UMI

A Bell & Howell Information Company
300 North Zeeb Road, Ann Arbor MI 48106-1346 USA
313/761-4700 800/521-0600

RICE UNIVERSITY

Exact and Inexact Newton Linesearch Interior–Point Algorithms for Nonlinear Programming Problems

by

Miguel Argáez Ramos

A Thesis Submitted
in Partial Fulfillment of the
Requirements for the Degree

Doctor of Philosophy

Approved, Thesis Committee:

Richard a Tapia

Richard A. Tapia, Chairman
Noah Harding Professor of Computational
and Applied Mathematics

Steve Cox
Associate Professor of Computational and
Applied Mathematics

George N. Phillips Jr.
Professor of Biochemistry and Cell Biology

Houston, Texas

May, 1997

UMI
300 North Zeeb Road
Ann Arbor, MI 48103

Abstract

Exact and Inexact Newton Linesearch Interior–Point Algorithms for Nonlinear Programming Problems

by

Miguel Argáez Ramos

In the first part of this research we consider a linesearch globalization of the local primal-dual interior-point Newton's method for nonlinear programming recently introduced by El-Bakry, Tapia, Tsuchiya, and Zhang. Our linesearch uses a new merit function that is a generalization of the standard augmented Lagrangian function and a new notion of centrality. We establish a global convergence theory and present rather promising numerical experimentation.

In the second part, we consider an inexact Newton's method implementation of the linesearch globalization algorithm given in the first part. This inexact method is designed to solve large scale nonlinear programming 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 projection method maintains linearized feasibility with respect to both the equality constraints and complementarity condition. 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 nonlinear programming problem plus a positive diagonal matrix. We report numerical experimentation for several large scale problems to illustrate the viability of the method.

Acknowledgments

I am grateful to my adviser, professor Richard Tapia. Professor Tapia give me a great opportunity to study at Rice and has been a fountain of valuables comments, suggestions through this research. I also thank him for many enriching conversations about research, teaching and life, and for extending his warmth and friendship to me and my family.

I would like to thank the other members of my committee, professors Steve Cox, and George N. Phillips Jr. for their interest in this research.

I dedicate this thesis to my wife Esperanza, my daughters Paola Andrea, and Diana Carolina for their love, patience, and understanding. I also dedicate this thesis to my mother Doña Lida. Ella me enseño desde muy pequeño el valor de una buena educación.

I would to thank all my friends at our department for their support and kindness. Specially, I enjoyed many fructiferal conversations with Amer-Elbakry, Zeferino Parada, Aurelio Oliveira and specially my close friend Leticia Velazquez about this research. I am also grateful to Hector Klie and Marcelo Rame for their encouragement that I received during the second part of this dissertation. Their interest and extremely valuable comments filled numerous gaps in my knowledge in iterative methods. Many thanks to Linda, Fran, and Daria who always provided valuable and opportune help.

Finally, I would like to make a specially acknowledge professor Alonso Takahaschi. I remember some great conversations with him at Cali, our homeland, that allow me grow and mature mathematically.

This research was supported by Universidad del Valle, Departamento de Matemáticas, Cali, Colombia, South America; the Center for Research on parallel Computation, Rice University- NSF Cooperative Agreement No. CCR-9120008; and DOE Grant DEFG05-86-ER25017.

Contents

List of Abbreviations

KKT	Karush-Kuhn-Tucker
NLP	Nonlinear Programming Problems
IPM	Interior Point Methods
LP	Linear Programming
$\mathcal{N}(B)$	Nullspace of B
$\mathcal{N}(B^T)$	Nullspace of B^T
$B^\dagger$	Moore-Penrose pseudo-inverse of B
SQMR	Symmetric Quasi-Minimal Residual
MINRES	Minimal Residual
GMRES	Generalized Minimal Residual

Chapter 1

Introduction

1.1 Historical background

The recent computational success of primal-dual interior-point methods for linear programming (for example Lustig, Marsten, and Shanno [31]) has motivated considerable research activity in the generally more difficult area of nonlinear programming. It is natural that this extension activity initially focused on quadratic programming and convex programming. Our concern here is the area of nonconvex programming, and in particular globally convergent formulations. Some papers in these areas include El-Bakry, Tapia, Tsuchiya, and Zhang [12], McCormick [18], Lasdon, Yu, and Plummer [29], Monteiro, Pang, and Wang [37], Nash and Sofer [38], Wright [50], Yamashita [53], R.H. Byrd, J.C. Gilbert, and J. Nocedal [6] and Forsgren and Gill [54]. We also acknowledge the conference presentation, Gay, Overton, and Wright [24].

The current work was influenced significantly by El-Bakry et al [12]. These authors extended the interior-point formulation from linear programming to general nonlinear programming, and then validated this extension by establishing local and Q-quadratic convergence for the local form of their Newton primal-dual interior-point method under no more assumptions than the standard Newton's method assumptions. In addition they considered a linesearch globalization strategy for their local algorithm that uses the l_2-norm of the residual function of the KKT conditions as merit function. While this choice for the merit function has obvious advantages, it also has obvious disadvantages. Hence, the primary objective of the present work is to construct a merit function which compares favorably with the choice of the l_2-norm of the KKT residual. We believe that we have succeeded, and in Chapter 7 offer numerical experimentation to reinforce this view.

The choice of merit function described in the current work was influenced by the merit function suggested by Anstreicher and Vial [1] for their convex programming application. Our philosophy of globalization was influenced by the approach taken by Gonzalez-Lima, Tapia, and Potra [27] in their application of designing an interior-point algorithm that would effectively calculate the analytic center of the solution set in linear programming. An elaboration of this philosophical point will help significantly in both motivating and understanding the inner-loop component of our global algorithm described in Chapter 5. The following comments are related closely to the material and interpretation contained in Sections 2 and 3 of El-Bakry et al [12]. For the sake of illustration it suffices to restrict our discussion to linear programming; however we do feel that these comments are particularly appropriate for nonconvex programming.

It is well-known that the logarithmic barrier function formulation promotes excellent global behavior at the expense of (theoretically) badly conditioned subproblems locally. Moreover, it is not difficult to see that the first-order condition for the logarithmic barrier function subproblem and the perturbed KKT conditions for the nonlinear program are equivalent. However, Proposition 2.3 of El-Bakry et al [12] demonstrates that the Newton iterates obtained from the equivalent subproblem (perturbed KKT conditions and the first-order conditions for the logarithmic barrier function subproblem) never coincide. This is indeed surprising since many authors tacitly assumed that they were the same. It follows that if we continue taking Newton steps (with an effective globalization strategy) on the perturbed KKT conditions with fixed perturbation parameter we will arrive at the solution of the logarithmic barrier subproblem. The critical issue here is that we will have obtained this solution via not necessarily badly conditioned linear systems (Newton defining relations). Hence we have a way of implementing the logarithmic barrier function method and circumventing the inherent bad conditioning. Far from the solution of the nonlinear program one may need to solve the logarithmic barrier function more accurately than would be needed near the solution. This can be accomplished by taking a variable number of Newton iterations on the perturbed KKT conditions with the perturbation parameter held fixed. Moreover, a linesearch globalization strategy may also be added to the Newton iteration procedure that attempts to solve the perturbed KKT conditions with fixed perturbation parameter. This philosophical approach is what was proposed in Gonzalez-Lima, Tapia, and Potra [27] in an effort to effectively calculate the analytic center of the solution set. Moreover, it is exactly this philosophy that we carry over into our global algorithm described in Chapter 5. A linesearch strategy with our merit function is used to reach an approximate solution of a new notion of centrality that is introduced in Chapter 4. We call this procedure the inner loop, and it can be viewed as an effective way of approximating the solution of the logarithmic barrier function subproblem. We point out that the El-Bakry et al [12] globalization strategy did not provide for the feature of performing an appropriate number of Newton iterates with the perturbation (barrier) parameter held fixed.

In the second part of this dissertation we are concerned with an inexact Newton's method for solving large scale nonlinear programming problems. A Newton's method applied to the nonlinear programming problem 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.,

$$\begin{pmatrix} A & B & -I \\ B^T & 0 & 0 \\ Z & 0 & X \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta y \\ \Delta z \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix}, \tag{1.1}$$

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 $Z, X \in \mathbb{R}^{n \times n}$ are positive diagonal matrices. The second and third equation are the linear equality constraint and linear complementarity condition, respectively. They will play an important role of the second part of this dissertation.

The use of primal-dual interior point methods for constrained minimization is relatively new (see [12], [24], [54], [6]). Recently, interest in large-scale applications of interior-point methodology motivated 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 a major challenge. Efforts in this direction are still sparse in the literature and a variety of alternatives have been examined in the area of inexact Newton solution of linear programming problems (see [25], [48],[52]), as well as of nonlinear problems (see [34], [6], [24]). 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 [38]. Most of these advances have relied on the use of Krylov subspace methods [45], [44]. Of particular interest for the optimizer is the formulation of robust iterative solution techniques for saddle-point problems, i.e., linear systems of the form

$$\begin{pmatrix} A & B \\ B^T & 0 \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} f \\ g \end{pmatrix}. \tag{1.2}$$

Interior-point method formulations can be cast in the form (1.2) by explicit elimination of the linear complementarity condition in (1.1). Consequently, design of robust inexact algorithms for (1.1) can rely on results from the formulation of saddle-point iterative solvers [20]. Other approaches have attempted to solve (1.1) 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 (1.2), 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 (1.2) 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 solution of the Schur complement, S of (1.2), i.e., $S = -B^TA^{-1}B$. 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 be replaced by general nonsymmetric solvers which potentially increase the computational cost.

Other optimization linear solvers of the reduction type are those restricting the search direction in (1.2) to the null space of B^T, denoted by $\mathcal{N}(\mathcal{B}^T)$. Most versions rely on finding a null space basis matrix for B^T. Other methods (this idea constitutes the core of the second part of this work) are based on the restriction of A to $\mathcal{N}(\mathcal{B}^T)$ as defined by an orthogonal projector $P = I - BB^\dagger$ onto $\mathcal{N}(\mathcal{B}^T)$ with $B^\dagger = (B^TB)^{-1}B^T$ if B is of full rank.
In this second part, we adopt and analize the orthogonal projection method proposed by Bramley [5] to solve problems of the form (1.2). We conclude that the orthogonal projection method exploited in an intelligent way offers the best compromise between effectiveness and robustness. We assert that this method removes the major difficulties introduced by the linear complementarity condition in the system (1.1).

One fundamental purpose of this second part is to combine the linear iterative solver obtained from the orthogonal projection method for (1.1) with the global nonlinear optimization algorithm presented in the first part in an efficient manner. Specifically, we prove that each iteration of the projection method applied to the linear system (1.1) produces a descent direction for our generalized augmented Lagrangian merit function introduced in the first part. 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 the 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.

1.2 Outline

This work is organized into two parts. The first part starts with Chapter 2 where we define the nonlinear program in a particular form and state its first-order necessary conditions. In Chapter 3, we present our generalized augmented Lagrangian function and establish several important properties that this function posseses. Next, in Chapter 4, we demonstrate that the Newton direction under consideration is a descent direction for our generalized augmented Lagrangian function, provided the penalty constant is chosen sufficiently large. The global algorithm is described in Chapter 5 and global convergence theory is established in Chapter 6. Preliminary numerical experimentation is the subject of Chapter 7.

The second part of this work is presented in Chapter 8 through Chapter 10. In Chapter 8, we focus our attention on the technical details supporting the orthogonal projection method. We stress the advantages of this orthogonal projection method and its relation with other methods. Chapter 9, is devoted to describing the algorithm and possible strategies for preconditioning the resulting projected linear system. In Chapter 10, we present numerical experiments that test the inexact global minimization method that we have introduced in the previous two sections. Concluding remarks are made in Chapter 11.

1.3 Notation

As is standard in the study of interior-point methods, given a vector $x = (x_1, \ldots, x_n)^T$ we write $x > 0$ to mean $x_i > 0$ for all i, X to denote the diagonal matrix, diag(x), and e to denote the vector whose components are all equal to one. We may also write x^{-1} to denote the vector whose i-th component is x_i^{-1}. Observe that we can also write $x^{-1} = X^{-1}e$.

Part I

Chapter 2

Formulation of the problem

2.1 Statement of the problem and first order KKT conditions

We study the general nonlinear program in the form

$$\begin{aligned} \text{minimize} \quad & f(x) \\ \text{subject to} \quad & h(x) = 0 \\ & x \geq 0, \end{aligned} \tag{2.1}$$

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

The Lagrangian function associated with problem (2.1) is

$$\ell(x, y, z) = f(x) + h(x)^T y - x^T z \tag{2.2}$$

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 this problem are

$$F(x, y, z) \equiv \begin{pmatrix} \nabla f(x) + \nabla h(x) y - z \\ h(x) \\ XZe \end{pmatrix} = 0, \qquad (x, z) \geq 0 \tag{2.3}$$

where $X = \text{diag}(x), Z = \text{diag}(z)$ and $e = (1, \ldots, 1)^T \in \mathbb{R}^n$. For a feasible point x of problem (2.1), we let $\mathcal{B}(x) = \{j : x_j = 0\}$. Clearly $\mathcal{B}(x)$ is the set of indices of active or binding inequality constraints. The set of gradients of active constraints is the set $\{e_j \in \mathbb{R}^n : j \in \mathcal{B}(x)\}$.

