# CENTER FOR PARALLEL OPTIMIZATION

**NONSMOOTH OPTIMIZATION METHODS FOR PARALLEL DECOMPOSITION OF MULTICOMMODITY FLOW PROBLEMS**

**by**

**Renato De Leone, Manlio Gaudioso & Maria Flavia Monaco**

**Computer Sciences Technical Report #1080**

**March 1992**

# NONSMOOTH OPTIMIZATION METHODS FOR PARALLEL DECOMPOSITION OF MULTICOMMODITY FLOW PROBLEMS

RENATO DE LEONE[1], MANLIO GAUDIOSO[2] AND MARIA FLAVIA MONACO[2]

**Abstract.** We develop an iterative algorithm based on right hand side decomposition for the solution of multicommodity network flow problems. At each step of the proposed iterative procedure the coupling constraints are eliminated by subdividing the shared capacity resource among the different commodities and a master problem is constructed which attempts to improve sharing of the resources at each iteration.

As the objective function of the master problem is nonsmooth, we apply to it a new optimization technique which does not require the exact solutions of the single commodity flow subproblems. This technique is based on the notion of $\epsilon$-subgradients, instead of subgradients and is suitable for parallel implementation. Extensions to the nonlinear, convex separable case are also discussed.

**1. Introduction.** The multicommodity network flow problem minimizes the cost of a set of network flows of different commodities sharing the same set of capacity resources.

Various decomposition techniques have been proposed and intensively studied for the solution of large–scale multicommodity flow problems. In fact, decomposition methods are particularly suitable for this class of problems because, removing the set of coupling constraints, the problem decomposes into separate minimum cost flow problems, one for each commodity.

An axcellent description of the different types of decomposition schemes considered in the literature may be found in [9]. Two major types of decomposition schemes are *price directive,* which is based on Dantzig-Wolfe decomposition approach and *resource directive,* which gives rise [7] to nondifferentiable optimization problems. Recently, Pinar and Zenios [13] and Zenios, Pinar and Dembo [17] proposed algorithms that use a linear–quadratic penalty function to eliminate the coupling constraints. In their approach the objective function for the master problem is differentiable but non-separable. Censor, Chajakis and Zenios [3] developed a decomposition algorithm for the quadratic multicommodity network flow problem based on a row-action primal–dual algorithm. Meyer and Schultz [15] and Schultz [14] proposed a decomposition method in conjunction with a shifting barrier method for the coupling constraints.

In this paper we tackle the multicommodity network flow problem by using a resource directive based decomposition technique. First, a set of decision variables representing a feasible sharing of the resource among the different commodities is introduced. Then, the problem is decomposed into minimum cost single commodity network flow

[1] Center for Parallel Optimization, Computer Sciences Department, University of Wisconsin Madison, 1210 West Dayton Street, Madison, WI 53706. The work of this author has been supported by the Air Force Office of Scientific Research Grant AFOSR-89-0410.

[2] Dipartimento di Elettronica, Informatica e Sistemistica, Università della Calabria, 87030 Rende, Cosenza, Italia.

problems. Finally, a master problem is solved in order to find an improved sharing of the resources.

Since the objective function for the master problem is nondifferentiable, we propose an iterative technique derived from the family of bundle methods [11] for finding its minimum. The main advantage of our approach is that the proposed algorithm does not require at each step *exact* evaluation of the objective function. The method differs from that presented in [10], because it does not require increasing precision in the objective function evaluation as the minimum is approached.

The decomposition scheme adopted is particularly suitable for parallel implementation, because the single commodity network flow problems can be solved independently by different processors working in parallel. Approximate solutions of the individual subproblems will be used to evaluate the objective function value of the master problem. Therefore, in a parallel environment, a good load balancing for the work of the individual processors can be achieved by stopping the computation performed by the single processors as soon as a "reasonable" approximation to the solution is obtained by all the processors. No processor will be idle waiting for the computation for the remaining processors to terminate as in general happens if an exact solution of the single commodity network flow problems is required.

The paper is organized as follows. In section 2 we state the problem. The decomposition algorithm is described in detail in Section 3. In particular, Section 3.1 describes the new bundle method for the master problem. Finally, in Section 4 we extend the results to the nonlinear, separable case.

