JOURNAL OF OPTIMIZATION THEORY AND APPLICATIONS: Vol. 55, No. 1, OCTOBER 1987

Dual Techniques for Constrained Optimization[1]

W. W. HAGER[2]

Communicated by R. A. Tapia

Abstract. Algorithms to solve constrained optimization problems are derived. These schemes combine an unconstrained minimization scheme like the conjugate gradient method, an augmented Lagrangian, and multiplier updates to obtain global quadratic convergence. Since an augmented Lagrangian can be ill conditioned, a preconditioning strategy is developed to eliminate the instabilities associated with the penalty term. A criterion for deciding when to increase the penalty is presented.

Key Words. Constrained optimization, duality, augmented Lagrangians, multiplier methods, preconditioning, null space methods.

1. Introduction

An algorithm to solve constrained optimization problems is developed. Each iteration in our algorithm has three steps: a restoration step which tries to satisfy the constraints, a multiplier update where an improved approximation to the Lagrange multiplier is computed, and a step which performs an unconstrained minimization of an augmented Lagrangian. Papers most closely related to our work include Refs. 1 and 2 by Rosen and Kreuser, Ref. 3 by Tapia, and Ref. 4 where Robinson presents a modification of the Rosen–Kreuser scheme. A fundamental difference between our algorithm and the Rosen–Kreuser scheme is that, in their algorithm, the restoration step is not separated from the minimization step—their minimization step also takes the constraints into account. An advantage to separating the restoration step from the minimization step is that feasibility can be maintained as the iterations progress. Note that, in the optimal control literature, Miele and his associates (Refs. 5–7) have also developed a successful family of algorithms for optimal control problems called the sequential gradient-restoration algorithms which involve a

[1] This work was supported by the National Science Foundation, Grant Nos. MCS-81-01892, DMS-84-01758, and DMS-85-20926, and by the Air Force Office of Scientific Research, Grant No. AFOSR-ISSA-860091.

[2] Professor, Department of Mathematics, Pennsylvania State University, University Park, Pennsylvania.

gradient phase where the cost is decreased and a restoration phase where the constraints are satisfied.

A new feature in our algorithm is that we utilize an augmented Lagrangian, while the Rosen-Kreuser-Robinson scheme utilizes the ordinary Lagrangian. Also, by including a penalty term in the Lagrangian, our algorithm has a natural extension to a globally convergent method. We show that algorithm 5.1 and Algorithm 5.2 in this paper are locally convergent while, on a global scale, convergent subsequences of iterations approach a Kuhn-Tucker point. Strategies for keeping the iterations bounded are discussed. Globally convergent algorithms based on the Rosen-Kreuser-Robinson scheme are presented in Refs. 8 and 9. However, the spirit of our method is somewhat different from Refs. 8 and 9 where a penalty method is used to generate a starting point for the locally convergent Rosen-Kreuser-Robinson scheme. Our scheme utilizes an augmented Lagrangian and starts with a small penalty. Depending on the direction of an inequality, the penalty is either increased or left the same. It is well known that, for large penalties, algorithms to solve the penalized problem converge slowly. We develop a preconditioning technique for eliminating the instabilities due to the penalty. Both numerical experiments and our theoretical analysis indicate that this preconditioning procedure will yield a striking improvement in the convergence speed.

Another important numerical issue studied in this paper is the accuracy that is needed in the restoration step and in the minimization step to preserve the quadratic convergence rate of our three-step algorithm. We show that quadratic convergence is preserved when restoration is accomplished with just one Newton step and when $n-m$ conjugate gradient iterations are used in the minimization step. Here, n denotes the number of unknowns and m denotes the number of constraints. Finally, note that, although there are some similarities between the Rosen-Kreuser-Robinson scheme and our scheme, the convergence theory that justified the Rosen-Kreuser-Robinson scheme does not seem to apply to our method. That is, proving that the Rosen-Kreuser-Robinson scheme is locally quadratically convergent is equivalent to showing that a derivative of an iteration map vanishes. Since this procesure does not seem to apply to our method, we develop a new analysis.

Algorithms in this paper are also related to Ref. 3 by Tapia. Tapia considers a multiplier update similar to ours and he minimizes an augmented Lagrangian in each iteration. Some differences between his approach and our approach are the following. His treatment of inequality constraints is different from our treatment, and his algorithm does not include adjustments to the penalty or preconditioning. Also, in each iteration, he performs one quasi-Newton iteration and updates the multiplier to obtain superlinear

convergence. In contrast, Algorithm 5.1 and Algorithm 5.2 perform several (at most n) conjugate gradient iterations before updating the multiplier. Algorithm 5.1 is linearly convergent, while Algorithm 5.2 is locally quadratically convergent. Other papers that are somewhat loosely related to our work includes Refs. 10–13. We highly recommend Ref. 11 by Nocedal and Overton. Besides developing a family of quasi-Newton schemes for constrained optimization, Ref. 11 provides an overview of recent work on quasi-Newton methods.

Our paper is organized as follows: Section 2 summarizes some properties of augmented Lagrangians and Section 3 incorporates these Lagrangians in algorithms which are locally quadratically convergent. Since an augmented Lagrangian can be ill conditioned due to the penalty term, Section 4 analyzes a preconditioning technique for eliminating the instabilities. Finally, Section 5 modifies the algorithms of Section 3 to obtain globally convergent algorithms and formulates a criterion for deciding when to increase the penalty in the augmented Lagrangian. An extended abstract of this paper appears in Ref. 14.

2. Augmented Lagrangians

A nonconvex problem often has a duality gap and the value of the dual problem is strictly less than the value of the primal problem. A strategy for bridging this gap emanates from work of Arrow and Solow (Ref. 15), Hestenes (Ref. 16), and Powell (Ref. 17). The basic idea is to augment the ordinary Lagrangian with a penalty term. To introduce this penalized dual approach, we first consider a finite-dimensional mathematical program with equality constraints:

$$\text{minimize } f(x), \quad \text{subject to } h(x)=0, x\in R^n, \tag{1}$$

where $f: R^n \to R$ and $h: R^n \to R^m$. Mathematical programs in a Hilbert space setting are studied in Refs. 18 and 19. The ordinary Lagrangian corresponding to (1) is $f(x)+\lambda^T h(x)$. Letting r be a positive scalar and letting $|\cdot|$ denote the Euclidean norm, the augmented Lagrangian corresponding to the penalty term $r|h(x)|^2$ is

$$\mathbf{L}(\lambda, x)=f(x)+\lambda^T h(x)+r|h(x)|^2. \tag{2}$$

The dual functional $\mathbf{L}(\lambda)$ corresponding to the augmented Lagrangian $\mathbf{L}(\lambda, x)$ is obtained by minimizing over x in R^n:

$$\mathbf{L}(\lambda)=\inf\{f(x)+\lambda^T h(x)+r|h(x)|^2: x\in R^n\}. \tag{3}$$

Observe that, for any $r\geq 0$, for any x that is feasible in (1), and for any λ in R^m, we have $f(x)\geq \mathbf{L}(\lambda)$, which implies the following optimality result.

Proposition 2.1. If $\lambda^* \in R^m$, $x^* \in R^n$, $h(x^*) = 0$, and $f(x^*) = \mathbf{L}(\lambda^*)$ for some r, then x^* is a solution to the primal problem (1), λ^* is a solution to the dual problem

$$\text{maximize } \{\mathbf{L}(\lambda): \lambda \in R^m\},$$

and x^* minimizes the augmented Lagrangian $\mathbf{L}(\lambda^*, x)$ over x in R^n.

As r increases, the gap between the value of the primal problem and the value of the dual problem decreases. Moreover, under reasonable assumptions, there exists a finite r for which the gap disappears. To illustrate the type of results that can be proved about the augmented Lagrangian, we state the following theorem which is extracted from Ref. 20. In stating this theorem, our convention is that the gradient ∇ is a row vector and the gradient ∇h of the vector-valued function h is an $m \times n$ matrix with i-th row ∇h_i, for $i = 1$ to m. Also, we let ∇^2 denote the Hessian matrix of second partial derivatives, and the phrase "x^* is a local minimizer for (1)" means that $h(x^*) = 0$ and $f(x^*) \leq f(x)$, whenever $h(x) = 0$ and x is near x^*.

Theorem 2.1. Suppose that x^* is a local minimizer for (1), both f and h are twice continuously differentiable in a neighborhood of x^*, and the rows of $\nabla h(x^*)$ are linearly independent. If $\lambda = \lambda^*$ is the solution to the equation

$$\nabla f(x^*) + \lambda^T \nabla h(x^*) = 0 \tag{4}$$

and $\nabla_x^2 (f(x^*) + h(x^*)^T \lambda^*)$ is positive definite in the null space of $\nabla h(x^*)$, then there exists a parameter s and a neighborhood $\mathbf{N}$ of x^* such that the problem

$$\text{minimize } \{\mathbf{L}(\mu, x): x \in \mathbf{N}\}$$

has a unique minimizer $x_{\mu,r}$ whenever $r \geq s$ and $|\lambda^* - \mu| \leq r/s$. Moreover, there exists a constant c, independent of r and μ, such that

$$|x_{\mu,r} - x^*| + |\lambda_{\mu,t} - \lambda^*| \leq c|\mu - \lambda^*|/r, \tag{5}$$

where $\lambda_{\mu,r} := \mu + 2rh(x_{\mu,r})$.

Remark 2.1. In some optimization problems, we wish to determine a saddle point of a functional, rather than a local minimum or a local maximum. For example, when expressing a partial differential equation in a variational framework, elliptic equations lead to minimization problems, while nonelliptic equations typically lead to saddle point problems. For a saddle-point problem, the assumption

$$\underset{\substack{\nabla h(x^*) y = 0 \\ \|y\| = 1}}{\text{minimum}} y^T \nabla_x^2 (f(x^*) + \lambda^T h(x^*)) y > 0 \tag{6}$$

of Theorem 2.1 will not be satisfied. In Ref. 18, we observe that (6) can be replaced by the weaker assumption

$$\underset{\substack{\nabla h(x^*)x=0 \\ \|x\|=1}}{\text{minimum}} \ \underset{\substack{\nabla h(x^*)y=0 \\ \|y\|=1}}{\text{maximum}} \ y^T \nabla_x^2 (f(x^*)+\lambda^T h(x^*))z > 0; \tag{7}$$

however, $x_{\mu,r}$ is a point where $\nabla_x \mathbf{L}(\mu, \cdot) = 0$, instead of a local minimizer for $\mathbf{L}(\mu, \cdot)$. Notice that (7) essentially requires that the Hessian of the ordinary Lagrangian evaluated at $\lambda = \lambda^*$ and at $x = x^*$ is nonsingular in the null space of $\nabla h(x^*)$, while (6) implies that the Hessian is positive definite in this null space.

As a corollary to Theorem 2.1, $\mathbf{L}(\lambda^*) = f(x^*)$ if the minimization in (3) is restricted to a neighborhood of x^*. Moreover, under a global uniqueness and growth assumption found in Ref. 21, this neighborhood can be extended to the entire space R^n. Now, let us consider the inequality constrained problem

$$\text{minimize } f(x), \quad \text{subject to } g(x) \leq 0, \ x \in R^n, \tag{8}$$

where $g: R^n \to R^l$, and let us employ the augmented Lagrangian suggested by Rockafellar (Ref. 22). To obtain this Lagrangian, problem (8) is expressed (as in Ref. 15) in the form

$$\text{minimize } f(x), \quad \text{subject to } g(x)+z=0, \ z \geq 0, \ x \in R^n. \tag{9}$$

Rockafellar's augmented Lagrangian is

$$\mathbf{L}(\lambda, x) = \inf\{f(x)+\lambda^T(g(x)+z)+r|g(x)+z|^2 : z \geq 0, z \in R^l\}.$$

After minimizing over z, we see that $\mathbf{L}$ can be written as

$$\mathbf{L}(\lambda, x) = f(x) + \sum_{i \in I_+} [\lambda_i g_i(x) + r g_i(x)^2] - (1/4r) \sum_{i \in I_-} \lambda_i^2, \tag{10}$$

where the sets I_+ and I_- are defined by

$$I_+ = \{i \in [1, l] : 2rg_i(x) + \lambda_i \geq 0\},$$

$$I_- = \{i \in [1, l] : 2rg_i(x) + \lambda_i < 0\}.$$

Thus, the part of the Lagrangian (10) corresponding to indices $i \in I_+$ resembles the equality Lagrangian (2), while the part of the Lagrangian corresponding to indices $i \in I_-$ is locally independent of x.