2.2 Standard assumptions

In the study of Newton's method, the standard assumptions for problem (2.1) are :

A2.1: (Existence) There exists x^* a solution to problem (2.1).

A2.2: (Smoothness) The Hessian operators $\nabla^2 f$, $\nabla^2 h_i$, $i = 1, \ldots, m$ are Lipschitz continuous in a neighborhood of x^*.

A2.3: (Regularity) The set $\{\nabla h_1(x^*), \ldots, \nabla h_m(x^*)\} \cup \{e_j : j \in \mathcal{B}(x^*)\}$ is linearly independent.

A2.4: (Second-order sufficiency) 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

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

A2.5: (Strict complementarity) For all j, $x_j^* + z_j^* > 0$.

The following interesting relationship between conditions A2.4-A2.5 and the invertibility of the Jacobian matrix can be found in Section 4 of El-Bakry et al [12].

Proposition 2.2.1 Let conditions A2.1 and A2.2 hold. Then the following statements are equivalent:

1. Conditions A2.3-A2.5 also hold.
2. The Jacobian matrix $F'(x^*, y^*, z^*)$ is nonsingular.

Proof. See El-Bakry et al [12].

2.3 Perturbed KKT conditions

We now motivate the perturbed KKT conditions. The complementarity conditions for problem (2.1) are

$$XZe = 0.$$

The linearized form of complementarity has a serious flaw. In a Newton's method formulation we will deal with the linearized complementary equation

$$Z\Delta x + X\Delta z = -XZe. \tag{2.4}$$

Now, if a component $x_j = 0$ and its corresponding Lagrange multiplier $z_j \neq 0$, then from (2.4)

$$\Delta x_j = 0.$$

Therefore the subsequent iterate is

$$(x_j)_+ = x_j + \alpha\Delta x_j = 0 \quad for\ \ any\ \ \alpha \in \mathbb{R}.$$

This means, if a component x_j becomes zero, with its corresponding Lagrange multiplier $z_j \neq 0$, then this component will remain zero in all future iterations. The analogous situation is true for the z variable. So, Newton's method forces iterates to stick to the boundary of the feasible region once they approach that boundary. Such an undesirable attribute clearly precludes the global convergence of the algorithm. This difficulty is overcome by perturbing the complementarity condition to obtain

$$XZe = \mu e, \quad \mu > 0.$$

Under the same situation studied above, we can prove

$$\Delta x_j \neq 0.$$

Hence this modification tends to keep the iterates away from the boundaries.

Therefore, the perturbed KKT conditions are

$$F_\mu(x,y,z) \equiv \begin{pmatrix} \nabla f(x) + \nabla h(x)y - z \\ h(x) \\ XZe - \mu e \end{pmatrix} = 0, \qquad \mu > 0 \tag{2.5}$$
$$(x,z) > 0$$

where $X = \text{diag}(x), Z = \text{diag}(z)$ and $e = (1,\ldots,1)^T \in \mathbb{R}^n$.

We will prove in the next section that the perturbed KKT conditions besides keeping the iterates away from the boundaries also promote the global convergence of the Newton interior-point method with no necessarily badly conditioned systems of equations. Therefore, we will base our global convergence technique on the perturbed KKT conditions.

2.4 Interpretation of the perturbed KKT conditions

As we have said in the introduction, it is well known that the logarithmic barrier function problem promotes excellent global behavior at the expense of badly conditioned subproblems locally. Our intention here is to demonstrate that the perturbed KKT conditions with a fixed perturbation parameter is a way to implement the logarithmic barrier function problem that circumvents the inherent bad conditioning.

Towards this end, we begin by considering the logarithmic barrier function problem associated with problem (2.1)

$$\text{minimize } f(x) - \mu \sum_{j=1}^{n} log(x_j), \quad x_j > 0 \tag{2.6}$$
$$\text{subject to} \quad h(x) = 0$$

An important property of the logarithmic barrier function is the following.
If $x_\mu > 0$ is a solution of (2.6) for $\mu > 0$ and if μ is regarded as a continuous parameter, then x_μ defines a smooth trajectory converging to a solution of problem (2.1), x^*, as $\mu \to 0$. The points $\{x_\mu\}$ are said to lie on the barrier trajectory. For proofs and additional details, see

Fiacco and McCormick [18].

The KKT conditions for (2.6) are

$$\hat{F}_\mu(x,y) \equiv \begin{pmatrix} \nabla f(x) + \nabla h(x)y - \mu x^{-1} \\ h(x) \end{pmatrix} \tag{2.7}$$

$$x > 0.$$

It is well known that Hessian matrices of the logarithmic barrier function become increasingly ill-conditioned at points on the barrier trajectory as the solution of problem (2.6) is approached. Also, the barrier Hessian is ill-conditioned in an entire region near the solution. See, Wright [23]. One way of overcoming this flaw is by introducing the auxiliary variables

$$z = \mu X^{-1}e$$

and then expressing these nonlinear defining relations in the form

$$XZe = \mu e.$$

Substituting the latter transformations of variables into the KKT conditions (2.7), we arrive at the perturbed KKT conditions (2.5) for the NLP problem (2.1). This is formalized in the next proposition.

> **Proposition 2.4.1** The KKT conditions for the logarithmic barrier function problem (2.6) given by (2.7), and the perturbed KKT conditions for problem (2.1) given by (2.5) are equivalent in the sense that they have the same solutions, i.e. , $\hat{F}_\mu(x,y) = 0$ if and only if $F_\mu(x,y,\mu x^{-1}) = 0$.

The perturbed KKT conditions (2.5) do not lead to inherent ill-conditioning. The introduction of the auxiliary variables (Lagrange multipliers associated with the inequality constraints) has been used to remove the ill-conditioning from the KKT condition (2.7) for the logarithmic barrier problem. Therefore, it is important to realize that if we take an appropriate number of Newton iterations (with an effective globalization strategy) on the perturbed KKT conditions with a fixed perturbation parameter, we will get a solution of the logarithmic barrier function problem by solving systems of equations that are not necessarily badly conditioned. The price that one has to pay is that instead of solving a system of $n+m$ equations now one has to solve a system of $2n+m$ equations.

Before we finish this discussion it is important to point out the following two facts:

First of all, the perturbed KKT conditions (2.5) for NLP problem (2.1) or any permutation of these equations, are not the KKT conditions for the logarithmic barrier function problem (2.6) or any other unconstrained or equality constrained optimization problem.

Second, the Newton step obtained from the nonlinear system $\hat{F}_\mu(x,y) = 0$ given by (2.7) never coincides with the Newton step obtained from the nonlinear system $F_\mu(x,y,z) = 0$ given by (2.5).

An excellent study of the last two facts is presented in Section 2 of El-Bakry et al [12].

Therefore one of the objectives of this dissertation is to present a strategy of globalization that takes advantages of the perturbed KKT conditions and retains the fast local rate of convergence that El-Bakry et al [12] established using this perturbed KKT conditions as a central framework.

As in any globalization technique the merit function plays a fundamental role, in the next chapter we will present a generalized augmented Lagrangian function associated with the general NLP problems (2.1). This function will be used as a merit function in our globalization framework.

Chapter 3

A new merit function for the general NLP

3.1 A generalized augmented Lagrangian function

One of the objectives of this research is to construct an appropriate merit function that couples the objective function with the constraints in such a way that progress in the merit function effectively means progress in solving problem (2.1). Our strategy is to modify the augmented Lagrangian function associated with the equality constrained optimization problem by adding to its penalty term the potential reduction function utilized in some linear programming applications. This modification leads to a new augmented Lagrangian function that retains stationarity of x_μ^* at any point $v_\mu^* = (x_\mu^*, y_\mu^*, z_\mu^*)$ that satisfies the perturbed KKT conditions (2.5), and adds positive curvature to convert x_μ^* from a stationary point to a minimizer.

In line with the objective stated above, we present the following generalized augmented Lagrangian function.

Definition 3.1.1 For any $\mu > 0$, we define our generalized augmented Lagrangian function by

$$M_\mu : \mathbb{R}^{n+m+n} \to \mathbb{R}$$

$$M_\mu(x, y, z; \rho) = \ell\,(x, y, z) + \rho\,\Phi_\mu(x, z) \qquad (3.1)$$

where $\ell\,(x, y, z)$ is the Lagrangian function associated with problem (2.1), i.e.,

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

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

$$\Phi_\mu(x, z) = \frac{1}{2}\, h(x)^T h(x) + x^T z - \mu \sum_{i=1}^{n} \ln(x_i z_i). \qquad (3.2)$$

By the way as we have formulated the problem the primary variables x , z are positive and therefore the function M_μ is well defined.

3.2 Global properties

Now, we are in a position to prove that our generalized augmented Lagrangian function for problem (2.1) possesses the same properties at a solution of the perturbed KKT conditions (2.5) as does the augmented Lagrangian function for an equality constrained problem at a solution.

Theorem 3.2.1 Consider $\mu > 0$. If $v_\mu^*=(x_\mu^*, y_\mu^*, z_\mu^*)$ satisfies the perturbed KKT conditions (2.5), then x_μ^* is a stationary point of $M_\mu(x, y_\mu^*, z_\mu^*; \rho)$ for any $\rho > 0$.

Proof The gradient of M_μ with respect to the variable x at (x, y_μ^*, z_μ^*) is

$$\nabla_x M_\mu(x, y_\mu^*, z_\mu^*; \rho) = (\nabla f(x) + \nabla h(x)\, y_\mu^* - z_\mu^*) + \rho[\nabla h(x) h(x) + z_\mu^* - \mu x^{-1}].$$

Evaluating the previous equation at x_μ^* we have

$$\nabla_x M_\mu(v_\mu^*; \rho) = 0$$

for any ρ. □

Theorem 3.2.2 Consider $\mu > 0$. If $v_\mu^* = (x_\mu^*, y_\mu^*, z_\mu^*)$ satisfies the perturbed KKT conditions (2.5), then there exists $\bar{\rho} > 0$ such that for $\rho \geq \bar{\rho}$ the Hessian of M_μ with respect to x is positive definite at $v_\mu^* = (x_\mu^*, y_\mu^*, z_\mu^*)$. i.e.,

$$\eta^T \nabla_x^2 M_\mu(v_\mu^*; \rho)\, \eta > 0$$

for any nonzero $\eta \in \mathbb{R}^n$.

Proof The Hessian of M_μ with respect to x evaluated at v_μ^* is

$$\nabla_x^2 M_\mu(v_\mu^*;\rho) = \nabla_x^2 \ell(v_\mu^*) + \rho\,[\nabla h(x_\mu^*)\nabla h(x_\mu^*)^T + \mu(X_\mu^*)^{-2}]$$

Now we can choose a $\bar{\rho} > 0$ such that $\nabla_x^2 \ell(x_\mu^*, y_\mu^*, z_\mu^*) + \bar{\rho} Z_\mu^*(X_\mu^*)^{-1}$ is positive definite. Therefore, for all $\rho \geq \bar{\rho}$ we have $\nabla_x^2 M_\mu(v_\mu^*;\rho)$ is positive definite in $\mathbb{R}^n$. □

The last two statements clearly lead us to the following corollary.

> **Corollary 3.2.1** Consider any $\mu > 0$. If $v_\mu^* = (x_\mu^*, y_\mu^*, z_\mu^*)$ satisfies the perturbed KKT conditions (2.5), then there exists a $\bar{\rho} > 0$ such that
>
> $$x_\mu^* = \mathit{argmin}\ M_\mu(x\,, y_\mu^*\,, z_\mu^*;\rho)$$
>
> for all $\rho \geq \bar{\rho}$.

3.3 An analogy

The purpose of this section is to present an analogy that exists between the augmented Lagrangian function defined for equality constrained optimization problems and the generalized augmented Lagrangian function, that we have presented in Definition 3.1.1, for logarithmic barrier function problems.

First, we establish the result for equality constrained optimization problems.

The equality constrained optimization problem is

$$\begin{aligned} &\text{minimize} \quad f(x) \\ &\text{subject to} \quad h(x) = 0 \end{aligned} \tag{3.3}$$

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

The Lagrangian function associated with problem (3.3) is

$$\ell(x, y, z) = f(x) + h(x)^T y \tag{3.4}$$

where $y \in \mathbb{R}^m$ is the Lagrange multiplier associated with the equality constraints.