In our notation the scalar product of two vectors $x$ and $y$ in $R^n$ is denoted by $x^T y$ and $e$ is the vector of ones of appropriate dimension. All vectors are column vectors. A superscript $^T$ indicates transpose.

**2. Problem definition and decomposition.** Given a directed graph $\mathcal{G}(N, A)$ represented by its $nxm$ node-arc incidence matrix $F$, the multicommodity network flow problem that we consider that we consider can be stated as follows:

$$
(1)\qquad
\begin{array}{rlcccccccl}
z = & \displaystyle\min_{x^1, x^2, .., x^K} & {c^1}^T x^1 & + & {c^2}^T x^2 & + & \dots & + & {c^K}^T x^K & \\
& & F x^1 & & & & & & & = r^1, \\
& & & & F x^2 & & & & & = r^2, \\
& & & & & & \ddots & & & \\
& & & & & & & & F x^K & = r^K, \\
& & x^1 & + & x^2 & + & \dots & + & x^K & \le d, \\
& & & & \multicolumn{5}{l}{x^k \ge 0, \;\; k = 1, \dots, K,} & 
\end{array}
$$

where $K$ is the number of different commodities, $x^k \in I\!R^m$ is the flow vector of the $k$-th commodity, $d \in R^m$, $d \ge 0$ is the shared resource vector (capacity). For each $k = 1, \dots, K,$ $r^k \in R^n$ is the supply–demand vector for commodity $k$ which defines, for each commodity, a partition of the nodes in supply, demand and intermediate nodes according to positive, negative or zero value of the corresponding component of $r^k$.

A feasible capacity allocation is a set of vectors $\{y^1, y^2, ..., y^K\}$, $y^k \in R^m$, $y^k \geq 0$, $k = 1, ..., K$ such that

$$\sum_{k=1}^{K} y^k = d \tag{2}$$

For each $k = 1, \ldots, K$, we define the following single commodity min-cost flow problem:

$$\begin{aligned} z_k(y^k) = \min_{x^k} \quad & c^{k^T} x^k \\ & F x^k = r^k, \\ & 0 \leq x^k \leq y^k. \end{aligned} \tag{3}$$

Consequently, the original problem (1) can be expressed as

$$\begin{aligned} z = \min_{y^1, y^2, ..., y^K} \quad & \sum_{k=1}^{K} z_k(y^k) \\ & \sum_{k=1}^{K} y^k = d, \\ & y^k \geq 0, \quad k = 1, ..., K. \end{aligned} \tag{4}$$

It is well known [9] that the function

$$\Phi(y^1, y^2, ..., y^K) = \sum_{k=1}^{K} z_k(y^k)$$

is convex and in general nondifferentiable. As any evaluation of function $\Phi$ requires exact solution of the flow problems (3), it seems reasonable to devise an iterative procedure for minimizing $\Phi$ which only requires *approximate* evaluation of the objective function, i.e. approximate solution to the single commodity flow problems. In order to define such minimization algorithm, is helpful to study the differential properties of the function $\Phi$.

Consider the following linear program $P(\alpha)$ depending on the parameter vector $\alpha$ which can be viewed as a perturbation on the resource vector $b$:

$$\begin{aligned} v(\alpha) = \min_{x} \quad & c^T x \\ & Ax = b + \alpha, \\ & x \geq 0 \end{aligned} \tag{5}$$

where $A$ is an $nxm$ real matrix, $b$ and $\alpha$ are $n$-dimensional real vectors. The multipliers associated with the constraint $Ax = b$ in the problem $P(0)$ will be indicated by $\lambda$.

It is well known [5] that the dual optimal multipliers $\lambda^*$ are subgradients of the function $v(\alpha)$ at $\alpha = 0$, i.e. that for any $\alpha \in R^n$ the following inequality holds:

$$v(\alpha) \geq v(0) + \lambda^{*T}\alpha.$$

In general, we can state the following proposition:

PROPOSITION 2.1. *Let $\lambda$ be any feasible solution to the dual of $P(0)$ Then $\lambda$ is an $\epsilon$-subgradient of $v(\alpha)$ at $\alpha = 0$ for $\epsilon = v(0) - b^T\lambda \geq 0$, i.e.*

$$v(\alpha) \geq v(0) + \lambda^T\alpha - \epsilon \quad \forall \alpha \in R^n.$$

