

ON THE SOLUTION OF A PARTIALLY STRUCTURED NONLINEAR PROGRAM

P. D. SIMMS

(Received 10 April 1978)

(Revised 6 December 1978)

Abstract

A class of partially structured nonlinear programming problems, containing the capacitated nonlinear minimum cost multicommodity flow problem, is considered. Such problems, although large, can often be solved efficiently and with minimal computational storage by gradient projection methods.

1. Introduction

It has been successfully demonstrated by Berry [1, 2] that the nonlinear minimum cost multicommodity network flow problem (which is a large scale structured nonlinear programming problem), arising from a study of designing economic alternative routing telephone networks, can be efficiently solved by using a modified version of Rosen's gradient projection method [11]. The success of this application of Rosen's method hinges about the fact that the structure of the problem gives rise to explicit analytic expressions for the direction vector and the estimates of the Lagrange multipliers, thus eliminating any matrix manipulations. This is particularly important as the programs have typically 5000–10,000 variables and further the second order derivatives are not always available.

From consideration of this success, we have considered techniques for the efficient solution of generalized programs which will now include the additional link capacity constraints. In doing this, we have attempted to utilize fully the structure of the original uncapacitated programs, with the aim of minimizing computational storage and the computation involved.

The mathematical programs under consideration are:

Minimize $f(x)$ subject to

$$\begin{aligned} \sum_{j=1}^{\phi(k)} x_j^k &= b^k, \quad k = 1, \dots, K, \\ \sum_{k=1}^K \sum_{j=1}^{\phi(k)} \gamma_{ij}^k x_j^k &\leq c_i, \quad i = 1, \dots, L, \\ x_j^k &\geq 0, \quad j = 1, \dots, \phi(k), \quad k = 1, \dots, K. \end{aligned} \quad (1.1)$$

where f is a differentiable function. In addition, it is assumed that K will often be much larger than L , and L will be relatively small (that is of the order of 200).

The structure that the constraints $\sum_{j=1}^{\phi(k)} x_j^k = b^k$, $k = 1, \dots, K$, have is commonly known as Generalized Upper Bounding (GUB). This structure has been thoroughly investigated in linear programming. The results of this investigation have been incorporated into the implementation of many commercial mathematical programming systems when $f(x)$ is linear. However, when $f(x)$ is nonlinear these results do not generalize in an obvious way. Thus it is necessary to find an alternative exploitation of the GUB structure.

2. Some properties of partitioned matrices

In the implementation of Rosen's method, where the matrix N has rows containing the normals of the currently active constraints, the vector $-\nabla f(x)$ is projected by the projection matrix

$$P = I - N^T [NN^T]^{-1} N \quad (2.1)$$

into the subspace satisfying the active constraints, giving the search direction

$$v = -P\nabla f(x). \quad (2.2)$$

In addition, at various stages during the application of Rosen's method, the estimates of the Lagrange multipliers

$$d = N^+ \nabla f(x), \quad (2.3)$$

where

$$N^+ = [NN^T]^{-1} N \quad (2.4)$$

will be required.

It can be seen that, for large scale problems, the storage required for the matrices $[NN^T]^{-1}$, N^+ or P becomes prohibitive.

We will now state a result that shows that an orthogonal projection can be computed in two parts, thus making the application of Rosen's method more practical.

LEMMA. Suppose the matrix $N = \begin{bmatrix} N_1 \\ N_2 \end{bmatrix}$ has full row rank. If P_0 and P are the projection matrices corresponding to N_1 and N , and if

$$T = N_2 P_0 N_2^T, \quad V = T^{-1} N_2 P_0, \quad W = I - N_2^T V,$$

then

$$(i) \quad P = P_0 W, \tag{2.5}$$

$$(ii) \quad N^+ = \begin{bmatrix} N_1^+ & W \\ & V \end{bmatrix}. \tag{2.6}$$

The proof of this lemma can be found in most textbooks dealing with pseudo-inverses (for example [4]).