The first-order necessary conditions for problem (3.3) are

$$F(x,y) \equiv \begin{pmatrix} \nabla f(x) + \nabla h(x) y \\ h(x) \end{pmatrix}.$$

The augmented Lagrangian function associated with problem (3.3) is

$$L(x,y;\rho) = l(x,y) + \rho\, h(x)^T h(x),$$

where $l(x,y)$ is the Lagrangian function defined by (3.4), and ρ is a penalty parameter that is greater than or equal to zero ($\rho \geq 0$).

This augmented Lagrangian function was suggested by Hestenes (1969) and has been used in different contexts for solving equality constrained problem (3.3). For example, the augmented Lagrangian function has been used as a merit function to globalize Newton's method for solving problem (3.3). See Alem [13].

The augmented Lagrangian function has the following fundamental property. If x^* is a solution of problem (3.3) with y^* its associated multiplier, then

$$L(x,y^*;\rho) = f(x) + h(x)^T y^* + \rho h(x)^T h(x)$$

has a local minimum at x^* for ρ sufficiently large.

That is, there exists a $\bar{\rho} \geq 0$ such that

$$x^* = arg\, min\ L(x,y^*;\rho),$$

for any $\rho \geq \bar{\rho}$.

Now, we show that our generalized augmented Lagrangian function has a local minimun in a solution of the logarithmic barrier function problem for a penalty parameter sufficiently large.

Let x_μ^* be a solution of the logarithmic barrier function problem (2.6), then there exists a Lagrange multiplier y_μ^* associated with the equality constraints such that (x_μ^*, y_μ^*) satisfies the KKT conditions $\hat{F}_\mu(x,y) = 0$ associated with (2.6). By Proposition 2.4.2 there exists a z^*_μ such that $(x^*_\mu, y^*_\mu, z^*_\mu)$ is a solution of the perturbed KKT conditions $F_\mu(x,y,z) = 0$ associated with NLP problem (2.1). Now, from Corollary 3.2.1 the generalized augmented Lagrangian function

$$M_\mu(x,y,z;\rho) = l(x, y^*_\mu, z^*_\mu) + \rho\, \Phi_\mu(x, z^*_\mu)$$

has a local minimum for $\mu > 0$ at x^*_μ for ρ suficiently large.

Therefore, if x^*_μ is a solution of the logarithmic barrier function problem, then there exists a $\bar{\rho} \geq 0$ such that

$$x^*_\mu = arg\, min\, M_\mu(x, y^*_\mu, \mu(x^*_\mu)^{-1}; \rho)$$

for any $\rho > \bar{\rho}$, where y^*_μ is the Lagrange multiplier associated with the equality constraints. In this way, we have established the analogy.

Therefore, from Proposition 2.4.1 and the latter result, the perturbed KKT conditions together with our generalized augmented Lagrangian function can be used to build an effective globalization strategy with does not require solution of badly conditioned systems of equations.

In this research, our generalized augmented Lagrangian function plays an important role in the globalized algorithm that we will propose in Chapter 5.

Chapter 4

The Newton direction as a descent direction

4.1 Fundamental definitions

In this section, we present the fundamental notions of interior and central points. In addition, we consider the primary variables associated with problem (2.1) and describe the manner in which we deal with these variables.

> **Definition 4.1.1** A point (x, y, z) is said to be an interior-point for problem (2.1) if $(x, z) > 0$.
>
> **Definition 4.1.2** An interior-point (x, y, z) is said to be a quasi-central point for problem (2.1) for a given $\mu > 0$ if
>
> $$h(x) = 0 \text{ and } (XZ)e = \mu e. \qquad (4.1)$$
>
> The quasi-central path associated with problem (2.1) is defined as the collection of quasi-central points (4.1) and is parameterized by μ.

This notion will play an important role in the design of our global algorithm.

We find it convenient to denote the triple (x, y, z) by v and $(\Delta x, \Delta y, \Delta z)$ by Δv. Recently Martinez, Parada, and Tapia [32] quite effectively demonstrated that in interior-point applications the variables (x, z) play a primary role and the variable y plays a secondary role. Observe that y does not enter into any of the constraints and at a solution $(x_\mu^*, y_\mu^*, z_\mu^*)$ one can readily obtain y_μ^* from x_μ^* and z_μ^*. This philosophical point of view is in strong alignment with the globalization strategy we are about to describe. We will treat the variable y essentially as a parameter, i.e., we will not differentiate our merit function with respect to y and we will exclude y from our descent considerations. This latter consideration will employ only the variables x and z. Hence we have a need to denote the (x, z) variables differently from the triple $v = (x, y, z)$. The notion $\tilde{v} = (x, z)$ accomplishes this objective.

Definition 4.1.3 For a given $\mu > 0$, the interior-point Newton step at the interior-point $v = (x, y, z)$ is $\Delta v = (\Delta x, \Delta y, \Delta z)$ obtained as the solution of the linear system

$$F'(v)\Delta v = -F_\mu(v). \tag{4.2}$$

It is not difficult to see that (4.2) has the structure

$$\begin{pmatrix} \nabla_x^2 l(v) & \nabla h(x) & -I \\ \nabla h(x)^T & 0 & 0 \\ Z & 0 & X \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta y \\ \Delta z \end{pmatrix} = - \begin{pmatrix} \nabla_x \ell(x, y, z) \\ h(x) \\ XZe \end{pmatrix} + \mu \begin{pmatrix} 0 \\ 0 \\ e \end{pmatrix} \tag{4.3}$$

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

Definition 4.1.4 For a given $\mu > 0$, the interior-point Newton step at $\tilde{v} = (x, z)$ is $\Delta\tilde{v} = (\Delta x, \Delta z)$ where Δx and Δz were obtained from (4.3).

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

$$\tilde{v} + \alpha\Delta\tilde{v} > 0. \tag{4.4}$$

A standard way to find α that satisfies this inequality is to let

$$\alpha = min(1, \tau\hat{\alpha}) \tag{4.5}$$

where

$$\hat{\alpha} = min\left(\frac{-1}{min\,(X^{-1}\Delta x, -1)}, \frac{-1}{min\,(Z^{-1}\Delta z_k, -1)}\right) \tag{4.6}$$

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

4.2 The Newton step as a descent direction

The results in this section concern descent in the Newton direction for the penalty term and the generalized augmented Lagrangian function defined in (3.1) and (3.2).

We will use the following proposition in the proof of Theorem 6.3.1.

Proposition 4.2.1 For $\mu > 0$, the penalty term $\Phi_\mu(x, z)$ is bounded below by $n\mu(1 - ln(\mu))$ in the class of all interior points. Moreover, it will be positive for $0 < \mu < \hat{e}$ (where $\hat{e}$ is the Euler constant).

Proof The proof follows directly from the observation that for $\mu > 0$ the function $g(w) = w - \mu\, ln(w)$ has $w = \mu$ as its global minimizer. □

Theorem 4.2.1 Consider $\mu > 0$. Let $v = (x, y, z)$ be an interior-point. Then the Newton step $\Delta\tilde{v}$ at $\tilde{v} = (x, z)$ is a descent direction for the penalty term Φ_μ, i.e.,

$$\nabla\Phi_\mu^T(\tilde{v})\Delta\tilde{v} < 0,$$

if and only if v is not a quasi-central point.

Proof The components of the gradient of Φ_μ with respect to x and z are

$$\nabla_x\Phi_\mu(\tilde{v}) = \nabla h(x)h(x) + z - \mu x^{-1},$$

and

$$\nabla_z\Phi_\mu(\tilde{v}) = x - \mu z^{-1}.$$

The directional derivative of $\Phi_\mu(\tilde{v})$ in the direction $\Delta\tilde{v}$ is

$$\nabla\Phi_\mu(\tilde{v})^T\Delta\tilde{v} = \nabla_x\Phi_\mu(\tilde{v})^T\Delta x + \nabla_z\Phi_\mu(\tilde{v})^T\Delta z.$$

If we set $w = (XZ)^{1/2}e$, using the equality constraint and the complementary equation that comes from (4.3) we obtain

$$\nabla\Phi_\mu(\tilde{v})^T\Delta\tilde{v} = -(\|h(x)\|^2 + \|w - \mu w^{-1}\|^2) \leq 0. \tag{4.7}$$

The latter equation establishes the theorem. □

Theorem 4.2.2 Consider $\mu > 0$. Let $v = (x, y, x)$ be an interior-point. If v is not a quasi–central point, then there exists a real number $\hat{\rho}$ such that for any $\rho > \hat{\rho}$ the Newton step $\Delta\tilde{v}$ at $\tilde{v} = (x, z)$ is a descent direction for our generalized augmented Lagrangian function M_μ in the sense that

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

Proof We have

$$\nabla_x M_\mu(v; \rho)^T\Delta x + \nabla_z M_\mu(v; \rho)^T\Delta z = \nabla_x\ell(v)^T\Delta x + \nabla_z\ell(v)^T\Delta z + \rho\nabla\Phi_\mu(\tilde{v})^T\Delta\tilde{v}. \tag{4.8}$$

From the previous theorem we know that $\nabla\Phi_\mu(\tilde{v})^T\Delta\tilde{v} < 0$. Hence our value for $\hat{\rho}$ is

$$\hat{\rho} = \frac{\nabla\,\ell(\tilde{v})^T\Delta\tilde{v}}{|\nabla\Phi_\mu(\tilde{v})^T\Delta\tilde{v}|}. \tag{4.9}$$

□

4.3 Sufficient decrease

In the statement of our algorithm we will find it convenient to consider the following transformation. If the penalty parameter ρ is written as

$$\rho = \hat{\rho} + c \tag{4.10}$$

then $\rho > \hat{\rho}$ is equivalent to stating that $c > 0$.

Also observe, if we substitute (4.10) into (4.8), then we obtain

$$\nabla_x M_\mu(v; \rho)^T\Delta x + \nabla_z M_\mu(v; \rho)^T\Delta z = c\nabla\Phi_\mu(\tilde{v})^T\Delta\tilde{v}. \tag{4.11}$$

Now, assume $\{M_\mu(\tilde{v} + \alpha\Delta\tilde{v}; y, \rho) | \alpha > 0\}$ is bounded below and as M_μ is a continuously differentiable function on $\mathbb{R}^{2n}$, then it is known from Dennis-Schnabel [9] that for any fraction $\beta \in (0, 1)$, there exists an $\alpha > 0$ that satisfies the following average rate of decrease (sufficient decrease condition)

$$M_\mu(x + \alpha\Delta x, y, z + \alpha\Delta z; \rho) \leq M_\mu(x, y, z; \rho) + c\,\alpha\,\beta\,\nabla\Phi_\mu(\tilde{v})^T\Delta\tilde{v}. \tag{4.12}$$

Now we are ready to formulate a global primal-dual interior-point Newton algorithm using our generalized augmented Lagrangian function as a merit function in the linesearch framework.

Chapter 5

Global linesearch interior–point algorithm

5.1 Description of the algorithm

Analogous to the well-known and highly useful situation in linear programming, we can define the central path associated with problem (2.1) as the collection of solutions of the perturbed KKT conditions (2.5) parameterized by μ. By the implicit function theorem we can guaranteed that such a path exists locally (i.e. for $0 < \mu < \hat{\mu}$ for some $\hat{\mu}$) in a neighborhood of a solution $v = (x^*, y^*, z^*)$ of (2.1) which satisfies the standard Newton's method theory assumptions A2.1-A2.5 listed in Chapter 2. Related to the central path notion, we introduced a new notion of centrality called the quasi–central path given by Definition (4.1.2). It is worth mentioning that the quasi-central path is really a surface and as before under the latter assumptions, we can guarantee that this surface exists close to the solution of the problem. The use of the quasi-central path as opposed to the central path gives us a definite advantage. Specifically, far from the solution it may be the case that the central path point corresponding to a parameter μ does not exist. When we consider the quasi-central path we have relaxed the requirements, i.e., we do not require $\nabla_x l = 0$, and consequently the chance that a point on the quasi-central path corresponding to this μ exists are dramatically improved. Indeed, observe that a point x is on the quasi-central path, i.e.,

$$(x, z) \in S^* = \{(x, z) \in \mathbb{R}^{2n} : h(x) = 0, XZe = \mu e, z > 0\}$$

if and only if x is strictly feasible, i.e.,

$$x \in S = \{x \in \mathbb{R}^n : h(x) = 0, x > 0\}.$$

This notion plays a fundamental role in the formulation of our global algorithm. Now, we follow the lead given by Gonzalez-Lima, Tapia, and Potra [27] in their linear programming application. We are looking for a notion of an effective neighborhood of our quasi-central path corresponding to μ. Toward this end we offer the following notion of closeness to a point on the quasi-central path.

