r)] + S_{\ell j}(\tilde{\phi}) \quad \forall\, \ell \notin \ell_i(j),\ \ell \in O(i).$$

Using (A.10) and the above equation

$$S_{ij}(\tilde{\phi}) < D'_{i\ell}[f_{i\ell}(\tilde{\phi},r) + \frac{\partial D(\tilde{\phi},r)}{\partial r_\ell(j)}$$

or equivalently

$$S_{ij}(\tilde{\phi}) < \delta_{i\ell}(j)(\tilde{\phi},r) \quad \forall\ \ell \notin \ell_i(j),\ \ell \in O(i).$$

By the assumption $D'_{i\ell} > 0$ and the fact $i \in I_{q+1}(j)$, we have

$$\ell_i(j) \subset I_q(j) \qquad j = 1,\ldots,N$$

Therefore by using hypothesis a) we have

$$\delta_{i\ell}(j)(\tilde{\phi},r) = D'_{i\ell}[f_{i\ell}(\tilde{\phi},r)] + \frac{\partial D(\tilde{\phi},r)}{\partial r_\ell(j)} = D'_{i\ell}[f_{i\ell}(\tilde{\phi},r)] + S_{\ell j}(\tilde{\phi}) \quad \text{(A.16)}$$

$$= S_{ij}(\tilde{\phi}) \qquad \forall\, \ell\varepsilon\ell_i(j), \qquad j = 1,\ldots,N \quad .$$

Since $0(i)$ is a finite set there exists $\varepsilon > 0$ such that

$$\delta_{iw}(j)(\tilde{\phi},r) - \varepsilon > \delta_{i\ell}(j)(\tilde{\phi},r) \quad \forall\, w \notin \ell_i(j),\ w\varepsilon 0(i),\ \ell \,\varepsilon\, \ell_i(j),\ j = 1,\ldots,N$$

Since $\delta_{i\ell}(j)(\phi,r)$ is continuous in ϕ and $\{\phi^{k-1}\}_{k\varepsilon\tilde{K}}$ is a convergent sequence, we get that for all k large enough, $k \,\varepsilon\, \tilde{K}$

$$\delta_{iw}(j)(\phi^{k-1},r) > \delta_{i\ell}(\phi^{k-1},r) + \frac{\varepsilon}{2} \qquad \forall\, w \notin \ell_i(j),\ w\varepsilon 0(i),\ \ell \,\varepsilon\, \ell_i(j),\ j = 1,\ldots,N.$$

Also $\frac{\partial D(\phi,r)}{\partial r_i(j)}$, $1 \leq i,\ j \leq N$, is continuous in ϕ and therefore by Lemma 2, (A.16) and hypothesis a), for all k large enough, $k \,\varepsilon\, \tilde{K}$

$$\frac{\partial D(\phi^{k-1},r)}{\partial r_i(j)} > \frac{\partial D(\phi^{k-1},r)}{\partial r_\ell(j)} \qquad \forall\, \ell \,\varepsilon\, \ell_i(j), \qquad j = 1,\ldots,N$$

which together with hypothesis b) and the definition of $B(\phi;i)(j)$ implies that for all k large enough, $k \,\varepsilon\, \tilde{K}$

$$\ell_i(j) \cap B(\phi^{k-1};i)(j) = \emptyset, \qquad j = 1,\ldots,N \quad . \qquad \text{(A.18)}$$

Lemma A.2 combined with (A.17) and (A.18) implies that for all k large enough $k \,\varepsilon\, \tilde{K}$

$$\phi^k_{iw}(j) = 0 \qquad \forall\, w \notin \ell_i(j), \qquad j = 1,\ldots,N \qquad \text{(A.19)}$$

and taking the limit

$$\hat{\phi}_{iw}(j) = 0 \qquad w \notin \ell_i(j), \qquad j = 1,\ldots,N \tag{A.20}$$

