SIAM J. CONTROL AND OPTIMIZATION
Vol. 14, No. 1, January 1976

EFFICIENTLY CONVERGING MINIMIZATION METHODS BASED ON THE REDUCED GRADIENT*

DANIEL GABAY AND DAVID G. LUENBERGER†

Abstract. This paper presents three computational methods which extend to nonlinearly constrained minimization problems the efficient convergence properties of, respectively, the method of steepest descent, the variable metric method, and Newton's method for unconstrained minimization. Development of the algorithms is based on use of the implicit function theorem to essentially convert the original constrained problem to an unconstrained one. This approach leads to practical and efficient algorithms in the framework of Abadie's generalized reduced gradient method. To achieve efficiency, it is shown that it is necessary to construct a sequence of approximations to the Lagrange multipliers of the problem simultaneously with the approximations to the solution itself. In particular, the step size of each iteration must be determined by a linesearch for a minimum of an approximate Lagrangian function.

1. Introduction. Many computational methods have been proposed to find the minimum of a real-valued function $f(x)$ over the n-dimensional real space R^n. When both the values of the function f and its derivatives $\partial f/\partial x_i$ for $i = 1, \cdots, n$ are available at every point x, gradient-related techniques are generally favored. These schemes iteratively construct, from an initial point x^0, a monotonically improving sequence of approximate solutions x^k according to a recurrence formula of the form

$$(*) \qquad x^{k+1} = x^k - \alpha_k p^k, \qquad k = 0, 1, \cdots,$$

where p^k is a direction determined on the basis of the gradient $\nabla f(x^k)$, and α_k is a positive scalar chosen to achieve a descent in the value of the objective:

$$f(x^{k+1}) \leqq f(x^k).$$

The parameter α_k controls the size of the step k and influences the convergence properties of the algorithm $(*)$.

The speed of convergence is strongly dependent on the step size choice. This dependence is well understood in the case of the method of steepest descent, which takes for p^k the direction of the gradient $\nabla f(x^k)$ itself, as originally proposed by Cauchy [6]. The best performance is obtained for the optimal steepest descent method in which α_k is chosen to achieve a local minimum of f along $\nabla f(x^k)$. As first exhibited by Kantorovitch [17], the sequence $\{x^k\}$ converges to x^* linearly, i.e., at least as fast as a geometric progression. The sharpest possible estimate for the ratio of this progression is asymptotically given by $(M - m/M + m)^2$, where M and m are, respectively, the largest and smallest eigenvalues of F^*, the matrix of second order partial derivatives of f at x^*. We refer to this as the *natural rate of convergence* of the problem. It represents the fastest possible speed for a steepest descent algorithm. It provides, therefore, a standard by which the performance of other schemes can be evaluated through comparison with this efficient natural

* Received by the editors April 23, 1973, and in revised form December 17, 1974.

† Department of Engineering-Economic Systems, Stanford University, Stanford, California 94305. The first author is now with Centre National de la Recherche Scientifique, Paris, France. This research was supported by the National Science Foundation under Grants GK32870 and GK 29237.

rate. And, indeed, the rate of convergence of other gradient-related algorithms can be expressed relative to this ratio [21].

In practice, many problems either arise or can be formulated as constrained optimization problems. In this case, the minimum of the function f is sought among the values it takes while the variable point x is restricted to a given subset $\mathcal{S}$. This subset is called the feasible region and is assumed to be described by a finite number of constraint equalities and inequalities. Without lack of generality, we can formally write

$$\mathcal{S} = \{x \in R^n | h_i(x) = 0,\ i = 1, 2, \cdots, m;\ a_j \leqq x_j \leqq b_j,\ j = 1, 2, \cdots, n\} \quad \text{with } m \leqq n,$$

where a_j and b_j are real numbers and can take the values $-\infty$ and $+\infty$.

One of the most successful gradient-related methods to solve this nonlinearly constrained problem is Abadie's generalized reduced gradient algorithm (GRG) [1]. As an extension to the nonlinear case of the upper bounding simplex method for linear programming [8], this method introduces a partition of the variables into *m basic variables*, denoted by the vector $x_B = (x_{B_1}, \cdots, x_{B_m})$, and $n - m$ remaining *independent variables* forming the vector $x_R = (x_{R_1}, \cdots, x_{R_{n-m}})$, such that

$$a_{B_i} < x_{B_i} < b_{B_i}, \qquad i = 1, \cdots, m,$$

$$a_{R_j} \leqq x_{R_j} \leqq b_{R_j}, \qquad j = 1, \cdots, n - m.$$

The independent variables are changed on the basis of the reduced gradient [28], [11], obtained by "pricing-out" the nonbasic components of the gradient $\nabla f(x)$, as the reduced costs are obtained in the simplex method. When the constraints h_i are nonlinear, Abadie's proposal consists in decomposing each iteration in two phases. Starting from a feasible point, a move is performed along a direction tangent to $\mathcal{S}$ based on the reduced gradient. It is followed by a restoration move, achieved by adjusting the m basic variables in order to satisfy the constraint equations. The resulting algorithm [2] is ranked first in efficiency among all available techniques in the comparison studies conducted by Colville on a series of test problems [7], [3]. The selection of the size of the tangent move is, in large part, responsible for the current complexity of the code, since, if this parameter is too large, the restoration may be impossible or may lead to a feasible point which does not constitute an improvement of the objective function. Therefore, following this approach, one is often forced to try several step lengths for the first phase in order to obtain a satisfactory point at the end of the second phase. Such repeated trials significantly increase the computation time and, even though a procedure is developed so as to insure convergence, the rate of convergence may be far from optimal.

