DOE/ER/25030--T1

DOE/ER/25030--T1

DE92 014086

Final technical report to
THE DEPARTMENT OF ENERGY
Contract No. DE-FG03-87ER25030

NEW APPROACHES TO LINEAR
AND NONLINEAR PROGRAMMING

January 1, 1987-December 31, 1989

Principal Investigators
Walter MURRAY
Michael A. SAUNDERS

Department of Operations Research
Stanford University
Stanford, California 94305-4022

Senior Research Personnel
Philip E. GILL

Department of Mathematics
University of California
San Diego, California

March 1990

MASTER

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

1 Project Description

This note summarizes technical progress during the past twelve months on DOE Contract DE-FG03-87ER25030. The project involves study of the theoretical properties and computational performance of techniques that solve linear and nonlinear programs by means of nonlinear transformations.

The group at the Systems Optimization Laboratory (SOL) were the first to recognize the connection between Karmarkar's (1984) projective method and the logarithmic barrier method (see Gill, Murray, Saunders, Tomlin and Wright, 1986). It is now generally recognized that essentially all interior-point methods for linear programming inspired by Karmarkar's method are closely related to application of Newton's method to a sequence of barrier functions (see e.g., Gonzaga, 1987; Renegar, 1988; Anstreicher, 1988). Each barrier function is defined from the objective function and a barrier term that is infinite along the boundary of the feasible region. As the weight on the barrier term is reduced to zero, the solution of the subproblem becomes closer to the solution of the original problem.

1.1 Summary

During the last twelve months, research has concentrated on barrier-function methods for *linear programming* (LP) and *quadratic programming* (QP). Some groundwork for the application of barrier methods to nonlinearly constrained problems has also begun.

In our previous progress report we drew attention to the difficulty of developing robust implementations of barrier methods for LP. We have continued to refine both the primal algorithm and the dual algorithm. We still do not claim that the barrier algorithms are as robust as the simplex method; however, the dual algorithm has solved all the problems in our extensive test set. We have also gained some experience with using the algorithms to solve aircrew scheduling problems.

Aeneas Marxen has now completed his thesis on the primal algorithm and a report describing his results has been published (see SOL 89-6). For a review of recent developments in interior-point methods for linear programming, with an emphasis on computational results, see SOL 88-14.

2 Linear Programming

In *all* interior-point methods for LP, the search direction is obtained from a system of the form

$$AD^2A^Tq = v, \quad (1)$$

where D is a diagonal matrix and v depends on the algorithm. If the right-hand side happens to be of the form $v = AD^2r$, this system is a set of "normal equations" equivalent to the linear least-squares problem

$$\min \|D(r - A^Tq)\|_2. \quad (2)$$

The matrix AD^2A^T is *large*, it could be quite *dense* compared to A , and in general it is *singular* or very nearly so.

The principal method developed at SOL uses the *exact* Cholesky factors of AD^2A^T , excluding if necessary a few dense columns of A . The main reason is that the sparsity pattern of the normal-equations matrix does not change as D changes; hence a single "analyze" can be performed on the sparsity pattern of AA^T to obtain an ordering of the rows of A that preserves the sparsity of the Cholesky factor. The same ordering is used for all subsequent factorizations $AD^2A^T = R^TR$.

The analyze, factorize and solve procedures are performed using the off-the-shelf equation solver SPARSPAK (see George and Liu, 1981). If necessary, a partitioning (Schur-complement) scheme with iterative refinement is used to deal with any dense columns removed from A before the Cholesky factorization.

2.1 LU Preconditioning

We have now implemented a dual algorithm based on solving (2) using a preconditioned conjugate-gradient method for least-squares problems. These results were presented at the SIAM Conference on Optimization in Boston (April 1989) and at the SIAM Annual Meeting in San Diego (July 1989).

Our first implementation of a primal barrier algorithm (Gill *et al.* 1986) also used preconditioned conjugate gradients. However, in that implementation the preconditioner was based on the Cholesky factor and (2) was solved to full accuracy. In the new dual implementation the preconditioner is based on an LU factorization of DA^T and only an approximate solution is obtained. We have experimented with a number of variants, one of which uses only a weighted basis from A ; i.e., LU factors of just a subset of the rows of DA^T .

The motivation behind this choice of preconditioner is the knowledge that LU factors are always sparse and the matrix DA^T tends towards a weighted basis. On any specific problem, an algorithm based on this preconditioner should use about the same memory as the simplex method. For a fuller description of why LU preconditioners should work well, see SOL 86-7.