Using (A.20), Lemma A.3 and hypothesis a) we have

$$\begin{aligned}
\frac{\partial D(\hat{\phi},r)}{\partial r_i(j)} &= \sum_{\ell} \hat{\phi}_{i\ell}(j) \left[D'_{ik}[f_{ik}(\hat{\phi},r)] + \frac{\partial D(\hat{\phi},r)}{\partial r_\ell(j)}\right] \\
&= \sum_{\ell \in \ell_i(j)} \hat{\phi}_{i\ell}(j) \left[D'_{ik}[f_{ik}(\hat{\phi},r)] + \frac{\partial D(\hat{\phi},r)}{\partial r_\ell(j)}\right] \\
&= \sum_{\ell \in \ell_i(j)} \hat{\phi}_{i\ell}(j) \left[D'_{ik}[f_{ik}(\tilde{\phi},r)] + \frac{\partial D(\tilde{\phi},r)}{\partial r_\ell(j)}\right] \\
&= S_{ij}(\tilde{\phi}) = S_{ij}(\hat{\phi})
\end{aligned}$$

This together with part a) of the hypothesis establishes a').

To see b'), notice that by continuity of $\frac{\partial D(\phi,r)}{\partial r_i(j)}$ in ϕ and the preceding equation we have that for all k la[illegible] nough, $k \in \tilde{K}$

$$\frac{\partial D(\phi^k,r)}{\partial r_i(j)} > \frac{\partial D(\phi^k,r)}{\partial r_\ell(j)} \qquad \forall\, \ell \in \ell_i(j), \quad j = 1,\ldots,N \ . \tag{A.21}$$

Equations (A.21) and (A.19) hold for all $i \in I_{q+1}(j)$ and b') follows.

Q.E.D.

By now we have developed all the machinery for the convergence proof of Proposition 2. We will simply make repeated application of Lemma A.4 for the proper sequences.

Proof of Proposition 2: Take $\bar{\alpha}$ to be as in Lemma A.1, let $\alpha\varepsilon(0,\bar{\alpha}]$ and let $\{\phi^k\}$ be a corresponding sequence generated by algorithm (8),(9). The sequence $\{\phi^k\}$ belongs to a compact set and therefore there exists a convergent subsequence $\{\phi^k\}_{k\varepsilon K} \to \phi$. The sequence $\{\phi^{k-1}\}_{k\varepsilon K}$ has a convergent subsequence $\{\phi^{k-1}\}_{k\varepsilon K_1} \to \phi_1$, $K_1 \subset K$. The sequence $\{\phi^{k-2}\}_{k\varepsilon K_1}$ has a convergent subsequence $\{\phi^{k-2}\}_{k\varepsilon K_2} \to \phi_2$, $K_2 \subset K_1$. Proceeding this way we get a convergent subsequence

$$\{\phi^{k-N+1}\}_{k\varepsilon K_{N-1}} \to \phi_{N-1},\ K_{N-1} \subset K_{N-2}\ .$$

We have $K_{N-1} \subset K_{N-2} \subset \ldots \subset K$ and

$$\{\phi^{k-N+1}\}_{k\varepsilon K_{N-1}} \to \phi_{N-1}\ , \ldots,\ \{\phi^{k-1}\}_{k\varepsilon K_{N-1}} \to \phi_1,\ \{\phi^k\}_{k\varepsilon K_{N-1}} \to \phi$$

By Lemma A.3 the shortest distances which correspond to $\phi_{N-1}, \phi_{N-2},\ldots,\phi$ are the same. As a result, in what follows, when we mention the set $I_q(j)$ we need not specify the limit point ϕ_i to which it corresponds.

Let $\tilde{K}$ in Lemma A.4 be K_{N-1}. For each destination j, the only element in $I_1(j)$ is j and therefore the assumptions of Lemma A.4 hold for $I_1(j)$ and the pairs of sequences

$$\{\{\phi^k\}_{k\varepsilon\tilde{K}},\ \{\phi^{k-1}\}_{k\varepsilon\tilde{K}}\},\ \{\{\phi^{k-1}\}_{k\varepsilon\tilde{K}},\ \{\phi^{k-2}\}_{k\varepsilon\tilde{K}}\},\ldots,\{\{\phi^{k-N+2}\}_{k\varepsilon\tilde{K}},\ \{\phi^{k-N+1}\}_{k\varepsilon\tilde{K}}\}.$$

