An Interior-Point Krylov-Orthogonal Projection Method for Nonlinear Programming

M. Argáez H. Klíe
M. Ramé R.A. Tapia

June 1997
(revised August 1997)

TR97-16

AN INTERIOR-POINT KRYLOV-ORTHOGONAL PROJECTION METHOD FOR NONLINEAR PROGRAMMING

MIGUEL ARGÁEZ *, HÉCTOR KLÍE †, MARCELO RAMÉ ‡ AND RICHARD A. TAPIA §

Abstract. In this work we consider an inexact Newton's method implementation of the primal-dual interior-point algorithm for solving general nonlinear programming problems recently introduced by Argáez and Tapia. This inexact method is designed to solve large–scale problems. The iterative solution technique uses an orthogonal projection - Krylov subspace scheme to solve the highly indefinite and nonsymmetric linear systems associated with nonlinear programming. Our iterative method is a projection method that maintains linearized feasibility with respect to both the equality constraints and the complementarity conditions. This guarantees that in each iteration the linear solver generates a descent direction, so that the iterative solver is not required to find a Newton step but rather cheaply provides a way to march toward an optimal solution of the problem. This makes the use of a preconditioner inconsequential except near the solution of the problem, where the Newton step is effective. Moreover, we limit the problem to finding a good preconditioner only for the Hessian of the Lagrangian function associated with the problem plus a positive diagonal matrix. We report numerical experimentation for several large–scale problems to illustrate the viability of the method.

Keywords: Interior-point, primal-dual, nonlinear programming, Newton step, large linear system, Krylov subspace methods, iterative methods, preconditioners, preconditioned conjugated gradient method

AMS(MOS) subject classification:

1. Introduction.

In this work we are concerned with an inexact method to solve large–scale nonlinear programming problems using the primal-dual interior-point framework. We study the general nonlinear problem in the form

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & h(x) = 0 \\ & x \geq 0, \end{array} \tag{1}$$

where $h(x) = (h_1(x), h_2(x), \ldots h_m(x))^T$ and $f, h_i : \mathbb{R}^n \rightarrow \mathbb{R}$, $i = 1, 2, \ldots, m$ $(n \geq m)$ are twice continuously differentiable functions.

A Newton method applied to problem (1) using the interior-point approach leads to a nonsymmetric and highly indefinite linear system that must be solved for the current Newton step, i.e.,

* Department of Computational and Applied Mathematics, Rice University, Houston Texas 77251, USA; e-mail: mar@rice.edu. Research supported by Universidad del Valle, Departamento de Matemáticas, Cali, Colombia, the Center for Research on Parallel Computation, Rice University under - NSF Cooperative Agreement No. CCR-9120008, and DOE Grant DEFG05-86-ER25017.

† Department of Computational and Applied Mathematics, Rice University, Houston Texas 77251, USA; e-mail: klie@rice.edu. Support of this author has been provided by Intevep S.A., Los Teques, Edo. Miranda, Venezuela.

‡ Department of Computational and Applied Mathematics, Rice University, Houston Texas 77251, USA; e-mail: marcelo@rice.edu.

§ Department of Computational and Applied Mathematics and the Center for Research on Parallel Computation, Rice University, Houston Texas 77251, USA; e-mail: rat@rice.edu. This author was supported in part by DOE Grant DEFG05-86-ER25017.

$$\begin{pmatrix} A & B & -I \\ B^T & 0 & 0 \\ Z & 0 & X \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta y \\ \Delta z \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix}, \tag{2}$$

where $A \in \mathbb{R}^{n \times n}$, is a symmetric matrix not necessarily positive definite; $B \in \mathbb{R}^{n \times m}$, is a full rank rectangular matrix; and Z, X are positive diagonal $n \times n$ matrices.

The use of primal-dual interior–point methods for constrained minimization is relatively new [1, 2, 14]. The success of interior-point methods applied to linear programming problems has generated great research interest in trying to extend these ideas to a more general framework, i.e., quadratic, convex programming and general nonlinear programming applications. Recently, interest in large–scale applications of interior-point methods brought about the idea of performing inexact Newton steps, in view of the expense of using direct solution techniques. However, mainly due to the fact that the linear operators are highly indefinite the construction of a robust iterative method for the repeated solution of such systems has presented as a major challenge computational. Efforts in this direction are still sparse in the literature and a variety of alternatives have been examined in the inexact Newton solution of linear programming problems (see, e.g., [9]), as well as of nonlinear problems (see [13]). All things considered, at this point there are no conclusive results on how to define robust inexact Newton schemes for general optimization problems and the field still offers plenty of learning opportunities.

In the arena of the solution of large–scale systems of nonlinear equations, the Newton theory has been extended in order to allow the inexact solution of the Newton linear system [8, 13]. Most of these advances have relied on the use of Krylov subspace methods [6, 7]. Of particular interest for the optimizer is the formulation of robust iterative solution techniques for saddle-point problems, i.e., linear equations of the form

$$\begin{pmatrix} A & B \\ B^T & 0 \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} f \\ g \end{pmatrix}. \tag{3}$$

Interior-point method formulations of (1) can be cast in the form of (3) by explicit elimination of the linear complementarity condition in (2). Consequently, design of robust inexact algorithms for (2) can rely on results from the formulation of saddle-point iterative solvers [9]. Other approaches have attempted to solve (2) by a preconditioned conjugate residual method. The obvious trouble here is that finding efficient or even mildly acceptable preconditioners for the entire matrix may prove to be a futile effort.

Turning our attention to the saddle-point problem (3), the two major approaches to its solution are the iterative solution of the entire system and the direct, or iterative, solution of the resulting decoupled or reduced components of the system. The latter may mitigate the high computational cost associated with the former. However, effective preconditioners for system (3) are hard to obtain and in some cases the resulting lack of robustness can lead to procedures as expensive as the reduced approaches.

One reduced-system approach leads to the construction of the Schur complement, S of the coefficient matrix in (3), i.e., $S = -B^T A^{-1} B$. Clearly, this approach is convenient

if A can be easily inverted or good approximations to its inverse can be computed. In most cases this is not possible and sometimes, in order to overcome this drawback, nested or inner-outer iterations have to be employed to generate fair approximations of the Schur complement. The inexactness induced by the inner iterations spoils the symmetry and positive definiteness present in the exact Schur complement, hence an outer procedure has to employ general nonsymmetric solvers which potentially increases the computational cost.

Other optimization linear solvers of the reduction type are those restricting the search direction to the null space of B^T, denoted by $\mathcal{N}(B^T)$. Most versions rely on finding a null space basis matrix for B^T. Other methods (this idea constitutes the core of this work) are based on the restriction of A to $\mathcal{N}(B^T)$ as defined by an orthogonal projector $P = I - BB^\dagger$ onto $\mathcal{N}(B^T)$ with $B^\dagger = (B^TB)^{-1}B^T$ if B is of full rank.

In this research, we adopt and analyze the orthogonal projection method proposed by Bramley [5] to solve problems of the form (2). We conclude that the orthogonal projection method exploited in an intelligent way offers the best compromise between effectiveness and robustness.

One fundamental purpose of this work is to combine in an efficient manner the linear iterative solver obtained from the orthogonal projection method for (3) with the global nonlinear optimization algorithm recently presented by Argaéz and Tapia. Specifically, we prove that each iteration of the projection method applied to the linear system (2) produces a descent direction for the generalized augmented Lagrangian merit function introduced by Argáez and Tapia . Therefore, the linear iterative solver is not used as a way to find a Newton step, but rather as a direct tool to march toward the optimal solution of the problem. This makes the use of a preconditioner inconsequential except near the solution, where Newton's method is very effective. Moreover, we limit the problem to finding a good preconditioner only for the upper block which, upon removing the complementarity condition, is given by the Hessian of the Lagrangian function associated with the nonlinear programming problem plus a positive diagonal matrix.

This paper is organized as follows. In the following section we describe the nonlinear programming problem and summarize the interior-point algorithm given by Argaéz and Tapia in [3]. In Section 3 we focus our attention on the technical details supporting the orthogonal projection method. We stress the advantages and disadvantages of this method and its relation with other methods. In Section 4 we introduce our algorithm for solving the linear system (2) and possible strategies for preconditioning the resulting projected system. Section 5 compares numerically the orthogonal projection method against other methods and covers computational experiments on the proposed inexact global minimization algorithm. We summarize the main results of the paper and propose further work in Section 6.

2. The nonlinear interior-point framework. In this section we formulate a globalized primal-dual Newton interior-point method for solving the optimization problem (1) and its resulting associated linear system. We present an exact Newton version

of the interior-point algorithm and discuss some practical implementation issues as well.

2.1. Description of the problem. The Lagrange function associated with problem (1) is given by $l(x,y,z) = f(x) + h(x)^T y - x^T z$ where $y \in \mathbb{R}^m$ and $z \geq 0 \in \mathbb{R}^n$ are Lagrange multipliers associated with the equality and inequality constraints, respectively.

