HOMENAJE AL PROFESOR SIXTO RIOS
TRABAJOS DE ESTADISTICA Y DE INVESTIGACION OPERATIVA
Vol. 36, Núm. 3, 1985, pp. 119 a 131

ON OPTIMIZING A MAXIMIN NONLINEAR FUNCTION SUBJECT TO REPLICATED QUASI-ARBORESCENCE-LIKE CONSTRAINTS.

L.F. Escudero

IBM Madrid Scientific Center.
Paseo de la Castellana 4. 28046 Madrid.

In this paper we present the motivation for using the Truncated Newton method in an algorithm that maximises a nonlinear function with additional maximin-like arguments subject to a network-like linear system of constraints. The special structure of the network (so-termed replicated quasi-arborescence) allows to introduce the new concept of independent superbasic sets and, then, using second-order information about the objective funtion withut too-much computer effort and storage.

Key words: Nonlinear programming; Numerical algorithms; Truncated Newton Method; Independent superbasic sets; Basic equivalent path.

AMS Classification (1980): Primary, 65K05.

Optimización de una función no-lineal Maximin sujeta a condiciones tipo Cuasi-Arbol Replicado

En este trabajo se presenta la motivación para utilizar el método Newton Truncado en un algoritmo que maximiza una función no lineal con argumentos adicionales de tipo maximin, sujeto a un sistema lineal de condiciones de tipo grafo. La estructura especial del grafo (denominada expansión horizontal de un cuasi-árbol) permite introducir el nuevo concepto de conjuntos superbásicos independientes y, por tanto, facillitar la utilización de información de segundo orden sobre la función objetivo sin requerir excesivo tiempo de cálculo, ni precisar demasiada capacidad de ordenador.

Palabras clave: Programa no lineal; Algoritmos numéricos; Método Newton Truncado; Conjuntos superbásicos independientes; Ruta equivalente básica.

Clasificación AMS (1980): Primaria, 65K05.

1. INTRODUCTION. PROBLEM DESCRIPTION

This paper reports the motivation for using second-order information about the objective function of a nonlinear network flow *problem* so that the computer effort and storage are affordable even for large-scale cases. The problem consists in the maximization of a nonlinear function subject to a linear system of constraints with a special network-like structure (so-termed *replicated quasi-arborescence*). Let J denote the set of single-segment nodes in a given quasi-arborescence, and P_j (res. Q_j) the set of single-segment nodes directly downstreeam (res. upstream) for single-segment node j for $j \in J$. Usually $|Q_j|=1$, but it is allowed $|Q_j| > 1$; assume that the latter case is not frequent, nor its cardinality is very big so that the word "quasi" makes sense. Let T denote the set of single-segments of the replicated quasi-arborescence along which the quasi-arborescence is expanded. The decision variables are denoted r_{tji} , the flow linking single-segment nodes j and i for $i \in Q_j$, $j \in J$ at segment t for $t \in T$; and s_{tj} , the flow linking segments t-1 and t at j for $t \in T$, $j \in J$.

The constraints form a linear system that can be represented by a special direct graph whose nodes correspond to the state of the single-segment nodes at a given segment, and the arcs correspond to the decision variables. The flow balance equations can be expressed

$$- \Sigma_{i \in P_j} r_{tij} - s_{tj} + \Sigma_{i \in Q_j} r_{tij} + s_{t+1,j} = b_{tj} \qquad \forall t \in T, \; j \in J \qquad (1)$$

where b_{tj} is the net exogenous inflow to single-segment node j at segment t . Letting X be a vector of all decision variables and b a vector of the exogenuous inflows, system (1) can be written AX= b , where A is the node-arc incidence matrix of the replicated network. The decision variables are bounded such that $m \le X \le M$. Each column of A corresponds to an arc and each row to a node. The nonzero elements in a column are +1 in the row corrresponding to the node where the arc originates and -1 in the row where the arc terminates. In addition, there is a *root* node which represents the Exit in all segments for single-segment node $j \in J$ such that $\exists$ $i \in Q_j$, $i = |J| + 1$.

Following a traditional approach, see Murtagh and Saunders (1978), matrix A can be partitioned such that A = (B,S,N) , where the columns of B form a basis and correspond to the basic arcs, and the columns of S and N correspond to the superbasic and nonbasic arcs, respectively; let **B** , **S** and **N** denote the related *basic* , *superbasic* and *nonbasic* sets of arcs. Nonbasic arcs are temporarily fixed to their bounds, and the flow in sets **B** and **S** vary between their bounds.