The object of this work is to propose generalized gradient related methods for nonlinearly constrained problems which properly extend the efficient convergence of the optimal methods of the unconstrained case. We restrict our analysis to problems without bound constraints on the variables and refer to [15] for the alterations necessary to treat the general case. The implicit function theorem [10] provides a natural and convenient framework to study the appropriate restrictions of the original methods of unconstrained minimization to the constraint set $\mathcal{S}$ itself (rather than to the subspace tangent to $\mathcal{S}$). The theorem

conceptually allows one to express the basic variables as functions of the independent variables, thus converting the original problem to an unconstrained one:

$$\text{Minimize } \phi(x_R) = f(x_B(x_R), x_R).$$

Solving this reduced problem by the gradient-related methods of unconstrained minimization leads, in the original space, to schemes in which the independent variables are moved on the basis of the gradient of the reduced function, which turns out to be Wolfe's reduced gradient. The basic variables are altered correspondingly to maintain feasibility. We thus extend the efficient convergence properties of the method of steepest descent, the variable metric method, and Newton's method to nonlinearly constrained minimization.

These ideal extended gradient-related methods cannot be implemented exactly, since it is not possible in practice to generate arcs along $\mathcal{S}$. We are led to consider more practical schemes which accurately approximate the arcs of the ideal methods.

In § 3 we define an implementable generalized reduced steepest descent algorithm, combining at each iteration a tangent phase and a restoration phase. To achieve efficiency, it is shown that it is necessary to construct a sequence of approximations to the Lagrange multipliers of the problem simultaneously with the approximations to the solution itself. Each combined step then accurately approximates the arc of the ideal scheme, provided that the step size is determined by a linesearch for a minimum of the approximate Lagrangian, a procedure which has been tentatively proposed on other occasions [19], [23].

In §§ 4 and 5, we show how the framework of the reduced unconstrained problem can establish guidelines to define practical and efficient algorithms extending, respectively, the variable metric method [14] and Newton's method [18] to nonlinearly constrained minimization and inheriting their superlinear and second order rates of convergence.

2. Notation. We denote n-dimensional vectors by notation such as $x = (x_1, \cdots, x_n)$. Unless otherwise specified, they are regarded as column vectors. For any matrix A, ${}'A$ denotes its transpose.

Given a function $f: R^n \to R$, its gradient at x is the n-row vector $\nabla f(x) = ((\partial f/\partial x_1)(x), \cdots, (\partial f/\partial x_n)(x))$. For any subset $K \subset \{1, \cdots, n\}$, we denote by $\nabla_K f(x)$ the vector of components $(\partial f/\partial x_i)(x)$ with $i \in K$. We denote the matrix of the second order partial derivatives, the Hessian, by $F(x)$.

For a mapping $h: R^n \to R^m$ with components h_i, $\nabla h(x)$ represents the $m \times n$ Jacobian matrix with element (i, j) given by $(\partial h_i/\partial x_j)(x)$. The second derivative of h is best regarded as the m-tuple $H = (H_1, H_2, \cdots, H_m)$, where H_i is the Hessian of h_i. We denote the m-tuple of the associated quadratic forms by ${}'x \cdot H \cdot x = ({}'xH_1x, {}'xH_2x, \cdots, {}'xH_mx)$ for any $x \in R^n$. We define the operator $\times$, associating an element λ of R^m and an m-tuple H of $R^{m \times n \times n}$ into an element of $R^{n \times n}$, by

$$\lambda \times H = \lambda_1 H_1 + \cdots + \lambda_m H_m;$$

for any $x \in R^n$, we have

$${}'x(\lambda \times H)x = \lambda({}'x \cdot H \cdot x).$$

We denote by (P) the problem

$$\text{(P)} \qquad \begin{aligned} &\text{Minimize } f(x) \\ &\text{Subject to } h(x) = 0. \end{aligned}$$

3. The generalized reduced steepest descent method for nonlinearly constrained minimization.

3.1. The idealized reduced gradient method. The implicit function theorem has historically played a fundamental role in the theory of constrained minimization problems, since it provides the tool required to establish the existence of Lagrange multipliers. Basically it reduces problem (P) to an unconstrained minimization (at least locally) by solving the implicit constraint equations. We assume in all the following that f and h_i are twice continuously differentiable and possess bounded third order derivatives.

Assuming that the constraints are regular, i.e., that the gradient vectors $\nabla h_1(x), \cdots, \nabla h_m(x)$ are linearly independent for all x, then the implicit function theorem guarantees the local existence of a mapping $\psi : R^{n-m} \to R^m$ such that $x_B = \psi(x_R)$. It is well known that

$$\nabla\psi(x) = -\nabla_B h(x)^{-1}\nabla_R h(x), \tag{1}$$

where the argument x stands indifferently for the independent variables $x_R \in R^{n-m}$ and for the n-tuple $(\psi(x_R), x_R)$; and it can be shown that the second derivative of ψ is given by

$$\Psi(x) = -\nabla_B h(x)^{-1} \times ['T(x) \cdot H(x) \cdot T(x)], \tag{2}$$

where $T(x)$ is the $n \times (n-m)$ matrix

$$T(x) = \begin{bmatrix} -\nabla_B h(x)^{-1}\nabla_R h(x) \\ I_{n-m} \end{bmatrix}. \tag{3}$$