*Proof.* From the definition of $v(\alpha)$ we have that

$$v(\alpha) = \lambda^T\alpha + \min_x \begin{array}{c} c^Tx - \lambda^T\alpha \\ Ax = b + \alpha, \\ x \geq 0 \end{array}$$

which can be rewritten as

$$v(\alpha) = b^T\lambda + \lambda^T\alpha + \min_x \begin{array}{c} (c - A^T\lambda)^T x \\ Ax = b + \alpha \\ x \geq 0 \end{array}$$

Hence, taking into account dual feasibility of $\lambda$ and nonnegativity of $x$, the proposition follows by adding and subtracting $v(0)$ to the right–hand–side of the above formula. ∎

Note that the function $\Phi(y) = \Phi(y^1, y^2, ..., y^K)$ is block separable and convex. Moreover, if $g^k \in \partial_{\epsilon_k} z_k(y^k)$ for $k = 1, \ldots, K$ the vector $g$ defined as

$$g^T = \left(g^1, g^2, \ldots, g^K\right)^T$$

is an $\epsilon$–subgradient for the function $\Phi$ at $y^T = (y^1, y^2, ..., y^K)^T$ with $\epsilon = \sum_{k=1}^{K} \epsilon_k$.

**3. The solution method.** In this section we focus our attention on the master problem (4). Let us consider any feasible resource allocation $y = (y^1, y^2, ..., y^K)$ satisfying (2) and let $x^k(y^k)$ be any feasible solution to (3). Assume we are also able to calculate a feasible solution to the dual of problem (3) and let $\chi_k$ be the corresponding dual objective function value and $\lambda^k$ the components of this dual feasible solution associated with the primal constraints $x^k \leq y^k$. From the observation

$$\epsilon_k = c^{k^T} x^k - \chi_k \geq z_k(y^k) - \chi_k \geq 0,$$

and from proposition (2.1) it follows that

$$\lambda^k \in \partial_{\epsilon_k} z_k(y^k).$$

Moreover the quantity

$$\sum_{k=1}^{K} c^{k^T} x^k$$

is an $\epsilon$-approximation to the objective function value $\Phi(y^1, y^2, ..., y^K)$, where

$$\epsilon = \sum_{k=1}^{K} \epsilon_k.$$

Therefore, we are able to calculate an approximation $h$ to $\Phi$

$$h = \sum_{k=1}^{K} c^{k^T} x^k \leq \Phi(y) + \epsilon, \tag{6}$$

and an $\epsilon$-subgradient $g$ of $\Phi$

$$g^T = (\lambda^1, \lambda^2, ..., \lambda^K)^T \in \partial_\epsilon \Phi(y). \tag{7}$$

In general, the descent methods for minimizing convex nondifferentiable functions are based on the possibility of calculating exactly the objective function and a subgradient at the current point. This requirement has been relaxed in [10] and [6].

In particular the method presented in [6] requires only the evaluation at each step of both an $\epsilon$–approximation to the objective function and an $\epsilon$–subgradient. The main iteration of this method will be described in detail in Section 3.1

**3.1. The algorithm for the master problem.** Suppose we wish to solve the following minimization problem:

$$\min \; f(y) \qquad y \in R^n$$

where $f$ is convex and not necessarily differentiable. Assume that $y_j, h_j, g_j$ are respectively the current point, an $\epsilon_j$-approximation to $f$ at the point $y_j$ and an $\epsilon_j$-subgradient of $f$ at $y_j$. We assume in particular that

$$f(y_j) \leq h_j \leq f(y_j) + \epsilon_j. \tag{8}$$

As usual in bundle methods [11] we assume also that a certain amount of information gathered at previous steps is available. In particular a set of triplets

$$(y_i, h_i, g_i) \quad i \in I$$

is given, where $y_i$, $i \in I$ are the points previously considered in the iterative procedure and $h_i$ and $g_i$ are respectively the estimate of the objective function value and an $\epsilon_i$-subgradient at $y_i$. We assume that

$$f(y_i) \leq h_i \leq f(y_i) + \epsilon_i. \tag{9}$$

The following proposition is a straightforward consequence of the $\epsilon$-subgradient inequality and of the definition of $h$ as an approximation to $f$. It provides the possibility of *transporting* $\epsilon_i$-subgradients from $y_i$, $i \in I$ to $y_j$.