Bertsekas observes that Theorem 2.1 also applies to inequality constrained problems, since an inequality is equivalent to an equality. In particular, (8) is equivalent to the problem

$$\text{minimize } f(x), \quad \text{subject to } g(x)+z^2=0, \ x \in R^n, \ z \in R^l, \tag{11}$$

where z^2 denotes the vector with components $(z^2)_i = z_i^2$, for $i = 1, \ldots, l$. This trick for converting an inequality constraint into an equality constraint is

found in Ref. 23. Letting $A(x)$ denote the active set defined by

$$A(x) = \{i \in [1, l]: g_i(x) = 0\},$$

we have the following lemma.

Lemma 2.1. Suppose that x^* is a local minimizer for (8), both f and g are twice continuously differentiable in a neighborhood of x^*, and the vectors $\nabla g_i(x^*)$ for $i \in A(x^*)$ are linearly independent. If $\lambda = \lambda^*$ satisfies the conditions

$$\nabla f(x^*) + \lambda^T \nabla g(x^*) = 0,$$

$$\lambda_i > 0, \quad \text{if } i \in A(x^*),$$

$$\lambda_i = 0, \quad \text{if } i \notin A(x^*),$$

and if $\nabla_x^2(f(x^*) + g(x^*)^T\lambda^*)$ is positive definite in the space of vectors orthogonal to $\nabla g_i(x^*)$ for every $i \in A(x^*)$, then the equality constrained problem (11) satisfies the assumptions of Theorem 2.1.

3. Local Algorithms

We now develop algorithms to solve constrained optimization problems. To simplify the discussion, these algorithms are formulated for equality constrained problems, although with slight modification these methods also apply to inequality constraints (see Section 5). Of course, in a neighborhood of an optimum, an inequality constrained problem can usually be treated as an equality constrained problem. Let us consider the mathematical program

$$\text{minimize } f(x), \quad \text{subject to } h(x) = 0, \; x \in R^n, \tag{12}$$

where $f: R^n \to R$ and $h: R^n \to R^m$. Let $\mathbf{L}$ denote the augmented Lagrangian defined by

$$\mathbf{L}(\lambda, x) = f(x) + \lambda^T h(x) + r|h(x)|^2.$$

Suppose that x^* is a local minimizer for (12) and $\lambda = \lambda^*$ satisfies (4). Letting x_k and λ_k be the kth approximation to x^* and λ^*, respectively, our first scheme for computing the $(k+1)$th approximation is the following algorithm.

Algorithm 3.1

Step 1. Compute a point y_k which satisfies $h(y_k) = 0$ using (for example) the Newton iteration

$$z_{j+1} = z_j - \nabla h(z_j)^{-1} h(z_j),$$

where the starting guess is $z_1 = x_k$ and $\nabla h(z_j)^{-1}$ denotes the pseudoinverse of $\nabla h(z_j)$.

Step 2. Let x_{k+1} be a solution to the linear equality constrained problem

$$\text{minimize } \{\mathbf{L}(\lambda_k, x): \nabla h(y_k)(x - y_k) = 0, x \in R^n\}. \tag{13}$$

Step 3. Set $\lambda_{k+1} = \lambda_k + 2rh(x_{k+1}) + \nu$, where ν is the vector of minimal Euclidean norm that satisfies

$$\nabla_x \mathbf{L}(\lambda_k, x_{k+1})^T + \nabla h(y_k)^T \nu = 0. \tag{14}$$

The vector ν generated in Step 3 is the multiplier associated with the Kuhn–Tucker conditions for (13). When the optimization problem (13) has more than one local minimizer, the x_{k+1} computed in Step 2 should be the local minimizer that is closest to x_k. Before analyzing Algorithm 3.1, let us review some facts about the pseudoinverse of a matrix. Consider the linear system $Ax = b$, where A is an $m \times n$ matrix and b is a vector in R^m. The pseudoinverse of A, denoted A^{-1}, is an $n \times m$ matrix with the property that $x = A^{-1}b$ is the vector with minimal Euclidean norm for which

$$|Ax - b| = \underset{y \in R^n}{\text{minimum}} |Ay - b|.$$

The pseudoinverse can be expressed using the singular-value decomposition. By the singular-value decomposition, A can be written as the product $Q\Lambda P$, where Q and P are square orthogonal matrices and Λ is an $m \times n$ matrix which is entirely zero except for its diagonal elements λ_{ii}, for $i = 1, \ldots, \min\{m, n\}$. Moreover, the ith diagonal element λ_{ii} is the square root of the ith largest eigenvalue of AA^T. It can be shown (see Ref. 24) that the pseudoinverse of A is given by

$$A^{-1} = P^{-1}\Lambda^{-1}Q^{-1} = P^T\Lambda^{-1}Q^T,$$

where the pseudoinverse of Λ is an $n \times m$ matrix which is entirely zero except for its diagonal. With the convention that $1/0$ is zero, the diagonal elements of Λ^{-1} are the reciprocal of the diagonal elements of Λ. If the rows of A are linearly independent, then the eigenvalues of AA^T are positive and the diagonal elements of Λ are positive. Hence, the formula

$$A^{-1} = P^T\Lambda^{-1}Q^T$$

implies that

$$AA^{-1} = I,$$

when the rows of A are linearly independent, even though A is not square. Observe that

$$(A^T)^{-1} = (P^T \Lambda^T Q^T)^{-1} = Q(\Lambda^T)^{-1} P = Q(\Lambda^{-1})^T P = (A^{-1})^T.$$

The matrix $(A^T)^{-1}$ is denoted A^{-T}.

The representation of the pseudoinverse in terms of the singular-value decomposition is a useful tool in analysis. However, there are more efficient ways to compute the pseudoinverse. Using the singular-value decomposition, it can be verified that

$$A^{-T} = (AA^T)^{-1} A. \tag{15}$$

But when the rows of A are linearly independent, AA^T is a nonsingular matrix and the pseudoinverse of AA^T is the ordinary inverse. Hence, (15) expresses the pseudoinverse of A^T in terms of the ordinary inverse. Also, the QR factorization can be used to compute both A^{-1} and A^{-T} efficiently. Assuming that $n \geq m$, the QR factorization implies that A can be written as

$$A = [L, 0]P = [L, 0]\begin{bmatrix} P_l \\ P_0 \end{bmatrix}, \tag{16}$$

where L is an $m \times l$ lower triangular matrix with linearly independent columns, 0 is an $m \times (n-l)$ matrix of zeros, and P is an $n \times n$ orthogonal matrix with its first l rows denoted P_l and with its last $n-l$ rows denoted P_0. It can be shown that

$$A^{-1} = P_l^T L^{-1}.$$

Since the columns of L are linearly independent, either L is nonsingular and L^{-1} is the ordinary inverse, or by (15) we have

$$L^{-1} = (L^T L)^{-1} L^T.$$

Similarly, the pseudoinverse of A^T can be written

$$A^{-T} = L^{-T} P_l.$$

In general, the vector ν, computed in Step 3 of Algorithm 3.1, is given by

$$\nu = -\nabla h(y_k)^{-T} \nabla_x \mathbf{L}(\lambda_k, x_{k+1})^T.$$

However, when the rows of $\nabla h(y_k)$ are linearly independent, it may be more convenient to express ν in the form

$$\begin{aligned} \nu &= -(\nabla h(y_k) \nabla h(y_k)^T)^{-1} \nabla h(y_k) \nabla_x \mathbf{L}(\lambda_k, x_{k+1})^T \\ &= -L^{-T} P_l \nabla_x \mathbf{L}(\lambda_k, x_{k+1})^T, \end{aligned}$$

where LP_l is the factorization of $A = \nabla h(y_k)$ given in (16). Efficient codes to compute either the singular-value decomposition or the QR factorization of a matrix are contained in LINPACK (Ref. 25).

Now, let us examine the convergence properties of Algorithm 3.1.

Theorem 3.1. Under the hypotheses of Theorem 2.1, there exists a neighborhood $\mathbf{N}$ of (x^*, λ^*) and there exists a constant c independent of (y_k, λ_k) in $\mathbf{N}$ such that

$$|\lambda_{k+1}-\lambda^*|+|x_{k+1}-x^*| \leq c|\lambda_k-\lambda^*|^2+c|y_k-x^*|^2+c|h(y_k)|, \tag{17}$$

where λ_{k+1} and x_{k+1} are generated by Steps 2 and 3 of Algorithm 3.1.

Proof. Since ν is equal to $\lambda_{k+1}-\lambda_k-2rh(x_{k+1})$, relation (14) can be written as

$$\nabla f(x_{k+1})^T+\nabla h(y_k)^T\lambda_{k+1}$$
$$+(\nabla h(x_{k+1})-\nabla h(y_k))^T(\lambda_k+2rh(x_{k+1}))=0. \tag{18}$$

Since x_{k+1} satisfies the constraints of (13), we have

$$\nabla h(y_k)(x_{k+1}-y_k)=0. \tag{19}$$

Defining the function $F: R^{n+m}\times R^{n+m}\to R^{n+m}$ by

$$F(\lambda, x, \mu, y)=\begin{bmatrix}\nabla f(x)^T+\nabla h(y)^T\lambda+(\nabla h(x)-\nabla h(y))^T(\mu+2rh(x))\\ \nabla h(y)(x-y)\end{bmatrix},$$

(18) and (19) are equivalent to the equation

$$F(\lambda_{k+1}, x_{k+1}, \lambda_k, y_k)=0.$$

Let us compute the gradient of F with respect to its first two arguments,

$$\nabla_{x,\lambda}F(\lambda^*, x^*, \lambda^*, x^*)=\begin{bmatrix}\nabla_x^2\mathbf{L}(\lambda^*, x^*) & \nabla h(x^*)^T\\ \nabla h(x^*) & 0\end{bmatrix}. \tag{20}$$

Since the rows of $\nabla h(x^*)$ are linearly independent and $\nabla_x^2\mathbf{L}(\lambda^*, x^*)$ is positive definite in the null space of $\nabla h(x^*)$, Ref. 26, Lemma 3.2 tells us that the Jacobian (20) is nonsingular for every $r\geq 0$. By the implicit function theorem, there exist neighborhoods $\mathbf{N}_1$ and $\mathbf{N}_2$ of (λ^*, x^*) such that the equation $F(\lambda, x, \mu, y)=0$ has a unique solution $(\lambda, x)=(\lambda(\mu, y), x(\mu, y))$ in $\mathbf{N}_1$ for every (μ, y) in $\mathbf{N}_2$. Moreover, by Ref. 18, Corollary 6.2, $\mathbf{N}_2$ can be chosen so that

$$|x(\mu, y)-x^*|+|\lambda(\mu, y)-\lambda^*|$$
$$\leq c|F(\lambda^*, x^*, \mu, y)-F(\lambda^*, x^*, \lambda^*, x^*)|, \tag{21}$$

for some constant c which is independent of $(\mu, y)\in\mathbf{N}_2$. By Ref. 18, Lemma 6.5, and by the second-order sufficiency condition (Ref. 27, page 226),

$x(\lambda_k, y_k)$ is the unique local minimizer for (13) near x^* when (λ_k, y_k) is near (λ^*, x^*). From the definition of F, we have

$$F(\lambda^*, x^*, \mu, y) - F(\lambda^*, x^*, \lambda^*, x^*)$$

$$= \begin{bmatrix} (\nabla h(x^*) - \nabla h(y))^T(\mu - \lambda^*) \\ \nabla h(y)(x^* - y) \end{bmatrix}. \tag{22}$$