Applying Lemma A.4 for q = 1, we obtain that its hypothesis holds for q = 2 and the pairs of sequences $([\{\phi^k\}_{k\varepsilon\tilde{K}},\ \{\phi^{k-1}\}_{k\varepsilon\tilde{K}}],\ldots,$ $[\{\phi^{k-N+3}\}_{k\varepsilon\tilde{K}},\ \{\phi^{k-N+2}\}_{k\varepsilon\tilde{K}}])$. Proceeding this way we note that the hypothesis of Lemma A.4 holds for $q = N-1$ and the pair $\left(\{\phi^k\}_{k\epsilon\tilde{K}}, \{\phi^{k-1}\}_{k\epsilon K}\right)$. Applying Lemma A.4 again we see that the conclusion of its part a') holds for $q = N-1$, i.e., equation (A.15) holds for $I_N(j)$, $j = 1,\ldots,N$. Since every node in the network belongs to $I_N(j)$, $j = 1,\ldots,N$, it follows that (A.9) is satisfied, and hence ϕ is optimal.

Q.E.D.

Appendix B

In this appendix we analyze the descent properties of the algorithm of Section 4. We assume a single destination but the proof extends trivially to the case where we have multiple destinations and the algorithm is operated in the one destination at a time mode. In view of the fact that each function $D_{i\ell}$ is strictly convex it follows that there is a unique optimal set of total link flows $\{f^*_{i\ell} \mid (i,\ell)\varepsilon L\}$. It is clear that given any $\varepsilon>0$ there exists a scalar γ_ε such that for all feasible total link flow vectors f satisfying

$$|f_{i\ell} - f^*_{i\ell}| \le \gamma_\varepsilon \quad , \quad \forall (i,\ell)\varepsilon L \tag{B.1}$$

we have

$$\frac{1}{1+\varepsilon} D''_{i\ell}(f^*_{i\ell}) \le D''_{i\ell}(f_{i\ell}) \le (1+\varepsilon)D''_{i\ell}(f^*_{i\ell}), \quad \forall (i,\ell)\varepsilon L. \tag{B.2}$$

The strict positivity assumption on $D''_{i\ell}$ also implies that for each $\gamma_\varepsilon>0$ there exists a scalar $\delta(\gamma_\varepsilon)$ such that every feasible f satisfying $\sum_{i,\ell} D_{i\ell}(f_{i\ell}) \le \delta(\gamma_\varepsilon)$ also satisfies (B.1) and hence also (B.2). Furthermore $\delta(\gamma_\varepsilon)$ can be taken arbitrarily large provided γ_ε is sufficiently large. We will make use of this fact in the proof of the subsequent result.

<u>Proposition B.1</u>: Let ϕ and $\bar{\phi}$ be two successive vectors of routing variables generated by the algorithm of Section 4 (with stepsize $\alpha=1$) and let f and $\bar{f}$ be the corresponding vectors of link flows. Assume that for some ε with $0 < \varepsilon < \frac{2}{\sqrt{3}} - 1$ we have

$$\sum_{i,\ell} D_{i\ell}(f_{i\ell}) \le \delta(\gamma_\varepsilon) \tag{B.3}$$

where γ_ε is the scalar corresponding to ε as in (B.1), (B.2), and $\delta(\gamma_\varepsilon)$ is such that (B.1) [and hence also (B.2)] holds for all feasible f satisfying (B.3). Then

$$D(\bar{\phi},r) - D(\phi,r) \leq - \rho(\varepsilon) \sum_{(i,\ell)\varepsilon L} t_i(D''_{i\ell}+\beta_{i\ell})(\bar{\phi}_{i\ell}-\phi_{i\ell})^2 \qquad (B.4)$$

where $\rho(\varepsilon) = \frac{1-4\varepsilon-2\varepsilon^2}{2} > 0$ for all ε with $0 < \varepsilon < \sqrt{\frac{3}{2}} - 1$.

