DISCLAIMER OF QUALITY

Due to the condition of the original material, there are unavoidable flaws in this reproduction. We have made every effort possible to provide you with the best copy available. If you are dissatisfied with this product and find it unusable, please contact Document Services as soon as possible.

Thank you.

Due to the poor quality of the original document, there is some spotting or background shading in this document.
PROJECTED NEWTON METHODS FOR
OPTIMIZATION PROBLEMS WITH SIMPLE CONSTRAINTS*

Dimitri P. Bertsekas
Department of Electrical Engineering and Computer Science
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

Abstract
We consider the problem \( \min \{f(x) \mid x > 0 \} \) and algorithms of the form \( x_{k+1} = \left[ x_k - \alpha_k \nabla f(x_k) \right]^+ \)
where \([\cdot]^+\) denotes projection on the positive orthant, \( \alpha_k \) is a stepsize chosen by an Armijo-like rule, and \( D_k \) is a positive definite symmetric matrix which is partly diagonal. We show that \( D_k \) can be calculated simply on the basis of second derivatives of \( f \) so that the resulting Newton-like algorithm has a typically superlinear rate of convergence. With other choices of \( D_k \) convergence at a typically linear rate is obtained. The algorithms are almost as simple as their unconstrained counterparts. They are well suited for problems of large dimension such as those arising in optimal control while being competitive with existing methods for low-dimensional problems.

1. Introduction

We consider the problem
\[
\begin{align*}
\text{minimize } & f(x) \\
\text{subject to } & x \geq 0
\end{align*}
\]

where \( f : \mathbb{R}^n \to \mathbb{R} \) is a continuously differentiable function and the vector inequality \( x > 0 \) is meant to be componentwise [i.e. for \( x = (x_1, x_2, \ldots, x_n) \in \mathbb{R}^n \), we write \( x > 0 \) if \( x_i > 0 \) for all \( i = 1, \ldots, n \)]. This type of problem arises very often in applications; for example when \( f \) is a dual functional relative to an original inequality constrained primal problem, and \( x \) represents a vector of nonnegative Lagrange multipliers corresponding to the inequality constraints, and when \( f \) represents an Augmented Lagrangian or exact penalty function taking into account other possibly nonlinear equality and inequality constraints. The analysis and algorithms that follow apply also with minor modifications to problems with rectangle constraints such as
\[
\begin{align*}
\text{minimize } & f(x) \\
\text{subject to } & b_1 \leq x \leq b_2
\end{align*}
\]

where \( b_1 \) and \( b_2 \) are given vectors. Problems (1) and (2) are referred to as simply constrained problems, and their algorithmic solution is the primary subject of this paper.

In view of the simplicity of the constraints, one would expect that solution of problem (1) is almost as easy as unconstrained minimization of \( f \). This expectation is partly justified in that the first order necessary condition for a vector \( x = (x_1, \ldots, x_n) \) to be a local minimum of problem (1) takes the simple form
\[
\begin{align*}
\frac{\partial f(x)}{\partial x_i} & > 0, & i = 1, \ldots, n & \quad (3a) \\
\frac{\partial f(x)}{\partial x_i} & = 0, & \text{if } x_i > 0, & i = 1, \ldots, n & \quad (3b)
\end{align*}
\]

Furthermore the direct analog of the method of steepest descent takes the simple form
\[
x_{k+1} = \left[ x_k - \alpha_k \nabla f(x_k) \right]^+, \quad k = 0, 1, \ldots
\]

where \( \alpha_k \) is a positive scalar stepsize and for any vector \( z = (z_1, \ldots, z_n) \in \mathbb{R}^n \) we denote
\[
[z]^+ = \left[ \begin{array}{c}
\max \{0, z_1\} \\
\vdots \\
\max \{0, z_n\}
\end{array} \right]
\]

The stepsize \( \alpha_k \) may be chosen in a number of ways.

In the original proposal of Goldstein [1], and Levitin and Poljak [2], \( \alpha_k \) is taken to be a constant \( \alpha \) (i.e. \( \alpha_k \equiv \alpha \), for all \( k \)) and a convergence result is shown under the assumption that \( \alpha \) is sufficiently small and \( \nabla f \) is Lipschitz continuous. In general a proper value for \( \alpha \) can be found only through experimentation. An alternative suggested by McCormick [3] is to choose \( \alpha_k \) by function minimization along the arc of points \( x_k(\alpha), \alpha \geq 0 \) where

To be presented at the 20th IEEE Conference on Decision and Control, San Diego, CA. December 1981.