This matrix represents the mapping of R^n onto $\mathcal{T}(x)$, the tangent subspace to $\mathcal{S}$ at x. We can view (P) in terms of the reduced problem (R) in R^{n-m}:

$$\text{(R)} \qquad \text{Minimize } \phi(x_R) = f[\psi(x_R), x_R],$$

defined at least in a neighborhood of a solution point x^* of (P). The gradient of ϕ is called the *reduced gradient* and its transpose is an $(n-m)$-dimensional column vector denoted by $r(x)$. The chain rule for derivatives leads to

$$\nabla\phi(x) = \nabla_B f(x)\nabla\psi(x) + \nabla_R f(x).$$

Using (1), we get

$$'r(x) = \nabla_R f(x) - \nabla_B f(x)[\nabla_B h(x)]^{-1}\nabla_R h(x). \tag{4}$$

Among the methods of steepest descent for solving (R), the optimal steepest descent algorithm provides the best performing algorithm. It consists of a series of moves in R^{n-m} along the reduced gradients at the successive iterates x_R^k. The size of each step is determined by a linesearch along $r^k = r(x_R^k)$ for a local minimum point of ϕ.

In practice, it is usually not possible to achieve explicitly the elimination of the dependent variables x_B and it is therefore necessary to solve problem (P) in the original space R^n. However, the above study of the reduced problem (R) shows that the most natural algorithm consists, for its kth iteration, starting from the feasible point $x^k = (x_B^k, x_R^k)$, in moving the independent variables x_R along the reduced gradient r^k, while maintaining feasibility by an alteration of the basic variables x_B. This defines an arc $x^k(\beta)$ on $\mathscr{S}$ emanating from x^k. The projection of this arc on the subspace R of the independent variables, parallel to the basic subspace B, is the straight line in the (negative) direction of r^k; hence

$$x_R^k(\beta) = x_R^k - \beta r^k. \tag{5}$$

To satisfy the constraint equations, the basic variables must satisfy

$$x_B^k(\beta) = \psi(x_R^k - \beta r^k).$$

Assuming that the constraints are uniformly regular (i.e., that there exists a scalar $\gamma > 0$ such that $\|\nabla_B h(x)\| \geqq \gamma$ for all x), we can write

$$x_B^k(\beta) = \psi(x_R^k) - \beta \nabla \psi^k r^k + (\beta^2/2)^t r^k \cdot \Psi^k \cdot r^k + O(\|r^k\|^3).$$

Using (1) and (2) we obtain

$$x_B^k(\beta) = x_B^k - \beta(-(\nabla_B h^k)^{-1} \nabla_R h^k) r^k - \beta^2 (\nabla_B h^k)^{-1} q^k + O(\|r^k\|^3), \tag{6}$$

where q^k is the m-dimensional column vector of components

$$q_i^k = \tfrac{1}{2} r^{kt} T^k H_i(x^k) T^k r^k. \tag{7}$$

This reduced gradient method, illustrated in Fig. 1, belongs to a class of techniques for nonlinearly constrained problems proposed by Altman [4] and was

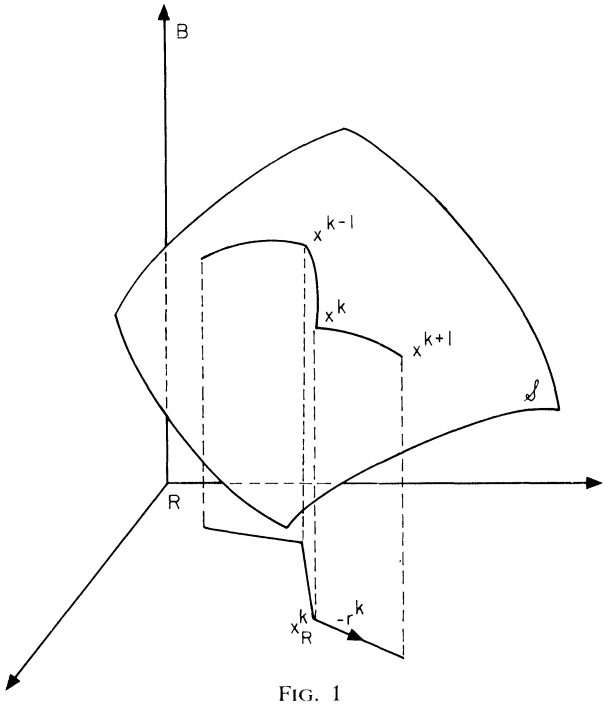

FIG. 1

presented first in [21] in this specific set-up. To efficiently extend the optimal steepest descent method, the step size parameter β_k must be chosen to achieve a local minimum of f along the curve $x^k(\beta)$:

$$\beta_k = \text{Argmin}\{f[x^k(\beta)] \mid \beta \geqq 0\}. \tag{8}$$

The speed of convergence of the sequence $\{x^k\}$ to x^* is then asymptotically given by the Kantorovitch-ratio $(M-m/M+m)^2$, where M and m are the extreme eigenvalues of Φ^*, the Hessian of ϕ at x^*. This defines the natural rate of convergence for reduced gradient methods, since this algorithm ideally extends the efficient performance of steepest descent methods for the unconstrained case. Our motivation is to find an efficient way to at least approximately find the parameter β_k of the ideal method, without actually searching the ideal curve. This, in turn, will lead to an algorithm achieving the natural rate.