The Karush-Kuhn-Tucker (KKT) conditions for the problem are given by

$$F(x,y,z) \equiv \begin{pmatrix} \nabla f(x) + \nabla h(x) y - z \\ h(x) \\ XZe \end{pmatrix} = 0 \tag{4}$$

$$(x,z) \geq 0,$$

where $X = \text{diag}(x), Z = \text{diag}(z)$ and $e = (1,1,\ldots,1)^T \in \mathbb{R}^n$. Equations associated with the bottom block of (4), i.e., the strict complementarity condition $XZe = 0$, are replaced by the perturbed complementarity condition $XZe = \mu e$ ($\mu > 0$) in order to produce steps pointing away from the boundary of the feasible region and to promote the global convergence of the Newton interior–point method. For further details see El-Bakry *et al.* [1]. Therefore, the global convergence is based on the perturbed KKT conditions given by

$$F_\mu(x,y,z) \equiv \begin{pmatrix} \nabla f(x) + \nabla h(x) y - z \\ h(x) \\ XZe - \mu e \end{pmatrix} = 0 \tag{5}$$

$$(x,z) \geq 0,$$

where the perturbation parameter $\mu \geq 0$.

DEFINITION 2.1. *A point (x,y,z) is said to be an interior-point for problem (1) if $(x,z) > 0$.*

We conveniently denote $v = (x,y,z)$ and $\Delta v = (\Delta x, \Delta y, \Delta z)$. For a given $\mu > 0$, we define the Newton step at the interior point $v = (x,y,z)$ as $\Delta v = (\Delta x, \Delta y, \Delta z)$, which is obtained as the solution of the linear system

$$F'(v)\Delta v = -F_\mu(v), \tag{6}$$

where $\mu > 0$. In block form this is equivalent to

$$\begin{pmatrix} \nabla_x^2 l(v) & \nabla h(x) & -I \\ \nabla h(x)^T & 0 & 0 \\ Z & 0 & X \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta y \\ \Delta z \end{pmatrix} = - \begin{pmatrix} \nabla_x \ell(x,y,z) \\ h(x) \\ XZe \end{pmatrix} + \mu \begin{pmatrix} 0 \\ 0 \\ e \end{pmatrix}, \tag{7}$$

where $X = diag(x), Z = diag(z)$ and $x, z > 0 \in \mathbb{R}^n$ and $\mu > 0$.

Let us denote $\tilde{v} = (x,z)$ corresponding to the point $v = (x,y,z)$. Accordingly, for a given $\mu > 0$, we define the Newton step at $\tilde{v}$ to be $\Delta\tilde{v} = (\Delta x, \Delta z)$, where Δx and Δz are obtained from (7).

To guarantee that the update point $v_+ = v + \alpha\Delta v$ is an interior point, we choose a steplength $\alpha \in (0,1]$ such that

$$\tilde{v} + \alpha\Delta\tilde{v} > 0.$$

A standard way to find α in order to satisfy the above inequality is to let

$$\alpha = min\,(1, \tau\hat{\alpha}),$$

where

$$\hat{\alpha} = min\,\{\frac{-1}{min\,(X^{-1}\Delta x, -1)}, \frac{-1}{min\,(Z^{-1}\Delta z_k, -1)}\}, \tag{8}$$

for some $\tau \in (0,1)$.

2.2. A nonlinear interior-point algorithm. Before proceeding to present the algorithm given by Argaéz and Tapia in [3], we briefly establish some of its supporting ideas.

For a given fixed $\mu > 0$, a nonlinear subproblem is solved whose optimal solution satisfies the perturbed KKT conditions (5). At this point, μ is reduced and the process is repeated. Thereby, the optimal solution of the problem is attained as μ asymptotically approaches zero. The exact solution of each subproblem may turn out to be expensive in most practical situations. Therefore, we prefer to work with an approximate solution to the subproblem and derive adequate tools toward this end. The following definitions are in order.

DEFINITION 2.2. *An interior point (x,y,z) is said to be a quasi-central point for problem (1) if $h(x) = 0$ and $XZe = \mu e$ for a given $\mu > 0$.*

In regards to this definition it is important to specify proximity to a quasi-central point.

DEFINITION 2.3. *The $(\mu,\gamma)-$neighborhood of the quasi-central point corresponding to μ is defined by*

$$\mathcal{N}_\mu(\gamma) = \{\, \tilde{v} = (x,z) > 0 \in \mathbb{R}^{2n} : \ \|h(x)\|^2 + \|w - \mu w^{-1}\|^2 \leq \gamma\mu \,\}, \tag{9}$$

where $w = (XZ)^{1/2}e$, and $(\mu,\gamma) > 0$. The parameter $\mu\gamma$ is called the neighborhood radius.

The previous definition provides an effective measure of how close an interior -point is from satisfying the perturbed KKT conditions (5) for a given $\mu > 0$.

DEFINITION 2.4. *The merit function $M_\mu(x,y,z;\rho)$ is defined by*

$$M_\mu(x,y,z;\rho) = \ell\,(x,y,z) + \rho\,\Phi_\mu(x,z), \tag{10}$$

where $\ell\,(x,y,z)$ is the Lagrangian function associated with problem (1), i.e.,

$$\ell\,(x,y,z) = f(x) + h(x)^T y - x^T z.$$

The parameter ρ is a nonnegative scalar called the penalty parameter, and $\Phi_\mu(x,z)$ is the penalty term

$$\Phi_\mu(x,z) = \frac{1}{2}\, h(x)^T h(x) + x^T z - \mu \sum_{i=1}^{n} \ln(x_i z_i).$$

For a perturbation $\mu > 0$ and for the corresponding $(x_\mu^*, y_\mu^*, z_\mu^*)$ satisfying the perturbed KKT conditions (5), this merit function satisfies the following properties:

- x_μ^* is a stationary point for any ρ.
- The Hessian of M_μ with respect to x is positive definite at $(x_\mu^*, y_\mu^*, z_\mu^*)$ (i.e., $\nabla_x^2 M_\mu(x_\mu^*, y_\mu^*, z_\mu^*) > 0$) for sufficiently large ρ.
- $x_\mu^* = \arg\min M_\mu(x, y_\mu^*, z_\mu^*; \rho)$ for sufficiently large ρ, and finally
- For any interior-point v that it is not a quasi-central point the Newton step $\Delta\tilde{v}$ at $\tilde{v}$ is a descent direction for M_μ, for sufficiently large ρ, at v. i.e.,

$$\nabla_x M_\mu(x,y,z;\rho)^T \Delta x + \nabla_z M_\mu(x,y,z;\rho)^T \Delta z < 0.$$

The penalty parameter ρ associated with the generalized augmented Lagrangian function that guarantees a Newton descent direction is given by

$$\rho_k = \begin{cases} \hat{\rho}_k + \hat{c} & \text{if } \hat{\rho}_k + \hat{c} > \rho_{k-1} \\ \hat{\rho}_k + c_k & \text{otherwise,} \end{cases} \tag{11}$$

where $\hat{\rho}_k = \frac{\tilde{\nabla}\ell(v)^T \Delta v}{|\nabla\Phi_\mu(\tilde{v})^T \Delta\tilde{v}|}$, $c_k = \rho_{k-1} - \hat{\rho}_k$, and $c_k \geq \hat{c}$.

Under this framework, we now present the global primal-dual interior-point Newton algorithm presented by Argáez and Tapia in [3] for nonlinear optimization problems.

ALGORITHM 2.1. *(Line-search interior-point Newton algorithm)*

1. Consider an initial interior-point $v_0 = (x_0, y_0, z_0)$. Choose β, p, $\gamma \in (0,1)$ and $\hat{c} > 0$.
2. For $k = 0, 1, 2, \ldots$ until convergence do (OUTER LOOP)
   2.1 Choose $\mu_k > 0$.
3. Repeat (INNER LOOP)
   3.1 Solve the linear system

$$F'(v_k)\Delta v_k = -F_{\mu_k}(v_k).$$

   3.2 (Maintain x and z positive.) Choose $\tau_k \in (0,1)$ and compute $\hat{\alpha}_k$ according to (8). Let $\tilde{\alpha}_k = \min(1, \tau_k \hat{\alpha}_k)$.
   3.3 (Force a descent direction.) Calculate a c_k and a ρ_k by (11) to ensure a Newton descent direction for M_μ.
   3.4 (Armijo's condition of sufficient decrease). Find $\alpha_k = p^t \tilde{\alpha}_k$ where t is the smallest positive integer such that α_k satisfies

$$M_{\mu_k}(x_k + \alpha_k \Delta x_k, y_k, z_k + \alpha_k \Delta z_k; \rho_k) \leq M_{\mu_k}(v_k; \rho_k) + c_k \alpha_k \beta \nabla\Phi_{\mu_k}(\tilde{v}_k)^T \tilde{v}_k.$$

3.5 Set $v_k = (x_k + \alpha_k \Delta x_k, y_k + \alpha_k \Delta y_k, z_k + \alpha_k \Delta z_k)$

4. (Proximity to the quasi-central path)

4.1 If $v_k \notin \mathcal{N}_{\mu_k}(\gamma)$ (see(9))

go to step 3

4.2 Else

go to step 2 (END OF INNER LOOP)

REMARK 2.1. *Details on some key parameters choices are given in [3] Section 7.*

3. An orthogonal projection method. The main purpose of this work is to extend Algorithm (2.1) so that it can handle large–scale problems which often arise in applications. The Jacobian of the KKT conditions for real problems is in general a large, sparse and nonsymmetric matrix. Therefore, considering a direct method for solving the linear system of equations (2) is an expensive computational choice. We prefer instead to use an iterative method to solve the linear system of equations. We do this by considering the Krylov subspace methods as inexact solvers. This section discusses two techniques for solving the linear system (3), that are especially useful for large scale problems. The objective is to reduce the problem to a problem in a space of smaller dimension where the solutions are obtainable in a more convenient way. We briefly introduce the projected Hessian method and discuss its advantages and drawbacks. Also we discuss the orthogonal projection method, whose idea constitutes the core of this paper. We denote the nullspace of B^T by $\mathcal{N}(B^T)$ and the nullspace of B by $\mathcal{N}(B)$.