*Work supported by Grant NSF ECS-79-20834 and by Alphatech, Inc. through DOE Contract No. DE-AC01-79ET29243.
Any attempt to construct a superlinearly convergent quadratic programming subproblem is characterized by linear convergence rate. Thus to attempt to construct a superlinearly convergent algorithm must be necessary involve a non-diagonal scaling matrix D_k which is an adequate approximation of the `inverse' Hessian V^2 J(x_k)^{-1}. At least along a suitable subspace. At this point we find that the algorithms available at present are far more complicated than their unconstrained counterparts, particularly when the problem has large dimension. Thus the most straightforward extension of Newton's method is given by

$$x_{k+1} = x_k + \alpha_k (x_k - x_k)$$

where x_k is a solution of the quadratic program

$$\min \{ \frac{1}{2} (x-x_k)^T V^2 J(x-x_k) : \mu(x) \leq 0 \}$$

subject to x \geq 0

and \alpha_k is a stepsize parameter. There are convergence and superlinear rate of convergence results in the literature regarding this type of method (Levitin and Poljak [2], Dunn [5]) and its Quasi-Newton versions (Garcia-Palomares and Mangasarian [6]), however its effectiveness is strongly dependent upon the computational requirements of solving the quadratic program (13). For problems of small dimension, problem (13) can be solved rather quickly by standard pivoting or manifold suboptimization methods but when the problem is of large dimension the solution of the quadratic program (13) by standard methods can be very time consuming. Indeed, there are large-scale quadratic programming problems arising in optimal control the solution of which by pivoting methods is unthinkable. In any case the facility or lack thereof of solving the quadratic program (13) must be accounted for when comparing method (12) against other alternatives.

Another possible approach for constructing superlinearly convergent algorithms for solving problem (1) stems from the original gradient projection proposal of Rosen [7], and is based on manifold suboptimization and active set strategies as in Gill and Murray [8], Goldfarb [9], Luenberger [10] and other sources, (see Lenard [11] for an up-to-date performance evaluation of various alternatives). Methods of this type are quite efficient for problems of relatively small dimension, but become typically unattractive for large-scale problems with a large number of constraints binding at a solution. The main reason is that typically at most one constraint can be added to the active set at each iteration, so if, for example, 1,000 constraints are binding at the point of convergence and an interior starting point is selected, then the method will require at least 1,000 iterations (and possibly many more) to converge. While several authors [8], [10] have alluded to the possibility of bending the direction of search along the constraint boundary, the only specific proposal known to the author that has been made in the context of the manifold suboptimization approach is the one of McCormick [12] and it does not seem particularly attractive for large-scale problems. (The Quasi-Newton methods proposed by Brayton and Cullum [13] incorporate bending but simultaneously require the solution of quadratic programming subproblems). Manifold suboptimization methods require also additional computation overhead in deciding which constraint to
drop from the currently active set. For the apparently most successful strategies (Lenard [11]) which attempt to drop as many constraints as possible, this overhead can be significant and must be taken into account when comparing the manifold suboptimization approach with other alternatives.

The algorithms proposed in this paper attempt to combine the basic simplicity of the steepest descent iteration (4), (7) with the sophistication and fast convergence of the constrained Newton’s method (12), (13). They do not involve solution of a quadratic program thereby avoiding the associated computational overhead, and there is no bound to the number of constraints that can be added to the currently active set thereby bypassing a serious inherent limitation of manifold suboptimization methods. The basic form of the method is

\[
x_{k+1} = x_k(a_k)
\]  

where

\[
x_k(a) = [x_k - \alpha_k D_k Vf(x_k)]^+ , \quad \alpha \geq 0.
\]  

\(D_k\) is a positive definite symmetric matrix which is partly diagonal, and \(\alpha_k\) is a stepsize determined by an Armijo-like rule similar to (7) that will be described later. The convergence and rate of convergence properties of this method are discussed in Section 2. A key property of the method is that under mild assumptions it identifies the manifold of binding constraints at a solution in finite number of iterations in the sense of (6). This means that eventually the method is reduced to an unconstrained method on this manifold and brings to bear the extensive methodology and analysis relating to unconstrained minimization algorithms.

In reference [17] we discuss how the method (14), (15) can form the basis for constructing algorithms for general linearly constrained problems of the form

\[
\text{minimize} \quad f(x)
\]

subject to \(b_1 \leq Ax \leq b_2\).