A familiar formulation of this result is obtained through the introduction of the *Lagrangian function* for (P), $l : R^n \times R^m \to R$ defined by

$$l(x, \lambda) = f(x) + \lambda h(x).$$

At every regular point x and for the partition of $R^n = B \oplus R$, we define the *reduced Lagrange multiplier* as the m-dimensional row vector

$$\lambda(x) = -\nabla_B f(x) \nabla_B h(x)^{-1}. \tag{9}$$

We can thus interpret the reduced gradient in terms of the gradient of the Lagrangian, since

$$\nabla_x l[x, \lambda(x)] = [\nabla_B l, \nabla_R l] = [0, {}^t r(x)].$$

We can also evaluate the Hessian $\Phi(x)$ of the function $\phi(x)$:

$$\begin{aligned}
\Phi(x) &= {}^t\nabla\psi(x) F_{BB}(x) \nabla\psi(x) + F_{BR}(x)\nabla\psi(x) + {}^t\nabla\psi(x) F_{RB}(x) \\
&\quad + F_{RR}(x) + \nabla_B f(x) \Psi(x) \\
&= {}^tT(x) F(x) T(x) + \lambda(x) \times (T(x) \cdot H(x) \cdot T(x)) \\
&\qquad\qquad \text{(using (1), (2), (3), (9))} \\
&= {}^tT(x) L[x, \lambda(x)] T(x),
\end{aligned}$$

where $L(x, \lambda)$ is the Hessian, with respect to x, of the Lagrangian l. ${}^tTLT$ represents a restriction to $\mathcal{T}(x)$ of the Hessian of the Lagrangian.

3.2. The step size selection. The reduced gradient method presented above is idealized from a computational viewpoint, since it is in practice impossible to generate the arcs $x^k(\beta)$. We can devise an implementable version of the reduced gradient algorithm, which is really an approximation, using first order information, of the idealized scheme. Calculation of the step along the curve from x^k to $x^k(\beta_k)$ is replaced by a combination of two phases, as depicted in Fig. 2. In the first phase, a move is made along the tangent to the ideal curve $x^k(\beta)$. (It has already been established that this tangent direction is given by $-T^k r^k$.) This step, charac-

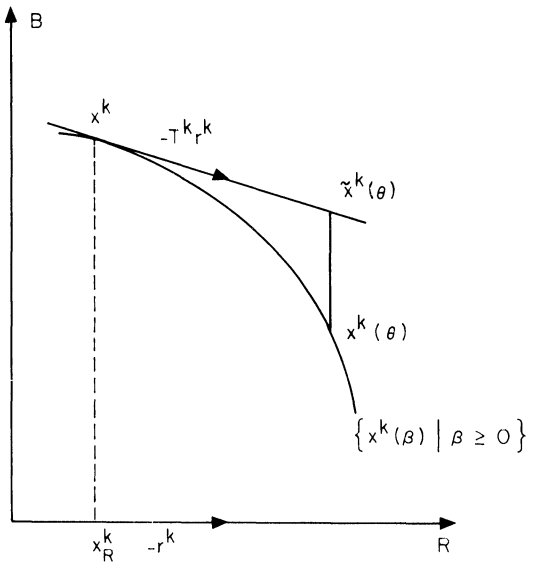

FIG. 2

terized by the step length parameter θ, leads to the point

$$\tilde{x}^k(\theta) = x^k - \theta T^k r^k,$$

which generally does not satisfy the constraint equations. A restoration phase back to a feasible point $x^k(\theta)$ of $\mathcal{S}$ is needed and is performed by adjusting the basic variables.

The efficiency of this algorithm depends, to a large extent, upon the selection of the step length θ_k. The study of the idealized reduced gradient method shows that, in order to achieve convergence at the natural rate, our practical algorithm must use a step size parameter which asymptotically satisfies

$$\theta_k = \beta_k + O(\|r^k\|).$$

We can compute an estimate of the value of the objective along the arc $\{x^k(\beta)\}$:

$$f[x^k(\beta)] = f(x^k) - \beta \nabla f^k T^k r^k + (\beta^2/2)^t r^{kt} T^k F^k T^k r^k$$

$$- \beta^2 \nabla_B f^k (\nabla_B h^k)^{-1} q^k + O(\|r^k\|^3).$$

Introducing $\lambda^k = \lambda(x^k)$ as defined by (9) and using the definition (7) of q^k, this can be written

$$f[x^k(\beta)] = f(x^k) - \beta \|r^k\|^2 + (\beta/2)^t r^{kt} T^k (F^k + \lambda^k \times H^k) T^k r^k + O(\|r^k\|^3). \tag{10}$$

Consider now the value of the Lagrangian function $l(\cdot, \lambda^k)$ along the direction

tangent at x^k to the arc $\{x^k(\beta)\} : \tilde{x}^k(\beta) = x^k - \beta T^k r^k$:

$$l[\tilde{x}^k(\beta), \lambda^k] = l(x^k, \lambda^k) - \beta \nabla l(x^k, \lambda^k) T^k r^k + \frac{\beta^2}{2} {}^t r^{kt} T^k L^k T^k r^k + O(\|r^k\|^3)$$

$$= f(x^k) - \beta \|r^k\|^2 + \frac{\beta^2}{2} {}^t r^{kt} T^k L^k T^k r^k + O(\|r^k\|^3).$$

Therefore the function $l(\cdot, \lambda^k)$ takes along the tangent direction $-T^k r^k$ up to second order the value of the objective f at the feasible point with the same independent coordinate.