The top component of (22) is estimated by a Taylor expansion:

$$|(\nabla h(x^*) - \nabla h(y))^T(\mu - \lambda^*)|$$

$$\leq |\mu - \lambda^*||\nabla h(x^*)^T - \nabla h(y)^T|$$

$$\leq (1/2)|\mu - \lambda^*|^2 + (1/2)|\nabla h(x^*)^T - \nabla h(y)^T|^2$$

$$\leq (1/2)|\mu - \lambda^*|^2 + (1/2)d^2|y - x^*|^2, \tag{23}$$

where

$$d = \left[\sum_{i=1}^{m} \underset{z_i \in [y, x^*]}{\text{maximum}} |\nabla^2 h_i(z_i)|^2 \right]^{1/2}.$$

Here, $[y, x^*]$ denotes the line segment connecting y and x^*. Similarly, expanding $h(x^*)$ in a Taylor series yields

$$0 = h(x^*) = h(y) + \nabla h(y)(x^* - y)$$

$$+ (1/2) \sum_{i=1}^{m} [(x^* - y)^T \nabla^2 h_i(\xi_i)(x^* - y)] e_i, \tag{24}$$

where e_i is the unit vector with every component zero except the ith component, which is one, and where ξ_i is a point on the line segment connecting y and x^*. Rearranging (24) and taking norms gives us

$$|\nabla h(y)(x^* - y)| \leq (1/2)d|y - x^*|^2 + |h(y)|. \tag{25}$$

Inserting $y = y_k$ and $\mu = \lambda_k$ and combining (21)–(25), the proof is complete. □

Remark 3.1. Since the x_{k+1} generated by Algorithm 3.1 is determined from y_k and λ_k, the inequality

$$|x_{k+1} - x^*| \leq c|\lambda_k - \lambda^*|^2 + c|y_k - x^*|^2 + c|h(y_k)|,$$

contained in (17), is valid regardless of how the multiplier is updated in Step 3 of Algorithm 3.1. Since

$$\lambda^* = -\nabla h(x^*)^{-T} \nabla f(x^*)^T,$$

by (4), an approximation λ_{k+1} to λ^* with accuracy comparable to the accuracy in x_{k+1} is given by the update

$$\lambda_{k+1} = -\nabla h(x_{k+1})^{-T} \nabla f(x_{k+1})^T.$$

The effect of the Newton iteration in Algorithm 3.1 is analyzed in the following theorem.

Theorem 3.2. Suppose that $x = x^*$ satisfies the equation $h(x) = 0$, h is twice continuously differentiable near x^*, and the rows of $\nabla h(x^*)$ are linearly independent. Then, there exists a neighborhood $\mathbf{N}$ of x^* and there exists a constant c such that, for every $x_0 \in \mathbf{N}$, the Newton iteration

$$x_{k+1} = x_k - \nabla h(x_k)^{-1} h(x_k), \tag{26}$$

based upon the pseudoinverse of $\nabla h(x_k)$, converges to a point y for which $h(y) = 0$ and

$$|x_k - y| \leq c|h(x_0)|(1/2)^{2^k}. \tag{27}$$

The fact that the iteration (26) converges to a solution of $h(x) = 0$ can be deduced from Ref. 28, Theorem 1. In the theorem stated above, we determine the convergence *rate* of the iteration. Using the terminology of Ortega and Rheinboldt (Ref. 29, page 290), (27) implies that the root convergence order of the iteration (26) is at least 2. This property will lead to a more practical implementation of Algorithm 3.1 (see Algorithm 3.2), where the Newton iterations are truncated and where the minimizer for (13) is approximated. Typically, the equation $h(x) = 0$ has an infinite number of solutions [otherwise, the optimization problem (12) is equivalent to minimizing over x in a finite set]. Consequently, the Newton iteration (26) usually does not converge to x^*.

Proof of Theorem 3.2. Let $\sigma(x)$ be the square root of the smallest eigenvalue of $\nabla h(x) \nabla h(x)^T$, and let $\mathbf{B}$ be a ball with center x^* and with radius so small that, inside $\mathbf{B}$, h is twice continuously differentiable and $\sigma(x)$ is uniformly bounded away from zero. That is, there exists a constant $\sigma_0 > 0$ such that $\sigma(x) \geq \sigma_0$ for every $x \in \mathbf{B}$. Since $h(x^*)$ is zero, $h(x)$ approaches zero as x approaches x^*. Let $x_0 \in \mathbf{B}$ be any point with the property that the ball with center x_0 and radius $2|h(x_0)|/\sigma_0$ lies in $\mathbf{B}$ and $d|h(x_0)| \leq \sigma_0^2$, where

$$d = \left[\sum_{i=1}^{m} \underset{z_i \in \mathbf{B}}{\text{maximum}} |\nabla^2 h_i(z_i)|^2 \right]^{1/2}.$$

Suppose that x_{k+1} and x_k lie in **B**, where x_{k+1} is given by (26). Expanding $h(x_{k+1})$ in a Taylor series about x_k yields

$$h(x_{k+1}) = h(x_k) + \nabla h(x_k)(x_{k+1} - x_k) + (1/2) \sum_{i=1}^{m} [(x_{k+1} - x_k)^T \nabla^2 h_i(\xi_i)(x_{k+1} - x_k)] e_i, \tag{28}$$

where ξ_i lies between x_{k+1} and x_k for each i and e_i is the unit vector with every component equal to zero, except for the ith component which is one. From the discussion of the pseudoinverse that precedes the statement of Theorem 3.1, we conclude that

$$\nabla h(x_k) \nabla h(x_k)^{-1} = I,$$

since the rows of $\nabla h(x_k)$ are linearly independent. Substituting for x_{k+1} using (26) and (28) gives

$$h(x_{k+1}) = (1/2) \sum_{i=1}^{m} [(\nabla h(x_k)^{-1} h(x_k))^T \nabla^2 h_i(\xi_i) \nabla h(x_k)^{-1} h(x_k)] e_i. \tag{29}$$

If $Q \Lambda P$ is the singular-value decomposition of $\nabla h(x)$ and $P^T \Lambda^{-1} Q^T$ is $\nabla h(x)^{-1}$, then, for the matrix norm induced by the Euclidean norm, we have

$$|\nabla h(x)^{-1}| = |\Lambda^{-1}| = 1/\sigma(x),$$

$$|\nabla h(x)^{-1} h(x)| \leq |h(x)|/\sigma(x).$$

It follows from (29) that

$$|h(x_{k+1})| \leq c|h(x_k)|^2, \qquad \text{where } c = d/2\sigma_0^2.$$

Suppose that $x_0, x_1, \ldots, x_k$ lie in **B**. Since

$$c|h(x_k)| \leq |ch(x_{k-1})|^2 \leq \cdots \leq |ch(x_0)|^{2^k},$$

(26) implies that

$$|x_{k+1} - x_k| \leq |h(x_k)|/\sigma_0 \leq |ch(x_0)|^{2^k}/c\sigma_0. \tag{30}$$

The triangle inequality and (30) yield

$$|x_{k+1} - x_0| \leq \sum_{j=0}^{k} |x_{j+1} - x_j| \leq (1/c\sigma_0) \sum_{j=0}^{k} |ch(x_0)|^{2^j} = [|h(x_0)|/\sigma_0] \sum_{j=0}^{k} |ch(x_0)|^{2^j - 1}. \tag{31}$$

By assumption,

$$c|h(x_0)| \leq 1/2.$$

Since the series

$$\sum_{j=0}^{\infty} (2/2^{2^j}) = 1+1/2+1/8+1/128+1/32768+\cdots$$

is bounded by 2, it follows from (31) that

$$|x_{k+1}-x_0| \leq 2|h(x_0)|/\sigma_0.$$

Since the ball with center x_0 and radius $2|h(x_0)|/\sigma_0$ is contained in **B**, we conclude that x_{k+1} lies in **B**. Hence, the entire sequence $\{x_k\}$ is contained in **B**. Similar to (31), it can be shown that the sequence $\{x_k\}$ is a Cauchy sequence which has a limit, say y. By the triangle inequality, (30), and the assumption $c|h(x_0)| \leq 1/2$, we have

$$\begin{aligned}
|x_k - y| &\leq \sum_{j=k}^{\infty} |x_{j+1}-x_j| \\
&\leq (1/c\sigma_0) \sum_{j=k}^{\infty} |ch(x_0)|^{2^j} \\
&\leq [|h(x_0)|/\sigma_0](1/2)^{2^{k-1}} \sum_{j=k}^{\infty} (1/2)^{2^j-2^k} \\
&\leq (4|h(x_0)|/\sigma_0](1/2)^{2^k},
\end{aligned}$$

which completes the proof. □

We now present a practical implementation of Algorithm 3.1 that employs a projected conjugate gradient scheme. Since one complete iteration of Algorithm 3.1 essentially squares the error, only one Newton iteration will be used in Step 1, and in Step 2 it is only necessary to find an approximation $\tilde{x}_{k+1}$ to x_{k+1} which satisfies

$$|x_{k+1}-\tilde{x}_{k+1}| \leq c|x_{k+1}-y_k|^2,$$

where c denotes a generic constant. Since $n-m$ iterations of the conjugate gradient method essentially squares the error, Algorithm 3.1 can be implemented in the following way.

Algorithm 3.2

Step 1. Set

$$y_k = x_k - \nabla h(x_k)^{-1} h(x_k),$$

and set

$$\lambda_k = -\nabla h(x_k)^{-T} \nabla f(x_k)^T.$$

Step 2. Apply $n-m$ projected conjugate gradient iterations to the function $\mathbf{L}(\lambda_k, \cdot)$, where the starting point is y_k and the gradient of $\mathbf{L}(\lambda_k, \cdot)$

is projected into the null space of $\nabla h(y_k)$ for each iteration. Let x_{k+1} denote the final conjugate gradient iteration.

Note that the multiplier update discussed in Remark 3.1 is used in Step 1. Of course, the conjugate gradient scheme in Step 2 can be replaced by a quasi-Newton scheme. The projection described in Step 2 can be constructed using either the singular-value decomposition $Q\Lambda P$ of $\nabla h(y_k)$ or the decomposition (16) of $A = \nabla h(y_k)$. In either case, the projection matrix H described in Step 2 is given by $H = P_0^T P_0$, where P_0 is the submatrix of P formed from the last $n-r$ rows and r is the rank of $\nabla h(y_k)$. Moreover, when the rows of $\nabla h(y_k)$ are linearly independent, the following representation of H may be convenient:

$$H = I - \nabla h(y_k)^T (\nabla h(y_k) \nabla h(y_k)^T)^{-1} \nabla h(y_k).$$

If the rows of $\nabla h(y_k)$ are linearly independent, then $\nabla h(y_k)\nabla h(y_k)^T$ is a nonsingular matrix and the pseudoinverse of $\nabla h(y_k)\nabla h(y_k)^T$ is the ordinary inverse. Explicitly, the projected conjugate gradient iteration of Algorithm 3.2 appears in (54).

Recall that each conjugate gradient iteration involves a one-dimensional minimization along a search direction. In stating our local convergence theorem for Algorithm 3.2, we assume implicitly that, in each conjugate gradient iteration, the closest local minimizer along the search direction is employed. Combining Theorem 3.1 and Theorem 3.2 with Cohen's quadratic convergence result (Ref. 30) for the conjugate gradient method, we have the following corollary.