PROPOSITION 3.1. *If $g_i$ is an $\epsilon_i$-subgradient of $f$ at $y_i$, then*

$$g_i \in \partial_{\xi_i} f(y_j)$$

*where*

$$0 \leq \xi_i = 2\epsilon_i + h_j - h_i - {g_i}^T(y_j - y_i).$$

*Proof.* From the definition of $\epsilon_i$-subgradient we have that

$$f(y) \geq f(y_i) + {g_i}^T(y - y_i) - \epsilon_i \quad \forall y \in R^n.$$

Adding and subtracting to the right hand side of the above formula $f(y_j)$ and considering that

$${g_i}^T(y - y_i) = {g_i}^T(y - y_j) + {g_i}^T(y_j - y_i),$$

we have that

$$f(y) \geq f(y_j) + {g_i}^T(y - y_j) - \left[f(y_j) - f(y_i) - {g_i}^T(y_j - y_i) + \epsilon_i\right].$$

But from the definition of $\epsilon_i$-subgradient and (8) and (9) we have that

$$0 \leq f(y_j) - f(y_i) - {g_i}^T(y_j - y_i) + \epsilon_i \leq h_j - h_i + \epsilon_i - {g_i}^T(y_j - y_i) + \epsilon_i = \xi_i$$

∎

Now we are ready to define the following problem whose optimal solution $s_j$ provides us with a tentative step from the current point $y_j$:

$$(10) \qquad \begin{array}{ll} \min\limits_{v,s} & v + \dfrac{1}{2t}\|s\|^2 \\ v \geq & {g_j}^T s - \epsilon_j, \\ v \geq & {g_i}^T s - \xi_i \qquad i \in I, \end{array}$$

where $t$ is a positive scalar. Problem (10) corresponds to finding the minimum of a piecewise linear approximation to $f$, obtained as the maximum of a finite number of affine functions which support from below the graph of $f$. The quadratic term $\frac{1}{2t}\|s\|^2$ is introduced both for stabilization purposes and to ensure that problem (10) is always well posed. The scalar parameter $t$ plays a role similar to the trust region parameter in differentiable optimization. It may be suitably updated as the algorithm proceeds. Individual scaling of the quantities $s_i$ is also possible. This scaling can, for example, reflect a different degree of trust in the individual components of $g_j$ corresponding to different degrees of accuracy in the solution of the individual subproblems.

Once the optimal solution $(v_j, s_j)$ to (10) has been found, the new point

$$y^+ = y_j + s_j$$

is considered and a check for sufficient decrease is done. The approximate value $h^+$ of $f$ at $x^+$ is calculated and the following descent condition is tested

$$h^+ \le h_j + \rho v_j \tag{11}$$

where $0 < \rho < 1$. If (11) is satisfied, then the point $y^+$ becomes the new iterate $y_{j+1}$. Otherwise, $y_j$ is unchanged and the bundle is enriched by the new triplet

$$(y^+, h^+, g^+)$$

where $g^+ \in \partial_{\epsilon_+} f(y^+)$ for the appropriate value of $\epsilon_+$ and a new problem (10), where the set $I$ has been enriched by including the new point $y^+$, is stated and solved.

The termination condition which can be proved to occur in a finite number of steps is of the type

$$| v_j | \le \delta$$

and it is analogous to the usual termination conditions of NDO methods. Of course, as the parameter $\delta$ dictates the tolerance in finding the minimum, once we have chosen $\delta$, we are forced to make an appropriate choice of the accuracy in evaluating $f$.

Of course, in applying the above described method to problem (4), we have to consider that actually the problem is of the constrained type. Thus we propose projection of the search direction $s_j$ onto the feasible set.