Definition 5.1.1 We define a (μ,γ)-neighborhood of a point on the quasi-central path corresponding to μ by

$$\mathcal{N}_\mu(\gamma) = \{\, v = (x,y,z) \in R^{n+m+n} : \; x > 0, z > 0, \|h(x)\|^2 + \|w - \mu w^{-1}\|^2 \leq \gamma\mu \,\} \tag{5.1}$$

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

In this work, the previous definition gives us an effective measure of how close an interior-point is from satisfying the perturbed KKT conditions for a corresponding $\mu > 0$.

It is of value to observe that a point on the central path for a corresponding μ is also on the quasi-central path for the corresponding μ; but the converse is not true. Moreover, both are contained in a (μ,γ)-neighborhood for $(\mu,\gamma) > 0$. Hence the (μ,γ)-neighborhood notion is fairly generous and hopefully will not be excessively restrictive.

The basic idea of our global algorithm is to apply a linesearch Newton's method to the perturbed KKT conditions (2.5) for fixed μ until we arrive to a specified (μ,γ)-neighborhood. This part of the algorithm is called the inner loop. Then we decrease μ, specify a new (μ,γ)-neighborhood and repeat the linesearch Newton's method. Clearly, if our (μ,γ)-neighborhood are excessively restrictive the algorithm will be quite costly. If the neighborhoods are excessively large, then global convergence can be threatened as mentioned in Section 2.3. We expect our algorithm to mimic the logarithmic barrier function method far from the solution and the El-Bakry et al [12] quadratically convergent Newton interior-point method near the solution.

5.2 Update of the penalty parameter

Before we propose a global algorithm, we begin by explaining how we will update the penalty parameter ρ associated with the generalized augmented Lagrangian function in order to force a descent direction.

For a corresponding $\mu > 0$, we update the current penalty parameter ρ by a nondecreasing update ρ_+.

Remember that from (4.10) the penalty parameter ρ is given by a value $\hat{\rho}$, where $\hat{\rho}$ is given by (4.9), plus a positive value $c > 0$. For our purpose, c will always be greater than or equal to a predetermined positive value $\hat{c}$ and is not varied during the entire minimization process.

In line with this objective, we update the penalty parameter in the following manner. Given $\hat{c} > 0$, and ρ the current penalty parameter, then the update is

$$\rho_+ = \begin{cases} \hat{\rho}_+ + \hat{c} & \text{if } \hat{\rho}_+ + \hat{c} > \rho \\ \hat{\rho}_+ + c & \text{otherwise,} \end{cases} \tag{5.2}$$

where $c = \rho - \hat{\rho}_+$, and $c \geq \hat{c}$.

This expression is the formula we use to update the penalty parameter ρ in the global linesearch interior-point algorithm that we will present in the next section.

5.3 Linesearch interior-point Newton algorithm

We propose the following global primal-dual interior-point Newton algorithm with a backtracking linesearch algorithm for the nonlinear optimization problem (2.1).

Algorithm 5.3.1 (Linesearch interior-point algorithm)

Step 0. Consider an initial interior-point $v_o = (x_o, y_o, z_o)$ (i.e. $(x_o, z_o) > 0$).

Choose $\beta, p, \gamma, \sigma \in (0,1)$, and $\hat{c} > 0$.

Step 1. For k=0,1,2,....until convergence do

1.1 Choose $\mu_k > 0$.

Step 2. Repeat (INNER LOOP)

2.1 Solve the linear system

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

2.2 (Maintain x and z positive). Choose $\tau_k \in (0,1)$ and calculate $\hat{\sigma}_k$ according to (4.6). Let $\tilde{\sigma}_k = min(1, \tau_k \hat{\sigma}_k)$.

2.3 (Force a descent direction). Calculate c_k and ρ_k by (5.2) to ensure a Newton descent direction for M_μ.

2.4 (Armijo's condition (sufficient decrease)). Find $\alpha_k = p^t \tilde{\sigma}_k$ where t is the smallest positive integer such that α_k satisfies

$$M_{\mu_k}(x_k + \alpha_k \Delta x_k, y_k, z_k + \alpha_k \Delta z_k; \rho_k) \leq M_{\mu_k}(v_k; \rho_k) + c_k \alpha_k \beta \nabla \Phi_{\mu_k}(\tilde{v}_k)^T \Delta \tilde{v}_k \tag{5.3}$$

2.5 Set $v_k = (x_k + \alpha_k \Delta x_k\, , y_k + \alpha \Delta y_k, z_k + \alpha_k \Delta z_k)$.

Step 3. (Proximity to the quasi-central path)

3.1 If $v_k \notin \mathcal{N}_{\mu_k}(\gamma)$ (see (5.1))

go to step 2

3.2 Else

go to step 1 (END OF INNER LOOP)

Remark 5.3.1 According to the material described in Equations 4.9-4.11, we know that our choice of ρ_k promotes descent in M_{μ_k} in the variables x and z. Moreover, Equation 4.12 explains the use of $\nabla\Phi_{\mu_k}$ in (5.3).

5.4 Updating the perturbation parameter

If an iterate, $v = (x, y, z)$, satisfies the (μ, γ)-neighborhood condition, but does not satisfy the stopping criteria for the inner loop (see (5.1)), then the perturbation parameter μ is updated by

$$\mu_{k+1} = \sigma\left(\|h(x_k)\|^2 + \|w_k - \mu_k w_k^{-1}\|^2\right) \tag{5.4}$$

where $\sigma < \gamma^{-1}$ and the new inner loop iteration is begun. In (5.4) $w_k = (X_k Z_k)^{1/2} e$.

Proposition 5.4.1 The sequence $\{\mu_k\}$ is convergent.

Proof By the way μ_k is defined, the sequence $\{\mu_k\}$ is monotonically decreasing and bounded below, therefore $\{\mu_k\}$ converges. □

Chapter 6

Global convergence theory

In this chapter, we present a global convergence theory for our linesearch interior-point Newton algorithm. In the first section, we give assumptions under which Algorithm 5.3.1 is well defined, then we present two global convergence theorems. The first theorem implies that any limit point of the sequence generated by the inner loop is on the quasi-central path, and the second theorem proves that if the sequence generated by Algorithm 5.3.1 is convergent, then it converges to a KKT point of Problem (2.1).

6.1 Well-definedness of Algorithm 5.3.1

In this section we give conditions that guarantee that Algorithm 5.3.1 is well defined starting with an arbitrary interior point v_0. Towards this end we will need the following two conditions.

A6.1: The set $\{\nabla h_1(x), \ldots, \nabla h_m(x)\}$ is linearly independent for positive x.

A6.2: For any interior-point (x, y, z),

$$\eta^T(\nabla_x^2 \ell\,(x, y, z) + X^{-1}Z)\,\eta > 0$$

for all $\eta \neq 0$ satisfying $\nabla h_i(x)^T\eta = 0$, i=1,...,m.

Lemma 6.1.1 Consider $\mu > 0$. Under Assumptions A.6.1 and A.6.2 the matrix $F_\mu'(x, y, z)$ is nonsingular for any interior-point (x, y, z).

Proof. The matrix $F_\mu'(x, y, z)$ can be written

$$F_\mu'(x, y, z) = \begin{pmatrix} A & B \\ C & D \end{pmatrix}$$

where

$$A = \begin{pmatrix} \nabla_x^2 \ell(x,y,z) & \nabla h(x) \\ \nabla h(x)^T & 0 \end{pmatrix}, \; B = \begin{pmatrix} -I \\ 0 \end{pmatrix}, \; C = (Z \;\; 0), \text{ and } D = X.$$

Since (x,y,z) is an interior-point, the diagonal matrix D is nonsingular. Therefore the matrix $F_\mu'(x,y,z)$ is nonsingular if $K = A - BD^{-1}C$ is a nonsingular matrix (see [16]). In our particular case K is given by

$$K = \begin{pmatrix} \nabla_x^2 \ell(x,y,z) + X^{-1}Z & \nabla h(x) \\ \nabla h(x)^T & 0 \end{pmatrix}.$$

From Assumptions A6.1 and A6.2, in a rather standard manner, it is possible to show that K is a nonsingular matrix. ◇

6.2 Global convergence theory

In order to state our global convergence theory, we start by proving that for a $\mu > 0$ (fixed) any limit point of the sequence generated by the subproblem inner loop is a quasi-central point.

Theorem 6.2.1 (Subproblem inner loop). Consider $\mu > 0$ (fixed). Let $v_\mu^* = (x_\mu^*, y_\mu^*, z_\mu^*)$ be a limit point of the sequence $\{v_k = (x_k, y_k, z_k)\}$ generated by the inner loop of Algorithm 5.3.1, with the stopping criterion deactivated. Assume that F' is continuous at v_μ^* and $F'(v_\mu^*)$ is nonsigular, then $(x_\mu^*, y_\mu^*, z_\mu^*)$ is on the quasi-central path. i.e.,

$$h(x_\mu^*) = 0, \quad X_\mu^* Z_\mu^* e = \mu e,$$

and

$$x_\mu^* > 0, \quad z_\mu^* > 0.$$

Proof. Since v_μ^* is a limit point of $\{v_k\}$, there exists a convergent subsequence $\{v_k, k \in K\}$ such that $v_k \to v_\mu^*$, $k \in K$. We first prove three important properties related to this subsequence.

First, we prove that v^*_μ is an interior-point. Since the penalty term $\Phi_\mu(\tilde{v}_k)$ is bounded below, by Proposition 4.2.1., and the fact that the Newton step $\Delta\tilde{v}_k$ is a descent direction for this quantity, then for $\beta^* \in (0,1)$ there exist a step length $\bar{\alpha}_k \in (0, \tilde{\alpha}_k]$, where $\tilde{\alpha}_k$ is given by (4.5), such that

$$\Phi_\mu(\tilde{v}_k + \bar{\alpha}_k \Delta\tilde{v}_k) \leq \Phi_\mu(\tilde{v}_k) + \beta^* \bar{\alpha}_k \nabla\Phi_\mu(\tilde{v}_k)^T \Delta\tilde{v}_k.$$

This assures that the sequence $\{\Phi_\mu(\tilde{v}_k + \bar{\alpha}_k\Delta\tilde{v}_k)\,; k \in K\}$ is monotone decreasing, therefore

$$\Phi_\mu(\tilde{v}_k + \bar{\alpha}_k\Delta\tilde{v}_k) \leq \Phi^o_\mu \; for \;\; k \; \in K,$$

where Φ^o_μ is the penalty term evaluated at the initial point. From the last inequality and by the way Φ_μ is defined, we have that all products $(x_j)_k\,(z_j)_k$, $k \in K$ are bounded away from zero. As $\{\tilde{v}_k, \;\; k \in K\}$ is bounded because the sequence $\{v_k\}$ is convergent, we conclude that any individual components $(x_j)_k$ and $(z_j)_k$ are bounded away from zero; implying that

$$x_k \rightarrow x^*_\mu > 0 \;\text{ and }\; z_k \rightarrow z^*_\mu > 0, \; k \in K.$$

Second, the sequence of search directions $\{\Delta v_k, k \in K\}$ is bounded. We conclude this because $E = \{v_k\,, v^*_\mu\,, k \in K\}$ is a compact set and $F_\mu\,, (F'_\mu)^{-1}$ are continuous functions on E.

Third, $\{\bar{\alpha}_k, k \in K\}$ is bounded away from zero. If it is not true, then without loss of generality we can say that for at least one j,

$$\lim_{k\rightarrow\infty} \frac{(x_j)_k}{|\Delta x_j|_k} = 0, \; k \in K.$$

Since the components $(x_j)_k$ are bounded away from zero, we must have that $|\Delta x_j|_k$ goes to infinity, but this contradicts the fact that $\{\Delta v_k, \;\; k \in K\}$ is bounded.

Now, we are ready to begin the proof of the theorem.

Let $\{\rho_k = \hat{\rho}_k + c_k;\; c_k > 0, \; k \in K\}$ be the sequence of penalty parameters associated with $\{\tilde{v}_k, \; k \in K\}$. Two possibilities can occur with this sequence.