3.1. Reducing a linear system of equations. One technique used for solving the linear system (3) consists of reducing it to

$$Z^T A Z w = Z^T f, \tag{12}$$

where $Z \in \mathbb{R}^{nx(n-m)}$ is a matrix whose columns form a basis for the nullspace of B^T, $\mathcal{N}(B^T)$. There are two traditional numerical approaches for forming the operator Z in (12). In the first approach one performs a QR factorization of B, say

$$B = QR = [Q_1\ Q_2] \begin{pmatrix} R_1 \\ 0 \end{pmatrix}$$

where Q_1 consists of the first m columns of Q, Q_2 is an orthonormal basis matrix for $\mathcal{N}(B^T) \in \mathbb{R}^{nx(n-m)}$, and $R_1 \in \mathbb{R}^{mxm}$ is an upper triangular matrix. This suggests choosing $Z = Q_2$. It is important to note that in this case the matrix Z could be dense even though the matrix B is sparse, and therefore the required storage can be prohibitive. This is certainly an inconvenient feature even for moderate scale problems. The second deals with partitioning the matrix $B^T = [B_1\ B_2]$ where matrix $B_1 \in \mathbb{R}^{mxm}$ is nonsingular, and the null space basis is formed by $Z = \left[-B_1^{-1}B_2\ \ I_{n-m}\right]^T$. This approach for calculating Z is more widely used than the QR factorization, but it can fail if B is not full rank. In contrast with the QR factorization the sparsity of the matrix

B can be further exploited here. However, there exists an extra cost for an appropriate selection of the columns B_1 in order to preserve the sparsity of the matrix B. When using the QR factorization the matrix Z satisfies $Z^TZ = I$, and by Lemma 2 [13] we can conclude that Z^TAZ is not more ill-conditioned than A. However if the basis matrix Z is obtained by partitioning the matrix B^T, the reduction does not guarantee that Z^TAZ is not worse conditioned than the original Hessian matrix A, see Lemma 10 [13]. More details about this technique can be found in [15].

3.2. The orthogonal projection method. We have discussed the strengths and weaknesses of reducing the linear system (3) for medium to large-scale implementations. Rather than generating a null space basis matrix for B^T, the computation of an orthogonal projector onto $\mathcal{N}(B^T)$ is preferable for reasons of efficiency. This requires using $B^\dagger$ (i.e., the Moore-Penrose pseudo-inverse of B) in order to form an orthogonal projector P onto the nullspace of B^T, specifically

$$P = I - BB^\dagger. \tag{13}$$

If the matrix B has full rank, then $B^\dagger = \left(B^TB\right)^{-1}B^T$ is its Moore-Penrose pseudo-inverse. The idea now is to reduce the linear system (3) to the following projected system,

$$PAPu = Pf. \tag{14}$$

The following theorem describes the orthogonal projection method as a numerical solution method.

THEOREM 3.1. *Let P be given by (13). Then (Pu, p) solves the linear system (3) with $g = 0$ if and only if $PAPu = Pf$ and $p = B^\dagger(f - APu) + w$, for any $w \in \mathcal{N}(B)$.*

Proof. Let (Pu, p) be the solution of the linear system (3), then

$$APu + Bp = f.$$

Solving the latter equation for p, then the general solution is $p = B^\dagger(f - APu) + w$ where the first term is a particular solution, and the second term w represents the solution of its associated homogeneous problem, i.e., $Bw = 0$.
Now we substitute p into the same equation and solve for u. This gives

$$APu + B\left[B^\dagger(f - APu) + w\right] = f.$$

which can be reduced to

$$PAPu = Pf.$$

The proof for the reverse implication is as follows. Let $PAPu = Pf$
and $p = B^\dagger(f - APu) + w$, $w \in \mathcal{N}(B)$. Substitute p into $APu + Bp = f$ to obtain

$$APu + Bp = APu + B\left[B^\dagger(f - APu) + w\right]$$

$$
\begin{aligned}
&= APu + BB^{\dagger}f - BB^{\dagger}APu \\
&= PAPu + BB^{\dagger}f \\
&= Pf + (I - P)f \\
&= f.
\end{aligned}
$$

Furthermore, $B^T p = 0$ since p is a projection onto $\mathcal{N}\left(B^T\right)$. □

This result plays an important role in solving the linear system (2) which arises in nonlinear programming problems, and it will be discussed further in the next section.

3.3. Advantages of the projection method. One of the main advantages of approach (13) over (12) consist of avoiding the explicit computation of a basis for $\mathcal{N}\left(B^T\right)$. Therefore, issues such as instability and high cost are somewhat alleviated. Another advantage comes from the minimax characterization of singular values. The smallest singular values of A and PAP satisfy $\sigma_{min}(A) \leq \sigma_{min}(PAP)$ while the largest singular values satisfy $\sigma_{max}(A) \geq \sigma_{max}(PAP)$. Therefore, the condition number of the matrix PAP is no worse than the condition number of the matrix A [5]. Figure

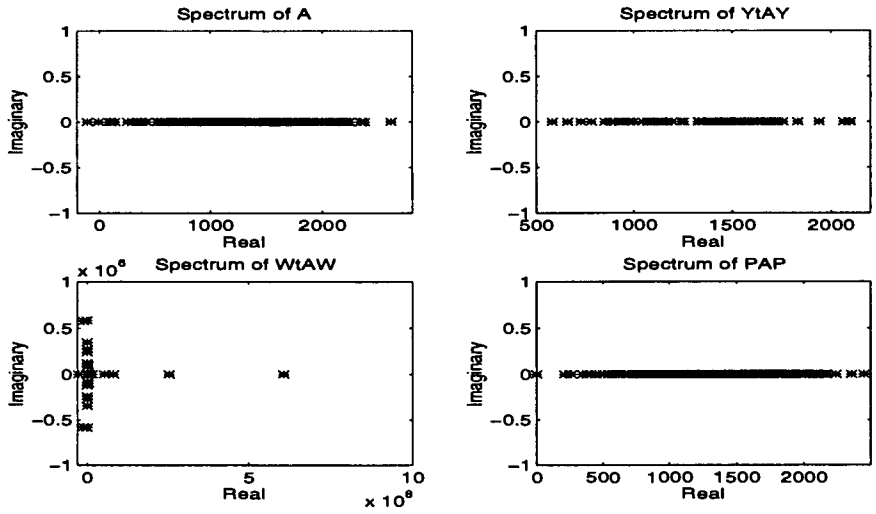

FIG. 1. *Spectra of A, Y^TAY ($Z = Y$, formed by the QR method), W^TAW, ($Z = W$, formed by the variable reduction method) and PAP, for a matrix A with random entries.*

1 illustrates the spectrums of A, both QR and variable reduction methods of Z^TAZ, and PAP for a random saddle-point matrix. Note that although the variable reduction method offers the opportunity to exploit matrix sparsity, the condition number obtained for Z^TAZ is much worse. Moreover, the possible number of distinct nonzero eigenvalues of PAP is at most $n - m$ compared to n for matrix A. This result suggests that the number of iterations required to solve equation (14) using any Krylov subspace method is not more than the number required to solve a single system with coefficient matrix A. In practice the number of iterations required by a Krylov subspace method to solve equation (14) depends on the distribution of its eigenvalues. Nevertheless, the projection method inherits the same drawbacks when B is highly sparse and the application of the

operator P can destroy the sparsity pattern.

The use of an orthogonal projection method brings to the foreground a subject that is often overlooked in the literature of preconditioning when solving systems like (3): the rank deficiency of B. This can happen during the process of solving the KKT linear systems. In order to overcome this problem, a positive constant ζ is introduced, which controls the amount of regularization of the system. This lends one to solve the following alternative problem

$$\begin{pmatrix} A & B \\ B^T & -\zeta I \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} f \\ g \end{pmatrix},$$

for $\zeta > 0$.

A careful selection of this parameter is required to prevent possible ill-conditioning and a deterioration of the rate of convergence. This adds another parameter to the interior-point method. So, it is important to notice that the use of a good preconditioner for the alternative problem has to take into consideration this parameter, introducing further complications for a good preconditioner. On the other hand, since rank deficiency of the matrix B can be admitted in the definition of the projector P, a robust singular value decomposition formula can be employed to carry out the orthogonal projection. Obviously, this introduces an extra computational cost in the whole procedure, but it does not represent an additional complication to the functionality of the projection method.