Proof: Let $\tilde{\Delta f}$ be the increment of flow corresponding to the increment $\tilde{\Delta\phi} = \bar{\phi}-\phi$. We have

$$D(\bar{\phi},r) - D(\phi,r) = \sum_{i,\ell} \Delta f_{i\ell} D'_{i\ell}(f_{i\ell}+\eta\Delta f_{i\ell})\Big|_{\eta=0} + \frac{1}{2}\sum_{i,\ell} (\Delta f_{i\ell})^2 D''_{i\ell}(f_{i\ell}+\eta\Delta f_{i\ell})\Big|_{\eta=\eta^*}$$

for some $\eta^*\varepsilon[0,1]$. Denoting $D''_{i\ell}(f_{i\ell}+\eta^*\Delta f_{i\ell}) = \hat{D}''_{i\ell}$ and using an argument similar to the one employed in Section 4 [cf. (28)-(31)] we obtain

$$D(\bar{\phi},r) - D(\phi,r) = \sum_{i,\ell} t_i\tilde{\Delta\phi}_{i\ell}(D'_{i\ell}+\bar{D}'_\ell) + \sum_{i,\ell} (\hat{D}''_{i\ell}-D''_{i\ell})\Delta t_i\bar{\phi}_{i\ell}t_i\tilde{\Delta\phi}_{i\ell}$$

$$+ \frac{1}{2}\sum_{i,\ell} \hat{D}''_{i\ell}[(\Delta t_i\bar{\phi}_{i\ell})^2 + (t_i\tilde{\Delta\phi}_{i\ell})^2]. \qquad (B.5)$$

We will derive upper bounds for each of the three terms in the right side of (B.5).

From the necessary condition for $\tilde{\Delta\phi}_i$ to minimize the function $Q_i(\Delta\phi_i)$ of (45) subject to the constraint (24) we obtain

$$\sum_\ell [D'_{i\ell}+\bar{D}'_\ell + (t_iD''_{i\ell}+\beta_{i\ell})\tilde{\Delta\phi}_{i\ell}]\tilde{\Delta\phi}_{i\ell} \leq 0$$

or

$$\sum_\ell (D'_{i\ell}+\bar{D}'_\ell)\tilde{\Delta\phi}_{i\ell} \leq - \sum_\ell (t_iD''_{i\ell}+\beta_{i\ell})(\tilde{\Delta\phi}_{i\ell})^2. \qquad (B.6)$$

There is no loss of generality in replacing each function $D_{i\ell}$ by a function $\bar{D}_{i\ell}$ which is continuously differentiable, is identical with $D_{i\ell}$ on the set of flows satisfying (B.1) and is quadratic outside this set, provided that, as part of the subsequent proof, we show that $\sum_{i,\ell} D_{i\ell}(f_{i\ell}+\eta\Delta f_{i\ell}) \leq \delta(\gamma_\varepsilon)$ for all $\eta\varepsilon[0,1]$. By using this device we can assume that $D''_{i\ell}$ satisfies (B.2) for all $f_{i\ell}$. Hence from (B.2)

$$\frac{\hat{D}''_{i\ell} - D''_{i\ell}}{D''_{i\ell}} \leq (1+\varepsilon)^2 - 1 \tag{B.7}$$

$$\frac{\hat{D}''_{i\ell}}{D''_{i\ell}} \leq (1+\varepsilon)^2. \tag{B.8}$$

Using (47), (B.7), the Cauchy-Schwartz inequality and the arithmetic-geometric inequality we have

$$\begin{aligned}
\sum_{i,\ell}(\hat{D}''_{i\ell}-D''_{i\ell})\Delta t_i\bar{\phi}_{i\ell}t_i\Delta\phi_{i\ell} &\leq [(1+\varepsilon)^2-1]\sum_{i,\ell} D''_{i\ell}|\Delta t_i|\bar{\phi}_{i\ell}t_i|\Delta\tilde{\phi}_{i\ell}| \\
&\leq [(1+\varepsilon)^2-1][\sum_{i,\ell} D''_{i\ell}(\Delta t_i\bar{\phi}_{i\ell})^2]^{\frac{1}{2}}[\sum_{i,\ell} D''_{i\ell}(t_i\Delta\tilde{\phi}_{i\ell})^2]^{\frac{1}{2}} \\
&\leq [(1+\varepsilon)^2-1][\sum_{i,\ell} \beta_{i\ell}t_i(\Delta\tilde{\phi}_{i\ell})^2]^{\frac{1}{2}}[\sum_{i,\ell} D''_{i\ell}(t_i\Delta\tilde{\phi}_{i\ell})^2]^{\frac{1}{2}} \qquad \text{(B.9)} \\
&\leq \frac{1}{2}[(1+\varepsilon)^2-1][\sum_{i,\ell}\beta_{i\ell}t_i(\Delta\tilde{\phi}_{i\ell})^2 + \sum_{i,\ell} D''_{i\ell}(t_i\Delta\tilde{\phi}_{i\ell})^2] \\
&= \frac{1}{2}[(1+\varepsilon)^2-1]\sum_i t_i \sum_\ell (t_iD''_{i\ell}+\beta_{i\ell}((\Delta\tilde{\phi}_{i\ell})^2.
\end{aligned}$$