First, the sequence $\{\hat{\rho}_k,\ k \in K\}$ can be unbounded. Then from Equation (4.9) and the fact that $\tilde{\nabla}\ell\,(v_k)^T\Delta\tilde{v}_k$ is bounded we have

$$\left|\nabla\Phi_\mu(\tilde{v}_k)^T\nabla\tilde{v}_k\right| \to 0,\ k \in K.$$

Therefore, Equation (4.7) directly implies that

$$\lim_{k\to\infty} h(x_k) = 0,\ k \in K$$

and

$$\lim_{k\to\infty} X_k z_k = \mu e,\ k \in K.$$

Our second possibility is that, the sequence $\{\hat{\rho}_k, k \in K\}$ is bounded above. Let $\rho^* = \sup\{\hat{\rho}_k, k \in K\}$. Since $\{\hat{\rho}_k; k \in K\}$ is bounded, then for every $c > 0$ there exists a $k_o \in K$ such that $\hat{\rho}_k \leq \hat{\rho}_{k_o}+c$ and $\rho^* \leq \hat{\rho}_{k_o}+c$ for $k \geq k_o, k \in K$. We can assume that $\rho_{k_o} = \hat{\rho}_{k_o}+c_{k_o}$, and by (5.2), it follows that $\rho_k = \rho_{k_o}$, for $k \geq k_o, k \in k$. Now, we can define the merit function as

$$M_\mu(\tilde{v}_k;\rho_{k_o}) = \ell\,(\tilde{v}_k) + \rho_{k_o}\,\Phi_\mu(\tilde{v}_k),\ k \geq k_o\,, k \in K$$

Since $M_\mu(\tilde{v}_k; y, \rho_{k_o})$ is a continuously differentiable function on $\mathbb{R}^{2n}$, bounded below, and we are considering an iterative scheme

$$\tilde{v}_{k+1} = \tilde{v}_k + \alpha_k\Delta\tilde{v}_k > 0$$

where $\nabla M_\mu(\tilde{v}_k;\rho_{k_o})^T\Delta\tilde{v}_k < 0$, $\alpha_k \in (0, \bar{\alpha}_k]$ satisfies the sufficient decrease condition given by Substep 2.4 of the Algorithm 5.3.1 with $\bar{\alpha}_k$ bounded away from zero, then on the sequence $\{\tilde{v}_k,\ k \geq k_o,\ k \in K\}$ we have, see [39],

$$\frac{\nabla M_\mu(\tilde{v}_k;\rho_{k_o})^T\Delta\tilde{v}_k}{\|\Delta\tilde{v}_k\|} \to 0.$$

Since $\{\Delta\tilde{v}_k\}$ is bounded and by (4.11), we obtain

$$\nabla M_\mu(\tilde{v}_k;\rho_{k_o})^T\Delta\tilde{v}_k = c_k\nabla\Phi_\mu(\tilde{v}_k)^T\Delta\tilde{v}_k \to 0$$

where $c_k = \rho_{k_o} - \hat{\rho}_k \geq \rho_{k_o} - \rho^* > 0$, $k \geq k_o$, $k \in K$. Since c_k becomes a constant greater than zero for $k \geq k_0$, we have

$$\nabla\Phi_\mu(\tilde{v}_k)^T \Delta\tilde{v}_k \to 0, \; k \geq k_o, \; k \in K.$$

Again, Equation (4.7) directly implies that

$$\lim_{k \to \infty} h(x_k) = 0, \; k \in K$$

and

$$\lim_{k \to \infty} X_k z_k = \mu e, \; k \in K.$$

◇

The next theorem states that, if the sequence generated by Algorithm 5.3.1 is convergent, then it converges to a KKT point of problem (2.1).

Theorem 6.2.2 If the sequence $\{v_k = (x_k, y_k, z_k)\}$ generated by Algorithm 5.3.1 converges to $v^* = (x^*, y^*, z^*)$ and the standard assumptions A2.1-A2.5 hold at x^*, then x^* is a KKT point of problem (2.1).

Proof. Any interior-point generated by Algorithm 5.3.1 is of the form

$$v_k = (x_k + \alpha_k \; \Delta x_k, y_k + \alpha_k \; \Delta y_k, z_k + \alpha_k \; \Delta z_k).$$

It is not difficult to see that the assumptions of this theorem imply the assumptions of the previous theorem. Hence the step length α_k is bounded away from zero. Therefore, we conclude that the sequence $\{\Delta v_k = (\Delta x_k, \Delta y_k, \Delta z_k)\}$ converges to zero.
Now, the subsequence

$$v_{kj} = (x_{kj} + \alpha_{kj} \; \Delta x_{kj}, y_{kj} + \alpha_{kj} \; \Delta y_{kj}, z_{kj} + \alpha_{kj} \; \Delta z_{kj})$$

such that $v_{kj} \in \mathcal{N}_{\mu_k}(\gamma)$ with $\mu_k \longrightarrow 0$ converges to the same point $v^* = (x^*, y^*, z^*)$, and in addition the subsequence $\{\Delta v_{k_j} = (\Delta x_{kj}, \Delta y_{kj}, \Delta z_{kj})\}$ converges to zero.

From the first equation of system (4.3) we have

$$\nabla f(x_{kj}) + \nabla h(x_{kj})(y_{kj} + \Delta y_{kj}) - (z_{kj} + \Delta z_{kj}) = -\nabla_x^2 \ell(v_{kj})\Delta x_{kj}.$$

Taking the limit as $k \to \infty$, and recalling that $\{\nabla_x^2 \ell(v_k)\}$ is bounded, we conclude that

$$\nabla f(x^*) + \nabla h(x^*)y^* - z^* = 0. \tag{6.1}$$

Now, since $v_{kj} = (x_{kj}, y_{kj}, z_{kj}) \in \mathcal{N}_{\mu_k}(\gamma)$, then its associated $\tilde{v}_{kj} = (x_{kj}, z_{kj})$ satisfies the inequality

$$\|h(x_{kj})\|^2 + \|w_{kj} - \mu_k w_{kj}^{-1}\|^2 \leq \gamma \mu_k.$$

Again, taking the limit as $k \to \infty$, we obtain

$$h(x^*) = 0, \tag{6.2}$$

$$X^* Z^* e = 0, \tag{6.3}$$

and

$$(x^*, z^*) \geq 0. \tag{6.4}$$

From (6.1)- (6.4), $F(x, y, z) = 0$ and $(x, z) \geq 0$ are satisfied by $v^* = (x^*, y^*, z^*)$. Therefore, v^* is a KKT point of problem (2.1). ◇

Chapter 7

Numerical results. Part I

7.1 Implementation of the algorithm

The numerical experiments were done on a SPARC station 5 running the SunOS Operating-System Release 4.1 with 64 Megabytes of memory. The programs were written in MATLAB version 4.2a. In the implementation of Algorithm 5.3.1 the parameters are chosen as follows. The initial perturbation parameter μ_o is $10^{-2}x_o^T z_o$. In Substep 1.1, we define μ_k by (5.5), with $\sigma = 10^{-2}$. In Substep 2.2, we choose the parameter τ_k (percentage of movement to the boundary) as

$$\tau_k = \max\left(.8, 1 - 100 * x_k^T z_k\right).$$

In Substep 2.3, the critical value for $\hat{c}$ in order to obtain a descent direction for the generalized augmented Lagrangian function is 2. Moreover, in Substep 2.4 we choose $\beta = 10^{-4}$ and set the backtracking factor $p = 0.5$. In Substep 3.1, we take $\gamma = .8$. We used a finite difference approximation to the Hessian of the Lagrangian function.

7.2 Numerical results

Our computations are directed at two main objectives. The first is to evaluate our generalized augmented Lagrangian function, $\mathbf{M}_\mu$, as a new merit function. The second objective is to compare the behavior of our merit function using our notion of centrality, which we will refer to as $\mathbf{M}_\mu$ − **CP**, with the strategy of using the ℓ_2 norm of the residual function as a merit function with the perturbation parameter μ changed at each iteration. This latter strategy was used by El-Bakry et al [12] and will be referred as ℓ_2 − **NCP**. Here **CP** denotes central point and **NCP** denotes no central point and signifies whether we use an inner loop or not.

The numerical experiments were performed on the Hock and Schittkowski set of test problems [15], and the Schittkowski test problems [47]. For all the problems, we used the standard starting points listed in [15] and [47]. The results of our numerical experiments are summarized in Tables 1-3. In each table, the first column gives the problem number as given in [15] and [47]. The second, third, and fourth columns give the dimension (number of variables, not including slack variables), the number of equality constraints and the number of inequality constraints, respectively. The fifth column gives the number of Newton iterations reported by El-Bakry et al [12], and the sixth column gives the number of Newton iterations obtained by replacing the ℓ_2 norm of the residual merit function with our generalized augmented Lagrangian merit function and following the strategy of El-Bakry et al [12] otherwise. We denote this strategy by $\mathbf{M}_\mu - \mathbf{NCP}$. The last column reports the number of Newton iterations taken by using the strategy of $\mathbf{M}_\mu - \mathbf{CP}$, described in Algorithm 5.3.1. In all cases, Newton iterations means the number of times that each algorithm solves the linear system associated with the problem until it obtains a point that satisfies the following stopping criterion

$$\frac{\|F(v_k)\|_2}{1+\|v_k\|_2} \leq \epsilon_{exit} = 10^{-8}.$$

We summarize the results obtained in the next three tables in the following way.

1. For the tested problems there is no doubt that our merit function M_μ gives good performance. In particular, our strategy M_μ-CP has an outstanding advantage when compared with ℓ_2-NCP. The 71 problems tested, without including problem 13, reported that the ℓ_2-NCP strategy required a total of 964 Newton iterations while our approach M_μ-CP required only a total of 770 iterations.

2. In problem 13, where strict complementarity does not hold at the solution, the approach ℓ_2-NCP reported slow convergence (after 100 iterations the norm of the residual was 3.21 10^{-2}). Yamashita [53], states that his algorithm takes 197 iterations to solve this problem in the sense that he obtains a good approximation to the solutions of the primal variables, but the norm of the Karush-Kuhn-Tucker conditions is not small. Using our merit function M_μ with the NCP and the CP strategies, we obtain the

solution of the problem. Moreover, with M_μ-CP strategy we obtain convergence in only 26 iterations.

7.3 Table notation

The abbreviations used in Tables 1-3 are collected in this section. The first three columns contain information about the problems:

n Number of variables

m Number of equality constraints

p Number of inequality constraints

The second set of three columns denotes the number of Newton iterations taken by the following algorithms:

ℓ_2-NCP Algorithm using the ℓ_2 norm of the residual function as a merit function without our centrality strategy. This is the algorithm presented by El-Bakry et al [12].

M_μ-NCP Algorithm using the generalized augmented Lagrangian function as a merit function with the strategy given by El-Bakry et al [12].

M_μ-CP Algorithm using the generalized augmented Lagrangian merit function with the quasi-central path as a notion of centrality condition.

Numerical Results

Problem	n	m	p	ℓ_2-NCP	M_μ-NCP	M_μ-CP
1	2	0	1	70	33	23
2	2	0	1	9	6	7
3	2	0	1	6	7	2
4	2	0	2	6	8	3
5	2	0	4	7	11	6
10	2	0	1	10	10	9
11	2	0	1	9	15	7
12	2	0	1	10	11	9
13*	2	0	3	>100	76	26
14	2	1	1	7	5	5
15	2	0	3	15	23	8
16	2	0	5	19	25	17
17	2	0	5	34	32	14
18	2	0	6	18	27	35
19	2	0	6	15	24	8
20	2	0	5	13	38	14
21	2	0	5	13	14	10
22	2	0	2	7	4	3
23	2	0	9	21	>100	20
24	2	0	5	8	15	10
25	3	0	6	9	17	8
26	3	1	0	26	28	8

Numerical Results (cont.)

Problem	n	m	p	ℓ_2-NCP	M_μ-NCP	M_μ-CP
29	3	0	1	13	11	8
30	3	0	7	13	12	10
31	3	0	7	9	18	6
32	3	1	4	15	19	15
33	3	0	6	10	20	7
34	3	0	8	9	17	8
35	3	0	4	7	12	9
36	3	0	7	9	14	6
37	3	0	8	8	14	6
38	4	0	8	11	16	11
41	4	1	8	12	16	15
43	4	0	3	12	12	10
44	4	0	10	9	17	6
45	5		10	9	17	11
53	5	3	10	6	12	7
55	6	6	8	62	28	4
60	3	1	6	9	11	9
62	3	1	6	9	14	14
63	3		3	8	10	13
64	3	0	4	24	24	17
65	3	0	7	20	22	18
66	3	0	8	10	21	7
71	4	1	9	18	16	18
72	4	0	10	12	39	12
73	4	1	6	17	17	17
75	4	3	10	16	22	11
76	4	0	7	8	12	7

Numerical Results (cont.)

Problem	n	m	p	ℓ_2-NCP	M_μ-NCP	M_μ-CP
80	5	3	10	6	8	13
81	9	13	13	13	11	13
83	5	0	16	23	25	16
84	5	0	16	17	33	22
86	5	0	15	18	16	14
93	6	0	8	10	22	8
100	7	0	4	10	17	8
104	8	0	22	12	16	10
226	2	0	4	7	12	8
227	2	0	2	7	8	5
231	2	0	2	57	16	21
233	2	0	1	6	16	8
250	3	0	8	8	14	6
251	3	0	7	9	14	5
262	4	1	7	8	12	16
263	4	2	2	19	15	16
325	2	1	2	7	9	8
339	3	0	4	8	12	21
340	3	0	2	8	5	4
341	3	0	4	9	11	5
342	3	0	4	14	25	12
353	4	1	6	10	16	17
354	4	0	5	11	10	6

Part II

Chapter 8

An orthogonal projection method

The main purpose of the second part of this dissertation is to extend Algorithm 5.3.1 so that it can handle large scale problems which arise often 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 (4.3) 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 chapter discusses two techniques for solving the linear system (1.2), 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 second part of this thesis. We denote the nullspace of B^T by $\mathcal{N}(B^T)$ and the nullspace of B by $\mathcal{N}(B)$.