Hence a simple and efficient procedure to find an approximation of the order of $\|r^k\|$ to the ideal step size β_k consists in selecting the step length parameter of the tangent phase in order to minimize $l(x, \lambda^k)$ along the direction $-T^k r^k$. This rule defines the parameter α_k:

$$\alpha_k = \text{Argmin}\,\{l(x^k - \alpha T^k r^k, \lambda^k) | \alpha \geqq 0\},$$

which we refer to as the *Lagrangian step size.* This selection rule can be easily incorporated in our algorithm. We describe below the detailed procedures to handle the possible computational difficulties associated with the restoration phase and to insure the convergence of the sequence of iterates.

3.3. The generalized reduced steepest descent algorithm. We must first notice that the choice of the Lagrangian step size does not represent any additional computational work since the evaluation of λ^k is a necessary step in the calculation of the reduced gradient r^k.

Under the assumption of uniform regularity of the constraints, the restoration phase from a point $\tilde{x}^k = [\tilde{x}^k_B, \tilde{x}^k_R]$ can always be performed, at least conceptually. A computationally efficient procedure consists of solving the system of equations

$$h_i(x_{B_1}, \cdots, x_{B_m}, \tilde{x}^k_{R_1}, \cdots, \tilde{x}^k_{R_{n-m}}) = 0, \qquad i = 1, \cdots, m,$$

for the unknown variables $x_{B_1}, \cdots, x_{B_m}$ using a modified Newton's method. Starting from $y^0 = \tilde{x}^k_B$, such a method constructs successive approximations $y^i \in R^m$, according to the recurrence

$$y^{i+1} = y^i - [\nabla_B h(x^k)]^{-1} h(y^i, \tilde{x}^k_R), \qquad i = 1, 2, \cdots.$$

(The inverse of the basic Jacobian at x^k has already been computed to evaluate λ^k and r^k.)

The convergence of this method for obtaining a feasible point has been established by Kantorovitch [18], provided that the starting point is sufficiently close to x^k and that the matrices H_i and $\nabla_B h(x)^{-1}$ are bounded in this neighborhood. It may be necessary to decrease the step length θ_k of the tangent phase, initially defined by the Lagrangian step size α_k, by scaling down θ_k by a factor $\rho_1 \in (0, 1)$, possibly several times, until the restoration phase is successful.

Assuming that the level sets of the Lagrangian function

$$\mathscr{L}[f(x^k)] = \{x \in R^n | l(x, \lambda^k) \leqq f(x^k)\}$$

are compact, the Lagrangian step sizes α_k are bounded. There exists, therefore, a neighborhood of a solution x^* of (P) and a corresponding integer N such that, for

all $k > N$, the iterates x^k are in this neighborhood and $\|r^k\|$ is small enough to guarantee the convergence of the modified Newton's method from the starting point $y^o = \tilde{x}_B^k(\alpha_k)$.

It is important to provide a rule for the step size which insures that the sequence $\{x^k\}$ is convergent. Each iteration must result in a descent in the value of the objective, and convergence can be established if this improvement is sufficient enough. Sufficient descent is achieved in our algorithm by enforcing the test for the step length θ_k first proposed by Armijo [5] in the framework of unconstrained optimization; namely, θ_k is scaled down by a factor $\rho_2 \in (0, 1)$ until

$$f[x^k(\theta_k)] < f(x^k) - \sigma\theta_k\|r^k\|^2, \tag{11}$$

where σ is a positive parameter chosen in $(0, \frac{1}{2})$. The Taylor expansion of f, considered as a function of θ, leads to

$$f[x^k(\theta_k)] - f(x^k) = -\theta_k\|r^k\|^2 + O(\theta_k^2).$$

Hence, after at most a finite number of scalings by the factor $\rho_2 \in (0, 1)$ from the initial determination $\theta_k = \alpha_k$, the test (11) will be satisfied.

We can give now a detailed description of the algorithm in a pseudo-ALGOL format. The method depends on the parameters ε, σ, ρ_1, and ρ_2 which must be specified in advance, with $\varepsilon > 0$, $\sigma \in (0, \frac{1}{2})$, $\rho_1 \in (0, 1)$, $\rho_2 \in (0, 1)$. The tolerance parameter ε expresses the accuracy required in the satisfaction of the constraints. The damping factors ρ_1, ρ_2 are selected according to the nonlinearity of the problem. (They are taken as $\frac{1}{2}$ or $\frac{1}{10}$ in the GRG method of Abadie [2]).

GRSD ALGORITHM (Generalized reduced steepest descent method).

Step 0. Select a feasible $x^0 \in R^1$; set $k = 0$.

Step 1. Procedure "check regularity assumption":
if x^k is not regular, then stop; else partition

$$x^k = (x_B^k, x_R^k) \quad \text{and} \quad \nabla h(x^k) = [B^k, D^k].$$

Step 2. Compute the reduced Lagrange multiplier:

$$\lambda^k = -\nabla_B f(x^k)(B^k)^{-1};$$

compute the reduced gradient:

$$'r^k = \nabla_R f(x^k) + \lambda^k D^k.$$

Step 3. Procedure "stopping rule":
if $r^k = 0$, then stop; *comment*: x^k is a solution candidate.

Step 4. Procedure "move in the tangent plane":
compute the direction $p^k = T^k r^k$;

Step 5. Compute the Lagrangian stepsize α_k such that

$$\alpha_k = \text{Argmin}\,\{l(x^k - \alpha p^k, \lambda^k) | \alpha \leqq 0\};$$

set $\theta = \alpha_k$ and $\tilde{x}^k(\theta) = x^k - \theta p^k$.

Step 6. Procedure "restoration of the constraints":
set $i = 0$; set $y^0 = \tilde{x}_B^k(\theta)$;
while $(\|h[y^i, \tilde{x}_R^k(\theta)]\| > \varepsilon)$ and $i <$ itermax do:
set $y^{i+1} = y^i - (B^k)^{-1}h(y^i, \tilde{x}_R^k)$ and set $i = 1 + 1$.