By construction of A it can be shown that the arcs corresponding to columns in any basis form a *spanning tree* of the network. A *maximal basis* spanning tree for a given feasible solution avoids a basic-superbasic degenerate pivot, see Dembo and Klincewicz (1981); otherwise, null steps are more frequent than in problems with a general structure.

Let Z be the *variable-reduction* matrix whose columns form a basis for the null space of A , given AZ= 0 , such that

$$Z = ((- B^{-1} S)^t , I^t , O^t)^t \qquad (2)$$

Let n denote the number of nodes (without including the root) and a denotes the number of arcs. Let *Basic-Equivalent-Path* (BEP) define the unique path in the basis spanning tree that leads from originating node, say i_k in superbasic or nonbasic arc k to termnating node, say j_k ; let β_k denote the set of arcs in the BEP of arc k . Arc $k' \in \beta_k$ has a forward orientation in the BEP of arc k if $p(i_{k'})_k = j_{k''}$, where $p(.)_k$ is the *predecessor* of node (.) in the BEP of arc k ; it has a *reverse* orientation if $p(j_{k'})_k = i_{k'}$. Let ρ_k denote the columun $(B^{-1} S)_k$ and then, $\rho \equiv (B^{-1} S)$. The nonzero elementos of ρ_k for $k \in S \cup N$ are +1 for a forward orientation and -1 for a reverse orientation. The inexpensive implicit computation of matrix Z (2), via specialized network data structures as in Rosenthal (1981), and the structure of the Hessian matrix G (6) (see below), makes affordable the using of second-order information even in large-scale problems.

The replicated arborescences have the property that a - n = n ; note that usually, a - n >> n in a general network. As a result, $| \beta_k |$ is very small. Any single-segment of our network is a quasi-arborescence and, then, $| \beta_k |$ is still relatively small; this property will be exploited in the related algorithm.

The paper is organized as follows. Seccion 2 describes the objective function. Seccion 3 summarises the variable-reduction environment of the algorithm. Seccion 4 presents the approach for obtaining the superbasic stepdirection, and introduces the new concept of independent superbasic sets. And, finally, Seccion 5 describes the de-activating process.

2. THE NONLINEAR OBJECTIVE FUNCTION

Let h_{tj} denote the nonlinear argument for $t \in T$ and $j \in J$ such that

$$h_{tj} = k_{tj} \Sigma_{i \in Qj} \min\{ r_{tji}, R_{ji} \} \quad (3)$$

where k_{tj} is a nonlinear function of the s- variables, such that

$$k_{tj} = f_{nl}(s_{tj}, s_{t+1,j}) \quad (4)$$

and R_{ji} denotes the upper bound for r_{tji} having any influence in h_{tj}. Let the objective function be expressed as follows.

$$\max\{ \Sigma_{t \in T} \Sigma_{j \in J} (h_{tj} - P_{tj} \max\{0, s_{tj} - T_{tj}\})\} \quad (5)$$

where $T_{tj} > 0$, and P_{tj} gives the unit penalty on excess $s_{tj} - T_{tj}$.

Let VL denote set of so-termed *linear arcs* (t, j, i) with *variable coefficients*, and NL the set of *nonlinear arcs* (t, j). Note that h_{tj} is a linear function of $\Sigma_i r_{tji}$ if s_{tj} and $s_{t+1,j}$ are fixed. The Hessian matrix G has the form

$$G = \begin{pmatrix} 0 & G_3 \\ G_3^t & G_4 \end{pmatrix} \begin{matrix} \{VL\} \\ \{NL\} \end{matrix} \quad (6)$$

such that G_3 is a two-diagonal matrix for $| Q_j | = 1 \ \forall j \in J$, and G_4 is a symmetric tri-diagonal matrix. Note that the nonzero r-elements related to the same pair (t, j) in a given column of matrix G_3 have the same value for all $i \in Q_j$, for $| Q_j |>1, j \in J$.

The *nondifferentiability* of the objective function does not introduce any difficulty on evaluating matrix G, provided that the basic spanning tree is kept maximal and the basic-based first-order estimation of the Lagrange multipliers of the *just-deactivated* nonbasic set (see Section 5) is used as its stepdirection.