8.1 Reducing a linear system of equations

One technique used for solving the linear system (1.2) consists of reducing it to

$$Z^T A Z w = Z^T f, \tag{8.1}$$

where $Z \in \mathbb{R}^{n\mathrm{x}(n-m)}$ 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 (8.1). 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 [38] 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 [38]. More details about this technique can be found in [36].

8.2 The orthogonal projection method

We have discussed the strengths and weaknesses of reducing the linear system 1.2 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

$$P = I - BB^\dagger. \tag{8.2}$$

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 (1.2) to the following projected system,

$$PAPu = Pf. \tag{8.3}$$

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

Theorem 8.2.1 Let P be given by (8.2). Then (Pu, p) solves the linear system (1.2) with $g = 0$ if and only if $PAPu = Pf$ and $p = B^{\dagger}(f - APu) + w$, *forany* w$\in \mathcal{N}(B)$.

Proof Let (Pu, p) be the solution of the linear system (1.2), 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

$$\begin{aligned} APu + Bp &= APu + B\left[B^{\dagger}(f - APu) + w\right] \\ &= APu + BB^{\dagger}f - BB^{\dagger}APu \\ &= PAPu + BB^{\dagger}f \\ &= Pf + (I - P)f \\ &= f. \end{aligned}$$

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 (4.2) which arises in nonlinear programming problems, and it will be discussed further in the next chapter.

8.3 Advantages of the projection method

One of the main advantages of approach (8.3) over (8.1) 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 somehow 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 8.1 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

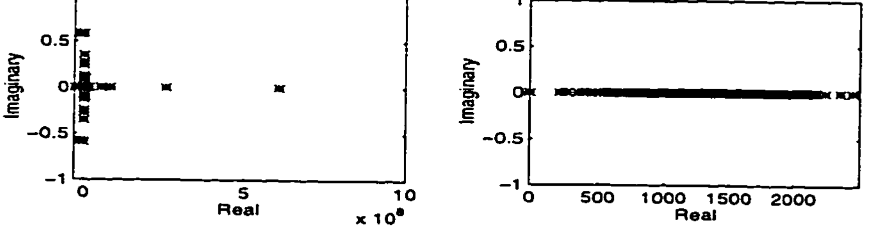

Figure 8.1 Spectra of A, Y^TAY ($Z = Y$, formed by the QR method), W^TAW, ($Z = W$, formed by the variable reduction method) and PAP, for a matrix A with random entries.

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 drawback when B is highly sparse and the computation 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 (1.2): 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.

Chapter 9

An orthogonal projection method for solving NLP linear systems

In this chapter, we explain how to reduce system (4.3) 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 (4.3). This procedure leads to satisfaction of both linearized equality constraints and linearized complementarity conditions associated with problem (2.1). 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.

9.1 Reduction of the interior-point linear system

We introduce an application of the projection method for solving the linear system (4.3). The following theorem formalizes this technique.

Theorem 9.1.1 Let P be given as in (8.2). Then $(\Delta x, \Delta y, \Delta z)$ is a solution of the linear system (4.3) if and only if

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

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

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 (4.3) can be written as follows,

$$\begin{pmatrix} A & -I & B \\ Z & X & 0 \\ B^T & 0 & 0 \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta z \\ \Delta y \end{pmatrix} = \begin{pmatrix} b_1 \\ b_3 \\ b_2 \end{pmatrix}. \tag{9.1}$$

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

$$\begin{pmatrix} G & \bar{B} \\ \bar{B}^T & 0 \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} f \\ g \end{pmatrix}. \tag{9.2}$$

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 (9.2), one obtains

$$\begin{pmatrix} G & \bar{B} \\ \bar{B}^T & 0 \end{pmatrix} \begin{pmatrix} u_h \\ p \end{pmatrix} = \begin{pmatrix} \bar{f} \\ 0 \end{pmatrix}. \tag{9.3}$$

where $\bar{f} = f - Gu_p$. At this point, the projection Theorem 3.1 can be directly applied to equation (9.3), whose solution is found by solving

$$\bar{P}G\bar{P}u_h = \bar{P}\bar{f} \tag{9.4}$$

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

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

It is straightforward to show that

$$\bar{B}^{\dagger} = \begin{pmatrix} B^{\dagger} & 0 \end{pmatrix} \text{ and } \bar{P} = \begin{pmatrix} P & 0 \\ 0 & I \end{pmatrix}, \tag{9.5}$$

where the projection operator P is defined by (8.2)

Substitution of $\bar{P}$ and $\bar{B}^{\dagger}$ into equation (9.3), yields

$$P(A + X^{-1}Z)P\Delta x_h = P(b_1 - A\Delta x_p + (X^{-1}b_3 + Z\Delta x_p)), \tag{9.6}$$

$$\Delta z = X^{-1}(b_3 - Z(\Delta x_p + P\Delta x_h)) \text{ and} \tag{9.7}$$

$$\Delta y = B^{\dagger}(b_1 - A(\Delta x_p + P\Delta x_h) + z) + w,\ w \in \mathcal{N}(B). \tag{9.8}$$

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.

9.2 Additional advantages of the orthogonal projection method

We notice that the computation of Δz and Δy, defined by equations (9.7) and (9.8), represent a small part of the overall computational cost for solving the system (4.3) 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 (9.6).

In Chapter 8, 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 (4.3) 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 (4.3) as described in Theorem 9.1.1 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^TB 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.

9.3 An iterative solution of the projected system

The standard second order sufficiency condition for Problem 2.1, A2.4, 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 (2.1) 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 (9.6). 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 [20], or MINRES [4] can be used.