Using again (47) and (B.8) we obtain for each i

$$\begin{aligned}
&\sum_{i,\ell}\hat{D}''_{i\ell}[(\Delta t_i\bar{\phi}_{i\ell})^2 + (t_i\Delta\tilde{\phi}_{i\ell})^2] \\
&\leq (1+\varepsilon)^2\sum_{i,\ell}[D''_{i\ell}(\Delta t_i\bar{\phi}_{i\ell})^2 + D''_{i\ell}(t_i\Delta\tilde{\phi}_{i\ell})^2] \qquad \text{(B.10)} \\
&\leq (1+\varepsilon)^2\sum[t_i\beta_{i\ell}(\Delta\tilde{\phi}_{i\ell})^2 + D''_{i\ell}(t_i\Delta\tilde{\phi}_{i\ell})^2] \\
&= (1+\varepsilon)^2\sum_i t_i\sum_\ell (t_iD''_i+\beta_{i\ell})(\Delta\tilde{\phi}_{i\ell})^2.
\end{aligned}$$

By combining now (B.5), (B.6), (B.9), and (B.10) we obtain

$$D(\bar{\phi},r) - D(\phi,r) \leq \left[-1+\frac{(1+\varepsilon)^2-1}{2} + \frac{(1+\varepsilon)^2}{2}\right] \sum_{(i,\ell)\varepsilon L} t_i(t_i D''_{i\ell} + \beta_{i\ell})(\Delta\tilde{\phi}_{i\ell})^2$$

$$= -\rho(\varepsilon) \sum_{(i,\ell)\varepsilon L} t_i(t_i D''_{i\ell} + \beta_{i\ell})(\bar{\phi}_{i\ell}-\phi_{i\ell})^2$$

and (B.4) is proved. It is also straightforward to verify that $\rho(\varepsilon) > 0$ for ε in the interval $(0,\sqrt{\frac{3}{2}} -1)$. Q.E.D.

The preceding proposition shows that the algorithm of Section 4 does not increase the value of the objective function once the flow vector f enters a region of the form $\{f| \sum_{i,\ell} D_{i\ell}(f_{i\ell}) \leq \delta(\gamma_\varepsilon)\}$, and that the size of this region increases as the third derivative of $D_{i\ell}$ becomes smaller. Indeed if each function $D_{i\ell}$ is quadratic then (B.2) is satisfied for all $\varepsilon>0$ and the algorithm will not increase the value of the objective for all f.

The preceding analysis can be easily modified to show that if we introduce a stepsize α as in (51) then the algorithm of Section 4 is a descent algorithm at all flows in the region $\{f| \sum_{i,\ell} D_{i\ell}(f_{i\ell}) \leq \delta(\gamma_\varepsilon)\}$ where

$$0 < \varepsilon < \sqrt{\frac{2 + \alpha}{2\alpha}} - 1.$$

From this it follows that given any starting point $\phi^o \varepsilon \Phi$, there exists a scalar $\bar{\alpha}>0$ such that for all stepsizes $\alpha\varepsilon(0,\bar{\alpha}]$ the algorithm of Section 4 does not increase the value of the objective function at each subsequent iteration.

References

[1] Gallager, R.G., "A Minimum Delay Routing Algorithm Using Distributed Computation", IEEE Trans. on Communication, Vol. COM-25, 1977, pp. 73-85.