There are two comments that can be made about the contents of this lemma. Firstly, it is important that efficient and stable methods for handling the matrix $T^{-1} = (N_2 P_0 N_2^T)^{-1}$ are used. To do this we will use the Cholesky decomposition of T and take advantage of the current state of the art for updating the factorization when the set of active constraints changes.

Secondly, it can be seen from Berry [1] that if the rows of N_1 contain the normals of the currently active constraints of the form

$$\sum_{j=1}^{\phi(k)} x_j^k = b^k, \quad x_j^k \geq 0,$$

then for an arbitrary vector $w = [w_1^1 \dots w_{\phi(1)}^1 \dots w_1^K \dots w_{\phi(K)}^K]^T$ we have

$$[P \quad w]_j^k = \begin{cases} 0 & \text{if } x_j^k = 0, \\ w_j^k - (1/p(k)) \sum_{i=1}^{\phi(k)} \delta_i^k w_i^k & \text{otherwise,} \end{cases} \tag{2.7}$$

where

$$\delta_i^k = \begin{cases} 0 & \text{if } x_i^k = 0, \\ 1 & \text{otherwise,} \end{cases}$$

and

$$p(k) = \sum_{i=1}^{\phi(k)} \delta_i^k.$$

Also, the elements of $N_1^+ w$ corresponding to $x_j^k = 0$ are numerically equal to

$$w_j^k - (1/p(k)) \sum_{i=1}^{\phi(k)} \delta_i^k w_i^k \tag{2.8}$$

and the elements of $N_1^+ w$ corresponding to $\sum_{j=1}^{\phi(k)} x_j^k = b^k$ are

$$(1/p(k)) \sum_{i=1}^{\phi(k)} \delta_i^k w_i^k. \tag{2.9}$$

Thus the operators P_0 and N_1^+ can be applied efficiently to an arbitrary vector w . This can be used to full advantage by letting the rows of N_2 contain the normals of the active constraints of the form:

$$\sum_{k=1}^K \sum_{j=1}^{\phi(k)} \gamma_{ij}^k x_j^k \leq c_i,$$

and calculating the projection vector and estimates of the Lagrange multipliers by (2.5) and (2.6) respectively.

3. Updating the factorization

3.1. Constraint added to N_2

If $N_2 P_0 N_2^T$ has the Cholesky decomposition LDL^T then

$$\begin{bmatrix} N_2 \\ \gamma_{r+1}^T \end{bmatrix} P_0 \begin{bmatrix} N_2^T & \gamma_{r+1} \end{bmatrix}$$

will have the Cholesky decomposition $\bar{L}\bar{D}\bar{L}^T$, where $\bar{L} = \begin{bmatrix} L & 0 \\ y^T & 1 \end{bmatrix}$ and

$\bar{D} = \begin{bmatrix} D & 0 \\ 0 & \delta^2 \end{bmatrix}$. The vector y is obtained as the solution to

$$LDy = N_2 P_0 \gamma_{r+1}$$

and the new diagonal element of \bar{D} is

$$\delta^2 = \|P_0 \gamma_{r+1}\|_2^2 - y^T Dy.$$

This simple updating procedure follows from the method described by Gill *et al.* [6, p. 532], and has the advantage that it only requires one transformation of a vector by the projection matrix P_0 . This updating procedure will be satisfactory if $\bar{L}\bar{D}\bar{L}^T$ is well conditioned, but otherwise it is not. This instability will occur if the new constraint is nearly linearly dependent on the existing set of constraints, in which case δ^2 is very small. The procedure will in fact fail if the quantity defining δ^2 is either zero or negative.

An alternative, although more expensive, method for computing δ^2 is the following,

solve $L^T w = y,$

compute $v = P_0(\gamma_{r+1} - N_2^T w),$

set $\delta^2 = \|v\|_2^2.$

This will ensure that δ^2 is non-negative but will still be unsatisfactory if δ^2 is small. This follows by comparison with the classical Gram-Schmidt procedure which is known to be numerically unreliable. If $A = P_0 N_2^T$, the Gram-Schmidt method would produce the factorization

$$A = ZL^T, \quad Z^T Z = D$$

and the updating procedure is equivalent to performing the next step of the Gram-Schmidt method, with the added disadvantage that the orthogonal matrix Z is not available but instead is replaced by $Z = A(L^{-1})^T$.