9.4 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 9.4.1 (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
    (a) $w_k = P\bar{A}Pd_k$
    (b) $\alpha_k = \frac{\rho_k}{d_k^T w_k}$.
    (c) $(\Delta x_h)_{k+1} = (\Delta x_h)_k + \alpha_k d_k$.
    (d) $r_{k+1} = r_k - \alpha_k w_k$.
    (e) $\rho_{k+1} = r_{k+1}^T r_{k+1}$.
    (f) If $(\rho_{k+1} < \varepsilon_p \; stop)$
    (g) $\beta_{k+1} = \frac{\rho_{k+1}}{\rho_k}$
    (h) $d_{k+1} = r_{k+1} + \beta_{k+1} d_k$
```

The following theorem states several properties of this algorithm.

Theorem 9.4.1 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 (8.2). 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 Step 4(a)) 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(b) is always well defined. Hence the convergence follows from the classical work of Hestenes and Stiefel [17].
(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 9.4.1 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

   $$P\,(\Delta x_h)_k = (\Delta x_h)_k. \tag{9.9}$$

   Therefore Equation (9.6) can be replaced by

   $$P(A + X^{-1}Z)\,(\Delta x_h)_k = P(b_1 - A\Delta x_p + (X^{-1}b_3 + Z\Delta x_p)). \tag{9.10}$$

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

   (a) $v_1 = B^Tv$,

   (b) Solve $Ly = v_1$,

   (c) Solve $L^Tz = y$,

   (d) $v_2 = Bz$,

(e) $w = v - v_2$,

where $w = Pv$. It is important to point out that the Cholesky decomposition of $B^T B$ can be made efficient by a reordering scheme based on the sparsity pattern of B.

9.5 Iterative solution of the complete system

Now, our fundamental purpose of this second part is to define an iterative solver for the linear system (4.3) using an iterative solver for equation (9.6) and to combine it with our Algorithm 5.3.1 in an efficient manner. Specifically, we will prove that a single iteration of our solver on the linear system (4.3) produces a descent direction for the penalty term and it is also a descent direction for our modified augmented Lagrangian function introduced in Chapter 3. Towards this objective, we present the following definition

Definition 9.5.1 Our iterative solver $(\Delta x_k, \Delta y_k, \Delta z_k)$ for the linear system (4.3) is defined as

$$\Delta z_k = X^{-1}(b_3 - Z\Delta x_k), \tag{9.11}$$

$$\Delta y_k = B^{\dagger}(b_1 - A\Delta x_k + z) + w. \; w \in \mathcal{N}(B) \tag{9.12}$$

$$\Delta x_k = (\Delta x_h)_k + \Delta x_p, \tag{9.13}$$

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 our strongest theoretical result for the second part of this dissertation in the following theorem.

Proposition 9.5.1 Any search direction $(\Delta x_k, \Delta z_k)$ given by (9.13) and (9.11) satisfies linearized equality constraints and linearized complementarity condition associated with Problem (2.1), i.e. ,

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

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

Proof Substituting equation (9.13) into the second equation of the linear system (4.3) 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 (9.11), we have

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

□

Remark 9.5.1

From this proposition, we conclude that the residual error for solving the linear system (4.3) depends only on the residual error, r_k, for solving the projected system (9.6). Therefore, we define the vector $(r_k, 0)$, $0 \in \mathbb{R}^{mxn}$, as the residual vector for the original nonsymmetric and indefinite system (4.3). Consequently, if we can control the tolerance of the projected system (9.6), we can control the tolerance of the entire system (4.3).

Theorem 9.5.1 Consider $\mu > 0$. Let $v = (x, y, x)$ be an interior-point. Then the search direction $(\Delta x_k, \Delta z_k)$ given by (9.11) and

(9.13) 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 4.2.1, 4.2.2 and previous propositions establish the theorem. □

A couple of observations are in order.

Remark 9.5.2 The latter theorem means that any single iteration $(\Delta x_k, \Delta y_k, \Delta z_k)$, defined by (9.11), (9.12) and (9.13), 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 (9.6). 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 9.5.3 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.

9.6 Preconditioning

In order to obtain reasonable convergence rates, preconditioning the projected system (9.6) 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 9.6.1 Let $\|PNP\| \left\|(P\bar{A}P)^\dagger\right\| = \gamma < 1$, then

$$\begin{aligned}\left\|(P\bar{A}P)^\dagger - (PMP)^\dagger\right\| &\leq \frac{1+\sqrt{5}}{2} \times \frac{\gamma}{1-\gamma} \left\|(P\bar{A}P)^\dagger\right\| \\ &\leq \frac{1+\sqrt{5}}{2} \times \frac{\|N\| \left\|(P\bar{A}P)^\dagger\right\|^2}{1-\|N\| \left\|(P\bar{A}P)^\dagger\right\|}.\end{aligned} \tag{9.14}$$

Proof The first inequality can be obtained as a particular case of Theorem 8.24 proved by Lawson and Hanson [30, 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

$$\begin{pmatrix} \bar{A} & B \\ B^T & 0 \end{pmatrix}. \tag{9.15}$$

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 9.6.1, 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 [26, 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.

Chapter 10

Numerical results. Part II

We consider two types of numerical experiments. The first is designed to test the ideas of the inexact global minimization that we have introduced in the last two chapters. We show from the numerical point of view that one iteration on the linear system (4.3) is effective at each step of the nonlinear minimization process until the iterates enter the region of fast convergence of Newton's method. Then it is necessary to switch in order to ask for more precision of the iterative linear solver for a better performance of our 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 (1.1) and SQMR acting on a system with coefficient matrix given by (9.15). Only simple (i.e., block or incomplete Cholesky) preconditioners are tested and the results are quite illustrative.

10.1 Experiments with the global minimization algorithm

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

$$\begin{aligned} \min \left(\frac{1}{2}x^T\mathbf{A}x - \mathbf{c}^Tx\right) \\ \text{subject to } \mathbf{B}x - \mathbf{b} =& 0, \qquad (10.1) \\ x \geq& 0. \end{aligned}$$

The full rank matrix **B** and vector **b** in the equality constraints and the vector **c** in the objective function were chosen randomly. The matrix in the quadratic part of the objective function, **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.

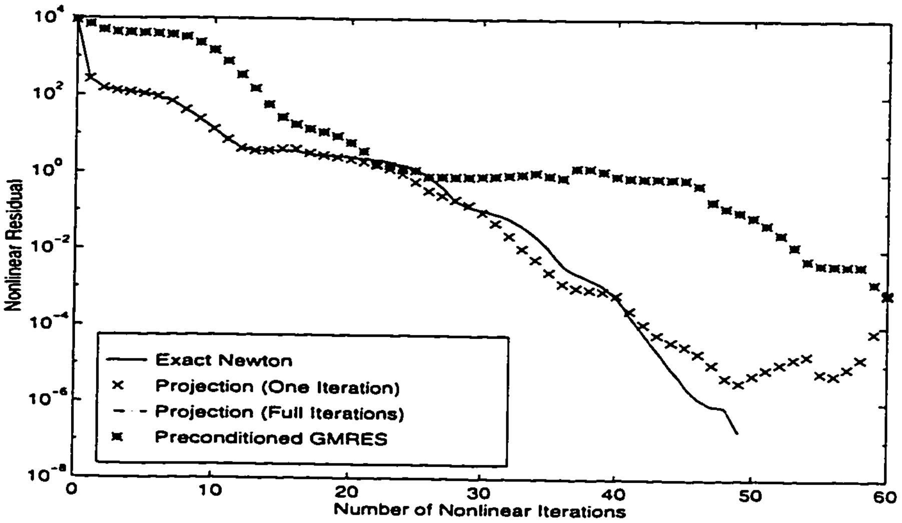

Figure 10.1 Convergence of the quadratic programming problem for exact Newton, orthogonal projection taking one iteration of CG on projected system throughout, orthogonal projection with dynamically adjusted linear tolerance and preconditioned GMRES.

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 (this was illustrated in Section 9.2.1).

In the first experiment, the Hessian matrix **A** is order 50 and the number of equality constraints is 10. This makes the coefficient matrix of (1.1) order 110. Figure 10.1 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

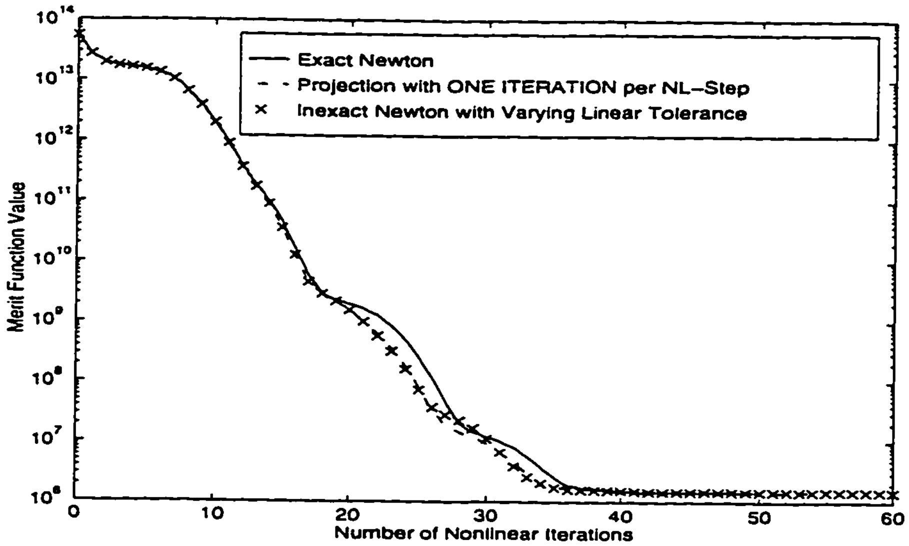

Figure 10.2 Merit function values for the quadratic programming problem for exact Newton, orthogonal projection taking one iteration of CG on projected system throughout and orthogonal projection with dynamically adjusted linear tolerance.

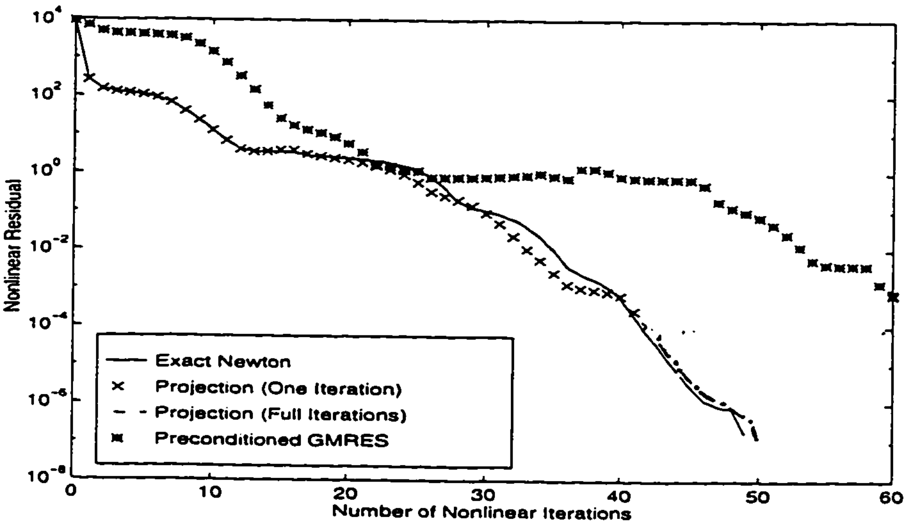

Figure 10.3 Convergence of the quadratic programming problem for exact Newton, an orthogonal projection taking one iteration of CG on projected system initially (x), and with dynamically adjusted linear tolerance near convergence to the optimum (·—).

figure: Newton's method, the projection method taking one iteration per nonlinear step, the projection method with a dynamically adjusted linear tolerance and preconditioned GMRES acting on the entire linear system (1.1) 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's 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's method. The linear tolerance was set according to

$$\mathsf{tol} = \min\left(\frac{1}{k+2}, \quad \left[\frac{|f^{(k)}|}{|f^{(0)}|}\right]^{1.5}\right), \tag{10.2}$$

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.,

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

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

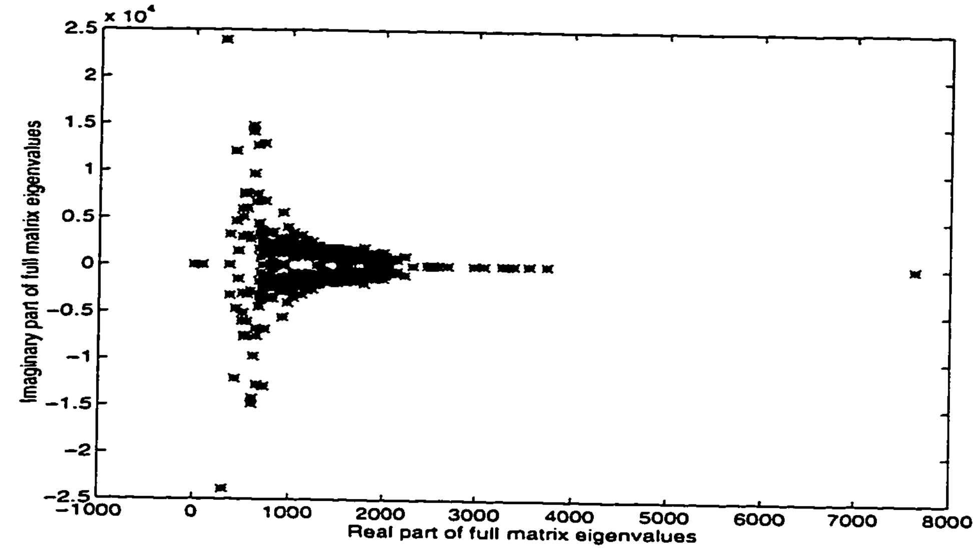

Figure 10.4 Eigenvalue distribution of coefficient matrix for interior point method formulation of the constrained quadratic minimization problem after 20 nonlinear iterations.

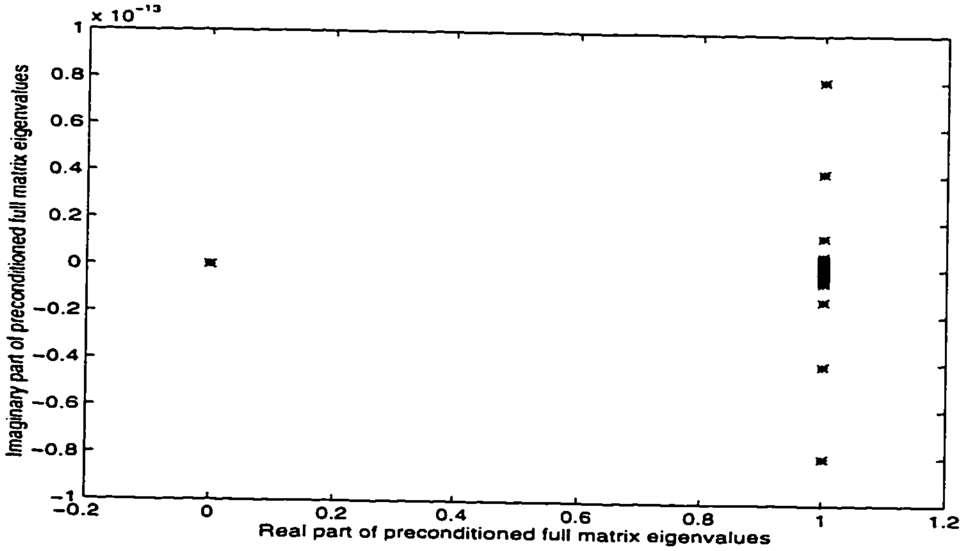

Figure 10.5 Eigenvalue distribution of preconditioned coefficient matrix for interior point method formulation of the constrained quadratic minimization problem after 20 nonlinear iterations.

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 after application of M^{-1} by comparison of Figures 10.4 and 10.5. 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 blocks 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 solution.

To see intuitively what it is going on, we direct the reader's attention to Figure 10.2, where the values of the merit function, given by Definition 3.1.1, are plotted versus the nonlinear iteration count for exact Newton and the projection method using full linear iterations or one iteration throughout. It is apparent that our merit function is no longer decreasing appreciably inside the region of quadratic convergence of Newton's method. This behavior separates, at least qualitatively, the region in which the merit function is driving the global convergence (this requires single iterations of the solver given by Definition 9.4.1) from that in which the merit function is no longer effective where 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. $P_{switch} = \frac{|\nabla M_\mu \cdot \Delta \bar{v}|}{M_\mu}$
2. IF ( $P_{switch} > \epsilon$ )
    (a) MAX LINEAR ITERATIONS = 1
3. ELSE
    (a) LINEAR TOLERANCE AS GIVEN BY (10.2)
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 the level 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 to be 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 entered repeatedly throughout the minimization process, the frequent reductions in μ can give an indication of how close one is to the region of the quadratic convergence of Newton's method. However, in many cases the neighborhood of the quasi-central path is entered only a few times near convergence and this produces too few instances of μ-reduction to make this a reliable scheme.

Finally in Figure 10.3, 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

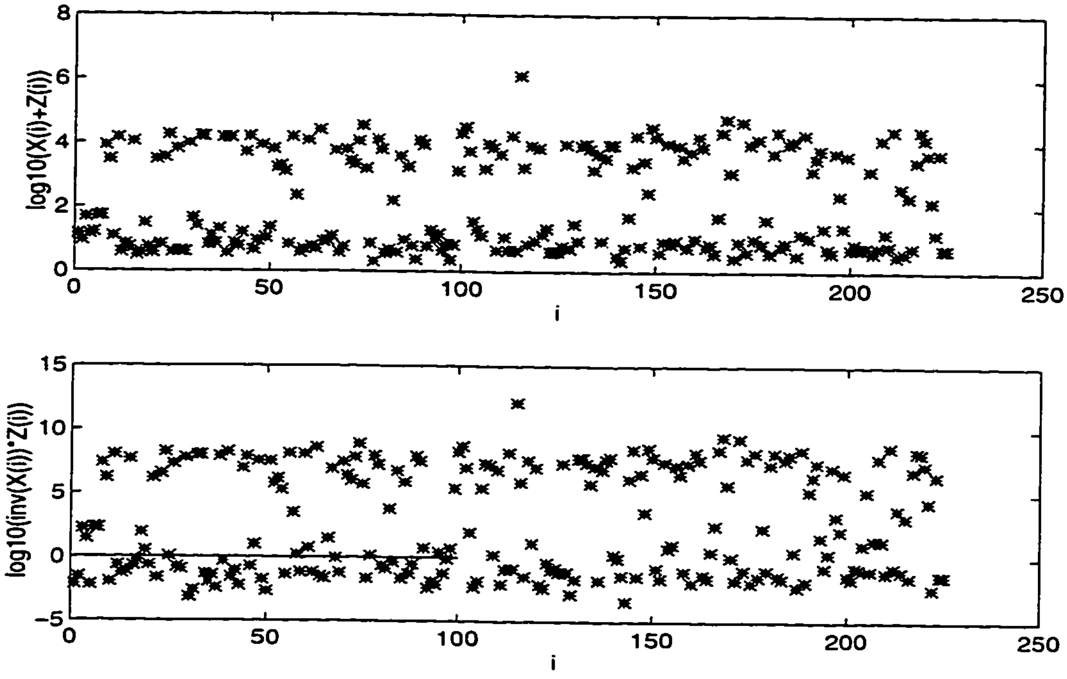

Figure 10.6 Entry distribution of the diagonal matrices X and Z.

quadratic convergence; then inside of this region, we dynamically adjusted the linear tolerance given by (10.2) 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.

Chapter 11

Conclusions and future research

11.1 Conclusions

In this dissertation, we have presented a new interior-point Newton algorithm for solving nonlinear programming problems. The algorithm utilizes the perturbed KKT conditions to promote global convergence. In order to obtain a good strategy of globalization, we have presented a generalization of the augmented Lagrangian function to be used as a new merit function, and considered a relaxation of the notion of the central path to the quasi-central path.

The algorithm attempts to follow the quasi-central path. For $\mu > 0$ (fixed), we solve an inner-loop subproblem that is equivalent to the logarithmic barrier subproblem. Moreover, our approach does not deal with ill- conditioned linear system of equations. As a stopping criterion for our inner-loop subproblem we introduced a notion of proximity to the quasi-central path. Our new merit function is used to guide inner-loop iterates to a prescribed neighborhood of the quasi-central path. The penalty parameter associated with our merit function is updated in a manner that guarentees descent in the Newton direction.