Corollary 3.1. In addition to the hypotheses of Theorem 2.1, assume that f and h are three times continuously differentiable in a neighborhood of x^*. Then, for either the Daniel, Fletcher-Reeves, or Polak-Ribière formulations of the conjugate gradient method, there exists a neighborhood $\mathbf{N}$ of (x^*, λ^*) and a constant c independent of (x_k, λ_k) in $\mathbf{N}$ such that

$$|\lambda_{k+1} - \lambda^*| + |x_{k+1} - x^*| \leq c|\lambda_k - \lambda^*|^2 + c|x_k - x^*|^2, \tag{32}$$

where λ_{k+1} and x_{k+1} are generated either by Algorithm 3.1 or by Algorithm 3.2.

See Ref. 27 for a statement of various formulations of the conjugate gradient method and for a brief discussion of Cohen's result. For reference, we state the Fletcher-Reeves and the Polak-Ribière formulations of the conjugate gradient method in Section 4.

Proof. We just prove (32) for Algorithm 3.2, since the proof for Algorithm 3.1 is similar. Throughout the proof, c denotes a generic constant which is uniformly bounded in a neighborhood of (x^*, λ^*). Let $\tilde{x}_{k+1}$ and

$\tilde{\lambda}_{k+1}$ denote the output of Steps 2 and 3 of Algorithm 3.1, and let x_{k+1} denote the output of Step 2 of Algorithm 3.2. Theorem 3.1 tells us that

$$|\tilde{\lambda}_{k+1}-\lambda^*|+|\tilde{x}_{k+1}-x^*|\leq c|\lambda_k-\lambda^*|^2+c|y_k-x^*|^2+c|h(y_k)|. \tag{33}$$

Since y_k is obtained from x_k by one Newton iteration, the proof of Theorem 3.2 reveals that

$$|h(y_k)|\leq c|h(x_k)|^2\leq c|x_k-x^*|^2. \tag{34}$$

Also, it follows from (30) that

$$|y_k-x_k|\leq c|h(x_k)|\leq c|x_k-x^*|, \tag{35a}$$

$$|y_k-x^*|\leq c|y_k-x_k|+c|x_k-x^*|\leq c|x_k-x^*|. \tag{35b}$$

Combining (33)–(35) yields

$$|\tilde{\lambda}_{k+1}-\lambda^*|+|\tilde{x}_{k+1}-x^*|\leq c|\lambda_k-\lambda^*|^2+c|x_k-x^*|^2. \tag{36}$$

Cohen's quadratic convergence result (Ref. 30) for the conjugate method implies that

$$|x_{k+1}-\tilde{x}_{k+1}|\leq c|\tilde{x}_{k+1}-y_k|^2. \tag{37}$$

The triangle inequalities

$$|x_{k+1}-x^*|\leq|x_{k+1}-\tilde{x}_{k+1}|+|\tilde{x}_{k+1}-x^*|,$$

$$\begin{aligned}|\tilde{x}_{k+1}-y_k|^2&\leq(|\tilde{x}_{k+1}-x^*|+|y_k-x^*|)^2\\&\leq 2|\tilde{x}_{k+1}-x^*|^2+2|y_k-x^*|^2\end{aligned}$$

combine with (35), (36), (37) to show that x_{k+1} satisfies (32). Since

$$\lambda^*=-\nabla h(x^*)^{-T}\nabla f(x^*)^T, \qquad \lambda_{k+1}=-\nabla h(x_{k+1})^{-T}\nabla f(x_{k+1})^T,$$

it follows that λ_{k+1} also satisfies (32). □

4. Preconditioning

A common criticism of the penalty approach to constrained optimization is that, as the penalty tends to infinity, algorithms for solving the penalized problem convergence slowly. In fact, some of the excitement surrounding augmented Lagrangians stems from the observation that good approximations to the primal solution are generated without a large penalty (see Theorem 2.1). Techniques for handling the instabilities due to the penalty are presented in Chapter 12 of Ref. 27, where the Luenberger suggests both a modified Newton scheme and an alteration of the penalty term. We now develop a preconditioning technique to eliminate the instabilities.

First, let us describe the instabilities associated with the penalty scheme. For simplicity, we consider the quadratic program

$$\text{minimize } x^T Ax + a^T x, \quad \text{subject to } Bx = 0,\ x \in R^n, \tag{38}$$

where A is an $n \times n$ positive definite matrix and B is an $m \times n$ matrix. The penalty approximation to (38) is

$$\text{minimize } x^T (A + rB^T B)x + a^T x, \quad \text{subject to } x \in R^n. \tag{39}$$

And as r tends to infinity, the solution to (39) approaches the solution to (38). On the other hand, the convergence rate of gradient methods for solving (39) is governed by the ratio λ_n/λ_1 between the largest eigenvalue λ_n and the smallest eigenvalue λ_1 of $A + rB^T B$. For example, the steepest descent iteration x_k generated by exact line search satisfies the inequality (Ref. 27, page 152)

$$E(x_{k+1}) \leq [(\lambda_n - \lambda_1)/(\lambda_n + \lambda_1)]^2 E(x_k), \tag{40}$$

where $E: R^n \to R$ is defined by

$$E(x) = x^T (A + rB^T B)x.$$

Akaike (Ref. 31) has shown that, except for special starting points, the inequality (40) is nearly an equality as k tends to infinity if λ_n/λ_1 is much larger than 1. Applying Loewner's (Ref. 32) interlocking eigenvalue result, λ_1 is bounded from above by the second largest eigenvalue of A; and, since λ_n is the maximum of the Rayleigh quotient $E(x)/x^T x$ over nonzero $x \in R^n$, we conclude that λ_n is bounded from below by the product between r and the spectral radius of $B^T B$. Combining the upper bound for λ_1 with the lower bound for λ_n yields

$$(\lambda_n - \lambda_1)/(\lambda_n + \lambda_1) = 1 + O(r^{-1}).$$

The notation $O(r^{-1})$ means a term whose norm is bounded by c/r, where c is a constant that is independent of r for r sufficiently large. Since the ratio $(\lambda_n - \lambda_1)/(\lambda_n + \lambda_1)$ approaches one as r tends to infinity, steepest descent converges arbitrarily slowly as r tends to infinity.

The eigenvalue ratio can be improved by changing variables. Let H denote the matrix $(C + rB^T B)^{-1}$, where C is any symmetric, positive-definite matrix. Since C is positive definite, H is a positive-definite matrix which can be factored,

$$H = UU^T.$$

Of course, these factors are not unique—Gaussian elimination applied to H generates a lower triangular U while a symmetric positive-definite U is

constructed from the eigenvalues and the eigenvectors of H. Let us change from the variable x to the variable

$$y = U^{-1}x.$$

Expressing (39) in terms of y, we have

$$\text{minimize } y^T U^T (A + rB^T B) Uy + a^T Uy, \quad \text{subject to } y \in R^n. \tag{41}$$

Again, the convergence rate of steepest descent applied to (41) is governed by the ratio between the largest and the smallest eigenvalue of $U^T(A + rB^T B)U$. The eigenvalues of $U^T(A + rB^T B)U$ are the roots of the characteristic equation

$$\det(U^T(A + rB^T B)U - \lambda I) = 0. \tag{42}$$

Let U^{-T} denote the matrix $(U^T)^{-1}$. Multiplying the left side of (42) by $\det(U^{-T})$ and multiplying the right side of (42) by $\det(U^T)$ gives us the equivalent equation

$$\det((A + rB^T B)H - \lambda I) = 0. \tag{43}$$

In analyzing the product

$$(A + rB^T B)H = (A + rB^T B)(C + rB^T B)^{-1},$$

it helps to remove redundant rows from B. If the rank of B is r, then, by the QR factorization, there exists an orthogonal matrix Q such that $B = QR$, where the first r rows of R are linearly independent and the next $m - r$ rows are completely zero. Letting S be the submatrix of R formed by the first r rows, we have

$$B^T B = R^T Q^T QR = R^T R = S^T S.$$

Since

$$B^T B = S^T S$$

and the rows of S are linearly independent, there is no loss of generality in assuming that the rows of B are linearly independent.

Lemma 4.1. If C is a symmetric positive-definite matrix and the rows of B are linearly independent, then, for any A, we have

$$(A + rB^T B)(C + rB^T B)^{-1} = D + O(r^{-1}), \tag{44}$$

where

$$D = AC^{-1} + (I - AC^{-1})B^T(BC^{-1}B^T)^{-1}BC^{-1}.$$

Moreover, D is nonsingular if and only if

$$\underset{\substack{By=0 \\ \|y\|=1}}{\text{minimum}} \ \underset{\substack{Bx=0 \\ \|x\|=1}}{\text{maximum}} \ x^T Ay > 0. \tag{45}$$

Proof. The Woodbury formula (Ref. 24, page 3) tells us that $(C+rB^TB)^{-1}$ can be expressed as

$$(C+rB^TB)^{-1} = C^{-1} - C^{-1}B^T(\epsilon I + BC^{-1}B^T)^{-1}BC^{-1}, \tag{46}$$

where ϵ denotes r^{-1}. Recall that if Q is a nonsingular matrix and the spectral radius of Q^{-1} is less than one, then we can expand $(I-Q)^{-1}$ in a geometric series,

$$(I-Q)^{-1} = -(Q^{-1}+Q^{-2}+Q^{-3}+Q^{-4}+\cdots).$$

Furthermore, if $|Q^{-1}|<1$ for some matrix norm $|\cdot|$, then the partial sums in the expansion satisfy the relation

$$\left|(I-Q)^{-1} + \sum_{i=1}^{p} Q^{-i}\right| \leq |Q^{-1}|^{p+1}/[1-|Q^{-1}|].$$

Since the rows of B are linearly independent,

$$Q = rBC^{-1}B^T$$

is nonsingular and, for $\epsilon = 1/r$ sufficiently small, we have the geometric series expansion

$$(\epsilon I + BC^{-1}B^T)^{-1} = (BC^{-1}B^T)^{-1} - \epsilon(BC^{-1}B^T)^{-2} + O(\epsilon^2). \tag{47}$$

Combining (46) and (47) yields

$$\begin{aligned}
&(A+rB^TB)(C+rB^TB)^{-1}\\
&= r^1[B^TBC^{-1} - B^TBC^{-1}B^T(BC^{-1}B^T)^{-1}BC^{-1}]\\
&+ r^0[AC^{-1} + (I-AC^{-1})B^T(BC^{-1}B^T)^{-1}BC^{-1}]\\
&- r^{-1}(I-AC^{-1})B^T(BC^{-1}B^T)^{-2}BC^{-1} + O(r^{-2}).
\end{aligned}$$

Observe that the coefficient of the r^1 term is zero, while the coefficient of the r^0 term is D.

To prove the last claim in the lemma, we first assume that C is the identity matrix I. From the definition of D, it follows that

$$DB^T = B^T.$$

Let P be a matrix whose rows are a basis for the null space of B. Since C is I, it also follows from the definition of D that

$$DP^T = AP^T.$$

The vectors forming the rows of B combined with the vectors forming the rows of P are a basis for R^n. Hence, for any x in R^n, there exists y and z such that

$$x = B^Ty + P^Tz.$$

Utilizing the identities

$$DB^T = B^T, \qquad DP^T = AP^T,$$

we have

$$Dx = D(B^T y + P^T z) = B^T y + AP^T z. \tag{48}$$

We must prove that D is nonsingular if and only if PAP^T is nonsingular. Suppose that PAP^T is nonsingular. If $Dx = 0$, then (48) and the orthogonality condition $PB^T = 0$ implies that $PAP^T z = 0$, and z is zero since PAP^T is nonsingular. Furthermore, if z is zero, then by (48), $B^T y = 0$. Since the rows of B are linearly independent, we conclude that y is zero. Since both y and z are zero,

$$x = B^T y + P^T z$$

is zero and D is nonsingular. Conversely, suppose that D is nonsingular or, equivalently, suppose that the range of D is R^n. Since the rows of B are orthogonal to the rows of P and since the rows of B and the rows of P are a basis for R^n, (48) implies that the range of AP^T is P^T and PAP^T is nonsingular.