Step 7. If $\|h[y^f, \tilde{x}_R^k(\theta)]\| > \varepsilon$, then
set $\theta = \rho_1 \theta$, go to Step 6;
else set $x^k(\theta) = (y^f, \tilde{x}_R^k(\theta))$.
Step 8. If $f[x^k(\theta)] > f(x^k) - \theta\sigma\|r^k\|^2$,
then set $\theta = \rho_2 \theta$, go to Step 6.
Step 9. Set $\theta_k = \theta$ and $x^{k+1} = x^k(\theta_k)$.
Step 10. Set $k = k+1$, and go to Step 1.

3.4. Convergence properties. We prove first that the rule given by (11), adopted to determine the step length, guarantees the convergence of the algorithm.

THEOREM 1 (Global convergence). *Assume that f is bounded from below, and that the level set $\mathscr{L}[f(x^0)]$ is compact. Let $\{x^k\}$ be the sequence of regular feasible points constructed by the GRSD algorithm. Then every cluster point of $\{x^k\}$ is a critical point.*

Proof. It follows from the assumptions that the sequence $\{f(x^k)\}$ is monotonically decreasing and that θ_k is positively bounded from below. □

The above theorem, as well as establishing global convergence, also guarantees that, after a finite number of iterations, $\|r^k\|$ is small enough so that the restoration phase does not offer any computational difficulties. We can now prove also that the Lagrangian step size α_k satisfies the test (11) for k large enough.

PROPOSITION 1. *Let $\{x^k\}$ be a sequence, constructed by the GRSD algorithm, converging to x^*, a critical point of* (P) *which satisfies the sufficient second order optimality conditions. There exists an integer N, such that, for all $k > N$, the step lengths θ_k are determined directly by the values of the Lagrangian step sizes α_k.*

Proof. The definition of the Lagrangian step size leads to

$$\alpha_k = \frac{\|r^k\|^2}{{}^t r^k {}^t T^k L^k T^k r^k} + O(\|r^k\|),$$

which yields

$$f[x^k(\alpha_k)] - f(x^k) = -\frac{1}{2}\frac{\|r^k\|^4}{{}^t r^k {}^t T^k L^k T^k r^k} + O(\|r^k\|^3). \tag{12}$$

Assuming that x^* fulfills the sufficient second order optimality conditions [13], the matrix ${}^t T^* L^* T^*$ is positive definite (at x^* and therefore in a domain around x^*); hence, for k large enough the first term of the right-hand side of (12) is negative and dominates the second term which is only of the order $O(\|r^k\|^3)$. Thus, for k large enough, the point $x^k(\alpha_k)$ satisfies the test of Step 8, and we can make the choice $\theta = \alpha_k$; then $x^{k+1} = x^k(\alpha_k)$. □

The introduction of the Lagrangian step size is a very powerful device. We have just established that, once the algorithm has approached close enough to a solution, this parameter defines a tangent move from which the restoration phase can be performed successfully without using complex scaling down procedures. This choice leads therefore to a computationally simple and convergent algorithm. We can further prove that the convergence itself is efficient by showing that the algorithm converges at the natural rate.

THEOREM 2 (Local rate of convergence). *Assume that the sequence* $\{x^k\}$ *constructed by the GRSD algorithm converges to* x^*, *an isolated local minimizer of* f, *subject to the constraints* $h(x)=0$. *Let* M *and* m *be, respectively, the largest and smallest eigenvalues of the matrix* ${}^tT^*L^*T^*$, *the restriction of the Hessian of the Lagrangian to the tangent subspace to* $\mathcal{S}$ *at* x^*. *Then the sequence* $\{x^k\}$ *converges linearly to* x^* *with asymptotic ratio* $(M-m/M+m)^2$.

Proof. The proof is a generalization of a similar estimate for the rate of convergence of the optimal steepest descent method.

For small r^k, equation (12) gives an estimate of the decrease of the objective function during the kth iteration:

$$f(x^{k+1})-f(x^k)=-\frac{1}{2}\frac{\|r^k\|^4}{{}^tr^{k}{}^tT^kL^kT^kr^k}+O(\|r^k\|^3). \tag{13}$$

Introducing the error vector $y^k=x^k-x^*$, we have

$$\begin{aligned} f(x^k)-f(x^*)&=l(x^k,\lambda^k)-l(x^*,\lambda^k)\\ &=\nabla l(x^k,\lambda^k)y^k-\tfrac{1}{2}{}^ty^kL^ky^k+O(\|y^k\|^3). \end{aligned}$$

Let z^k be the vector of independent components of y^k. Using a first order Taylor expansion of the implicit function $y^k=\psi(z^k)$, we obtain

$$y^k=T^kz^k+O(\|z^k\|^2)=O(\|z^k\|).$$

Using a first order Taylor expansion of ∇l, we derive an estimate for r^k:

$${}^tr^k=[{}^ty^kL^k+O(\|y^k\|^2)]T^k={}^tz^{k}{}^tT^kL^kT^k+O(\|z^k\|^2).$$

Hence

$$f(x^k)-f(x^*)=\tfrac{1}{2}{}^tz^{k}{}^tT^kL^kT^kz^k+O(\|z^k\|^3),$$

or, in terms of r^k,

$$f(x^k)-f(x^*)=\tfrac{1}{2}{}^tr^k({}^tT^kL^kT^k)^{-1}r^k+O(\|y^k\|^3).$$

Denoting by F^k the matrix ${}^tT^kL^kT^k$, we have, for the ratio of the successive errors,

$$\frac{f(x^{k+1})-f(x^*)}{f(x^k)-f(x^*)}=\left(1-\frac{({}^tr^kr^k)^2}{{}^tr^kF^k{}^tr^k(F^k)^{-1}r^k}\right)(1+O(\|y^k\|)).$$