3.2. Constraint added to N_1

If the new N_1 matrix is $\begin{bmatrix} N_1 \\ q^T \end{bmatrix}$ and \bar{P}_0 is the corresponding projection matrix, then from (2.5)

$$\bar{P}_0 = P_0 - \frac{P_0 q q^T P_0}{q^T P_0 q}.$$

Thus

$$\begin{aligned} N_2 \bar{P}_0 N_2^T &= N_2 P_0 N_2^T - \frac{N_2 P_0 q q^T P_0 N_2^T}{q^T P_0 q} \\ &= LDL^T + \sigma z z^T, \end{aligned}$$

where $z = N_2 P_0 q$ and $\sigma = -1/(q^T P_0 q)$.

Techniques for calculating $\bar{L} \bar{D} \bar{L}^T$ from $LDL^T + \sigma z z^T$ have been extensively reported in the literature. A recent comprehensive report of such techniques can be found in Gill and Murray [7].

3.3. Constraints deleted from N_1

If the l th constraint from N is deleted, then it can be shown (see Appendix 1) that the resulting projection matrix is

$$\bar{P} = P + \frac{N^{+T} e_l e_l^T N^+}{(e_l^T N^+ N^{+T} e_l)}$$

where e_l is the l th column of the identity matrix.

If, in particular, the l th constraint is deleted from N_1 , then

$$\bar{P}_0 = P_0 + \sigma y y^T,$$

where $y = N_1^{+T} e_1$ and $\sigma = 1/\|y\|_2$, so that

$$N_2 \bar{P}_0 N_2^T = LDL^T + \sigma z z^T,$$

where $z = N_2 y$. Again suitable techniques can be used to form the Cholesky decomposition $\bar{L} \bar{D} \bar{L}^T$ of $N_2 \bar{P}_0 N_2^T$.

3.4. Constraint deleted from N_2

Consider the case when the l th constraint is to be deleted. For convenience we will partition L and D as follows

$$L = \begin{bmatrix} L_1 & 0 & 0 \\ L_2 & 1 & 0 \\ L_3 & L_4 & L_5 \end{bmatrix} \quad \text{and} \quad D = \begin{bmatrix} D_1 & & \\ & d & \\ & & D_2 \end{bmatrix}$$

where L_1 and D_1 are $(l-1) \times (l-1)$, $L_2^T \in R^{l-1}$, L_3 is $(r-l) \times (l-1)$, $L_4 \in R^{r-l}$ and L_5 and D_2 are $(r-l) \times (r-l)$. It can be shown [for example 7] that the new Cholesky decomposition is $\bar{L} \bar{D} \bar{L}^T$, where

$$\bar{L} = \begin{bmatrix} L_1 & 0 \\ L_3 & \bar{L}_5 \end{bmatrix}, \quad \bar{D} = \begin{bmatrix} D_1 & \\ & \bar{D}_2 \end{bmatrix}$$

and

$$\bar{L}_5 \bar{D}_2 \bar{L}_5^T = L_5 D_2 L_5^T + d L_4 L_4^T.$$

4. Discussion

It was stated in Section 3.1 that both methods for extending the LDL^T factorization will be unsatisfactory if δ^2 is small. Consequently for practical problems where LDL^T is not well-conditioned the updating procedure will not be satisfactory. However, if the QR factorization of $P_0 N_2^T$ were available a more stable factorization could be obtained [6, p. 532].

In this case the orthogonal matrix is not available and although no satisfactory algorithm is currently available there are two methods offering possible remedies to this situation. These methods are both based on the method of iterative refinement. This method has been used previously to refine least-squares solutions and can be extended easily to include refinement of y and δ^2 .

The problem of updating R is intimately connected with the least-square problem, $\min \|Ax - b\|_2$. A method that does not require Q has been given recently by

Bjorck [3], in his Algorithm 2'. This may be described as follows with a slight extension to obtain $p = D^{\frac{1}{2}}y$ and δ .