4. An orthogonal projection method for solving NLP linear systems. In this section, we explain how to reduce system (2) to a block symmetric system by a reordering of the unknowns and by a subsequent application of a orthogonal projection method. Then we apply a Krylov subspace method to solve the reduced block symmetric system which will allow us to define an iterative procedure to solve the entire system (2). This procedure leads to satisfaction of both linearized equality constraints and linearized complementarity conditions associated with problem (1). We discuss some main properties and details about this iterative method and its implementation.

Let the matrix A denote the Hessian of the Lagrangian function, the matrix B the gradients of the equality constraints, the vector b_1 the negative gradient with respect to x of the Lagrangian function, the vector b_2 the negative equality constraint functions, and finally the vector b_3 the negative perturbed complementarity conditions.

4.1. Reduction of the interior-point linear system. We introduce an application of the projection method for solving the linear system (2). The following theorem formalizes this technique.

THEOREM 4.1. *Let P be given as in (13). Then $(\Delta x, \Delta y, \Delta z)$ is a solution of the linear system (2) if and only if*

$$\Delta z = X^{-1}\left(b_3 - Z\Delta x\right),$$

$$\Delta y = B^{\dagger}(b_1 - A\Delta x + \Delta z) + w, \quad for \quad some \quad w \in \mathcal{N}(B),$$

and $\Delta x = P\Delta x_h + \Delta x_p$ *where* Δx_p *is a particular solution of* $B^T\Delta x = b_2$, *and* Δx_h *satisfies*

$$P(A + X^{-1}Z)P\Delta x = P(b_1 - A\Delta x_p + (X^{-1}b_3 + Z\Delta x_p)).$$

Proof. By a block row and column switch, equation (2) can be written as follows,

$$\begin{pmatrix} A & -I & B \\ Z & X & 0 \\ B^T & 0 & 0 \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta z \\ \Delta y \end{pmatrix} = \begin{pmatrix} b_1 \\ b_3 \\ b_2 \end{pmatrix}. \tag{15}$$

We denote the matrix blocks by

$$G = \begin{pmatrix} A & -I \\ Z & X \end{pmatrix}, \quad \bar{B} = \begin{pmatrix} B \\ 0 \end{pmatrix}, \quad u = \begin{pmatrix} \Delta x \\ \Delta z \end{pmatrix},$$

$$p = \Delta y, \quad f = \begin{pmatrix} b_1 \\ b_3 \end{pmatrix} \quad \text{and } g = b_2$$

so that the linear system can be written

$$\begin{pmatrix} G & \bar{B} \\ \bar{B}^T & 0 \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} f \\ g \end{pmatrix}. \tag{16}$$

The projection method can be readily applied when the equations corresponding to the lower matrix blocks are homogeneous. In order to fit this framework, we express u as $u = u_h + u_p$, where $u_h \in \mathcal{N}\left(\bar{B}^T\right)$ and u_p is a particular solution of $\bar{B}^T u = g$. Upon substituting into equation (16), one obtains

$$\begin{pmatrix} G & \bar{B} \\ \bar{B}^T & 0 \end{pmatrix} \begin{pmatrix} u_h \\ p \end{pmatrix} = \begin{pmatrix} \bar{f} \\ 0 \end{pmatrix}. \tag{17}$$

where $\bar{f} = f - Gu_p$. At this point, the projection Theorem 3.1 can be directly applied to equation (17), whose solution is found by solving

$$\bar{P}G\bar{P}u_h = \bar{P}\bar{f} \tag{18}$$

$$p = (\bar{B})^{\dagger}(\bar{f} - G\bar{P}u_h) + \bar{w}$$

, where $\bar{w} \in \mathcal{N}\left(\bar{B}^T\right)$.

It is straightforward to show that

$$\bar{B}^{\dagger} = \begin{pmatrix} B^{\dagger} & 0 \end{pmatrix} \text{ and } \bar{P} = \begin{pmatrix} P & 0 \\ 0 & I \end{pmatrix}, \tag{19}$$

where the projection operator P is defined by (13).
Substitution of $\bar{P}$ and $\bar{B}^{\dagger}$ into equation (17), yields

$$P(A + X^{-1}Z)P\Delta x_h = P(b_1 - A\Delta x_p + (X^{-1}b_3 + Z\Delta x_p)), \tag{20}$$

$$\Delta z = X^{-1}(b_3 - Z(\Delta x_p + P\Delta x_h)) \text{ and} \tag{21}$$

$$\Delta y = B^{\dagger}(b_1 - A(\Delta x_p + P\Delta x_h) + \Delta z) + w,\ w \in \mathcal{N}(B)\,. \tag{22}$$

In order to clarify the notation used in these equations, we mention that the decomposition of u into a homogeneous and a particular solution, which formally affects both Δx and Δz, leaves Δz unchanged because of the special structure of $\bar{B}$. □

We remark that the same formulation can be attained by eliminating the complementarity equation, thus reducing the problem to a 2×2 block symmetric system, and then applying the projection method.

4.2. Additional advantages of the orthogonal projection method. We notice that the computation of Δz and Δy, defined by equations (21) and (22), represent a small part of the overall computational cost for solving the system (2) because X is a diagonal matrix and the operator $B^{\dagger}$ has been previously computed to form the projector P, in contrast to solving the projected system (20).

In previous section, we mentioned that the projection method is attractive from the standpoint of its algebraic properties because the condition number of $P(A + X^{-1}Z)P$ is not worse than the condition number of the original matrix $A + X^{-1}Z$. Besides, we will show latter that if M is a good preconditioner for the matrix $\bar{A}$ then PMP is a good preconditioner for the matrix $P\bar{A}P$.

Indeed, several attempts have been made to find good preconditioners for the entire linear system (2) and have resulted in little or no success. When this system is reduced to the saddle point problem by eliminating the complementarity equation, it has also been difficult to define good preconditioners for the reduced system since the first block $A + X^{-1}Z$ will become ill-conditioned near the solution of the problem. Therefore, the use of the projection method for solving the linear system (2) as described in Theorem 4.1 puts us at a great advantage in terms of robustness.

When the dimension m (number of equality constrains) is small compared with the dimension n (number of variables), i.e. for instance $m \leq \frac{1}{4}n$, the projection method is strongly recommended. Also the method is of value, when m is large and the sparsity of the matrix B is such that the operator $B^T B$ has a high degree of sparsity. When this is not the case an option that deserves investigation to avoid the high cost associated with the computation of the projector P, is to establish a heuristic design such that the projector is fixed for some number of iterations during the minimization process. However, in both linear and quadratic programming, the projection operator is fixed and therefore need only be computed once, thus making the projection method very appealing even for large-scale problems for both of these classes of problems.

4.3. An iterative solution of the projected system. The standard second order sufficiency condition for Problem 1, states that at the solution x^* : For all $\eta \neq 0$ satisfying $\nabla h_i(x^*)^T\eta = 0$, $i = 1,\ldots,m$; $e_j^T\eta = 0$, $j \in \mathcal{B}(x^*)$ we have

$$\eta^T\nabla_x^2\ell(x^*,y^*,z^*)\eta > 0.$$

The set $\mathcal{B}(x^*)$ consists of the indices of the components of the primal variable that vanish at the solution, i.e., $\mathcal{B}(x^*) = \{j : x_j^* = 0\}$. This set is called the active set of indices. If the active set is nonempty, one should not expect the Hessian of the Lagrangian function, $\nabla_x^2\ell(x^*,y^*,z^*)$, associated to problem (1) to be positive definite in $\mathcal{N}\left(B^T\right)$ at the solution or, by continuity, in a neighborhood of the solution. Therefore, assuming the Hessian matrix A is positive definite in $\mathcal{N}\left(B^T\right)$ or the matrix $(A+X^{-1}Z)$ is positive definite over the entire space (as is assumed in some studies), compromises the success of general scope theoretical efforts. We assume that $(A+X^{-1}Z)$ is positive semidefinite in $\mathcal{N}\left(B^T\right)$. In this way, we set the stage for the second order sufficiency condition to be more closely met since the positive diagonal contribution (possibly large) added to the Hessian matrix A assists in shifting the spectrum of A towards the positive real axis. With this assumption we can apply the conjugate gradient method to solve the projected system (20). From a mathematical point of view, we can assert that the method converges in at most $(n-m)$ steps because the projected system has at most $(n-m)$ nonzero eigenvalues. However, in practice, the number of iterations that the conjugate gradient method requires depends also on the distribution of the eigenvalues of the projected system. Moreover, if the matrix $A+X^{-1}Z$ is not positive semidefinite on $\mathcal{N}\left(B^T\right)$, then a more general Krylov subspace method, such as SQMR [9], or MINRES [4] can be used.

4.4. Conjugate gradient method for solving the projected system. In this subsection we outline the conjugate gradient algorithm used for solving the projected linear system

$$P\bar{A}P\Delta x_h = P\bar{b_1},$$

where $\bar{A} = A + X^{-1}Z$, and $\bar{b_1} = b_1 - A\Delta x_p + (X^{-1}b_3 + Z\Delta x_p)$.