There are numerous examples of the application field of replicated quasi-arborescences. Probably, the most typical is the management of hydro-electric power systems where the single-segment nodes are the reservoirs, the river's stream can be represented by the quasi-arborescence, and the segments are the time periods along the planning horizon; see Escudero (1983). Other examples lie in the electrical circuits design, water distribution systems, urban traffic systems and so on.

3. SKELETAL ALGORITHM FOR OBTAINING FEASIBLE-INCREASING SOLUTIONS.

Let **d** define the stepdirection from feasible solution, say **x** such that the new iterate can be expressed

$$x_k := x_k + \alpha_k d_k \quad \forall k \in \mathbf{B} \cup \mathbf{S} \tag{7}$$

where $\{\alpha_k\}$ is the steplength vector. Given system (1) and the matrix partition $A = (B,S,N)$, by linearity it results

$$(B, S, N)\,(d^t_B,\ d^t_S,\ d^t_N)^t = 0 \tag{8}$$

being $\mathbf{d} \equiv (d^t_B,\ d^t_S,\ d^t_N)^t$. The basic stepdirection d_B is used to satisfy the constraints system (1), the nonbasic stepdirection d_N is temporarily fixed to zero, and the superbasic stepdirection d_S is used to maximise the objective function (5).

At each iteration, the problem then becomes determining vectors **d** and α, such that $\{\alpha_k d_k\}$ is feasible and *increasing enough* . Direction **d** is feasible if system (8) is satisfied. Since $d_N = 0$ and d_S is allowed to be free, it results $d_B = -\ B^{-1} S\, d_S$, such that $\mathbf{d} = Z\, d_S$.

The ascent enough stepdirection d_S can be obtained by "solving" the problem $\max\{h^t d_S + 1/2\, d_S{}^t H\, d_S\}$, where the reduced Hessian H can be expressed $H = Z^tGZ$, and the reduced gradient h can de written

$$h \equiv Z^t g = g_S - S^t \mu_B \tag{9}$$

such that the *basic estimation* μ_B of the constraints Lagrange multipliers solves the system

$$g_B = B^t \mu_B \tag{10}$$

and $\mathbf{g} = (g^t_B,\ g^t_S,\ g^t_N)^t$. Note that the solution d_S of system

$$Hd_S = -h \tag{11}$$

is feasible-ascent for a positive definite matrix - H and a maximal basis spanning tree.

Solving the n-system (10) when the arcs corresponding to the columns of B form a spanning tree does not need a greet computational effort, but the LP simple rules for updating μ_B do not apply when the objective function is nonlinear (even if basic set **B** does not change) . From other point of view, using sytem (10) in expression (9) is computational advantegous, since

1) a - n = m for any replicated arborescence and, then, $|\beta_k|$ for $k \in \mathbf{S} \cup \mathbf{N}$ is small,

2) The cardinality of the set to be used while iteratively solving the above quadratic problem is much smaller than a - n, and

3) β_k must be used, in any case, for obtaining the stepdirection d_S and the steplength α . Then, it can be written

$$h_k = g_k - \Sigma_{j \in \beta_k} \rho_{jk} g_j \qquad (12)$$

Matrix H is likely very dense even for sparse matrices Z and G. Since we are dealing with large-scale problems, we cannot afford to use matrix H , nor any of its approximations suggested in the literature. We suggest to use the Truncated Newton method (see Dembo and Steihaug (1983), and Escudero (1984)) at independent series of iterations, such that matrix H does not need to be stored and the computer effort and storage are within affordable limits. Note that system (11) is not needed to be completely solved at every iteration for getting, under mild conditions, a Q-superlinear rate of convergence (see above references).

The steplength $\{\alpha_k\}$ must be feasible and $\{\alpha_k d_k\}$ must be increasing enough; it is interesting it may allow to activate as many as possible superbasic arcs; see Escudero (1983).