The preliminary results confirm our hypothesis that a "good" approximate solution can be obtained in a relatively small number of conjugate-gradient steps. Moreover, the total number of iterations is not unduly affected by using only an approximate solution.

The CPU times for the LU -based algorithm have initially been disappointing. This is entirely due to the relative time of performing an LU factorization compared to that of the Cholesky factorization. Since the location of the nonzeros in the LU factors changes at each iteration, and since stability depends on careful row and column orderings, LU factorization is inherently more complex than Cholesky.

As it happens, the Cholesky factors for our test set are nearly all very sparse. A basic assumption we are making is that on at least some problems, the LU factors will be sparse when the Cholesky factors are not. (One 700-row problem in the test set did run 13 times faster using the LU approach.) Nevertheless, we clearly need to improve the speed of the LU factorization.

3 A PRIMAL BARRIER ALGORITHM FOR QUADRATIC PROGRAMMING 3

The LU approach may prove to be more beneficial on shifted barrier functions, since it can be shown that the shifts and the weights may be chosen so that the factors converge to those of B , a basis of A , and not those of a diagonal weighted basis. This is significant because DA^T is usually much more ill-conditioned than B itself. Such an approach would also fit well in a hybrid algorithm that incorporates the simplex method.

2.2 Primal-Dual Barrier Methods for LP

The usual view of barrier methods is to categorize them according to how they were derived. For example, if the method was derived by applying a barrier transformation to the primal LP problem, it is normally viewed as a primal barrier algorithm.

Recently we have shown that all of the different barrier algorithms may be derived from any single approach (e.g. applying the barrier transformation to the primal or the dual or the primal-dual LP). In some sense *all* the algorithms are primal-dual algorithms in that they all generate estimates to both the primal and dual solutions. What has distinguished say a primal barrier method is that only primal feasibility was maintained.

We have now shown how to maintain in a so-called primal barrier or dual barrier algorithm both primal feasible and dual feasible estimates of a solution. Hence these algorithms can, if it proves desirable, mimic the distinguishing property of a primal-dual barrier algorithm. A talk describing the new approach was presented at the Asilomar meeting on "Recent Advances in Mathematical Programming" (February 1990).

3 A Primal Barrier Algorithm for Quadratic Programming

A quadratic program may be stated in the following format:

$$\begin{array}{ll} \text{QP} & \begin{array}{l} \text{minimize} \\ \text{subject to} \end{array} \end{array} \quad \begin{array}{l} c^T x + \frac{1}{2} x^T H x \\ x \in \mathbb{R}^n, \quad A x = b, \\ x \geq 0, \end{array}$$

where H is an $n \times n$ symmetric matrix and A is an $m \times n$ matrix with $m \leq n$. There are many mathematically equivalent forms of a QP problem. Our analysis shows that for barrier methods it is critical to use the above form.

The primal barrier methods developed for LP are readily extended to quadratic programming. This is not the case, however, for dual or primal-dual methods, since in general the dual of QP (although it may formally exist) does not have a useful relationship to the primal problem.

Eliminating the inequality constraints $x \geq 0$ by the use of a barrier term leads

3 A PRIMAL BARRIER ALGORITHM FOR QUADRATIC PROGRAMMING⁴

to the problem

$$\begin{aligned} \text{BQP} \quad & \underset{x \in \mathbb{R}^n \in \mathbb{R}^n}{\text{minimize}} \quad c^T x + \frac{1}{2} x^T H x - \mu \sum_{j=1}^n \ln x_j \\ & \text{subject to} \quad Ax = b, \end{aligned}$$

where μ is the barrier parameter. Let $x^*(\mu)$ be a minimizer for BQP. It follows from standard theory (see Fiacco and McCormick, 1968) that under mild conditions,

$$\lim_{\mu \rightarrow 0} d(\mu) = 0,$$

where $d(\mu)$ is the shortest distance from $x^*(\mu)$ to a minimizer of QP. An immediate difference from the LP case is that *both* QP and BQP may not have unique minima.