The main idea here is to view problem (16) locally as a simply constrained problem via a transformation of variables. For example, if the matrix \(A\) is square and invertible, problem (16) is equivalent to the problem

\[
\text{minimize} \quad h(y) \quad \text{subject to} \quad b_1 \leq y \leq b_2 \quad \text{via the transformation} \quad y = Ax.
\]

A similar approach based on an active set strategy is employed when \(A\) is not square and invertible. The ideas are similar to those involved in manifold suboptimization methods where a linear manifold is selected as a "local universe" for the purposes of the current iteration. In our algorithms we take a suitably chosen iteration (i.e. a set described by upper and lower bounds on the variables) as a "local universe" instead of a manifold.

Throughout the paper we emphasize Newton-like methods as prototypes for broad classes of superlinearly converging algorithms that fit the framework of the paper. We often make positive definiteness assumptions on the Hessian matrix of \(f\) in order to avoid getting bogged down in technical details relating to modifications of Newton's method such as those employed in unconstrained minimization [14]-[16] to account for the possibility that \(V^2 f\) is not positive definite. Quasi-Newton versions of the Newton-like methods presented are possible but the discussion of specific implementations is beyond the scope of the paper. More generally it may be said that the nature of the algorithms proposed is such that almost every useful idea from unconstrained minimization can be fruitfully adapted within the constrained minimization framework considered here, however, the precise details of how this should be done may involve considerable further research and experimentation.

The notation employed throughout the paper is as follows. All vectors are considered to be column vectors. A prime denotes transposition. The standard norm in \(R^n\) is denoted by \(|\cdot|\), i.e. for \(x = (x_1, \ldots, x^n)\) we write \(x = \sqrt{\sum_{i=1}^{n} (x_i^2)}^{1/2}\). The gradient and Hessian of a function \(f; R^n + R\) are denoted by \(Vf\) and \(V^2 f\) respectively. Proofs of all results stated may be found in reference [17].

2. Algorithms for Minimization Subject to Simple Constraints

We consider first the problem \(\min \{f(x) \mid x \geq 0\}\) of (1). Any vector \(x\) satisfying the first order necessary condition (3) will be referred to as a critical point with respect to problem (1). We focus attention at iterations of the form

\[
x_{k+1} = [x_k - \alpha_k D_k Vf(x_k)]^+ , \quad \alpha \geq 0.
\]

where \(D_k\) is a positive definite symmetric matrix and \(\alpha_k\) is chosen by search along the arc of points

\[
x_k(0) = [x_k - \alpha_k D_k Vf(x_k)]^+ , \quad \alpha \geq 0.
\]

It is easy to construct examples where an arbitrary positive definite choice of the matrix \(D_k\) leads to situations where it is impossible to reduce the value of the objective by suitable choice of the stepsize \(\alpha\) (i.e. \(f(x_k(0)) \geq f(x_k), \forall \alpha \geq 0\)). The following proposition identifies a class of matrices \(D_k\) for which an objective function reduction is possible. Define for all \(x \geq 0\)
We say that a symmetric $n \times n$ matrix $D$ with elements $d_{ij}$ is diagonal with respect to a subset of indices $I \subseteq \{1, 2, \ldots, n\}$ if
\[ d_{ij} = 0, \quad \forall i \in I, \quad j \neq i. \]

Proposition 1: Let $x > 0$ and $D$ be a positive definite symmetric matrix which is diagonal with respect to $I^+(x)$ and denote
\[ I(x) = \{ i | x_i > 0 \}. \]

a) The vector $x$ is a critical point with respect to problem (1) if and only if $x = x(a)$, $\forall a \geq 0$.

b) If $x$ is not a critical point with respect to problem (1) there exists a scalar $\alpha > 0$ such that $f(x(a)) < f(x)$, $\forall a \in (0, \alpha]$.

Based on Proposition 1 we are led to the conclusion that the matrix $D_k$ in the iteration
\[ x_{k+1} = \left[ x_k - \alpha_k D_k \nabla f(x_k) \right]^+ \]

should be chosen diagonal with respect to a subset of indices that contains
\[ I^+(x_k) = \{ i | x_i^k > 0, \frac{\partial f(x_k)}{\partial x_i^k} > 0 \}. \]

Unfortunately the set $I^+(x_k)$ exhibits an undesirable discontinuity at the boundary of the constraint set whereby given a sequence $\{x_k\}$ of interior points that converges to a boundary point $x$ the set $I^+(x_k)$ may be strictly smaller than the set $I^+(x)$. This causes difficulties in proving convergence of the algorithm and may have an adverse effect on its rate of convergence. (This phenomenon is quite common in feasible direction algorithms and is referred to as zigzagging or jamming). For this reason we will employ certain enlargements of the sets $I^+(x_k)$ with the aim of bypassing these difficulties.