Let us introduce the normalized vectors $u^k=r^k/\|r^k\|$, which converge to u^*; the Kantorovitch inequality [17] gives for the positive definite matrix F^*,

$$\frac{({}^tu^*u^*)^2}{{}^tu^*F^*u^*{}^tu^*F^{*-1}u^*}\geqq\frac{4mM}{(M+m)^2}.$$

Hence we obtain the desired result:

$$\lim_{k\to\infty}\frac{f(x^{k+1})-f(x^*)}{f(x^k)-f(x^*)}\leqq\left(\frac{M-m}{M+m}\right)^2. \qquad \square$$

We have thus been able to establish that the convergence characteristics of the GRSD algorithm are simple and complete extensions of the corresponding

properties of the unconstrained steepest descent method. The algorithm has been run satisfactorily on several examples, derived from Colville's tests [7] (where only the constraints active at the optimum were considered and treated as equality constraints), and its computational performance has been consistently comparable with the results obtained with a GRG algorithm of similar sophistication. We describe in the next subsection certain situations where the new method performs even better than the original scheme.

3.5. Comparison with the generalized gradient method. Since the GRSD method and Abadie's GRG algorithm [2] use the same procedures to determine the direction of the tangent phase and to perform the restoration, the step length determination is the key difference between the two methods. In the latter, the point in the tangent direction is initially chosen to achieve a local minimum of the objective function. Although the resulting performance of the GRG algorithm is often satisfactory [3], this selection rule does not constitute the proper extension of the optimal steepest descent method and does not exhibit the efficient properties achieved by the Lagrangian step size in the GRSD algorithm.

For example, if the initial choice of the step length is systematically small compared to the ideal step size, the convergence of the GRG algorithm may be slowed. Such a situation arises when the restricted Hessian $'TFT$ of the objective is ill-conditioned compared to the corresponding Hessian $'TLT$ of the Lagrangian. This phenomenon is illustrated by the following problem:

$$\text{Minimize } 5x_1^2+3x_2^2+5x_3^2+x_4^2-9x_1+7x_2-x_3-6x_4$$

$$\text{Subject to } h_1(x)=x_1^2+x_2^2+x_3^2+x_4^2+x_1-7x_2+3x_3-5x_4+4=0,$$

$$h_2(x)=2x_1^2+x_2^2+2x_3^2+3x_2+5x_3-4x_4-9=0.$$

The objective achieves its minimum value 5 at the solution point $x^*=(1,1,1,1)$; the corresponding Lagrange multiplier vector is $\lambda^*=(1,-2)$. The Hessians at x^* of the objective and of the Lagrangian are respectively

$$F^*=\begin{bmatrix}10 & & & \\ & 6 & & \\ & & 10 & \\ & & & 2\end{bmatrix} \text{ and } L^*=\begin{bmatrix}4 & & & \\ & 4 & & \\ & & 4 & \\ & & & 4\end{bmatrix}.$$

The starting point is taken as the feasible point $x^0=(3,2,-1,4)$. The partition chosen treats x_2 and x_3 as the basic variables and x_1 and x_4 as the independent ones. While it takes 19 iterations of the GRG algorithm to reach an approximate solution where the reduced gradient is in norm less than 10^{-3}, the GRSD algorithm reaches the same precision in only 6 iterations. Since each step of both methods consists of the same operations and necessitates about the same amount of computational work, the GRSD method is about three times faster.

4. A generalized reduced variable metric method for nonlinearly constrained minimization. For unconstrained minimization, the conjugate gradient method [14] or the variable metric algorithm [13], which exhibit superlinear rates

of convergence, are sometimes preferred to the method of steepest descent. It is, therefore, natural to seek a way of combining these efficient schemes with the reduced gradient technique in order to solve constrained problems. But the appealing properties of these methods rely, to a substantial degree, on the fact that, at each step, the objective function is accurately minimized along the direction of search. This is not an obstacle when the constraints are linear, and simple as well as efficient combination schemes have been proposed in this framework [25].

The only available extension of the Fletcher–Powell method to nonlinearly constrained minimization has been proposed by Davies [9] in the context of Rosen's gradient projection [27]. The restoration phase is, however, a source of difficulty, ignored by Davies but acknowledged by Murtagh and Sargent [24], since the new feasible iterate is not likely to exactly achieve a local minimum of f. This leads to a possible deterioration of the convergence properties of the algorithm.

4.1. The idealized reduced variable metric method. A natural and efficient generalization to the constrained case can be provided within the implicit function framework we have already adopted to extend the method of steepest descent. The key idea consists again in viewing problem (P) in terms of the reduced unconstrained problem (R). The minimization of $\phi(x_R)$ is then, at least ideally, performed by the variable metric method in the subspace R of the independent variables. The kth iteration of this scheme proceeds from x_R^k by searching for the minimum of $\phi(x_R)$ along a direction s^k, defined by

$$s^k = G^k r^k,$$

where G^k is an $(n-m)\times(n-m)$ matrix updated according to the formula

$$G^{k+1} = G^k - \frac{G^k(r^{k+1}-r^k)'(r^{k+1}-r^k)G^k}{'(r^{k+1}-r^k)G^k(r^{k+1}-r^k)} + \frac{(x_R^{k+1}-x_R^k)'(x_R^{k+1}-x_R^k)}{'(x_R^{k+1}-x_R^k)(r^{k+1}-r^k)},$$

which approximates the inverse Φ, the Hessian of ϕ.