Now, let C be an arbitrary, symmetric, positive-definite matrix. Recall that a symmetric positive-definite matrix can be expressed as the square of a symmetric positive-definite matrix. That is, there exists a symmetric positive-definite matrix S such that

$$C = SS.$$

Replacing C by SS, the left side (44) can be written as

$$(A + rB^T B)(C + rB^T B)^{-1} = S(\tilde{A} + r\tilde{B}^T \tilde{B})(I + r\tilde{B}^T \tilde{B})^{-1} S^{-1}, \tag{49}$$

where

$$\tilde{B} = BS^{-1}, \qquad \tilde{A} = S^{-1} A S^{-1}.$$

Let $\tilde{D}$ be the same as D, except that C is replaced by I, A is replaced by $\tilde{A}$, and B is replaced by $\tilde{B}$. By (49),

$$D = S\tilde{D}S^{-1}.$$

Since S is positive definite, we conclude that D is nonsingular if and only if $\tilde{D}$ is nonsingular. However, it was just demonstrated that $\tilde{D}$ is nonsingular if and only if $\tilde{P}\tilde{A}\tilde{P}^T$ is nonsingular, where $\tilde{P}$ is the matrix whose rows are a basis for the null space of

$$\tilde{B} = BS^{-1}.$$

Since

$$\tilde{P} = PS,$$

it follows that

$$\tilde{P}\tilde{A}\tilde{P}^T = PAP^T,$$

which completes the proof. □

Remark 4.1. Observe that hypothesis (45) of the lemma is the same as hypothesis (7) involved in the analysis of an augmented Lagrangian. Also note that the matrix A in Lemma 4.1 is not necessarily symmetric or positive definite. Moreover, in the proof of (44), the crucial property is that $BC^{-1}B^T$ is nonsingular. Hence, (44) can be established in the following more general setting: Suppose that A and C are $n \times n$ matrices where C is nonsingular, and suppose that V and W are $m \times n$ matrices for which $WC^{-1}V^T$ is nonsingular. Then, we have

$$(A + rV^TW)(C + rV^TW)^{-1} = E + O(r^{-1}),$$

where

$$E = AC^{-1} + (I - AC^{-1})V^T(WC^{-1}V^T)^{-1}WC^{-1}.$$

Now, let us examine the convergence speed of gradient methods for solving (41) when

$$UU^T = H = (C + rB^TB)^{-1}.$$

If A is positive definite, then $A + rB^TB$ is positive definite. Since the eigenvalues of the product of positive-definite matrices are positive, the eigenvalues of $(A + rB^TB)(C + rB^TB)^{-1}$ are positive when both A and C are positive definite. If (45) holds, then the smallest eigenvalue λ_1 is bounded away from zero and the largest eigenvalue λ_n is bound away from infinity uniformly in r. Since λ_n grew with r before changing variables, the ill conditioning associated with large r has been eliminated. Clearly, the best choice for C is A. In this case, we have

$$H = (A + rB^TB)^{-1}, \qquad (A + rB^TB)H = I, \qquad \lambda_1 = \lambda_n = 1,$$

and by (40), steepest descent converges in one iteration. Of course, this one iteration is equivalent to solving the equation

$$2(A + rB^TB)x + a = 0,$$

which characterizes the solution to (39). For nonquadratic problems, the analog of A is a Hessian of the ordinary Lagrangian evaluated at the optimal solution. This Hessian is often difficult to evaluate and a more convenient C is the identity matrix I. With this choice for C, the Woodbury formula (46) implies that

$$\lim_{r \to \infty} (I + rB^TB)^{-1} = I - B^T(BB^T)^{-1}B.$$

In other words, as r tends to infinity,

$$H = (I + rB^TB)^{-1}$$

approaches the matrix that projects a vector into the null space of B.

To show that the preconditioner developed above also applies to nonquadratic problems, it must be demonstrated that preconditioning improves the eigenvalue structure for the Hessian of the augmented Lagrangian. Let $\mathbf{L}$ be the augmented Lagrangian introduced in (2), and let $\mathscr{D}$ denote the matrix defined by

$$\mathscr{D}(\lambda, x, y) = \nabla_x^2 \mathbf{L}(\lambda, y)(C + r\nabla h(x)^T \nabla h(x))^{-1}.$$

Lemma 4.2. Under the hypotheses of Theorem 2.1, there exist neighborhoods $\mathbf{N}_x$ of x^* and $\mathbf{N}_\lambda$ of λ^* (which depend on r) and there exists a constant c (which is independent of r for r sufficiently large) such that $\mathscr{D}(\lambda, x, y)$ is nonsingular and

$$|\mathscr{D}(\lambda, x, y)^{-1}| + |\mathscr{D}(\lambda, x, y)| \leq c,$$

for every x and y in $\mathbf{N}_x$ and for every λ in $\mathbf{N}_\lambda$.

Proof. $\mathscr{D}$ can be written in the form

$$\mathscr{D}(\lambda, x, y) = (A + E_1 + rB^TB)(C + E_2 + rB^TB)^{-1},$$

where

$$A = \nabla_x^2(f(x^*) + h(x^*)^T\lambda^*), \qquad B = \nabla h(x^*),$$

and E_1 and E_2 approach zero as x and y approach x^* and as λ approaches λ^*. Now, proceed as we did in the proof of Lemma 4.1. The terms generated by E_1 and E_2 can be made arbitrarily small by taking x and y near x^* and by taking λ near λ^*. □

We now state the preconditioned versions of some standard optimization algorithms. Of course, the notion of preconditioning is widely used in numerical analysis (especially in the context of numerical techniques for partial differential equations—see Ref. 33, pages 157–160). However, different authors seem to treat preconditioning (or scaling) in different ways. So, to be perfectly clear, we now state preconditioned versions of steepest descent, conjugate gradient, and quasi-Newton schemes. Let us consider the general unconstrained mathematical program

$$\text{minimize } \{f(x): x \in R^n\}. \tag{50}$$

We replace x by Uy to obtain the equivalent problem

$$\text{minimize } \{f(Uy): y \in R^n\}. \tag{51}$$

Each iteration of steepest descent is given by

$$y_{k+1} = y_k - \alpha_k U^T g_k, \tag{52}$$

where α_k is the step length, $g_k = \nabla f(x_k)^T$ is the gradient transpose at x_k, and $x_k = Uy_k$. Multiplying (52) by U gives us the iteration in terms of x_k:

$$x_{k+1} = x_k - \alpha_k H g_k,$$

where $H = UU^T$ is the *preconditioning matrix*. Since the standard steepest descent iteration for (50) is $x_{k+1} = x_k - \alpha_k g_k$, the effect of preconditioning is to replace g_k by Hg_k.

For problem (51), the Fletcher-Reeves conjugate gradient scheme is

$$y_{k+1} = y_k + \alpha_k g_k, \tag{53a}$$

$$q_{k+1} = -U^T g_{k+1} + \beta_k q_k, \tag{53b}$$

where

$$\alpha_k = \arg\min_{\alpha \geq 0} f(Uy_k + \alpha U q_k),$$

$$\beta_k = g_{k+1}^T UU^T g_{k+1} / g_k^T UU^T g_k.$$

Efficient schemes for computing α_k appear in Refs. 34 and 35. The initial search direction q_1 is the negative gradient $-U^T g_1$. Multiplying (53) by U and defining $p_k = Uq_k$, we obtain the conjugate gradient iteration in terms of x,

$$x_{k+1} = x_k + \alpha_k p_k, \tag{54a}$$

$$p_{k+1} = -Hg_{k+1} + \beta_k p_k, \tag{54b}$$

where

$$\alpha_k = \arg\min_{\alpha \geq 0} f(x_k + \alpha p_k),$$

$$\beta_k = g_{k+1}^T Hg_{k+1} / g_k^T Hg_k,$$

and the initial search direction p_1 is $-Hg_1$. The Polak-Ribière formulation (Ref. 36, page 54) of the conjugate gradient method just differs from the Fletcher-Reeves formulation in the choice for β_k. Corresponding to Polak and Ribière's choice, we have

$$\beta_k = (g_{k+1} - g_k)^T Hg_{k+1} / g_k^T Hg_k.$$

Finally, let us consider the Davidon-Fletcher-Powell quasi-Newton method (see refs. 37 and 38). The DFP iteration applied to (50) is

$$x_{k+1} = x_k - \alpha_k S_k g_k, \tag{55a}$$

$$S_{k+1} = S_k - \alpha_k (S_k g_k g_k^T S_k / g_k^T S_k q_k) - S_k q_k q_k^T S_k / q_k^T S_k q_k, \tag{55b}$$

where the starting matrix S_1 is symmetric and positive definite,

$$q_k = g_{k+1} - g_k,$$

$$\alpha_k = \arg\min_{\alpha \geq 0} f(x_k - \alpha S_k g_k).$$

Applying the DFP iteration to (51) and expressing y in terms of x as we did in the previous examples, it can be shown that the preconditioned iteration is identical to the iteration (55). The effect of preconditioning is to replace the starting matrix S_1 by US_1U^T. In particular, if S_1 is the identity matrix, then the starting matrix for the preconditioned scheme is

$$UU^T = H.$$

5. Global Algorithms

Algorithm 3.1 and Algorithm 3.2 are locally convergent. That is, they converge when the starting guess is sufficiently near a local minimizer for (12). We now modify these algorithms to obtain globally convergent schemes. For completeness, these globally convergent algorithms are stated in the context of a general optimization problem containing both equality and inequality constraints:

$$\text{minimize } f(x), \tag{56a}$$

$$\text{subject to } g_i(x) \leq 0, \quad i = 1 \text{ to } l, \tag{56b}$$

$$g_i(x) = 0, \quad i = l+1 \text{ to } m, \; x \in R^n, \tag{56c}$$

where $f: R^n \to R$ and $g_i: R^n \to R$, for $i = 1$ to m. We employ Rockafellar's augmented Lagrangian which is given by

$$\mathbf{L}(\lambda, x) = f(x) + \sum_{i \in I_+} [\lambda_i g_i(x) + r g_i(x)^2] - (1/4r) \sum_{i \in I_-} \lambda_i^2,$$

where the sets I_+ and I_- are defined by

$$I_+(\lambda, x) = \{i \in [1, l]: 2r g_i(x) + \lambda_i \geq 0\} \cup \{l+1, \ldots, m\},$$

$$I_-(\lambda, x) = \{i \in [1, l]: 2r g_i(x) + \lambda_i < 0\}.$$

Recall that λ and x satisfy the Kuhn–Tucker conditions for (56) if

$$g_i(x) \leq 0, \qquad \lambda_i \geq 0, \qquad i = 1 \text{ to } l,$$

$$g_i(x) = 0, \qquad i = l+1 \text{ to } m,$$

$$\lambda^T g(x) = 0, \qquad \nabla f(x) + \lambda^T \nabla g(x) = 0.$$

Any x and λ which satisfies the Kuhn-Tucker conditions will be called a Kuhn-Tucker point for (56). We monitor convergence to a point which satisfies the Kuhn-Tucker conditions using the function $K: R^{m+n} \to R$ defined by

$$K(\lambda, x) = [|\lambda_-|^2/4r^2 + |g_+(x)|^2 + |\nabla_x \mathbf{L}_0(\lambda, x)|^2]^{1/2}. \tag{57}$$

Here, $\lambda_\pm$ and $g_\pm(x)$ denote the components of λ and $g(x)$ corresponding to indices $i \in I_\pm(\lambda, x)$, and $\nabla_x \mathbf{L}_0(\lambda, x)$ is defined by

$$\nabla_x \mathbf{L}_0(\lambda, x) = \nabla f(x) + \sum_{i=1}^{m} \lambda_i \nabla g_i(x).$$

Some properties of K are established in the following proposition.

Proposition 5.1. The zeros of the function K defined in (57) are the Kuhn-Tucker points of (56). If f and g are continuously differentiable at y, then $K(\lambda, x)$ is continuous at $x = y$ and $\lambda = \mu$ for any $\mu \in R^m$.