4. INDEPENDENT SETS OF SUPERBASIC ARCS.

4.1 DEFINITIONS.

The Truncated-Newton method does not require the calculation of any Hessian matrix, but the product

$$q^{(i)} \equiv H\delta_S^{(i)} = Z^t G Z \delta_S^{(i)} \qquad (13)$$

For obtaining $q^{(i)}$, the superbasic set $\mathbf{S}$ is partitioned into, say $|P|$ disjoint so-termed *independent superbasic* sets, such that $\mathbf{S} \Delta \cup \mathbf{S}^{(p)} \ \forall\ p \in P$, and $\mathbf{S}^{(p)} \cap \mathbf{S}^{(p)} = \{\phi\}\ \forall\ p, q \in P$. Let $\mathbf{B}^{(p)} \subset \mathbf{B}$ define the set of basic arcs covered by the superbasic arcs included in set $\mathbf{S}^{(p)}$, such that $\mathbf{B}^{(p)} \Delta \cup \beta_k \quad \forall\ k \in \mathbf{S}^{(p)}$; let $\mathbf{C}^{(p)} \Delta\ \mathbf{B}^{(p)} \cup \mathbf{S}^{(p)}$ define the set of basic and superbasic arcs to be used for obtaining $d_S^{(p)}$.

Superbasic arc k will be included set $\mathbf{S}^{(p)}$ *iff* the following condition is satisfied.

$$\mathbf{B}^{(p)} \cap \beta_k \neq \{\phi\} \ \vee\ (\exists\, G_{gg'} \neq 0 \mid (g \in \{k\} \cup \beta_k) \wedge (g' \in \mathbf{C}^{(p)})) \qquad (14)$$

That is, two superbasic arcs will belong to the same independent superbasic set if any flow change in one of them *affects the other's solution feasibility or objective function coefficient.*

Sets, say $\mathbf{S}^{(p)}$ and $\mathbf{S}^{(q)}$ will be joined in one single set *iff* the following condition is satisfied

$$(\mathbf{B}^{(p)} \cap \mathbf{B}^{(q)} \neq \{\phi\}) \vee (G_{gg'} \neq 0 \mid g \in \mathbf{C}^{(p)} \wedge g' \in \mathbf{C}^{(q)}) \qquad (15)$$

Note that $|\beta_k|$ is small, G very sparse, and $|\mathbf{S}^{(p)}|$ and $|\mathbf{S}^{(q)}|$ can be chosen to be small in the de-activating process.

Let $\mathbf{C}\ \Delta \cup \mathbf{C}^{(p)} \ \forall\ p \in P$. Let $\mathbf{VL} \subset \mathrm{VL} \cap \mathbf{C}$ define the set of basic and superbasic linear arcs whose variable-coefficientes are not fixed at the current iteration. Let $\mathbf{NL}\ \Delta\ \mathrm{NL} \cap \mathbf{C}$ define the set of nonbasic arcs in set $\mathbf{C}$.

Note that at major iteration say l, $|P|$ independent iterations are consecutively executed; note also that there are $\{i\}$ minor iterations to be executed at each iteration p for $p \in P$.

The advantages of using independent sets at sucessive major iterations are as follows.

1) The computational effort for obtaining vector $q^{(i)}$ (13) is drastically reduced.

2) Faster minor iterations at the price of more (but much cheaper) major iterations. Note that the elements of matrix G out of set $\mathbf{C}^{(p)}$ are not modified

after obtaining $d_S^{(p)}$. Note also that only the elements of matrices G_3 and G_4 related to set **C** and the gradient related to set **VL** $\cup$ **NL** are to be evaluated at a given major iteration.

3) Independent steplength upper bound for each set C^p $\forall$ p $\in$ P. Then, it allows a deeper step along the search direction; (see Escudero (1985)).

4) Strong reduction on the number of arcs (i.e., cardinality of set $C^{(p)}$) to be used for obtaining the steplength related to set $C^{(p)}$. Note also that only the terms of the objective function (5) related to set $C^{(p)}$ are to be recomputed for obtaining the objective function value $F(X_{BS}^{(p)})$ related to each trial step.