ALGORITHM 4.1. *(Conjugate gradient algorithm)*

1. Initialize $k = 0$, *given* $(\Delta x_h)_0$ initial guess.
2. Compute $r_0 = d_0 = P(\bar{b_1} - \bar{A}(\Delta x_h)_0)$.
3. Compute $\rho_0 = r_0^T r_0$.
4. For $k = 0,1,2,\ldots,$ do
   - 4.1 $w_k = P\bar{A}Pd_k$
   - 4.2 $\alpha_k = \frac{\rho_k}{d_k^T w_k}$.
   - 4.3 $(\Delta x_h)_{k+1} = (\Delta x_h)_k + \alpha_k d_k$.
   - 4.4 $r_{k+1} = r_k - \alpha_k w_k$.
   - 4.5 $\rho_{k+1} = r_{k+1}^T r_{k+1}$.
   - 4.6 If $(\rho_{k+1} < \varepsilon_p$ *stop*).

4.7 $\beta_{k+1} = \frac{\rho_{k+1}}{\rho_k}$.

4.8 $d_{k+1} = r_{k+1} + \beta_{k+1} d_k$.

The following theorem states several properties of this algorithm.

THEOREM 4.2. *Let $(\Delta x_h)_0 \in \mathcal{N}(B^T)$. If $\bar{A}$ is positive definite in $\mathcal{N}(B^T)$, and P is given by (13). Then*

(i) The conjugate gradient algorithm converges to the unique minimum norm solution of $P\bar{A}P\Delta x = P\bar{b_1}$.

(ii) The iterates $(\Delta x_h)_k$, the conjugate directions d_k, and the residuals r_k remain in $\mathcal{N}(B^T)$ for all k.

(iii) The matrix-vector product $w_k = PAPd_k$ (see substep 4.1) can be calculated instead by $w_k = PAd_k$ for all k.

Proof. (i) The system is consistent because the left and right side of the projected system are preceded by the same projection operator P.
Since $\bar{A}$ is positive definite on $\mathcal{N}(B^T)$ then

$$d_k^T w_k = (Pd_k)^T \bar{A}(Pd_k) > 0.$$

Therefore the step length α_k given by the conjugate gradient algorithm 4.2 is always well defined. Hence the convergence follows from the classical work of Hestenes and Stiefel [11].
(ii) The proof is done by induction. Since $(\Delta x_h)_0$, $d_0 = P(\bar{b_1} - \bar{A}P(\Delta x_{h_0})) \in \mathcal{N}(B^T)$ and assuming $(\Delta x_h)_k, d_k, r_k \in \mathcal{N}(B^T)$, then $w_k = PAPd_k \in \mathcal{N}(B^T)$, and

$$(\Delta x_h)_{k+1} = (\Delta x_h)_k + \alpha_k d_k \in \mathcal{N}(B^T).$$

$$r_{k+1} = r_k - \alpha_k w_k \in \mathcal{N}(B^T).$$

$$d_{k+1} = r_{k+1} + \beta_{k+1} d_k \in \mathcal{N}(B^T).$$

The unique minimum norm solution to $P\bar{A}P = P\bar{b_1}$ is one with $\Delta x_h \in \mathcal{N}(B^T)$.
(iii) Since the conjugate directions d_k are in the nullspace of B^T, we have $Pd_k = d_k$. Therefore the first projection operator, P, in the calculation of w_k can be omitted. i.e.,

$$w_k = PAd_k$$

□

REMARK 4.1. *The latter theorem shows that with a proper initialization, i.e.* $(\Delta x_h)_0 \in \mathcal{N}(B^T)$

1. *One projection P per conjugate gradient iteration need be compute in step 4 (a)*
2. *Since the iterates $(\Delta x_h)_k \in \mathcal{N}(B^T)$, we have*

$$P(\Delta x_h)_k = (\Delta x_h)_k. \tag{23}$$

Therefore Equation (20) can be replaced by

$$P(A + X^{-1}Z)(\Delta x_h)_k = P(b_1 - A\Delta x_p + (X^{-1}b_3 + Z\Delta x_p)). \tag{24}$$

3. *Also, it is important to comment, that the projector P does not need to be computed explicitly. Its action on a vector is given by*

$$Pv = (I - BB^{\dagger})v = (I - B(B^TB)^{-1}B^T)v.$$

Therefore by forming the Cholesky decomposition $LL^T = B^TB$ one can compute the action of P on a vector v by

3.1 $v_1 = B^Tv$,
3.2 Solve $Ly = v_1$,
3.3 Solve $L^Tz = y$,
3.4 $v_2 = Bz$,
3.5 $w = v - v_2$,
where $w = Pv$. It is important to point out that the Cholesky decomposition of B^TB can be made efficient by a reordering scheme based on the sparsity pattern of B.

4.5. Iterative solution of the complete system. Now, our fundamental purpose is to define an iterative solver for the linear system (2) using an iterative solver for equation (20) and to combine it with Algorithm 2.1 in an efficient manner. Specifically, we will prove that a single iteration of our solver on the linear system (2) produces a descent direction for the penalty term and it is also a descent direction for the modified augmented Lagrangian function presented in definition 2.4. Towards this objective, we present the following definition

DEFINITION 4.1. *Our iterative solver $(\Delta x_k, \Delta y_k, \Delta z_k)$ for the linear system (2) is defined as*

$$\Delta z_k = X^{-1}(b_3 - Z\Delta x_k), \tag{25}$$

$$\Delta y_k = B^{\dagger}(b_1 - A\Delta x_k + z) + w,\ w \in \mathcal{N}(B) \tag{26}$$

$$\Delta x_k = (\Delta x_h)_k + \Delta x_p, \tag{27}$$

where $(\Delta x_h)_k$ is given by conjugate gradient algorithm 9.3.1, and Δx_p is a particular solution of $B^T\Delta x = b_2$
Now, we present two strong theoretical result in the following theorems.

THEOREM 4.3. *Any search direction $(\Delta x_k, \Delta z_k)$ given by (27) and (25) satisfies linearized equality constraints and linearized complementarity condition associated with Problem (1), i.e. ,*

$$B^T\Delta x_k = b_2,$$

$$Z\Delta x_k + X\Delta z_k = b_3.$$

Proof. Substituting equation (27) into the second equation of the linear system (2) we obtain

$$B^T \Delta x_k = B^T((\Delta x_h)_k + \Delta x_p) \ ,$$

$$B^T \Delta x_k = B^T(\Delta x_h)_k + B^T \Delta x_p \ .$$

Since $(\Delta x_h)_k \in \mathcal{N}(B^T)$ and Δx_p is a particular solution of the equality constraint then

$$B^T \Delta x_k = b_2$$

Now, from equation (25), we have

$$Z\Delta x_k + X\Delta z_k = b_3$$

□

REMARK 4.2. *From this proposition, we conclude that the residual error for solving the linear system (2) depends only on the residual error, r_k, for solving the projected system (20). Therefore, we define the vector $(r_k, 0)$, $0 \in \mathbb{R}^{mxn}$, as the residual vector for the original nonsymmetric and indefinite system (2). Consequently, if we can control the tolerance of the projected system (20), we can control the tolerance of the entire system (2).*

THEOREM 4.4. *Consider $\mu > 0$. Let $v = (x, y, x)$ be an interior-point. Then the search direction $(\Delta x_k, \Delta z_k)$ given by (25) and (27) is a descent direction for the penalty term $\nabla\Phi_\mu$ and it is also a descent direction for the modified augmented Lagrangian function M_μ, for sufficiently large ρ, at v. i.e.,*

$$\nabla_x \Phi_\mu(\tilde{v})^T \Delta x + \nabla_z \Phi_\mu(\tilde{v})^T \Delta z < 0$$

and

$$\nabla_x M_\mu(x, y, z; \rho)^T \Delta x + \nabla_z M_\mu(x, y, z; \rho)^T \Delta z < 0,$$

for ρ sufficiently large.

Proof. Theorems 4.1, 4.2 given in [3] and previous theorem establishes the theorem. □

A couple of observations are in order.

REMARK 4.3. *The latter theorem means that any single iteration $(\Delta x_k, \Delta y_k, \Delta z_k)$, defined by (25), (26) and (27), is sufficient to march towards the solution of the problem using the modified augmented Lagrangian function M_μ as a merit function. Therefore, it should be made clear that no preconditioner is needed except near the solution. Numerical experiments show that, for a large fraction of the total number of nonlinear iterations required for converging to the solution of the problem, single iterations are enough to obtain a good step. The remaining few iterations will no doubt require an accurate iterative solution of the projected system (20). This is not seen as a surprise since once the iterates are inside the region of quadratic convergence of Newton's method the merit*

function does not play an important role, and therefore single iterations are not enough to obtain a good step. It is only at this stage of the procedure that, we will require an accurate iterative quasi-Newton step. It is also at this stage that preconditioning becomes important. Some investigation on this issues is given below.