Proof. Clearly, a Kuhn-Tucker point for (56) is a zero of K. Conversely, suppose that $K(\lambda, x) = 0$. Since $\lambda_- = 0$ and $g_+(x) = 0$, it follows from the definition of $\lambda_\pm$ and $g_\pm$ that $g_-(x) < 0$ and $\lambda_+ \geq 0$. Since $\nabla_x \mathbf{L}_0(\lambda, x) = 0$, we conclude that (λ, x) is a Kuhn-Tucker point for (56). To prove the second part of the proposition, observe that

$$\sum_{i=1}^{l} (g_i(x) + z_i)^2 + \sum_{i=1+1}^{m} g_i(x)^2 = |\lambda_-|^2/4r^2 + |g_+(x)|^2, \tag{58}$$

when z_i is the solution to

$$\underset{z_i \geq 0}{\text{minimize}}\ \lambda_i(g_i(x) + z_i) + r(g_i(x) + z_i)^2. \tag{59}$$

The z_i which attains the minimum in (59) depends on both λ and x and will be denoted $z_i(\lambda, x)$. From the definition of z_i, it follows that

$$|z_i(\lambda, x) - z_i(\mu, y)| \leq |g_i(x) - g_i(y) + (\lambda_i - \mu_i)/2r|.$$

Therefore, $z(\lambda, x)$ is a continuous function of λ and x when $g(x)$ is a continuous function of x. Hence, (58) implies that $|\lambda_-|^2 + |g_+(x)|^2$ is a continuous function of λ and x when $g(x)$ is a continuous function of x. Since K is the square root of the sum $|\lambda_-|^2 + |g_+(x)|^2 + |\nabla_x \mathbf{L}_0(\lambda, x)|^2$, it follows that K is continuous when f and g are continuously differentiable. □

There are many other ways of defining a function like K which just vanishes at Kuhn-Tucker points for (56). For example, an alternative definition is obtained by substituting $\mathbf{L}$ for $\mathbf{L}_0$ in the definition of K. We now show that K measures the distance to a Kuhn-Tucker point in the neighborhood of that point.

Proposition 5.2. Let λ^* and x^* denote a Kuhn–Tucker point for (56). Suppose that both f and g_+ are twice Fréchet differentiable at x^*, g is continuous at x^*, $\lambda_i^* > 0$ for $i \in I_+(\lambda^*, x^*) \cap [1, l]$, and (7) is satisfied when h is identified with g_+ and λ is identified with λ_+. Then, there exists a neighborhood $\mathbf{N}$ of (λ^*, x^*) and there exist positive constants c_1 and c_2 such that

$$c_1 K(\lambda, x) \leq |\lambda - \lambda^*| + |x - x^*| \leq c_2 K(\lambda, x),$$

for every $(\lambda, x) \in \mathbf{N}$.

Proof. Suppose that $F: R^N \to R^N$, $F(z^*) = 0$ for some $z^* \in R^N$, F is differentiable at z^*, and the matrix $\nabla F(z^*)$ is nonsingular. Note that

$$[1/|\nabla F(z^*)^{-1}|]|z - z^*| \leq |\nabla F(z^*)(z - z^*)| \leq |\nabla F(z^*)|\,|(z - z^*)|. \tag{60}$$

Let Δ be defined by

$$\Delta(z) = F(z) - F(z^*) - \nabla F(z^*)(z - z^*).$$

Given by $\epsilon > 0$, there exists a neighborhood $\mathbf{N}$ of z^* with the property that

$$|\Delta(z)| \leq \epsilon |z - z^*|, \quad \text{whenever } z \in \mathbf{N}.$$

The identity

$$F(z) = \Delta(z) + \nabla F(z^*)(z - z^*)$$

and (60) imply that

$$|F(z)|/[\epsilon + |\nabla F(z^*)|] \leq |z - z^*| \leq |F(z)|\,|\nabla F(z^*)^{-1}|/[1 - \varepsilon |\nabla F(z^*)|]. \tag{61}$$

We apply (61), where z is identified with the pair (λ, x) and where the equation $F(z) = 0$ is identified with the following system:

$$\nabla f(x) + \lambda^T \nabla g(x) = 0,$$

$$g_i(x) = 0, \qquad i \in I_+(\lambda^*, x^*),$$

$$\lambda_i / 2r = 0, \qquad i \in I_-(\lambda^*, x^*).$$

Since

$$\lambda_i^* > 0, \qquad \text{for } i \in I_+(\lambda^*, x^*) \cap [1, l],$$

we conclude that

$$I_+(\lambda, x) = I_+(\lambda^*, x^*),$$

for (λ, x) in a neighborhood of (λ^*, x^*). As a consequence,

$$K(\lambda, x) = |F(\lambda, x)|,$$

for (λ, x) in a neighborhood of (λ^*, x^*). By Ref. 39, Proposition 1.1, the gradient of F evaluated at (λ^*, x^*) is nonsingular. Since the hypotheses used in the derivation of (61) are fulfilled, the proof is complete. □

Before stating the first of our globally convergent algorithms, we need some notation. Let $\mathbf{L}_y(\lambda, x)$ be the same as $\mathbf{L}(\lambda, x)$, except that $I_+(\lambda, x)$ is replaced by the set $I_+(\lambda, x) \cup I_+(\lambda, y)$ and $I_-(\lambda, x)$ is replaced by the set $I_-(\lambda, x) \cap I_-(\lambda, y)$. That is,

$$\mathbf{L}_y(\lambda, x) = f(x) + \sum_{i \in I_+(\lambda,z) \cup I_+(\lambda,y)} [\lambda_i g_i(x) + r g_i(x)^2] - (1/4r) \sum_{i \in I_-(\lambda,x) \cap I_-(\lambda,y)} \lambda_i^2.$$

Let $B(\lambda, x)$ denote the active constraint matrix $\nabla g_+(x)$. The active constraint matrix depends upon both λ and x, since the indices in I_+ depend upon both λ and x. If x_k and λ_k denote the kth approximation to a solution and a corresponding multiplier for (56), then the matrix $B(\lambda_k, x_k)$ is abbreviated B_k, the matrix $B(\lambda_{k-1}, x_k)$ is abbreviated B_{k-}, and the Lagrangian $\mathbf{L}_{x_k}$ is abbreviated $\mathbf{L}_k$. Let π_k denote a vector with one component corresponding to each index $i \in I_+(\lambda_{k-1}, x_k)$, and let ν_k denote a vector with one component corresponding to each index $i \in I_-(\lambda_{k-1}, x_k)$. Let λ_0 and x_1 denote the starting approximation to a Kuhn–Tucker point for (56), and assign the parameter β the initial value $K(\lambda_0, x_1)$. The following rules are used to generate x_{k+1} and λ_k from x_k and λ_{k-1}.

Algorithm 5.1

Step 1. Set $\nu_k = 0$, set $\pi_k^T = -\nabla f(x_k) B_{k-}^{-1}$, and let λ be the m component vector formed from the components of π_k and ν_k. If $K(\lambda, x_k) \leq \beta/2$, then go to Step 3 after making the assignments $\beta \leftarrow K(\lambda, x_k)$ and $\lambda_k \leftarrow \lambda$. Otherwise, make the assignment $\lambda_k \leftarrow \lambda_{k-1}$ and proceed to Step 2.

Step 2. If $|\nabla_x \mathbf{L}(\lambda_k, x_k)| \leq |g_+(x_k)| + |(\lambda_k)_-|/2r$, then $r \leftarrow 10r$.

Step 3. Apply n iterations of the preconditioned conjugate gradient method to the function $\mathbf{L}_k(\lambda_k, \cdot)$, where the iteration starting point is x_k and the preconditioning matrix is $(I + rB_k^T B_k)^{-1}$. If a nonbinding constraint becomes binding after a conjugate gradient iteration, then stop the iterations and x_{k+1} is the latest approximation to x^*. Otherwise, x_{k+1} is the point generated by the complete set of n conjugate gradient iterations.

The factor 1/2 in Step 1 and the factor 10 in Step 2 are somewhat arbitrary. Also, the conjugate gradient method can be replaced by a quasi-Newton scheme. With regard to Step 3, the phrase "nonbinding constraint

i becomes binding at y" means that i lies in both $I_-(\lambda_k, x_k)$ and $I_+(\lambda_k, y)$. By the Woodbury formula (46), the preconditining matrix employed in Step 3 can be expressed as

$$H = I - B_k^T(r^{-1}I + B_k B_k^T)^{-1} B_k.$$

Also in Step 3, we minimize $\mathbf{L}_k(\lambda_k, \cdot)$ instead of $\mathbf{L}(\lambda_k, \cdot)$ to prevent the chattering or zigzagging phenomenon that occurs when a constraint oscillates between binding and nonbinding. As will be seen shortly, $\mathbf{L}_k$ can be replaced by $\mathbf{L}$ in Step 3 of Algorithm 5.1 without affecting the global convergence. On the other hand, it is observed in practice that the convergence in Step 3 can be slow when $\mathbf{L}_k$ is replaced by $\mathbf{L}$, since constraints oscillate between binding and nonbinding. The preconditioning aspect of Algorithm 5.1 is crucial. For the problem of optimally coating a surface discussed in Ref. 40, Algorithm 5.1 exhibits practically no convergence unless the preconditioner is included in Step 3.

Algorithm 5.1 is similar to Algorithm 3.2, since the same estimate is used for the multiplier and nearly the same preconditioning matrix is employed. Algorithm 5.1 differs from Algorithm 3.2 in two respects. First, the constraint feasibility Step 1 of Algorithm 3.2 is omitted, and feasibility is enforced through the penalty term in the cost functional. Second, the multipliers generated in Step 1 are only utilized if they yield a smaller value for K. In analyzing the global convergence of Algorithm 5.1, it is assumed that the iterations are well defined; that is, f and g are differentiable at each point generated by the algorithm, and the one-dimensional minimizer computed in each iteration of the conjugate gradient method exists. It is also assumed that a convergent subsequence of the iterations can be extracted. Often, it can be deduced from the nature of the cost or from the nature of the constraints that the iterations are bounded and a convergent subsequence of the iterations can be extracted. If the iterations are unbounded, then Algorithm 5.1 can be modified slightly to keep the iterations bounded. For example, if the update $\lambda_k \leftarrow \lambda$ in Step 1 is only performed when $|\lambda|$ is smaller than some predetermined bound [hopefully larger than the norm of the multiplier associated with the desired solution of (56)], then the λ_k are uniformly bounded. To keep the x_k bounded, add an extra inequality constraint of the form

$$g_0(x) := a(|x|^2 - \rho^2) \leq 0,$$

where ρ is a positive constant chosen so large that the desired solution of (56) has magnitude less than ρ and a is another positive constant. Justification for this strategy to keep the x_k bounded appears in Lemma 5.4. The global convergence proof for Algorithm 5.1 is partitioned into three cases.

Lemma 5.1. Let $\{(\mu_k, y_k)\}$ denote any convergent subsequence of the iterations $\{(\lambda_k, x_k)\}$ generated by Algorithm 5.1, and let (x^*, λ^*) denote the subsequence limit. If f and g are continuously differentiable at x^*, the penalty parameter r generated by Algorithm 5.1 is uniformly bounded, and $K(\mu_k, y_k)$ tends to zero, then x^* and λ^* satisfy the Kuhn–Tucker conditions for (56).

Proof. This follows immediately from Proposition 5.1. □

Lemma 5.2. Suppose that $K(\lambda_k, x_k)$ is bounded away from zero for the iterations generated by Algorithm 5.1. Let $\{(\mu_k, y_k)\}$ denote any convergent subsequence of the iterations and let (λ^*, x^*) denote the subsequence limit. If f and g are twice continuously differentiable in a neighborhood of x^*, then the penalty parameter r generated by Algorithm 5.1 tends to infinity as k increases.