4.2 REDUCED GRADIENT USED AS A SUPERBASIC STEPDIRECTION.

Assume that the p-th stepdirection is being obtained for p $\in$ P. Assume that $\mathbf{VL}^{(p)} \cup \mathbf{NL}^{(p)} = \{\phi\}$; in this case, a LP-network subproblem must be maximised, such that the related stepdirection $d^{(p)}$ can be expressed

$$d^{(p)}_k := h_k = g_k - \Sigma_{j \in \beta_k} \rho_{jk}\, g_j \qquad \forall\ k \in \mathbf{S}^{(p)} \tag{16}$$

$$d^{(p)}_j := - \Sigma_{k \in S^{(p)}} \rho_{jk}\, d_k \qquad \forall\, k' \in \mathbf{B}^{(p)} \tag{17}$$

4.3 OBTAINING VECTOR $q^{(i)}$ IN THE TRUNCATED NEWTON METHOD.

Assumed that $q^{(i)}$ is related to the superbasic stepdirection $d_S^{(p)}$. Let $\mathbf{VL}\Delta\mathbf{VL}^{(p)}$, $VL^{(p)}\,\Delta\ C^{(p)} \cap VL$, and $\mathbf{NL}\ \Delta \cup \mathbf{NL}^{(p)}$ for $\forall$ p $\in$ P . Let $\mathbf{VL}^{(p)}_n\ \Delta\ VL^{(p)} / \mathbf{VL}^{(p)}$. Let $\mathbf{C}_n^{(p)}$ define the complement of set $\mathbf{C}^{(p)}$ in set **B** $\cup$ **S** $\cup$ **N**.

1) Obtain intermediate vector $\boldsymbol{\delta}^{(i)} := Z^{(p)}\delta_S^{(i)}$, such that

$$\begin{aligned} &\boldsymbol{\delta}_l^{(i)} := \Sigma_{k \in S^{(p)}} \rho_{lk}\delta_k^{(i)} && \forall\ l \in \mathbf{B}^{(p)} \\ &\boldsymbol{\delta}_l^{(i)} := \delta_l^{(i)} && \forall\, l \in \mathbf{S}^{(p)} \\ &\boldsymbol{\delta}_l^{(i)} := 0 && \forall\, l \in \mathbf{C}_n^{(p)} \end{aligned} \tag{18}$$

2) Obtain intermediate vector $\delta^{(i)} := G\, \boldsymbol{\delta}^{(i)}$, such that

$$\begin{aligned}
&\delta_l^{(i)} := 0 && \forall\, l \in C_n^{(p)} \cup VL_n^{(p)} \\
&\delta_l^{(i)} := \Sigma_{j \in NL^{(p)}} G_{3lj}\, \boldsymbol{\delta}_j^{(i)} && \forall\, l \in VL^{(p)} \qquad (19)\\
&\delta_l^{(i)} := \Sigma_{j \in VL^{(p)}} G_{3jl}\, \boldsymbol{\delta}_j^{(i)} + \Sigma_{j \in NL^{(p)}} G_{4lj}\, \boldsymbol{\delta}_j^{(i)} && \forall\, l \in NL^{(p)}
\end{aligned}$$

Computation of vector $\delta^{(i)}$ is very fast since G_4 is a tridiagonal symmetric matrix, each column of matrix G_3 has only two different nonzero values, the elements of $\boldsymbol{\delta}^{(i)}$ related to set $VL_n^{(p)}$ are not used (since the related elements in matrix G_3 are zero), and only the rows of the Hessian matrix related to set $VL^{(p)} \cup NL^{(p)}$ are used.

3) Obtain vector $q^{(i)} := Z^{(p)t}\boldsymbol{\delta}^{(i)} = \boldsymbol{\delta}_S^{(i)} - (B^{-1}S)^{(p)t}\boldsymbol{\delta}_B^{(i)}$, such that

$\boldsymbol{\delta}^{(i)} = (\boldsymbol{\delta}_B^{(i)t}, \delta_S^{(i)t}, \delta_N^{(i)t})^t.$

$$q_k^{(i)} = \boldsymbol{\delta}_k^{(i)} - \Sigma_{j \in \beta_k} \rho_{jk}\, \boldsymbol{\delta}_j^{(i)} \qquad \forall\, k \in S^{(p)} \qquad (20)$$

5. DE-ACTIVATING STRATEGY

5.1 PRICING NONBASIC ARCS.

Let us define indicator γ_k $k \in \mathbf{N}$ as follows. $\gamma_k = 0$ means that nonbasic arc k is not a *candidate* to be de-activated; otherwise, it takes the sign of its de-activating direction (+ for up-direction and - for down-direction). A nonbasic arc will not be

a candidate to be de-activated if the pricing is not favorable or it is a blocked arc; see below.