1. Solve

$$LD^{\frac{1}{2}}p_0 = N_2P_0\gamma_{r+1}, \quad D^{\frac{1}{2}}L^T x_0 = p_0, \quad r_0 = P_0(\gamma_{r+1} - N_2^T x_0).$$

2. For $i = 0, 1, 2, \dots$ compute

$$\left. \begin{aligned} f_i &= P_0(\gamma_{r+1} - N_2^T x) - r_i, \\ g_i &= N_2 r_i, \end{aligned} \right\} \text{using double precision accumulation.}$$

Solve

$$LD^{\frac{1}{2}}\delta p_i = N_2P_0f_i + g_i, \quad D^{\frac{1}{2}}L^T \delta x_i = \delta p_i.$$

Compute

$$\delta r_i = f_i - P_0N_2^T \delta x_i,$$

$$p_{i+1} = p_i + \delta p_i, \quad x_{i+1} = x_i + \delta x_i, \quad r_{i+1} = r_i + \delta r_i.$$

Exit if $\|g_i\|_2$ is sufficiently small.

3. Set $\delta = \|r_i\|_2$.

Another known possibility is similar but a little simpler. This follows from a suggestion by Kahan in the paper by Golub and Wilkinson [10].

1. Solve

$$LD^{\frac{1}{2}}p_0 = N_2P_0\gamma_{r+1}, \quad D^{\frac{1}{2}}L^T x_0 = p_0.$$

2. For $i = 0, 1, 2, \dots$ compute

$$r_i = P_0(\gamma_{r+1} - N_2^T x_i) \quad \left\{ \begin{array}{l} \text{using double precision} \\ \text{(and saving } r_i \text{ as a double precision vector),} \end{array} \right.$$

$$g_i = N_2 r_i \quad \left\{ \begin{array}{l} \text{using double precision accumulation} \\ \text{(single precision } N_2 \text{ times double } r_i). \end{array} \right.$$

Solve

$$LD^{\frac{1}{2}}\delta p_i = g_i, \quad D^{\frac{1}{2}}L^T \delta x_i = \delta p_i.$$

Compute

$$p_{i+1} = p_i + \delta p_i, \quad x_{i+1} = x_i + \delta x_i.$$

Exit if $\|g_i\|_2$ is sufficiently small.

3. Set $\delta = \|r_i\|_2$.

In both methods it could be that $\|g_0\|_2$ is negligible, so that the extra work involved will not necessarily be substantial. It is common practice to use only single precision the first time through, which allows further economy for the likely event that $\|g_0\|_2$ does prove small.

We now have a solution technique for large partially structured problems that has a reasonable chance of success except that it is based on Rosen's gradient projection method. This is nothing more than the method of steepest descent within various subspaces, and the rate of convergence is typically extremely slow. However, the algorithm that this solution technique is based on uses a modified version of the gradient projection method, in that it allows the deletion of more than one constraint at the beginning of each iteration with due regard to preventing zig-zagging.

Acceleration devices have also proved to be successful in reducing the number of iterations needed to obtain a good approximation to the optimal solution. The conjugate gradient method of Goldfarb [9] can be used whenever the set of active constraints remains unchanged from one iteration to the next. This method can be described as follows:

1. $p^0 = -P\nabla f(x^0)$,

2. for $k = 0, 1, \dots, n-1$ define

$$x^{k+1} = x^k + \alpha^k p^k,$$

$$\beta^k = \frac{\|P\nabla f(x^{k+1})\|_2^2}{\|P\nabla f(x^k)\|_2^2},$$

$$p^{k+1} = -P\nabla f(x^{k+1}) + \beta^k p^k.$$

Another acceleration device that has proved to be successful is that of PARTAN acceleration. This has been discussed for the unconstrained case in [12]. The main feature of this device is that it can effectively compensate for the frequent stalling of the gradient projection method on steep ridges or in steep valleys.