**3.2. Solving the min-cost flow problem.** Now we focus our attention to problem (3), which, by dropping for simplicity of notation the superscript $k$ may be rewritten as

$$\begin{array}{lll} \min_x & \sum_{(i,j)\in\mathcal{A}} c_{ij}x_{ij} & \\ & \sum_j x_{ij} - \sum_j x_{ji} = r_i & i \in \mathcal{N}, \\ & 0 \le x_{ij} \le y_{ij} & (i,j) \in \mathcal{A}. \end{array} \tag{12}$$

The dual of problem (12) has the form

$$\begin{array}{lll} \max_{w,\mu} & \sum_{i\in\mathcal{A}} r_i w_i - \sum_{(i,j)\in\mathcal{A}} y_{ij}\mu_{ij} & \\ & w_i - w_j - \mu_{ij} \le c_{ij} & (i,j) \in \mathcal{A}, \\ & \mu_{ij} \ge 0 & (i,j) \in \mathcal{A}. \end{array} \tag{13}$$

If a primal feasible solution $x$ and a dual feasible solution $(w, \mu)$ are available, the difference between the primal and the dual objective function values provides us with an estimate of the approximation $\epsilon$ in solving problem (12).

However, if, in solving the single commodity problem, a method such as the network simplex which maintains at each step a primal basic feasible solution is adopted, no explicit dual feasible solution is available until the algorithm reaches the optimum.

Our aim now is to show how to obtain a dual feasible solution once a primal basic feasible solution is available. Suppose we have calculated a primal basic feasible solution $x$ with $n$ basic variables associated with a rooted spanning tree [9] and $(m-n)$ non–basic variables fixed either at their lower bound or upper bound.

The primal–dual complementarity conditions are

$$\begin{aligned} x_{ij}(c_{ij} - (w_i - w_j - \mu_{ij})) &= 0 \quad \forall (i,j), \\ \mu_{ij}(y_{ij} - x_{ij}) &= 0 \quad \forall (i,j). \end{aligned} \tag{14}$$

Therefore, the following system of linear equations must be satisfied by the dual variables $w$:

$$w_i - w_j = c_{ij} \quad \forall (i,j) \in N_B \tag{15}$$

where $N_B$ is the set of the indices of the basic variables. Moreover, by defining $N_l$ and $N_u$ as the sets of the indices of the (non–basic) primal variables which are respectively at their lower and upper bounds, we have that the dual solution which satisfies the complementarity conditions (14) may be obtained by solving (15) and letting

$$\begin{aligned} \mu_{ij} &= 0 & (i,j) \in N_B \cup N_l, \\ \mu_{ij} &= -\hat{c}_{ij} & (i,j) \in N_u \end{aligned} \tag{16}$$

where $\hat{c}_{ij}$ is the reduced cost defined as

$$\hat{c}_{ij} = c_{ij} - (w_i - w_j). \tag{17}$$

Unless the primal solution is optimal, such a dual solution is not feasible. In fact it may happens that $\hat{c}_{ij} < 0$ for some $(i,j) \in N_l$ and/or $\hat{c}_{ij} > 0$ for some $(i,j) \in N_u$.

To get a feasible solution "as close as possible" to the above defined solution we operate in the following way. First, the dual variables $w$ are chosen to satisfy the system of linear equations (15). Then, we define the variables $\mu$ in the following way:

$$\begin{aligned} \mu_{ij} &= 0 & (i,j) \in N_B, \\ \mu_{ij} &= \max(0, -\hat{c}_{ij}) & (i,j) \notin N_B. \end{aligned} \tag{18}$$

Obviously, dual feasibility is achieved but complementarity is lost. It is easy to verify that the corresponding value of $\epsilon$, obtainable as the difference between the dual "complementary" solution and the dual feasible one is the following:

$$\epsilon = \sum_{(i,j) \in N_u, \hat{c}_{ij} > 0} \hat{c}_{ij} y_{ij} - \sum_{(i,j) \in N_l, \hat{c}_{ij} < 0} \hat{c}_{ij} y_{ij}. \tag{19}$$

**4. Extension to the nonlinear case.** In this section we extend some of the results obtained for the multicommodity network optimization problem with linear objective function to the nonlinear case. We will focus our attention to nonlinear convex, differentiable objective functions.