Let $\mathbf{D}$ define the set of nonbasic arcs to be de-activated ; that is, the arcs that will be moved from the nonbasic set to the superbasic set. Let $\mathbf{D}\ \Delta \cup \mathbf{D}^{(p)}\ \forall\, p \in P$ and $\mathbf{D}^{(p)} \cap \mathbf{D}^{(q)} = \{\phi\}$, where $\mathbf{D}^{(p)}$ is the independent nonbasic set to be de-activated and, then, joined with the independent superbasic set $\mathbf{S}^{(p)}$. A candidate nonbasic are will not be de-activated if $|\,\mathbf{D}\,|$ is at its given (upper) bound and there is, at least, any other candidate arc with higher (first-order) guarantee of a stronger increase in the objective function.

Let ${}_{(l)}a_k$, ${}_{(i)}a_k$ and ${}_{(u)}a_k$ denote the lower bound m_k, "intermediate" bound (R_{ji} or T_{ji} for related arc k) (see Section 2) and upper M_k of the feasible flow in arc k , respectively. The nonbasic Lagrange mutipliers estimation λ is obtained as follows.

Let

$$\lambda_k^i = g_k^i = -\Sigma_{j \in \beta_k} \rho_{jk} g_j^i \quad \forall k \in \mathbf{N} \tag{21}$$

where λ^i_k is the Lagrange multiplier estimation related to the i-direction of the potential move of nonbasic arc k , and g_k^i gives the gradient element related to the i-direction of nonbasic arc k . The directions can be + (up), - (down) and 0 (no move).

Note that $g_k^i = g_k^+$ for $\mathbf{x}_k = {}_{(l)}a_k$ (lower bound) and $g_k^i = g_k^-$ for $\mathbf{x}_k = {}_{(u)}a_k$ (upper bound) such that $g_k^i = g_k$, where g_k is the usal gradient element related to arc k . Both gradient elements g_k^+ and g_k^- (and, then, λ_k^+ and λ_k^-) are required for the nonbasic arc whose current solution is at its "intermediate" bound ${}_{(i)}a_k$. Note also that $g_{k'}^+ = g_{k'}^- = g_{k'}$ for $k' \in \beta_k$ such that $\mathbf{x}_{k'} \neq {}_{(i)}a_{k'}$.

Gradient elements $g_l^+ = g_l^-$ for a basic or nonbasic arc, say l such that $\mathbf{x}_l = {}_{(i)}a_l$ are obtained as follows: $g_l^+ = 0$ and $g_l^- = k_{tj}$ for $l \equiv (t, j, i)$; $g_l^+ = g_l$ for $l \equiv (t, j)$ such that $\max\{0, s_{tj} - T_{tj}\} := s_{tj} - T_{tj}$ while obtaining, in the usual way, gradient element g_l , and $g_l^- = g_l^+ + P_{tj}$; see (5). Element $g_{k'}^j$ for $k' \in \beta_k$ and $\mathbf{x}_{k'} = {}_{(i)}a_{k'}$ is expressed as follows.

$$\begin{aligned} g_{k'}^j &:= g_{k'}^+ \mid (g_k^i = g_k^+ \wedge \rho_{k'k} = -1) \vee (g_k^i = g_k^- \wedge \rho_{k'k} = +1) \\ g_{k'}^j &:= g_{k'} \ , \quad \text{otherwise} \end{aligned} \tag{22}$$

Finally, indicator γ_k for nonbasic arc k is assigned as follows. For $\mathbf{x}_k = {}_{(l)}a_k$, $\gamma_k = +$ if $\lambda_k^i > \varepsilon$, where i = + ; otherwise, $\gamma_k = 0$ where ε is a positive tolerance (typically,10E-07) . For $\mathbf{x}_k = {}_{(u)}a_k$, $\gamma_k = -$ if $\lambda_k^i < -\varepsilon$, where i = - ; otherwise, $\gamma_k = 0$. For $\mathbf{x}_k = {}_{(i)}a_k$, it results

$$\begin{aligned} \gamma_k &:= 0 \mid \lambda_k^+ \leq \varepsilon \ \wedge \ \lambda_k^- \geq -\varepsilon \\ \gamma_k &:= + \mid \lambda_k^+ > \varepsilon \ \wedge \ \lambda_k^- \geq -\varepsilon \\ \gamma_k &:= - \mid \lambda_k^+ \leq \varepsilon \ \wedge \ \lambda_k^- < -\varepsilon \\ \gamma_k &:= j \mid \mid \lambda_k^j \mid = \mathrm{Max}\, \{ \mid \lambda_k^+ \mid, \mid \lambda_k^- \mid \} \text{, otherwise} \end{aligned} \tag{23}$$

Thus , assign $\lambda_k := \lambda_k^i$ for $i = \gamma_k$.