Our global convergence theorem, Theorem 6.2.2, presents a fundamental; yet rather minimal global convergence result. It is now our considered opinion that those features of the merit function that help promote the demostrated good numerical behaviour of the method, also contribute significantly to complicating the mathematical analysis of global convergence. This investigation will be the focus of a future research effort. At the present we are content with presenting the new algorithm, the impressive numerical experimentation, and a rather basic global convergence analysis.

For the second part, dealing with large-scale problems, this research is part of an ongoing project that will be continued motivated by promising numerical results. We have considered an orthogonal projection method to reduce the linear system, of order $2n + m$, associated with nonlinear programming problems to one of order n such that its condition number is not worse than the initial linear system. We solve the reduced linear system using only matrix-vector operations. In particular we use the conjugate gradient method, and then we defined an iterative solver for the entire linear system associated with nonlinear programming problem. The iterative solver has the property that a single iteration produces a descent direction with respect to our merit function, producing extremely cheap steps. This makes the use of preconditioners not needed except inside the region of quadratic convergence of Newton's method. We presented numerical experiments that confirm this fact. The projection method that we have proposed only requires the formulation of a preconditioner for a block matrix formed by the Hessian of the Lagrangian function plus a positive diagonal matrix. We report numerical experimentation for several large scale problems to illustrate the viability of the method that we have proposed.

11.2 Future research

We present the following research issues as future work:

1. Further numerical and theoretical research is needed to establish the role that the quasi-central path plays for solving nonlinear programming problems.
2. Further study of our generalized augmented Lagrangian function for solving nonlinear programming problems using different approaches, for example as a merit function in the trust region framework.

   For large-scale general nonlinear programming problems.
3. Establish a design that allows the projector operator P to be fixed for some number of iterations during the minimization process when the number of constraints is very large to avoid the high computational cost.

Bibliography

[1] K. M. Anstreicher and J. Vial, *On the convergence of an infeasible primal-dual interior-point method for convex programming*, Optimization Methods and Software, 1993.

[2] M. Argaez and R. A. Tapia, *On the global convergence of a modified augmented Lagrangian linesearch interior point Newton method for nonlinear programming.* Talk presented at the SIAM Annual Meeting, North Carolina, 1995.

[3] M. Argaez, H. Klie, M. Rame, and R. A. Tapia, *An interior point Krylov-orthogonal projection method for nonlinear programming.* Talk presented at the Fifth SIAM Conference on Optimization, Victoria, Canada, 1996.

[4] R. Barrett, M. Berry, T. Chan, J. Demmel, J. Dunato, J. Dongarra, V. Eikkhout, R. Pozo, 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*, Tech. Rep. TR-1190, Center of Supercomputing Research and Development, University of Illinois.

[6] R. Byrd, J. C. Gilbert, and J. Nocedal, *A trust region method based on interior point techniques for nonlinear programming*, Tech. Rep. 96-02, Optimization Technology Center, Northwest University, 1996.

[7] P. N. Brown, Y. Saad, and H. F. Walker, *Preconditioning with low rank updates*, in preparation, 1995.

[8] D. Cores and R. A. Tapia, *A robust choice of the Lagrange multipliers in the successive quadratic programming method*, Tech. Rep. TR94-37, Dept. of Computational and Applied Mathematics, Rice University, 1994.

[9] J. E. DENNIS JR. AND R. B. SCHNABEL, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Prentice-Hall Inc., Englewood cliffs, New Jersey, 1983.

[10] S. C. EISENSTAT AND H. F. WALKER, *Choosing the forcing terms in an inexact Newton method*, Tech. Rep. TR94–25, Dept. of Computational and Applied Mathematics, Rice University, 1994.

[11] S. C. EISENSTAT AND H. F. WALKER, *Globally convergent inexact Newton methods*, SIAM J. Optim., Vol. 4 (1994), pp. 393–422.

[12] A. S. EL BAKRY, R. A. TAPIA, T. TSUCHIYA, AND Y. ZHANG, *On the formulation and theory of primal-dual Newton interior-point method for nonlinear programming problems*, J. Optim. Theory Appl., Vol. 89 (1996), pp. 507-541.

[13] M. EL ALEM, *A global convergence theory for a class of trust region algorithms for constrained optimization*, Tech. Rep. TR88-5, Dept. of Computational and Applied Mathematics, Rice University, 1988.

[14] H. C. ELMAN AND G. H. GOLUB, *Inexact and preconditioned Uzawa algorithms for saddle point problems*, SIAM J. Numer. Anal., Vol. 31 (1994), pp. 1645-1661.

[15] W. HOCK AND K. SCHITTKOWSKI, *Test Examples for Nonlinear Programming Codes*, Springer-Verlag, New York, 1981.

[16] V. N. FADDEEVA, *Computational Methods of Linear Algebra*, Dover Publications Inc., New York, 1959.

[17] M. HESTENES AND E. STIEFEL, *Methods of conjugate gradients for solving linear systems*, Journal of Research National Bureau of Standards, Vol. 49 (1952), pp. 409-436.

[18] A. V. FIACCO AND G. P. MCCORMICK, *Nonlinear Programming: sequential unconstrained minimization techniques*, John Wiley and Sons, New York, 1968.

[19] A. Forsgren. P. E. Gill, and J. R. Shinnerl, *Interior point methods for large-scale nonconvex optimization.* Talk presented at the fifth SIAM conference on optimization, Victoria. Canada, 1996.

[20] 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.

[21] R. W. Freund and N. M. Nachtigal, *QMR: a quasi-minimal residual method for non-Hermitian linear systems*, Numer. Math., Vol. 60 (1991), pp. 315-339.

[22] R. W. Freund, G. H. Golub, and N. M. Nachtigal, *Iterative solution of linear systems*, Acta Numerica, Cambridge University Press, New York. 1991, pp.57-100.

[23] M. H. Wright, *Some properties of the Hessian of the logarithmic barrier function*, Math. Programming, Vol. 67 (1994), pp 265-295.

[24] D. Gay, M. Overton, and M. H. Wright, *An interior-point algorithm for solving general nonlinear programming problems.* Talk presented at the 15th International Symposium in Mathematical Programming. Ann Arbor, Michigan, 1994.

[25] P. E. Gill, W. Murray, D. B. Ponceleón, and M. A. Saunders, *Preconditioners for indefinite systems arising in optimization*, SIAM J. Mat. Anal. Appl., Vol. 13 (1992), pp. 292–311.

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

[27] M. Gonzalez Lima, R. A. Tapia, and F. Potra, *On effectively computing the analytic center of the solution set by primal-dual interior point methods*, Tech. Rep. TR95-38, Dept. of Computational and Applied Mathematics, Rice University, 1995.

[28] H. Klie, M. Rame, and M. Wheeler, *Krylov-secant methods for solving systems of nonlinear equations*, Tech. Rep. TR95–27, Dept. of Computational and Applied Mathematics, Rice University, 1995.

[29] L. LASDON, G. YU AND J. PLUMMER, *Primal-dual and primal interior point algorithms for general nonlinear programs*, ORSA J. on Computing, Vol. 7 (1995).

[30] C.L. LAWSON AND R.J. HANSON, *Solving Least Squares Problems*, Classics in Applied Mathematics, SIAM, Philadelphia, 1995.

[31] I. LUSTIG, R. MARSTEN, AND D. SHANNO, *On implementing Mehrotra's predictor-corrector interior point method for linear programming*, SIAM J. Optim., Vol. 2 (1992), pp. 435-449.

[32] H. J. MARTINEZ, Z. PARADA, AND R. A. TAPIA, *On the characterization of Q-superlinear convergence of quasi-Newton interior-point methods for nonlinear programming*, Bol. Soc. Mat. Mexicana Vol.1 (1995), pp. 137-148.

[33] A. V. FIACCO, AND G. P. MCCORMICK, *The sequential unconstrained minimization technique for nonlinear programming: a primal-dual method*, Management Sci., Vol. 10 (1964), pp. 360-364.

[34] S. G. NASH, *Preconditioning reduced matrices*, SIAM J. Matrix Anal. Appl., Vol. 17 (1996), pp. 47-68.

[35] S. G. NASH AND A. SOFER, *Linear and Nonlinear Programming*, McGraw–Hill, 1996.

[36] J. NOCEDAL AND M. L. OVERTON, *Projected Hessian updating algorithms for nonlinearly constrained optimization*, SIAM J. Numer. Anal., Vol. 22 (1985), pp. 821-850.

[37] R. C. MONTEIRO, J. PANG, AND T. WANG, *A positive algorithm for the nonlinear complementarity problem*, Tech. Rep., Dept. of Systems and Industrial Engineering, University of Arizona, 1992.

[38] S.NASH AND A. SOFER, *Preconditioning reduced matrices*, SIAM J. Matrix Anal. Appl., Vol. 17 (1994), pp. 47-68.

[39] J. M. ORTEGA AND W. C. RHEINBOLDT, *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, New York, 1970.

[40] R. A. TAPIA, Y. ZHANG, AND Y. YE, *On the convergence of the iteration sequence in primal-dual interior-point methods*, Math. Programming, Vol. 68, (1995), pp. 141–154.

[41] R. A. TAPIA *An introduction to the algorithms and theory of constrained optimization*, Tech. Rep. TR90036, Center for Research on Parallel Computation, or CAAM554 Class notes, Rice University, 1990.

[42] Y. SAAD AND M. H. SCHULTZ, *GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Stat. Comput., Vol. 7 (1986), pp. 856-869.

[43] Y. SAAD, *Krylov subspace methods in supercomputers*, SIAM J. Sci. Stat. Comput., Vol. 10 (1989), pp. 1200-1232.

[44] P. N. BROWN AND Y. SAAD, *Convergence theory of nonlinear Newton-Krylov algorithms*, SIAM J. Optim., Vol. 4 (1994), pp. 297-1232.

[45] P. N. BROWN AND Y. SAAD, *Hybrid Krylov methods for nonlinear system of equations*. SIAM J. Sci. Statist. Compt., Vol. 11 (1990), pp. 450-481.

[46] Y. SAAD, *On the Lanczos method for solving symmetric linear systems with several right-hand-sides*, Mathematics of Computation, Vol. 48 (1987), pp. 651-662.

[47] K. SCHITTKOWSKI, *More test Examples for Nonlinear Programming Codes*. Springer-Verlag, New York, 1987.

[48] R. J. VANDEREI AND T. J. CARPENTER, *Symmetric indefinite systems for interior-point methods*, Math. Programming, Vol. 58 (1993), pp. 1-32.

[49] J.P. VIAL, *Computational experience with a primal-dual interior-point method for smooth convex programming*, Tech. Rep. 1992.7, Dept. of Management Studies. University of Geneva, Switzerland, 1992.

[50] M. H. WRIGHT, *Interior methods for constrained optimization*, Numerical Analysis Manuscript 91-10, AT&T Bell Laboratories, Murray Hill, New Jersey, 1991.

[51] S. J. WRIGHT, *A superlinear infeasible-interior-point algorithm for monotone nonlinear complementarity problems*, Tech. Rep. MCS-P344-1292. Mathematics and Computer Science Division, Argonne National Laboratory, Illinois, 1992.

[52] S. J. WRIGHT, *An interior point algorithm for linearly constrained optimization*, SIAM J. Optim., Vol. 2 (1992), pp. 450-473.

[53] H. YAMASHITA, *A globally convergent primal-dual interior point method for constrained optimization*, Tech. Rep., Mathematical Systems Institute, Inc., Japan, 1992.

[54] A. FORSGREN AND P. GILL, *Primal-dual interior methods for nonconvex nonlinear programming*, Tech. Rep. NA-3, Dept. of Mathematics, University of California, San Diego, 1996.

[55] L. N. VICENTE, *Trust-region interior-point algorithms for a class of nonlinear programming problems*, Ph.D thesis, Tech. Rep. TR96-05, Dept. of Computational and Applied Mathematics, Rice University, 1996.

[56] H. KLIE, *Krylov-secant methods for solving large scale system of couple nonlinear parabolic equations*, Ph.D thesis,Tech. Rep. TR96-28, Dept. of Computational and Applied Mathematics, Rice University, 1996.