REMARK 4.4. *In the event that $A + X^{-1}Z$ is not positive semidefinite on $\mathcal{N}(B^T)$, the last two propositions hold for any other Krylov subspace method.*

4.6. Preconditioning . In order to obtain reasonable convergence rates, preconditioning the projected system (20) is very important for large-scale applications. In the next section, it will become apparent that full iterations on the projected system are not needed except near the solution. Nevertheless, for overall algorithmic efficiency, one is still interested in the fast convergence of the Newton step near the solution generated by the iterative method. It can be shown that if M is a good preconditioner for $\bar{A}$, then PMP is a good preconditioner for $P\bar{A}P$. We can characterize the quality of $(PMP)^\dagger$ in terms of how close the preconditioner M is to $\bar{A}$. If we consider the splitting $P\bar{A}P = P(M - N)P = PMP - PNP$, then we obtain the following result.

THEOREM 4.5. *Let $\|PNP\| \left\|(P\bar{A}P)^\dagger\right\| = \gamma < 1$, then*

$$\begin{aligned} \left\|(P\bar{A}P)^\dagger - (PMP)^\dagger\right\| &\leq \frac{1+\sqrt{5}}{2} \times \frac{\gamma}{1-\gamma} \left\|(P\bar{A}P)^\dagger\right\| \\ &\leq \frac{1+\sqrt{5}}{2} \times \frac{\|N\| \left\|(P\bar{A}P)^\dagger\right\|^2}{1 - \|N\| \left\|(P\bar{A}P)^\dagger\right\|}. \end{aligned} \tag{28}$$

Proof. The first inequality can be obtained as a particular case of Theorem 8.24 in Lawson and Hanson [12, page 46]. The second inequality follows trivially from the fact that the orthogonal projector P does not increase the norm of a matrix, i.e., $\|PNP\| \leq \|N\|$. □

Note that

$$(P\bar{A}P)^\dagger = \bar{A}^{-1} - A^{-1}B(B^T\bar{A}^{-1}B)^{-1}B^T \; bar A^{-1},$$

which is exactly the upper left block in the inverse of

$$\begin{pmatrix} \bar{A} & B \\ B^T & 0 \end{pmatrix}. \tag{29}$$

It is a remarkable result that the pseudo-inverse of the coefficient matrix of the projected system appears naturally in the computation of the inverse coefficient matrix of the saddle-point problem. Hence, it is clear now why the projection methods are so well suited for our application, they support the choice of preconditioners for the projected system based solely on the properties of the operator $\bar{A}$. In Theorem 4.5, the Golden Mean $\frac{1+\sqrt{5}}{2} \approx 1.618$ is replaced by 1 when the operators involved are nonsingular, as is suggested by Golub and Van Loan [10, Theorem 2.3.4]. The theorem shows that a reduction in $\|N\|$ improves the quality of the preconditioner for the projected system

$P\bar{A}P$. This also implies a quality improvement of M as a preconditioner for $\bar{A}$, and therefore the properties of $\bar{A}$ by themselves determine the choice of preconditioners for the projected system.

5. Numerical experiments. We consider two types of numerical experiments. The first is designed to test the ideas of inexact global minimization that we have introduced in the last two sections. We show from the numerical point of view that one iteration on the linear system (2) is required at each step of the nonlinear minimization process until the iterates fall in the region of convergence of the Newton's method. Then it is necessary to consider a switch in order to ask for more precision of the iterative linear solver for a better performance of the orthogonal projection method. It is in this step that a good preconditioner is needed. In this context, we present a comparison of performances of the orthogonal projection method, GMRES acting on the whole linear system given by equation (2) and SQMR acting on a system with coefficient matrix given by (29). Only simple (i.e., block or incomplete Cholesky) preconditioners are tested and the results are very illustrative.

5.1. Experiments on the global minimization algorithm. The first model problem is the minimization of a quadratic objective function subject to linear equality constraints and to nonnegativety constraints on the primal variables, i.e.,

$$\begin{aligned} &\min \left(\frac{1}{2}x^T\mathbf{A}x - \mathbf{c}^T x\right) \\ &\text{subject to } \mathbf{B}x - \mathbf{b} = 0, \\ &\qquad\qquad x \geq 0. \end{aligned} \tag{30}$$

The full rank matrix $\mathbf{B}$ and vector $\mathbf{b}$ in the equality constraints and the vector $\mathbf{c}$ in the objective function were chosen randomly. The matrix in the quadratic part of the objective function, $\mathbf{A}$, is also chosen randomly but so it is positive definite in $\mathcal{N}(B^T)$. It should be noted that the Hessian matrix, the matrix of linearized constraints and the orthogonal projector $P = I - BB^{\dagger}$ are constant in quadratic programming problems. The only blocks that change throughout the minimization process are those (diagonal) blocks corresponding to the entries of the primal and dual variable vectors.

Problems of this type put the projection method immediately at an advantage in that the potentially costly step of computing the projection operator is done only once. However, the projector does not need to be computed explicitly but can be applied to a vector by working with the current form of the linearized equality constraints, not only for quadratic programming but also for general nonlinear programming problems.

In the first experiment, the Hessian matrix $\mathbf{A}$ is order 50 and the number of equality constraints is 10. This makes the coefficient matrix of (2) order 110. Figure 2 shows the norm of the KKT conditions (labeled the nonlinear residuals) as a function of the number of nonlinear steps. Four methods of solving the linear systems are compared in this figure: Newton's method, the projection method taking one iteration per nonlinear step, the projection method with a dynamically adjusted linear tolerance and

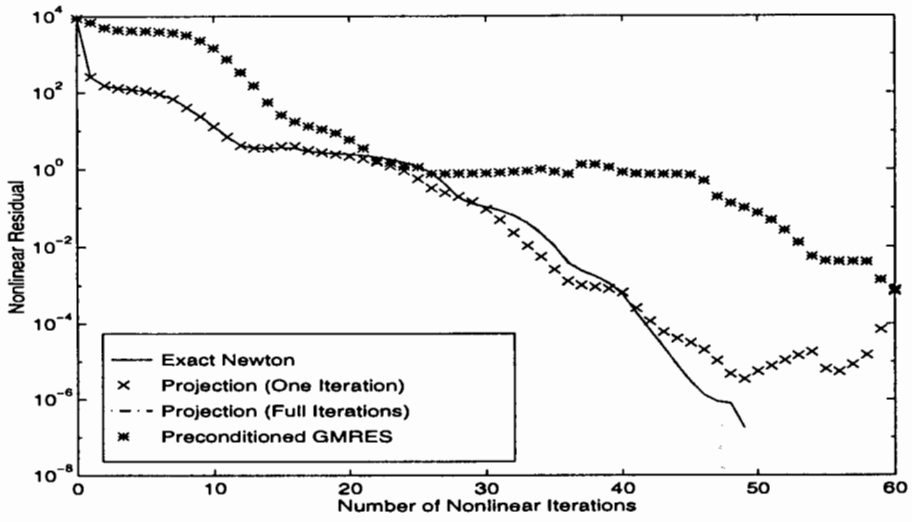

FIG. 2. *Convergence of quadratic programming problem for exact Newton, orthogonal projection taking one iteration of CG on projected system throughout, orthogonal projection with dynamically adjusted linear tolerance and preconditioned GMRES.*

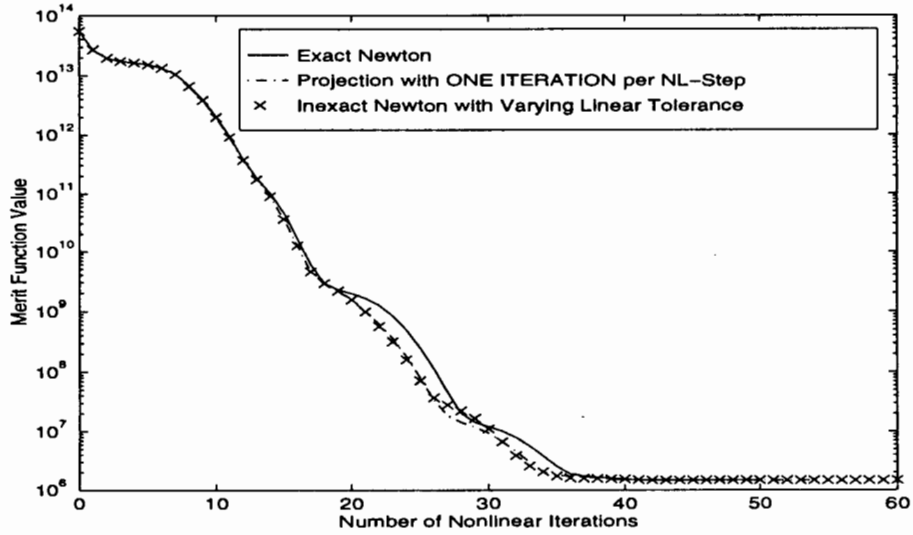

FIG. 3. *Merit function values for the quadratic programming problem for exact Newton, orthogonal projection taking one iteration of CG on projected system throughout and orthogonal projection with dynamically adjusted linear tolerance.*