Note that expression (22) is based on the direction i and orientation $\rho_{k'k}$. The ambiguity on $g_{k'}^j$ for $k' \in \beta_k \cap \beta_l \mid \mathbf{x}_{k'} = {}_{(i)}\mathbf{a}_{k'}$ is solved by blocking the nonbasic arc, k or l that satisfies test t1 (see below); note that there is not a null step for t_1 since λ is used as the stepdirection of set **D** , the solution of a superbasic arc is not, by definition, at any of its bounds, and a maximal basis spanning tree is assumed (i.e., there is not any j in $\mathbf{B}^{(p)}$ $\forall$ p $\in$ P such that $\mathbf{x}_j$ is at any of its bounds).

5.2 BLOCKING NONBASIC ARCS.

A maximal basis spanning tree avoids degenerate basic-superbasic pivots, but it does not prevent null steps when a non basic arc is de-activated. Therefore, a mecanism is needed for testing whether a nonbasic arc, say k must be considered as a candidate to be de-activated. It may be carried out at the same time the nonbasic arc is priced (i.e., its Lagrange multiplier estimate is calculated) and then γ_k is set to 0 if otherwise a null step could not be prevented. Thus, $\gamma_k = 0$ if t1$\vee$t2$\vee$t3 , where t1, t2 and t3 are the result of the following blocking tests:

t1: $|\lambda_k| = \min\{ |\lambda_k|, |\lambda_l| \mid \mathbf{x}_j = {}_{(i)}a_j$ for $j \in \beta_k \cap \beta_l \wedge l \in \mathbf{D} \wedge \gamma_l \neq 0 \wedge \gamma_k \gamma_l \rho_{jk} \rho_{jl} = - \}$

Note that Γt1 if the flow in basic arc j changes in the same direction for any flow change in the appropriate direction of arcs k and l (given by γ_k and γ_l, respectively).

t2: (case $\gamma_k = +$) : $\exists$ $j \in \beta_k$ such that
$\rho_{jk} = -$ 1(reverse) $\wedge$ ($\mathbf{x}_j = {}_{(u)}a_j$) or
$\rho_{jk} = +$ 1(forward)$\wedge$ ($\mathbf{x}_j = {}_{(l)}a_j$)

t3: (case $\gamma_k = -$) : $\exists$ $j \in \beta_k$ such that
$\rho_{jk} = -1 \wedge$ ($\mathbf{x}_j = {}_{(l)}a_j$) or
$\rho_{jk} = +1 \wedge$ ($\mathbf{x}_j = {}_{(u)}a_j$)

If t1$\vee$t2$\vee$t3 we refer to arc k as a *blocked* arc and, then, it will not be a candidate to be de-activated.

5.3 OBTAINING INDEPENDENT SET $\mathbf{D}^{(p)}$ TO BE DE-ACTIVATED.

Let $\mathbf{C}^{(p)}{}_d \,\Delta\, \mathbf{B}^{(p)}{}_d \cup \mathbf{D}^{(p)}$ where $B^{(p)}{}_d \,\Delta \cup \beta_k$ $\forall$ $k \in \mathbf{D}^{(p)}$. An arc, say k to be de-activated must be included in set $\mathbf{D}^{(p)}$ if any move $d_k \neq 0$ effects the solution feasibility or the objective function coefficient of any arc from set $\mathbf{C}^{(p)} \cup \mathbf{C}^{(p)}{}_d$;

formally, $D^{(p)} \Delta D^{(p)} \cup \{k\}$ for $k \in \mathbf{D}$ if $t4 \wedge (t5 \vee t6)$, where t4,t5, and t6 are the result of the following including tests:

t4: $| \mathbf{S}^{(p)} | + | \mathbf{D}^{(p)} | < \tau$

t5: $(\mathbf{B}^{(p)} \cup \mathbf{B}^{(p)}{}_d) \cap \beta_k \neq \{\phi\}$

t6: $\exists\, G_{gg'} \neq 0 \mid (g \in \{k\} \cup \beta_k) \wedge (g' \in \mathbf{C}^{(p)} \cup \mathbf{C}^{(p)}{}_d)$

where τ is a given upper bound (typically, 60).