[2] Goldstein, A.A., "Convex Programming in Hilbert Space", Bull. Amer. Math. Soc., Vol. 70, 1964, pp. 709-710.

[3] Levitin, E.S. and B.T. Polyak, "Constrained Minimization Problems", USSR Comput. Math. Math. Phys., Vol. 6, 1966, pp. 1-50.

[4] Bertsekas, D.P., "Algorithms for Nonlinear Multicommodity Network Flow Problems", in Proc. of International Symposium on Systems Optimization and Analysis, A. Bensoussan and J.L. Lions (eds.), Springer-Verlag, 1979, pp. 210-224.

[5] Bertsekas, D.P., Gafni, E., and Vastola, K.S., "Validation of Algorithms for Optimal Routing of Flow in Networks", Proc. of IEEE Conf. on Decision and Control, San Diego, Calif., Jan. 1979, pp. 220-227.

[6] Vastola, K.S., "A Numerical Study of Two Measures of Delay for Network Routing", M.S. Thesis, Dept. of Electrical Engineering, Univ. of Ill, Urbana, Ill., Sept. 1979.

[7] Bertsekas, D.P., "A Class of Optimal Routing Algorithms for Communication Networks", Proc. of the Fifth International Conference on Computer Communication (ICCC-80), Atlanta, Ga., Oct. 1980, pp. 71-76.

[8] Bertsekas, D.P., and Gafni, E., "Projection Methods for Variational Inequalities with Application to the Traffic Assignment Problem", Report LIDS-P-1043, Mass. Institute of Technology, Cambridge, Mass., June 1980, Math. Programming Studies, (to appear).

[9] Golestaani, S.J., "A Unified Theory of Flow Control and Routing in Data Communication Networks", Report LIDS-TH-963, Laboratory for Information and Decision Systems, Mass. Institute of Technology, Cambridge, Mass., Jan. 1980.

[10] Gallager, R.G., and Golestaani, S.J., "Flow Control and Routing Algorithms for Data Networks", Proc. of the Fifth International Conference on Computer Communication (ICCC-80), Atlanta, Ga., Oct., 1980, pp. 779-784.

[11] Gafni, E., "Convergence of a Routing Algorithm", Report LIDS-TH-907, Laboratory for Information and Decision Systems, Mass. Institute of Technology, Cambridge, Mass., May 1979.

Distribution List

Defense Documentation Center Cameron Station Alexandria, Virginia 22314	12 Copies
Assistant Chief for Technology Office of Naval Research, Code 200 Arlington, Virginia 22217	1 Copy
Office of Naval Research Information Systems Program Code 437 Arlington, Virginia 22217	2 Copies
Office of Naval Research Branch Office, Boston 495 Summer Street Boston, Massachusetts 02210	1 Copy
Office of Naval Research Branch Office, Chicago 536 South Clark Street Chicago, Illinois 60605	1 Copy
Office of Naval Research Branch Office, Pasadena 1030 East Greet Street Pasadena, California 91106	1 Copy
Naval Research Laboratory Technical Information Division, Code 2627 Washington, D.C. 20375	6 Copies
Dr. A. L. Slafkosky Scientific Advisor Commandant of the Marine Corps (Code RD-1) Washington, D.C. 20380	1 Copy

Office of Naval Research Code 455 Arlington, Virginia 22217	1 Copy
Office of Naval Research Code 458 Arlington, Virginia 22217	1 Copy
Naval Electronics Laboratory Center Advanced Software Technology Division Code 5200 San Diego, California 92152	1 Copy
Mr. E. H. Gleissner Naval Ship Research & Development Center Computation and Mathematics Department Bethesda, Maryland 20084	1 Copy
Captain Grace M. Hopper NAICOM/MIS Planning Branch (OP-916D) Office of Chief of Naval Operations Washington, D.C. 20350	1 Copy
Advanced Research Projects Agency Information Processing Techniques 1400 Wilson Boulevard Arlington, Virginia 22209	1 Copy
Dr. Stuart L. Brodsky Office of Naval Research Code 432 Arlington, Virginia 22217	1 Copy
Captain Richard L. Martin, USN Commanding Officer USS Francis Marion (LPA-249) FPO New York 09501	1 Copy