Proof. We assume that r is uniformly bounded and we show that this leads to a contradiction. Since $K(\lambda_k, x_k)$ is bounded away from zero, the update $\lambda_k \leftarrow \lambda$ in Step 1 of Algorithm 5.1 is performed just a finite number of times. Without loss of generality, we can assume that $\lambda_k = \lambda$ for every k and that the same penalty r is used for each k. By the definition of $\mathbf{L}_k$, it follows that $\mathbf{L}_k(\lambda, \cdot) \geq \mathbf{L}(\lambda, \cdot)$. Also, since each step of the conjugate gradient method involves a one-dimensional minimization, we conclude that

$$\mathbf{L}(\lambda, x_k) = \mathbf{L}_k(\lambda, x_k) \geq \mathbf{L}_k(\lambda, x_{k+1}) \geq \mathbf{L}(\lambda, x_{k+1}) = \mathbf{L}_{k+1}(\lambda, x_{k+1}). \tag{62}$$

Suppose that $\nabla_x \mathbf{L}_k(\lambda, y_k)$ is bounded away from zero. Since f and g are twice continuously differentiable near x^*, it follows (see Ref. 41, Corollary A.6) that, in a neighborhood of x^*, the second derivative of $\mathbf{L}_k(\lambda, \cdot)$ exists almost everywhere and is essentially bounded. Since the first iteration of the conjugate gradient method is a steepest descent step, there exists $\epsilon > 0$ such that

$$\mathbf{L}_k(\lambda, x_k) \geq \mathbf{L}_k(\lambda, x_{k+1}) + \epsilon, \tag{63}$$

when $x_k = y_j$ and j is sufficiently large. Since $\mathbf{L}(\lambda, x_k)$ is bounded from below by $\mathbf{L}(\lambda, x^*)$, (62) and (63) combine to yield a contradiction. Therefore, $\nabla_x \mathbf{L}_k(\lambda, y_k)$ approaches zero as k increases. Since

$$\nabla_x \mathbf{L}_k(\lambda, y_k) = \nabla_x \mathbf{L}(\lambda, y_k),$$

we conclude that $\nabla_x \mathbf{L}(\lambda, y_k)$ approaches zero as k increases. By assumption, the same penalty is used in each iteration. Thus Step 2 of Algorithm 5.1 implies that

$$\nabla_x \mathbf{L}(\lambda, y_k) > |g_+(y_k)| + |\lambda_-|/2r, \tag{64}$$

for each k. Since the left side of (64) approaches zero as k increases, we conclude that $|g_+(y_k)|$ approaches zero as k increases and $\lambda_i = 0$, if $i \in I_-(\lambda, y_k)$ and k is sufficiently large. Therefore, $\nabla_x \mathbf{L}(\lambda, y_k)$ approaches $\nabla_x \mathbf{L}_0(\lambda, y_k)$ as k increases. Since $\nabla_x \mathbf{L}_0(\lambda, y_k)$ approaches zero, we conclude that $K(\lambda, y_k)$ tends to zero as k increases. This contradicts the lemma's assumption that $K(\lambda_k, x_k)$ is bounded away from zero. □

If the penalty generated by Algorithm 5.1 tends to infinity, then the inequality contained in Step 2 is satisfied an infinite number of times. Given any infinite sequence $\{(\lambda_k, x_k)\}$ of iterations generated by Algorithm 5.1, also note that a subsequence, say $\{(\mu_k, y_k)\}$, can be chosen so that $I_\pm(\mu_i, y_i) = I_\pm(\mu_j, y_j)$ for every i and j.

Lemma 5.3. Suppose that the penalty generated by Algorithm 5.1 tends to infinity as k increases, and let $\{y_k\}$ denote any convergent subsequence of the iterations with the property that $I_+(\mu_k, y_k)$ is equal to a fixed set I_+ for every k and

$$|\nabla_x \mathbf{L}(\mu_k, y_k)| \leq |g_+(y_k)| + |(\mu_k)_-|/2r_k, \tag{65}$$

for every k. Here μ_k denotes the multiplier approximation corresponding to y_k and r_k is the associate penalty. If x^* denotes the limit of the y_k, f and g are continuously differentiable at x^*, the vectors $\nabla g_i(x^*)$ for $i \in I_+$ are linearly independent, and μ_k is uniformly bounded, then the quantity $(\mu_k)_+ + 2rg_+(y_k)$ approaches a limit, say λ_+^*. Moreover, if λ^* is the vector formed from λ_+^* by inserting zeros for those components corresponding to indices $i \notin I_+$, then λ^* and x^* satisfy the Kuhn–Tucker conditions for (56).

Proof. Applying the inequality

$$|a+b| \geq |a| - |b|$$

to the left side of (65), we obtain the relation

$$|B_k^T((\mu_k)_+ + 2r_k g_+(y_k))| \leq |\nabla f(y_k)| + |g_+(y_k)| + |(\mu_k)_-|/2r_k.$$

By our hypotheses, the row of B_k are uniformly linearly independent for k sufficiently large. Therefore, there exists a positive constant γ such that

$$2r_k\gamma|g_+(y_k)| \leq \gamma|(\mu_k)_+| + |\nabla f(y_k)| + |g_+(y_k)| + |(\mu_k)_-|/2r_k, \tag{66}$$

for k sufficiently large. Since r_k tends to infinity while the ratio $|\mu_k|/2r_k$ tends to zero, (66) implies that $g_+(y_k)$ tends to zero as k increases. From the continuity of g, it follows that

$$g_+(x^*) = 0.$$

Since

$$2r_k g_i(y_k) + \mu_{k,i} < 0, \qquad \text{for } i \in I_-,$$

and since μ_k/r_k tends to zero, we also conclude that

$$g_i(x^*) \leq 0, \qquad \text{for every } i \in I_-.$$

Since the right side of (65) approaches zero and the rows of $\nabla g_+(x^*)$ are linearly independent, $|\nabla_x \mathbf{L}(\mu_k, y_k)|$ approaches zero and the nonnegative vector $(\mu_k)_+ + 2r_k g_+(x_k)$ approaches a limit λ_+^*. If λ^* is the vector formed from λ_+^* by inserting zeros for those components corresponding to indices $i \in I_-$, then the vectors λ^* and x^* satisfy the Kuhn-Tucker conditions for (56). □

Now, let us consider the local convergence of Algorithm 5.1. Since the nonbinding inequality constraints can be dropped in a neighborhood of an optimum, we assume for simplicity that $l = 0$. In stating this local convergence result, it is implicitly assumed that in each conjugate gradient iteration, we employ the closest local minimizer along the search direction.

Theorem 5.1. Suppose that $l = 0$ and the hypotheses of Theorem 2.1 are satisfied, where the h of Theorem 2.1 is identified with the g of (56). If f and g are three times continuously differentiable in a neighborhood of x^*, then for either the Daniel, Fletcher-Reeves, or Polak-Ribière formulations of the conjugate gradient method, there exists a neighborhood $\mathbf{N}$ of x^* (this neighborhood depends on r) and there exists a constant c independent of x_k in $\mathbf{N}$ and independent of r for r sufficiently large such that

$$|\lambda_{k+1} - \lambda^*| + |x_{k+1} - x^*| \leq c|x_k - x^*|(|x_k - x^*| + r^{-1}), \tag{67}$$

provided

$$\lambda_k^T = \nabla f(x_k) B_k^{-1}, \qquad \lambda_{k+1}^T = \nabla f(x_{k+1}) B_{k+1}^{-1}.$$

Proof. Throughout the proof, c denotes a generic constant which is bounded uniformly in a neighborhood of x^*. Theorem 2.1 asserts the existence of a unique local minimizer y_k for $\mathbf{L}(\lambda_k, \cdot)$ which satisfies the inequality

$$|y_k - x^*| \leq c|\lambda_k - \lambda^*|/r,$$

when r is sufficiently large. By Cohen's quadratic convergence result (Ref. 30) for the conjugate gradient method, we have

$$|x_{k+1} - y_k| \leq c|x_k - y_k|^2. \tag{68}$$

The c in (68) depends on upper and lower bounds for the eigenvalues of the preconditioned Hessian of the augmented Lagrangian. By Lemma 4.2, these eigenvalues are bounded away from zero and away from infinity uniformly in r for some neighborhood of x^*. Therefore, the c in (68) is

independent of r for a neighborhood of (λ^*, x^*). From the definition of λ_k and λ_{k+1}, we have

$$|\lambda_k - \lambda^*| \leq c|x_k - x^*|,$$

$$|\lambda_{k+1} - \lambda^*| \leq c|x_{k+1} - x^*|.$$

Combining these inequalities, the proof is complete. □

Earlier, we noted that the x_k can be kept in a bounded set if an extra inequality of the form

$$g_0(x) := a(|x|^2 - \rho^2) \leq 0$$

is added to the constraints. Let us now expand upon this observation. The goal in Step 3 of Algorithm 5.1 is to minimize over x the expression

$$C_k(\lambda_k, x) = (1/r)L_k(\lambda_k, x) = \sum_{i \in I_{k+}(x)} g_i(x)^2 + O(r^{-1}), \tag{69}$$

where

$$I_{k+}(x) = I_+(\lambda_k, x_k) \cup I_+(\lambda_k, x).$$

Since Step 3 is a descent step, it tends to reduce the constraint violation until the summation appearing in (69) is comparable in size to the adjacent $O(r^{-1})$ term. To show that some descent is possible even with preconditioning, let us consider the first conjugate gradient iteration performed in Step 3. This first iteration is a steepest descent step which minimizes $C_k(\lambda_k, x_k + \alpha d)$ over $\alpha \geq 0$, where

$$d = -H(p_k + 2r_k B_k^T g_+(x_k)), \tag{70a}$$

$$p_k^T = \nabla f(x_k) + \lambda_k^T B_k, \tag{70b}$$

$$H = (I + rB_k^T B_k)^{-1}. \tag{70c}$$

Lemma 5.4. If f and g are differentiable at x_k and the rows of B_k are linearly independent, then we have

$$(d/d\alpha)C_k(\lambda_k, x_k + \alpha d)|_{\alpha=0} = -4|g_+(x_k)|^2 + O(r^{-1}). \tag{71}$$

Moreover, the norm of the direction d is bounded by a constant which depends on f and g, but does not depend on r.

Proof. From the definition of d, it follows that

$$\begin{aligned}&(d/d\alpha)C_k(\lambda_k, x_k + \alpha d_k)|_{\alpha=0}\\ &= (r^{-1}p_k + 2B_k^T g_+(x_k))^T d\\ &= -(r^{-1}p_k + 2B_k^T g_+(x_k))^T H(p_k + 2rB_k^T g_+(x_k)).\end{aligned} \tag{72}$$

Applying the Woodbury formula to H and expanding the resulting inverse as we did in the proof of Lemma 4.1 yields

$$H = I - B_k^T(B_k B_k^T)^{-1}B_k + (1/r)B_k^T(B_k B_k^T)^{-2}B_k + O(r^{-2}). \tag{73}$$

Combining (72) and (73) yields (71). Finally, inserting the expansion (73) into (70), we see that d is bounded by a constant that depends on f and g, but which is independent of r. □

By Lemma 5.4, the first conjugate gradient iteration in Step 3 of Algorithm 5.1 reduces the value of $C_k(\lambda_k, \cdot)$ by $\epsilon(|g_+(x_k)|^2 + O(r^{-1}))$, where $\epsilon > 0$ and the size of ϵ depends on the second derivatives of f and g near x_k. As these second derivatives become large, ϵ becomes small. We now improve Algorithm 5.1 in two ways. First, the number of conjugate gradient iterations will be reduced by the number of binding constraints. Second, a Newton step will be inserted to obtain quadratic convergence. Again, the parameter β is assigned the initial value $K(\lambda_0, x_1)$.