Now, to assure that sets $D^{(j)}$ $\forall\, j \in P$ are independent, it is required to analyse if any move $d_k \neq 0$ $k \in \mathbf{D}^{(p)}$ will effect the solution feasibility or the objective function coefficient of any arc from the q-th set $\forall\, q \in P/\{p\}$; in that case, both sets $\mathbf{D}^{(p)}$ and $\mathbf{D}^{(q)}$ must be joined. Formally, $\mathbf{D}^{(p)} \Delta \mathbf{D}^{(p)} \cup \mathbf{D}^{(q)} \cup \{k\}$ and $\mathbf{D}^{(q)} \Delta \{\phi\}$ if arc k simultaneously satisfies $t5 \vee t6$ for the p-th and q-th current independent sets and, besides, the following *joining* test is satisfied.

t7: $\Sigma_{i \in \{p,q\}} | \mathbf{S}^{(i)} | + | \mathbf{D}^{(i)} | < \tau$

If $\Gamma t7$ then arc k is not de-activated and, then, $\mathbf{D} \Delta \mathbf{D} / \{k\}$; if as a result, $\mathbf{D}=\{\phi\}$, then the set $\mathbf{C}$ must be revisited and partitioned in as many as possible independent sets and, as a final solution, the upper bound τ must be temporarily incremented to a suitable value. Note that the procedure is executed during the de-activating process and, then, hopefully, after many basic-superbasic arcs have been activated on the (sub) optimization of the previous manifold.

CONCLUSIONS.

In this paper we have presented a rough algorithm that takes into account second-order information for solving a type of large nonlinear network problems; its main ideas may be easily extended to the general sparse case.

Taking advantage of the structure of the Hessian matrices G_3 and G_4 and being the constraints system a replicated quasi-arborescence, the main features are as follows. Null steps are prevented, even although the objective function has some discontinuities, since the basis spanning tree is kept maximal and an ad-hoc blocking de-activating strategy is used. The new concept of independent superbasic sets is introduced so that the Truncated-Newton method and the linesearch procedure can be used for optimizing "in parallel" the manifold of each independent basic-superbasic set. Given the special structure of the Hessian matrix G and the variable-reduction matrix Z , the computer effort for obtaing the vector $q^{(i)}$ (13), at each minor iteration i , is within affordable limits, since the cardinality of each independent set is usually small. One of the main reasons for not using the basic estimation of the Lagrange multipliers of the nodes (apart the nonlinearity of the objective function), is precisely the size of the basic equivalent path of each superbasic or nonbasic arc.

ACKNOWLEDGEMENT

I wish to thank Ron Dembo for various guiding comments and suggesstions that have strongly improved the paper and Rick Rosenthal for making available to the author his code on the same subject.

REFERENCES.

DEMBO, R.S. AND J.G. KLINCEWICZ (1981). A scaled reduced gradient algorithm for network flow problems with convex separable costs. *Mathematical Programming* **15**, pp. 125-147

DEMBO, R.S. AND T. STEIHAUG (1983). Truncated-Newton algorithms for large-scale unconstrained optimization. *Mathematical Programming* **26**, pp. 190-212.

ESCUDERO, L.F. (1983). An structural-based motivation for using the Truncated Newton approach in the hydropower generation management. A nondifferentiable nonlinear network flow problem, *Questiio* **7**, pp. 437-450.

ESCUDERO, L.F. (1984). On diagonally-preconditioning the Truncated Newton method for super-scale linearly constrained nonlinear programming. *European J. of Operational Research* **17**, 401-414.

ESCUDERO, L.F.(1985). Sobre la amplitud de paso multivariante en programación no lineal con condiciones lineales. E.T.S.I. de Minas, Madrid. Special issue on honor of Prof. E. Chacón (in press).

MURTAGH, B. AND M. SAUNDERS (1978). Large-scale linearly constrained optimization. *Mathematical Programming* **14**, pp 41-72.

ROSENTHAL, R.E. (1981). A nonlinear network flow algorithm for maximization of benefit in a hydroelectric power system. *Operations Research* **29**, pp. 763-786.