Applying Newton's method for linearly constrained problems to BQP leads to the following subproblem for the search direction:

$$\begin{aligned} & \underset{p \in \mathbb{R}^n \in \mathbb{R}^n}{\text{minimize}} \quad g_B^T p + \frac{1}{2} p^T H_B p \\ & \text{subject to} \quad Ap = 0, \end{aligned}$$

where g_B and H_B are the gradient and Hessian of the objective function in BQP. It is well known that p may be obtained from the KKT system

$$K \begin{pmatrix} -p \\ \pi \end{pmatrix} = \begin{pmatrix} g_B \\ 0 \end{pmatrix} \quad \text{where} \quad K = \begin{pmatrix} H_B & A^T \\ A & 0 \end{pmatrix}. \quad (3)$$

The same equations arise for the LP problem, but H_B is then a positive definite diagonal matrix and the system defining p can be transformed into a linear least-squares problem of dimension m . Unfortunately this is not generally possible for (3). Equations of this form arise in many algorithms for *constrained* nonlinear problems, and the efficient solution of such equations is clearly vital.

3.1 Ill-conditioning

It was shown by Murray (1969) that if the number of constraints active at the solution is greater than zero but less than n , then

$$\lim_{\mu \rightarrow 0} \kappa(\mu) = \infty,$$

where $\kappa(\mu)$ is the condition number of $K(x^*(\mu))$. It was partly due to this undesirable property that interest in barrier functions waned. We can now show that under certain special circumstances this property, although still true, is benign.

It is easy to see that the ill-conditioning is not benign if the problem is not in the form proposed. Suppose the format admits some general inequality constraints, say $\hat{A}x \geq \hat{b}$, as well as the bounds $x \geq 0$. The Hessian of the barrier function is then of the form

$$H_B = H + \mu \hat{A}^T \hat{D} \hat{A} + \mu D,$$

3 A PRIMAL BARRIER ALGORITHM FOR QUADRATIC PROGRAMMING 5

where $\hat{D}$ and D are diagonal matrices whose diagonal elements are $1/(\hat{a}_i^T x - \hat{b}_i)^2$ and $1/x_i^2$ respectively. If any of the general inequality constraints are active at the solution, it is easily appreciated that sensitivity to the precision of H is lost when forming H_B , since it is a property of active constraints that

$$\lim_{\mu \rightarrow 0} \mu / (\hat{a}_i^T x - \hat{b}_i)^2 = \infty.$$

The same situation must arise if a simple bound is active; however, it is only the precision of the corresponding diagonal of H that is affected. A formal analysis shows that this is benign. Specifically we can show that the relevant condition number governing the sensitivity of (3) is κ_{FR} , the condition number of the KKT matrix of the original QP in just the variables that are not on their bound at the solution x^* .

A talk describing the new algorithm was presented at the SIAM Conference on Optimization (Boston, April 1989). An implementation of the algorithm outlined is under development (Ponceleón, 1990).

3.2 Solution of the KKT System

The equations (3) may be solved using MA27 (Duff and Reid, 1983), a sparse implementation of the Bunch-Parlett factorization. In the current implementation of the QP algorithm, this is the approach adopted.

Recently, we have investigated solving these equations using the conjugate-gradient-like algorithm SYMMLQ (Paige and Saunders, 1975) with preconditioning. Finding good preconditioners for indefinite systems like (3) has in the past proved more difficult than finding preconditioners for positive definite systems such as (1). However, we have now discovered a means of constructing a preconditioner from the factors generated by MA27. This may be likened to the construction of preconditioners for positive definite systems using Cholesky's algorithm. Since the preconditioner has to be positive definite it is not possible to solve (3) in one iteration even when the preconditioner is exact. However, we have shown that under ideal circumstances only two iterations are required by SYMMLQ.

We may of course still solve systems of the form (3) even when solving LP's. Ordinarily this is computationally expensive when compared to solving the normal equations. However, in the special case of LP the preconditioning scheme we propose for QP reduces to finding the LU factors of a basis of A , i.e., we no longer require MA27.

A talk describing the new preconditioning schemes was presented at the Asilomar meeting on "Recent Advances in Mathematical Programming" (February 1990). From the point of view of numerical accuracy it is much better to solve (3) than (1) or (2). Moreover, the occurrence of dense columns or free variables is of little consequence when solving (3).

4 Barrier Methods for Nonlinear Programming

Preliminary work has begun on barrier methods to solve the large-scale nonlinear programming problem

$$\begin{array}{ll} \text{NP} & \begin{array}{l} \text{minimize} \quad F(x) \\ x \in \mathbb{R}^n \in \mathbb{R}^n \\ \text{subject to} \quad h(x) = 0, \\ x \geq 0, \end{array} \end{array}$$

where $F(x)$ is a smooth nonlinear function and $h(x)$ is a vector of smooth nonlinear functions. (We assume that any nonlinear inequality constraints have been converted to equalities by the addition of non-negative slack variables. For the same reasons as those given in the QP case, this form of the problem is crucial.)