Algorithm 5.2

Step 1. Set $x = x_k - B_k^{-1} g_+(x_k)$, set $\nu_k = 0$, set $\pi_k^T = -\nabla f(x_k) B_k^{-1}$, and let λ be the m component vector formed from the components of π_k and ν_k. If $K(\lambda, x) \leq \beta/2$, then go to Step 3 after making the assignments $\beta \leftarrow K(\lambda, x)$, $\lambda_k \leftarrow \lambda$, $x_k \leftarrow x$, $H \leftarrow$ the matrix that projects a vector into the null space of B_k, and $b \leftarrow$ the number of elements in $I_+(\lambda_k, x_k)$. Otherwise, make the assignments $\lambda_k \leftarrow \lambda_{k-1}$, $H \leftarrow (I + rB_k^T B_k)^{-1}$, $b \leftarrow 0$, and proceed to Step 2.

Step 2. If $|\nabla_x \mathbf{L}(\lambda_k, x_k)| \leq |g_+(x_k)| + |(\lambda_k)_-|/2r$, then $r \leftarrow 10r$.

Step 3. Apply $n - b$ iterations of the preconditioned conjugate gradient method to the function $\mathbf{L}_k(\lambda_k, \cdot)$, where the iteration starting point is x_k and the preconditioning matrix is H. If a nonbinding constraint becomes binding after a conjugate gradient iteration, then stop the iterations and x_{k+1} is the latest approximation to x^*. Otherwise, x_{k+1} is the point generated by the complete set of $n - b$ conjugate gradient iterations.

As observed earlier, the matrices A^{-1} and A^{-T} and the matrix H that projects a vector into the null space of A can be constructed from either the singular-value decomposition $Q\Lambda P$ of A or the decomposition $LP_l + 0P_0$ given in (16). In particular, we have

$$A^{-1} = P^T \Lambda^{-1} Q^T = P_l^T L^{-1},$$

$$A^{-T} = Q\Lambda^{-T} P = L^{-T} P_l,$$

$$H = P_0^T P_0.$$

Also, the Woodbury formula can be used to write the matrix $(I + rA^T A)^{-1}$ in the form

$$\begin{aligned}(I + rA^T A)^{-1} &= I - A^T (r^{-1} I + AA^T)^{-1} A \\ &= I - P_l^T L^T (r^{-1} I + LL^T)^{-1} L P_l \\ &= P^T [I - \Lambda^T (r^{-1} I + \Lambda\Lambda^T)^{-1} \Lambda] P.\end{aligned}$$

If λ_i is the ith diagonal element of Λ, then the ith diagonal element of the diagonal matrix $I-\Lambda^T(r^{-1}I+\Lambda\Lambda^T)^{-1}\Lambda$ is equal to $r^{-1}/(\lambda_i^2+r^{-1})$, for $i=1,\ldots,l$, and it is equal to one for $i>l$.

The global convergence results for Algorithm 5.1 also apply to Algorithm 5.2. From the analysis of Algorithm 3.2, it follows that Algorithm 5.2 is locally quadratic. In stating this local convergence result, it is implicitly assumed that in each conjugate gradient iteration, we employ the closest local minimizer along the search direction.

Theorem 5.2. Suppose that x^* is a local minimizer of (56), both f and g_+ are three times continuously differentiable in a neighborhood of x^*, g is continuous at x^*, $\lambda_i^*>0$ for $i\in I_+(\lambda^*, x^*)\cap[1, l]$, and (6) is satisfied when h is identified with g_+ and λ is identified with λ_+. Then, for either the Daniel, Fletcher-Reeves, or Polak-Ribière formulations of the conjugate gradient method, there exists a neighborhood $\mathbf{N}$ of (x^*, λ^*) and there exists a constant c independent of (x_k, λ_k) in $\mathbf{N}$ such that

$$|(\lambda_{k+1}-\lambda^*)_+|+|x_{k+1}-x^*|\leq c|(\lambda_k-\lambda^*)_+|^2+c|x_k-x^*|^2,$$

when λ_{k+1} and x_{k+1} are generated by Algorithm 5.2 and

$$(\lambda_{k+1})_+^T=-\nabla f(x_{k+1})B_{(k+1)}^{-1}.$$

References

1. ROSEN, J. B., and KREUSER, J. L., *A Gradient Projection Algorithm for Nonlinear Constraints*, Numerical Methods for Nonlinear Optimization, Edited by F. A. Lootsma, Academic Press, London, England, pp. 297–300, 1972.
2. ROSEN, J. B., and KREUSER, J. L., *A Quadratically Convergent Lagrangian Algorithm for Nonlinear Constraints*, University of Wisconsin, Computer Sciences Department, Report No. 166, Madison, Wisconsin, 1972.
3. TAPIA, R. A., *Diagonalized Multiplier Methods and Quasi-Newton Methods for Constrained Optimization*, Journal of Optimization Theory and Applications, Vol. 22, pp. 135–194, 1977.
4. ROBINSON, S. M., *A Quadratically Convergent Algorithm for General Nonlinear Programming Problems*, Mathematical Programming, Vol. 3, pp. 145–156, 1972.
5. MIELE, A., PRITCHARD, R. E., and DAMOULAKIS, J. N., *Sequential Gradient-Restoration Algorithm for Optimal Control Problems*, Journal of Optimization Theory and Applications, Vol. 5, pp. 235–282, 1970.
6. MIELE, A., DAMOULAKIS, J. N., CLOUTIER, J. R., and TIETZE, J. L., *Sequential Gradient-Restoration Algorithm for Optimal Control Problems with Nondifferential Constraints*, Journal of Optimization Theory and Applications, Vol. 13, pp. 218–225, 1974.
7. GONZALEZ, S., and MIELE, A., *Sequential Gradient-Restoration Algorithm for Optimal Control Problems with General Boundary Conditions*, Journal of Optimization Theory and Applications, Vol. 26, pp. 395–425, 1978.

8. BEST, M. J., BRÄUNINGER, J., RITTER, K., and ROBINSON, S. M., *A Globally and Quadratically Convergent Algorithm for General Nonlinear Programming Problems*, University of Wisconsin, Mathematics Research Center, Report No. 1973, 1979.
9. ROSEN, J. B., *Two-Phase Algorithm for Nonlinear Constraint Problems*, University of Minnesota, Computer Science Department, Report No. 77-8, 1977.
10. HAN, S. P., *Superlinearly Convergent Variable-Metric Algorithms for General Nonlinear Programming Problems*, Mathematical Programming, Vol. 11, pp. 263–282, 1976.
11. NOCEDAL, J., and OVERTON, M., *Projected Hessian Updating Algorithms for Nonlinearly Constrained Optimization*, SIAM Journal on Numerical Analysis, Vol. 22, pp. 821–850, 1985.
12. POWELL, M. J. D., *Algorithms for Nonlinear Constraints That Use Lagrangian Functions*, Mathematical Programming, Vol. 14, pp. 224–248, 1978.
13. YAMASHITA, H., *A Globally Convergent Constrained Quasi-Newton Method with an Augmented Lagrangian-Type Penalty Function*, Mathematical Programming, Vol. 23, pp. 75–86, 1982.
14. HAGER, W. W., *Dual Techniques for Constrained Optimization*, Proceedings of the 24th Conference on Decision and Control, Fort Lauderdale, Florida, pp. 616–621, 1985.
15. ARROW, K. J., and SOLOW, R. M., *Gradient Methods for Constrained Maxima, with Weakened Assumptions*, Studies in Linear and Nonlinear Programming, Edited by K. Arrow, L. Hurwicz, and H. Uzawa, Stanford University Press, Stanford, California, 1958.
16. HESTENES, M. R., *Multiplier and Gradient Methods*, Journal of Optimization Theory and Applications, Vol. 4, pp. 303–320, 1969.
17. POWELL, M. J. D., *A Method for Nonlinear Constraints in Minimization Problems*, Optimization, Edited by R. Fletcher, Academic Press, New York, New York, 1972.
18. HAGER, W. W., *Approximations to the Multiplier Method*, SIAM Journal on Numerical Analysis, Vol. 22, pp. 16–46, 1985.
19. POLYAK, B. T., and TRET'YAKOV, N. V., *The Method of Penalty Estimates for Conditional Extremum Problems*, Zhurnal Vychislitel'noi Matematiki i Matematicheskoi Fiziki, Vol. 13, pp. 34–46, 1973 (Translated into USSR Computational Mathematics and Mathematical Physics, Vol. 13, pp. 42–58, 1973).
20. BERTSEKAS, D. P., *Constrained Optimization and Lagrange Multiplier Methods*, Academic Press, New York, New York, 1982.
21. ROCKAFELLAR, R. T., *Augmented Lagrange Multiplier Functions and Duality in Nonconvex Programming*, SIAM Journal on Control, Vol. 12, pp. 268–285, 1974.
22. ROCKAFELLAR, R. T., *The Multiplier Method of Hestenes and Powell Applied to Convex Programming*, Journal of Optimization Theory and Applications, Vol. 12, pp. 555–562, 1973.
23. VALENTINE, F. A., *The Problem of Lagrange with Differential Inequalities as Added Side Conditions*, Contributions to the Calculus of Variations, University of Chicago Press, Chicago, Illinois, pp. 407–448, 1937.

24. GOLUB, G. H., and VAN LOAN, C. F., *Matrix Computations*, Johns Hopkins University Press, Baltimore, Maryland, 1983.
25. DONGARRA, J. J., MOLER, C. B., BUNCH, J. R., and STEWART, G. W., *LINPACK Users' Guide*, Society for Industrial and Applied Mathematics, Philadelphia, Pennsylvania, 1979.
26. HAGER, W. W., *Lipschitz Continuity for Constrained Processes*, SIAM Journal on Control and Optimization, Vol. 17, pp. 321–338, 1979.
27. LUENBERGER, D. G., *Introduction to Linear and Nonlinear Programming*, Addison-Wesley, Reading, Massachusetts, 1973.
28. BEN-ISRAEL, A., *A Newton-Raphson Method for the Solution of Systems of Equations*, Journal of Mathematical Analysis and Applications, Vol. 15, pp. 243–252, 1966.
29. ORTEGA, J. M., and RHEINBOLDT, W. C., *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, New York, New York, 1970.
30. COHEN, A., *Rate of Convergence for Root Finding and Optimization Algorithms*, University of California at Berkeley, PhD Thesis, 1970.
31. AKAIKE, H., *On a Successive Transformation of Probability Distribution and Its Application to the Analysis of the Optimum Gradient Method*, Annals of the Institute of Statistical Mathematics, Vol. 11, pp. 1–16, 1959.
32. LOEWNER, C., *Uber Monotone Matrixfunktionen*, Mathematische Zeitschrift, Vol. 38, pp. 177–216, 1934.
33. MITCHELL, A. R., and GRIFFITHS, D. F., *The Finite Difference Method in Partial Differential Equations*, John Wiley, Chichester, England, 1980.
34. GHOSH, N., and HAGER, W. W., *A Derivative-Free Bracketing Scheme for Univariate Minimization* (to appear).
35. HAGER, W. W., *A Derivative-Based Bracketing Scheme for Univariate Minimization* (to appear).
36. POLAK, E., *Computational Methods in Optimization: A Unified Approach*, Academic Press, New York, New York, 1971.
37. DAVIDON, W. C., *Variable-Metric Method for Minimization*, Report No. 5990, Argonne National Laboratory, Lemont, Illinois, 1959.
38. FLETCHER, R., and POWELL, M. J. D., *A Rapidly Convergent Descent Method for Minimization*, Computer Journal, Vol. 6, pp. 163–168, 1963.
39. BREZZI, F., *On the Existence, Uniqueness, and Approximation of Saddle-Point Problems Arising from Lagrange Multipliers*, RAIRO Analyse Numérique, Vol. 8, pp. 129–151, 1974.
40. HAGER, W. W., and ROSTAMIAN, R., *Optimal Coatings, Bang-Bang Controls, and Gradient Techniques*, Optimal Control: Applications and Methods, Vol. 8, pp. 1–20, 1987.
41. HAGER, W. W., and IANCULESCU, G. D., *Dual Approximations in Optimal Control*, SIAM Journal on Control and Optimization, Vol. 22, pp. 423–465, 1984.