The algorithm that we describe utilizes a scalar $\varepsilon > 0$ (typically small), a fixed diagonal positive definite matrix $M$ (for example the identity), and two parameters $\delta \varepsilon (0,1)$ and $\delta \varepsilon (0,\frac{1}{2})$ that will be used in connection with an Armijo-like stepsize rule. An initial vector $x_0 \geq 0$ is chosen and at the $k$th iteration of the algorithm we have a vector
\[ x_k - \alpha_k D_k \nabla f(x_k) \]

Actually the results that follow can be shown also for the case where $H$ is changed from one iteration to the next in a way that its diagonal elements are bounded above and away from zero.
The gradient \( \nabla f \) is Lipschitz continuous on each bounded set of \( \mathbb{R}^n \), i.e., given any bounded set \( S \subseteq \mathbb{R}^n \), there exists a scalar \( L \) (depending on \( S \)) such that
\[
|\nabla f(x) - \nabla f(y)| \leq L|x - y|, \quad \forall x, y \in S.
\] (28)

There exist positive scalars \( \lambda_1, \lambda_2 \) and nonnegative scalars \( q_1, q_2 \) such that
\[
\lambda_1 |z|^2 \leq z^T D_k z \leq \lambda_2 |z|^2, \quad \forall z \in \mathbb{R}^n, \quad k = 0, 1, \ldots
\] (30)

where \( D_k \) is the inverse of the Hessian of \( f \) with respect to the indices \( \{i_k \} \). Then there exists a scalar \( \delta > 0 \) such that if \( \{x_k\} \) is a sequence generated by iteration (24) and for some index \( k \) we have
\[
|x_k - x^*| < \delta,
\] (36)

then \( \{x_k\} \) converges to \( x^* \) and we have
\[
s_k = B(x_k) = B(x^*), \quad \forall k \geq k + 1.
\] (37)

Under the assumptions of Proposition 3 we see that if the algorithm converges to a local minimum \( x^* \) satisfying Assumption (C) then it reduces eventually to an unconstrained minimization method restricted to the subspace
\[
T = \{x | x^i = 0, v \in B(x^*) \}.
\] (38)

Furthermore as shown in Proposition 3 for some index \( k \) we will have
\[
s_k = B(x^*), \quad \forall k \geq k.
\] (39)

This shows that if the portion of the matrix \( D_k \) corresponding to the indices \( s_k^+ \) is chosen in a way that approximates the inverse of the portion of the Hessian of \( f \) corresponding to these same indices,

Proposition 3: Let \( x^* \) be a local minimum of problem (1) satisfying Assumption (C). Assume also that
(1) \( \frac{\partial x}{\partial x} \) holds in the stronger form whereby, in addition to (41), the diagonal elements \( d_{ii} \) of the matrix \( D_k \) satisfy for some scalar \( \lambda > 0 \)
\[
1 \leq d_{ii}, \quad \forall k \geq 0, \quad i \in s_k^+.
\] (35)

Then there exists a scalar \( \delta > 0 \) such that if \( \{x_k\} \) is a sequence generated by iteration (24) and for some index \( k \) we have
\[
|x_k - x^*| < \delta,
\] (36)

then \( \{x_k\} \) converges to \( x^* \) and we have
\[
s_k = B(x_k) = B(x^*), \quad \forall k \geq k + 1.
\] (37)

More specifically, by rearranging indices if necessary, assume without loss of generality that
\[
s_k^+ = \{r_{k+1}, \ldots, n\},
\] (40)

where \( r_k \) is some integer. Then \( D_k \) has the form
\[
D_k = \begin{bmatrix}
0 & d_k & \cdots & d_k \\
-1 & -1 & \cdots & -1 \\
0 & 0 & \cdots & 0 \\
0 & 0 & \cdots & 0
\end{bmatrix}
\] (41)

where \( d_i > 0, i = r_{k+1}, \ldots, n \) and \( D_k \) can be an arbitrary positive definite matrix. Suppose we choose \( D_k \) to be the inverse of the Hessian of \( f \) with respect to the indices \( s_k^+ \), i.e., the elements
\[
[\tilde{D}_k]_{ij}
\] of \( \tilde{D}_k^{-1} \) are
By Assumption (C), $V^2 f(x^*)$ is positive definite on $T$ so it follows from (39) that this choice is well defined and satisfies the assumption of Proposition 3 for $k$ sufficiently large. Since the conclusion of this proposition asserts that the method eventually reduces to Newton's method restricted on the subspace $T$ a superlinear convergence rate result follows. This type of argument can be used to construct a number of Newton-like and quasi-Newton methods and prove corresponding convergence and rate of convergence results.

References