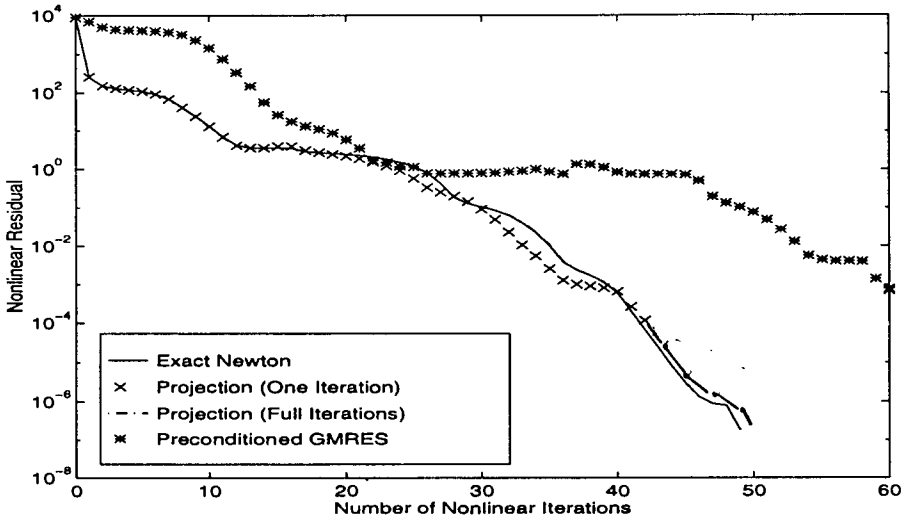

FIG. 4. *Convergence of quadratic programming problem for exact Newton and orthogonal projection taking one iteration of CG on projected system initially (x), and with dynamically adjusted linear tolerance near convergence to the optimum (·−).*

preconditioned GMRES acting on the entire linear system (2) with a linear tolerance two orders of magnitude smaller than that imposed on the KKT conditions (i.e., 10^{-7}). This is an example of modest size and complexity but the results are enlightening. The exact Newton method takes 49 iterations to find the optimum. The final value of the perturbation parameter at the solution is of order 10^{-8}. The curve given by the projection method with full iterations reproduces the convergence path of the exact Newton method near the convergence to the optimum. The linear tolerance was set according to

$$\text{tol} = \min\left(\frac{1}{k+2}, \left[\frac{|f^{(k)}|}{|f^{(0)}|}\right]^{1.5}\right), \tag{31}$$

where, k denotes the nonlinear iteration number and f denotes the objective function. It should be mentioned that the linear tolerance was $\mathcal{O}(10^{-10})$, in spite of this, the projection method never exceeded 12 iterations per solve even though no preconditioner was used for the conjugate gradient method.

It is worth noting that GMRES does not do a good job of approximating the Newton steps or of finding a descent direction for the merit function M_μ, even though it is using a *tailor-made* preconditioner given by

$$M = \begin{pmatrix} A & I_B & -I \\ I_B^T & I & 0 \\ Z & 0 & X \end{pmatrix},$$

where I_B is such that its columns are the first m canonical vectors $\mathbf{e}_i \in \mathbb{R}^n$, $i = 1, \ldots, m$, i.e.,

$$I_B = \begin{pmatrix} I_m \\ 0 \end{pmatrix},$$

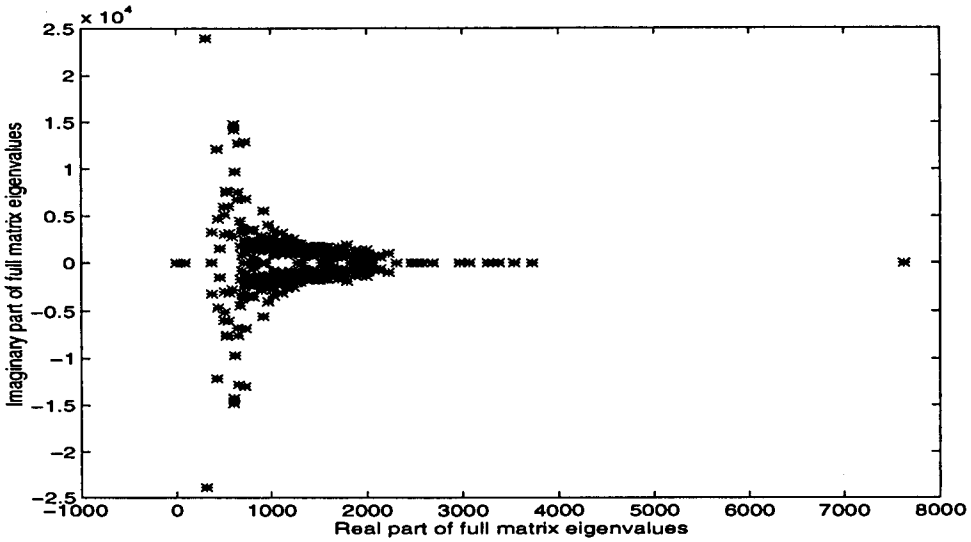

FIG. 5. *Eigenvalue distribution of coefficient matrix for interior point method formulation of the constrained quadratic minimization problem after* 20 *nonlinear iterations.*

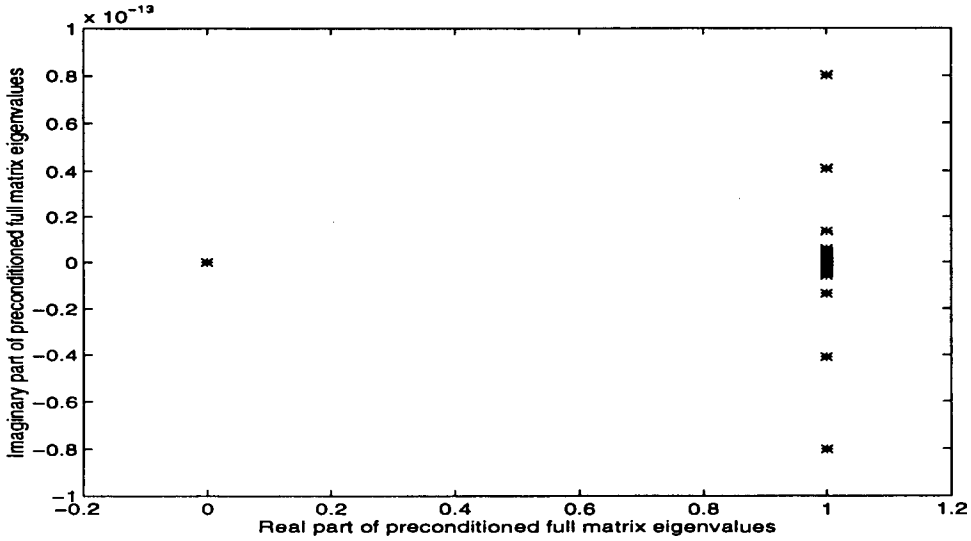

FIG. 6. *Eigenvalue distribution of preconditioned coefficient matrix for interior point method formulation of the constrained quadratic minimization problem after* 20 *nonlinear iterations.*

where I_m is the identity operator of order m. This preconditioner is extremely rich, i.e., the only differences between the preconditioner and the original coefficient matrix are in the absence of the exact blocks B and B^T and in the (necessary) nonzero block inserted in the main diagonal. Note the dramatic clustering of the eigenvalues under application of M^{-1} by comparison of figures 5 and 6. This situation is not realizable in practice, however, but the rationale for its use here is that one can hope to have a reasonable preconditioner for the Hessian block (given some problem structure), the remaining diagonal blocks of the coefficient matrix are easy to handle but formulation of effective preconditioner block corresponding to B and B^T may not be so obvious. In summary, our choice of M gives a *best-case* scenario in which to illustrate the poor performance of preconditioned GMRES for these problems.

The remaining curve on this graph corresponds to the projection method taking *one* iteration per linear system. For a little over 40 iterations this extremely cheap way of finding descent directions for the given merit function is also extremely effective in decreasing the nonlinear residual toward the optimum.

To see intuitively what it is going on, we turn the reader's attention to Figure 3, where the values of the merit function, given by equation (10), are plotted versus the nonlinear iteration count for the exact Newton and the projection method using full linear iterations or one iteration throughout. It is apparent that the merit function is no longer decreasing appreciably near the region of convergence to the optimum. This behavior separates, at least qualitatively, the region in which the merit function is driving the global convergence (this requires only one iteration of the projection method on each linear system) from that in which the merit function is no longer effective but the nonlinear iterates have presumably fallen into the region of quadratic convergence of Newton's method. This behavior is in agreement with the objective of a globalization technique where the merit function together with a descent direction is proposed to help the minimization process to carry out the steps inside of the convergence region of Newton's method. When the iterates fall into this region, it is necessary to make a switch and ask for a better approximation to the Newton step, in order to retain the faster rate of convergence of Newton's method.