An algorithm has been written to solve a large partially structured problem which is, in fact, an extension of the problem solved by Berry [2]. The problem had over 3400 variables and over 1100 GUB constraints. The new problem has the additional feature of 50 link capacity constraints. The modified gradient projection method as well as the acceleration devices have subsequently proved to offer a clear advantage over the standard gradient projection method. Near the approximate optimal solution 45 of these 50 constraints were active and no numerical instability was noticed so that the merits of the iterative refinement schemes have not yet been investigated.

5. Conclusions

It has been seen that a solution technique can be constructed for a large partially structured problem. The class of problems considered was that with GUB structure with additional linear inequality constraints. The viability of this approach is

mainly a result of (2.7), (2.8) and (2.9). In effect the information needed about a large part of the active constraint matrix (N_1) has been incorporated into two simply stated operators. This means that dense matrix methods are only needed to be used on a matrix which is the same size as N_2 . This approach is however not only limited to problems with GUB structure. For any problem for which similar operators can be constructed, a similar solution technique can be used.

Another feature of the solution technique is that any sparseness that N_2 might have can be fully exploited.

Acknowledgement

The author wishes to thank the referees for numerous suggestions and for additional references.

Appendix

From Fletcher's paper [5],

$$\begin{bmatrix} N_k^+ \\ 0 \end{bmatrix} = N_{k+1}^+ - N_{k+1}^+ \frac{VV^T}{V^T V},$$

where V is the $(k+1)$ st row of N_{k+1}^+ . It then follows that

$$\begin{aligned} [N_k^T \quad 0] \begin{bmatrix} N_k^+ \\ 0 \end{bmatrix} &= N_{k+1}^T \begin{bmatrix} N_k^+ \\ 0 \end{bmatrix} \\ &= N_{k+1}^T N_{k+1}^+ - N_{k+1}^T N_{k+1}^+ \frac{VV^T}{V^T V}. \end{aligned}$$

Hence

$$P_k = P_{k+1} + \frac{VV^T}{V^T V}.$$

Thus, if the l th constraint in N is deleted, the new projection matrix \bar{P} is given by

$$\bar{P} = P + \frac{N^{+T} e_l e_l^T N^+}{e_l^T N^+ N^{+T} e_l},$$

where e_l is the l th column of the identity matrix.

References

- [1] L. T. M. Berry, "On the solution of a structured non-linear programme", *J. Austral. Math. Soc. Ser. B* 19 (1975), 242-248.
- [2] L. T. M. Berry, *A mathematical model for optimizing telephone networks* (Ph.D. thesis, University of Adelaide, 1971).

- [3] A. Bjorck, "Comment on the iterative refinement of least-squares solutions", *J. American Stat. Assoc.* 73, 361 (1978), 161–166.
- [4] T. L. Boullion and P. L. Odell, *Generalized inverse matrices* (Wiley–Interscience, 1971).
- [5] R. Fletcher, "A technique for orthogonalisation", *J. Inst. Maths Applics* 5 (1969), 162–166.
- [6] P. E. Gill, G. H. Golub, W. Murray and M. A. Saunders, "Methods for modifying matrix factorization", *Math. of Comp.* 28 (1974), 505–535.
- [7] P. E. Gill and W. Murray, "Modification of matrix factorization after a rank one change", in *The state of the art in numerical analysis* (ed. D. Jacobs) (Academic Press, 1976).
- [8] P. E. Gill and W. Murray, "Numerically stable methods for quadratic programming", *Math. Programming* 14 (1978), 349–372.
- [9] D. Goldfarb, "Extension of Davidon's variable metric method to maximization under linear inequality and equality constraints", *SIAM J. Appl. Math.* 17 (1969), 739–764.
- [10] G. H. Golub and J. H. Wilkinson, "Note on the iterative refinement of least squares solutions", *Numerische Mathematik* 9 (1966), 139–148.
- [11] J. B. Rosen, "The gradient projection method for nonlinear programming. Part I. Linear constraints", *SIAM J. Appl. Math.* 8 (1960), 181–217.
- [12] B. V. Shah, R. J. Buehler and O. Kempthorne, "Some algorithms for minimizing a function of several variables", *SIAM J. Appl. Math.* 12 (1964), 74–92.

Department of Applied Mathematics
University of Adelaide
Adelaide, S.A. 5000