First of all we extend Proposition 2.1 to the nonlinear case. We define the problem $P(\alpha)$ as

$$\begin{array}{rll} v(\alpha) = & \min\limits_{x} & f(x) \\ & & Ax = b + \alpha, \\ & & x \geq 0 \end{array} \tag{20}$$

where $f : x \in I\!R^n \mapsto I\!R$ is a convex, differentiable function. The dual of the above problem is [12]:

$$\begin{array}{ll} \max\limits_{x,\lambda,v} & f(x) - \lambda^T(Ax - b - \alpha) - v^T x \\ & \nabla f(x) - A^T\lambda - v = 0, \\ & v \geq 0. \end{array}$$

PROPOSITION 4.1. *Let $(\bar{x}, \bar{\lambda}, \bar{v})$ be a feasible solution to the dual of $P(0)$ Then $\bar{\lambda}$ is an $\epsilon$-subgradient of $v(\alpha)$ at $\alpha = 0$ for $\epsilon = v(0) - f(\bar{x}) + \bar{\lambda}^T(A\bar{x} - b) + \bar{v}^T\bar{x} \geq 0$, i.e.*

$$v(\alpha) \geq v(0) + \bar{\lambda}^T\alpha - \epsilon \quad \forall \alpha \in R^n.$$

*Proof.* Following similar steps as in the proof of Proposition 2.1 we have:

$$\begin{array}{rcl} v(\alpha) & = & \bar{\lambda}^T\alpha + \left[f(\bar{x}) - \bar{\lambda}^T(A\bar{x} - b) - \bar{v}^T\bar{x}\right] + \\ & & \left[\begin{array}{ll} \min\limits_{x} & f(x) - \bar{\lambda}^T\alpha - f(\bar{x}) + \bar{\lambda}^T(A\bar{x} - b) + \bar{v}^T\bar{x} \\ & Ax = b + \alpha \\ & x \geq 0 \end{array}\right]. \end{array}$$

Observe now that

$$f(x) - \bar{\lambda}^T\alpha - f(\bar{x}) + \bar{\lambda}^T(A\bar{x} - b) + \bar{v}^T\bar{x} = f(x) - f(\bar{x}) - \bar{\lambda}^T\left(b + \alpha\right) + \bar{\lambda}^T A\bar{x} + \bar{v}^T\bar{x}$$

$$= f(x) - f(\bar{x}) - \left(A^T\bar{\lambda}\right)^T(x - \bar{x}) + \bar{v}^T\bar{x} \tag{21}$$

$$= f(x) - f(\bar{x}) - (\nabla f(\bar{x}) - \bar{v})^T(x - \bar{x}) + \bar{v}^T\bar{x} \tag{22}$$

$$= f(x) - f(\bar{x}) - \nabla f(\bar{x})^T(x - \bar{x}) + \bar{v}^T x \geq 0 \tag{23}$$

where (21) follows for the primal feasibility of $x$, (22) follows from the dual feasibility of $(\bar{x}, \bar{\lambda}, \bar{v})$ and (23) from the convexity of $f$.

Therefore exactly as for Proposition 2.1, taking into account that

$$\epsilon = v(0) - \left[f(\bar{x}) - \bar{\lambda}^T(A\bar{x} - b) - \bar{v}^T\bar{x}\right] \geq 0,$$

we have:

$$v(\alpha) \geq \bar{\lambda}^T \alpha + v(0) - \epsilon.$$

∎

The nonlinear multicommodity network flow problem can be stated as follows:

$$
(24) \quad
\begin{array}{llllllll}
z = \min\limits_{x^1, x^2, \ldots, x^K} & f_1(x^1) & + & f_2(x^2) & + \; \ldots \; + & f_K(x^K) & & \\
& Fx^1 & & & & & = r^1, & \\
& & & Fx^2 & & & = r^2, & \\
& & & & \ddots & & & \\
& & & & & Fx^K & = r^K, & \\
& x^1 & + & x^2 & + \; \ldots \; + & x^K & \leq d, & \\
& & & x^k \geq 0 & & & k = 1, \ldots, K, &
\end{array}
$$

where $f_k(x^k)$ is nonlinear differentiable and convex.

Problems of this form appear in several areas of applications including logistic, transportation planning, water distribution, financial modeling and air-traffic control (see [9] and [8, Chapter 11] )

Recently, various nonlinear programming algorithms for the nonlinear case have been proposed. In particular, relaxation methods ([1]) and row–action–type algorithms have been studied (see [16]) for problems with objective functions that belong to the Bregman class [2] as characterized by Censor and Lent [4]. These method are well suited for massively parallel implementation.