Based on this observation, the objective now is to propose a criteria that allows us to determine in what moment it is necessary to make a switch from a single linear iteration to full iterations. The control of the behavior of the linear solver at different stages of the nonlinear minimization process is given by,

1. $P_{switch} = \frac{|\nabla M_\mu(\tilde{v})^T \Delta(\tilde{v})|}{M_\mu(\tilde{v}}$
2. IF ($P_{switch} > \epsilon$)
   2.1 MAX LINEAR ITERATIONS = 1
3. ELSE
   3.1 LINEAR TOLERANCE AS GIVEN BY (31)
4. END,

where ϵ hopefully is given by a safe choice valid for a large range of problems. The choice of the numerator of the switching parameter P_{switch} is a natural one, since it measures

by how much the inexact (or exact, for that matter) nonlinear step produces a descent direction for the merit function M_μ. This value is further normalized by the currently assumed value of M_μ. In our experimentation, $\epsilon = \mathcal{O}(10^{-2})$ was found acceptably safe.

An alternative idea to determine the switching point between single linear iterations and full iterations is to follow the values of the perturbation parameter μ. If the neighborhood of the quasi-central path is hit repeatedly throughout the minimization process, the frequent reductions in μ can give an indication of how close one is to the region of Newton's quadratic convergence. However, in many cases the neighborhood of the quasi-central path is hit only a few times near convergence and this produces too few instances of μ-reduction to make this a reliable scheme.

Finally in Figure 4, we show the behavior between Newton's method and the projection method taking one iteration per nonlinear step until the iterates fall into the region of quadratic convergence; then inside of this region, we dynamically adjusted the linear tolerance given by (31) in order to obtain a better approximation to the Newton step and therefore retain a fast rate of convergence. From the numerical results obtained, we conclude that our method is viable.

In general nonlinear programming problems, the main point of concern of the approach proposed here is the recomputation of the projection. As was mentioned above, the application of the projector P to a vector amounts to computing the Cholesky decomposition of B^TB, which requires $\mathcal{O}(m^3)$ floating point operations (recall m is the number of equality constraints). On the other hand, the final termination property of Krylov subspace iterative methods guarantees that a solution to the projected system can be obtained in $\mathcal{O}(n-m)$ floating point operations in exact arithmetic (recall that the projected system has $n-m$ nonzero eigenvalues). Therefore, at instances of the algorithm when the projected system must be resolved accurately, as the number of equality constraints, m, grows closer to the dimension of the vector of variables $x \in \mathbb{R}^n$, the cost of applying the projector grows as m^3 and the cost of solving the projected system decreases linearly as m approaches n. This suggests that one should schedule updates to the projection operator at intervals longer than after every nonlinear iteration. Since this is a crucial issue in order to promote the proposed inexact method for application in general nonlinear programming problems, a study of it and related implementation problems is proposed for future research.

6. Conclusions and further research. An inexact method based on orthogonal projections on $\mathcal{N}(B^T)$ and iterative solution of the generated linear systems by Krylov subspace iterative methods is presented. This method produces extremely cheap nonlinear iterations by enforcing linear feasibility to generate a descent direction for the adopted merit function M_μ with one solver iteration on the projected systems for all instances of the global minimization algorithm for which ∇M_μ is steep. Near convergence to the global optimum, the linear systems have to be resolved progressively more accurately and a criterion for dynamically selecting linear tolerances is given here, as well as a criterion for when to stop relying on single iterations of the iterative linear solver. The issue of preconditioning is also addressed. In the proposed algorithm, only a preconditioner based on the algebraic properties of $\nabla_x^2 \ell + X^{-1}Z$ is needed. We believe

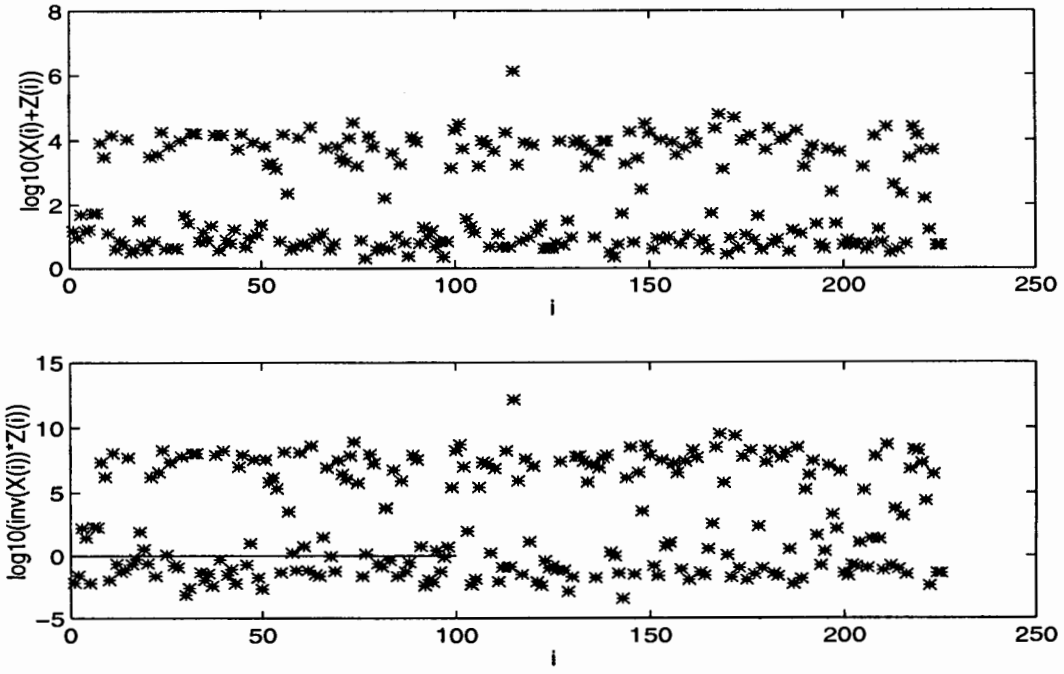

FIG. 7. *Entry distribution of the diagonal matrices X and Z.*

that this is a tractable problem that can be solved satisfactorily for a large class of general nonlinear programming problems. A study of preconditioners for the inexact method introduced in this work as well as for some other inexact method proposed in recent literature is given, which suggests that the preconditioned projection method performs well when compared to its immediate competitors.

Further research on repeated updates of the the projection operator P for general nonlinear programming problems as well as on quasi-Newton ideas incorporated to this algorithm will be given in a subsequent paper.

Acknowledgments. The authors thank Leticia Velázquez for reading an earlier draft of this paper and suggesting many improvements in wording and style. They also acknowledge the Rice interior–point optimization group for their collective comments and suggestions.

REFERENCES

[1] A. S. El-Bakry, R. A. Tapia, T. Tsuchiya and Y. Zhang, *On the formulation and theory of the Newton interior-point method for nonlinear programming*, J. of Optim. Theory and Appl., 89(1996), pp. 507–541.

[2] K. M. Anstreicher and J. P. Vial, *On the convergence of an infeasible primal-dual interior-point method for convex programming*, Optimization Methods and Software, 3(1994), pp. 273-283.

[3] M. Argáez, and R. A. Tapia, *On the global convergence of a modified augmented Lagrangian linesearch interior-point Newton method for nonlinear programming*, Tech. Rep. TR95-38, Rice University, Dept. of Computational and Applied Mathematics, 1995.

[4] R. Barrett and M. Berry and T. Chan and J. Demmel and J. Dunato and J. Dongarra and V. Eikkhout and R. Pozo and C. Romine and H. van der Vorst, *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*, SIAM, Philadelphia, 1994,

[5] R. Bramley, *An orthogonal projection algorithm for linear Stokes problems*, Center of Supercomputing Research and Development, University of Illinois, Urbana-Champaign, CSRD-1190, 1992.

[6] P.N. Brown and Y. Saad, *Hybrid Krylov methods for nonlinear systems of equations*, SIAM J. Sci. Statist. Comput., 11(1990), pp. 450–481.

[7] P. N. Brown and Y. Saad, *Convergence theory of nonlinear Netwon-Krylov algorithms*, SIAM J. Optim., 4(1994), pp. 297–330.

[8] R.S. Dembo and S.C. Eisenstat and T. Steihaug, *Inexact Newton methods*, SIAM J. Numer. Anal., 19(1982), pp. 400–408.

[9] R.W. Freund and F. Jarre, *A QMR-based interior-point algorithm for solving linear programs*, AT&T Numerical Analysis Manuscript, Bell Laboratories, Murray Hill, NJ, 1995.

[10] G.H. Golub and C.F. Van Loan, *Matrix Computations*, John Hopkins University Press, 1989.

[11] M. R. Hestenes and E. Stiefel, *Methods of conjugate gradients for solving linear systems*, J. Res. Nat. Bur. Standards, 49(1952), pp. 409–436.

[12] C.L. Lawson and R.J. Hanson, *Solving Least Squares Problems*, Classics in Applied Mathematics, SIAM, Philadelphia, 1995.

[13] S. Nash, *Preconditioning reduced matrices*, SIAM J. Matrix Anal. Appl., 17(1996), pp. 47–68.

[14] S. Nash and A. Sofer, *Linear and Nonlinear Programming*, McGraw Hill, 1996.

[15] J. Nocedal and M. L. Overton, *Projected Hessian updating algorithms for nonlinearly constrained optimization*, SIAM J. Numer. Anal., 22(1985), pp. 821–850.

[16] H. Yamashita, *A globally convergent primal-dual interior point method for constrained optimization*, Mathematical Systems Institute, Inc., Tokyo, Japan, 1992.