Sequential quadratic programming (SQP) methods are widely considered the most effective general techniques for solving nonlinearly constrained problems; see Fletcher (1981) and Gill, *et al.* (1981). Our approach has been to synthesize SQP methods with the barrier approach. This may be done in two ways. In an SQP algorithm it is necessary to solve a QP at each iteration. Clearly, one way we could do the synthesis is to solve the QP subproblems using the barrier algorithm described in the previous section. Alternatively, we could eliminate the inequality constraints $x \geq 0$ in NP using a barrier function, then solve the resulting equality-constrained problem using an SQP algorithm. The QP subproblems that arise also have only equality constraints and may be solved via a single KKT system (3).

There are virtues to both approaches and both have been tried on problems arising in optimal power flow. The preliminary results have been very encouraging when compared to a pure SQP algorithm. We have confirmed our theoretical analysis that the ill-conditioning of (3) is of no consequence. Moreover, many of the numerical difficulties common in barrier methods for LP did not arise on the problems solved. We stress that this is only a preliminary forage into a very difficult area. However, the results we have obtained have far surpassed our expectations.

The current theoretical basis of SQP methods requires the precise solution of the QP subproblems. To overcome this limitation we have begun a long-range study of using just an incomplete solution of the QP subproblem. The recently completed thesis of Francisco Prieto (SOL 89-7) was a significant step in accomplishing our goal. Aspects of this work were presented in an invited talk at the SIAM conference in Boston and at the CORS/ORSA/TIMS meeting in Vancouver (October 1989).

References

- I. S. Duff and J. K. Reid (1983). The multifrontal solution of indefinite sparse symmetric linear equations, *ACM Transactions on Mathematical Software* 9, 302-325.
- A. V. Fiacco and G. P. McCormick (1968). *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*, John Wiley and Sons, New York and Toronto.

- R. Fletcher (1981). *Practical Methods of Optimization. Volume 2: Constrained Optimization*, John Wiley and Sons, Chichester and New York.
- P. E. Gill, W. Murray and M. A. Saunders (1988). A single-phase dual barrier method for linear programming, Report SOL 88-10, Department of Operations Research, Stanford University, Stanford, CA.
- P. E. Gill, W. Murray and M. A. Saunders (1988). Interior-point methods for linear programming: a challenge to the simplex method, Report SOL 88-14, Department of Operations Research, Stanford University, Stanford, CA.
- P. E. Gill, W. Murray, M. A. Saunders, J. A. Tomlin and M. H. Wright (1986). On projected Newton barrier methods for linear programming and an equivalence to Karmarkar's projective method, *Mathematical Programming* 36, 183-209.
- P. E. Gill, W. Murray, M. A. Saunders and M. H. Wright (1986). A note on nonlinear approaches to linear programming, Report SOL 86-7, Department of Operations Research, Stanford University, Stanford, CA.
- P. E. Gill, W. Murray and M. H. Wright (1981). *Practical Optimization*, Academic Press, London and New York.
- C. C. Gonzaga (1987). Search directions for interior linear programming methods, Memorandum UCB/ERL M87/44, Electronics Research Laboratory, College of Engineering, University of California, Berkeley, CA.
- A. Marxen (1989). *Primal Barrier Methods for Linear Programming*, Ph.D. Dissertation, Report SOL 89-6, Department of Operations Research, Stanford University, Stanford, California.
- W. Murray (1969). *Constrained Optimization*, Ph.D. Thesis, University of London, London, England.
- C. C. Paige and M. A. Saunders (1975). Solution of sparse indefinite systems of linear equations, *SIAM J. on Numerical Analysis* 12, 4, 617-629.
- D. B. Ponceleón (1990). *Barrier Methods for Large-Scale Quadratic Programming*, Doctoral dissertation, Computer Science Department, Stanford University, Stanford, CA.
- F. Prieto (1989) *Sequential Quadratic Programming Algorithms*, Ph.D. Dissertation, Report SOL 89-7, Department of Operations Research, Stanford University, Stanford, California.

END

**DATE
FILMED**

7 / 6 / 92