The decomposition technique discussed in the previous sections for the linear case, applies also to the nonlinear separable case. Proposition 4.1 allows us to calculate $\epsilon$–subgradients of the $K$ subproblems corresponding to the different commodities once a dual feasible solution for each subproblem is available. Dual ascent methods (see [1]), applied to the single commodity nonlinear subproblems, iteratively construct a sequence of primal–dual pairs of vectors satisfying slackness conditions until primal feasibility is obtained. Therefore, at each iteration of a dual ascent method for the subproblems, an $\epsilon$–subgradient is available to the master program and the bundle method as discussed in Section 3.1 can be used to detect optimality or determine a new feasible capacity allocation.

**5. Conclusions.** We presented an iterative algorithm based on right hand side decomposition for the solution of linear multicommodity network flow problems. The coupling constraints are eliminated and the shared capacity resource is subdivided among the different commodities. The master problem that determines the the sharing of the resources is solved by using a new nondifferentiable optimization technique based on the bundle method. Extensions to the nonlinear, convex separable case are also discussed.

REFERENCES

[1] D.P. Bertsekas and J.N. Tsitsiklis. *Parallel and Distributed Computation.* Prentice–Hall, Inc, Englewood Cliffs, New Jersey, 1989.

[2] L.M. Bregman. The relaxation method for finding the common point of convex sets and its application to the solution of problems in convex programming. *USSR Comput. Math. and Math. Phys.*, 7:200–217, 1967.

[3] Y. Censor, Chajakis E.D., and S.A. Zenios. A primal-dual algorithm for large scale multicommodity network optimization problems and its parallel implementation. Technical Report 91-09-03, Decision Sciences Department, The Wharton School, Philadelphia, PA 19104, September 1991.

[4] Yair Censor and A. Lent. An iterative row-action method for interval convex programming. *Journal of Optimization Theory and Applications*, 34:321–353, 1981.

[5] A. V. Fiacco. *Introduction to Sensitivity and Stability Analysis in Nonlinear Programming.* Academic Press, New York, 1983.

[6] M. Gaudioso. Nonsmooth optimization with inexact calculation of the objective function. Technical report, Dip. di Elettronica, Informatica e Sistemistica, Universita' della Calabria, 1992.

[7] A. M. Geoffrion. Primal resource-directive approaches for optimizing nonlinear decomposable systems. *Operations Research*, 18(3):375–403, 1970.

[8] P.A. Jensen and W. Barnes, J. *Network Flow Programming.* John Wiley & Sons, New York, 1980.

[9] J. L. Kennington. *Algorithms for Network Programming.* John Wiley & Sons, New York, 1980.

[10] K. C. Kiwiel. Approximations in decomposition of large scale convex programs via a nondifferentiable optimization method. In V. Utkin U. Jaaksoo, editor, *Proc. of the 11th IFAC World Cong.*, Tallin, August 1990.

[11] C. Lemarechal. Nondifferentiable optimization. In G.L. Nehmauser, A.H.G. Rinnooy Kan, and M.J. Todd, editors, *Handbooks in Opns. Res. and Mgmt. Sc., Vol 1 : Optimization.* North-Holland, Amsterdam, 1989.

[12] O.L. Mangasarian. *Nonlinear Programming.* McGraw–Hill, New York, 1969.

[13] M.C. Pinar and S.A. Zenios. Parallel decomposition of multicommodity network flows using smooth penalty functions. Technical Report 90-12-06, Decision Sciences Department, The Wharton School, Philadelphia, PA 19104, December 1990.

[14] Gary L. Schultz. *Barrier Decomposition for the parallel optimization of block-angular programs.* PhD thesis, University of Wisconsin-Madison, 1991.

[15] Gary L. Schultz and Rober R. Meyer. A structured interior point method. Technical Report 934, University of Wisconsin, Computer Sciences Dept., 1990.

[16] S.A. Zenios and Y. Censor. Massively parallel row–action algorithms for some nonlinear transportation problems. *SIAM Journal on Optimization*, 1(3):373–400, 1991.

[17] S.A. Zenios, M.C. Pinar, and R.S. Dembo. A smooth penalty function algorithm for network–structured problems. Technical Report 90-12-05, Decision Sciences Department, The Wharton School, Philadelphia, PA 19104, January 1991.