In practice it is necessary to solve the problem in the original space R^n, since the reduction to the form (R) can generally be achieved only conceptually. The ideal scheme consists, therefore, in defining a curve $\{x^k(\beta)\}$ on $\mathcal{S}$ emanating from $x^k = (x_B^k, x_R^k)$, its projection on R, parallel to the basic subspace B, being the straight line in the negative direction of s^k. To extend the Fletcher–Powell method, the next point x^{k+1} must be chosen to achieve a local minimum of f along the arc $\{x^k(\beta)|\beta \geqq 0\}$.

By construction, this method exhibits the convergence properties of the variable metric method in the $(n-m)$-dimensional subspace R. In particular, the rate of convergence of this variable metric method is actually superlinear. Moreover, the conceptual framework adopted shows that we have only to construct a sequence of $(n-m)\times(n-m)$ matrices G^k instead of $n\times n$ matrices as proposed in [9].

4.2. A generalized variable reduced metric method. The method developed in the previous paragraph is an idealized version, since it is computationally

impossible to generate the curves $\{x^k(\beta)\}$. But we can derive from it a practically implementable algorithm which asymptotically generates the same points. Again, this is achieved by a move along the direction $p^k = T^k s^k$ of the tangent subspace $\mathcal{T}^k$, followed by a restoration.

To obtain the best possible approximation of the arc of the idealized scheme, we are led, as in § 3.2, to define the step length parameter in terms of the Lagrangian step size, achieving a local minimum of the updated Lagrangian $l(x, \lambda^k)$ along p^k. We have established in § 3.3 that this provides a first order approximation to the step size of the ideal search for the point achieving a local minimum of f along the arc $x^k(\beta)$.

An even better method would be to adapt this generalization technique in conjunction with the version of the variable metric method proposed recently for the unconstrained situation by Oren and Luenberger [26], the self-scaling variable metric algorithm. It exhibits rapid convergence even when the minimization step is performed only approximately, while the Fletcher–Powell algorithm is adversely affected by even a small error in the step size.

5. A generalized Newton's method for nonlinearly constrained minimization. In spite of the very appealing fast convergence of Newton's method for the minimization of unconstrained convex functions (when second order information is available and when the dimension n of the problem is not too large to prohibit storage and inversion of an $n \times n$ matrix), very little effort has been devoted to extend the method to constrained situations. Levitin and Polyak [20] were the first to study a Newton's scheme for such cases. They proposed an implementable algorithm which considers only a linearized version of the constraints and which uses the inverse of the $n \times n$ Hessian F of the objective function to compute each iteration. This does not seem to be the most suitable approach, since it ignores the nonlinearity of $\mathcal{S}$ and therefore does not fully capture the essence of the problem to second order. It is preferable to explicitly incorporate the second order information available.

5.1. The idealized reduced Newton's method. An ideal method can again be conceived by viewing problem (P) in terms of the reduced problem (R) and by adopting Newton's method in the subspace R to find the unconstrained minimum of $\phi(x_R)$. The derivatives of ϕ have already been computed in § 3.1:

$$\nabla\phi(x) = {}^t r(x),$$
$$\Phi(x) = {}^t T(x) L[x, \lambda(x)] T(x).$$

Hence the kth iteration consists of a move from x_R^k along the (negative of the) direction

$$p^k = ({}^t T^k L^k T^k)^{-1} r^k. \tag{14}$$

To guarantee a descent in the value of the objective, we must assume that ${}^t TLT$ is positive definite and we must sometimes use a damping parameter $\theta \in (0, 1]$ to reduce the size of the step along $-p^k$, until the point

$$x_R^{k+1} = x_R^k - \theta_k p^k$$

satisfies a descent condition; a test like Armijo's rule, for example [5]. There is no need, however, to determine the step length by an accurate minimization procedure, as in the previous gradient-related methods, to obtain efficient convergence, since asymptotically $\theta_k = 1$ will yield convergence of order 2. If the matrix ${}^tT^kL^kT^k$ is not positive definite, p^k must be modified to preserve the descent character of the algorithm. Computationally efficient schemes [12], [16] for the unconstrained case can be applied in this case as well.

In the original space R^n, a step of this ideal scheme consists in moving along an arc $\{x^k(\beta)|0 \leqq \beta \leqq 1\}$ *of* $\mathscr{S}$, the projection on R of which is the straight line in the negative direction of p^k. By introducing this ideal scheme, we conclude that it is necessary to invert only $(n-m)\times(n-m)$ matrix. We can also derive from it a practical algorithm, by approximating the search along the arc to second order.

5.2. The generalized reduced Newton's algorithm. Let us consider the move of the independent variables

$$x_R^k(\theta) = x_R^k - \theta p^k, \quad \text{with} \quad \theta \in (0, 1].$$

To satisfy the constraint equations, the basic variables must be altered to

$$x_B^k(\theta) = \psi(x_R^k - \theta p^k) = \psi(x_R^k) - \theta \nabla\psi^k p^k + \frac{\theta^2}{2} {}^tp^k \cdot \Psi^k \cdot p^k + O(\|p^k\|^3),$$

$$x_B^k(\theta) = x_B^k - \theta(-B^{-1}D)p^k - \theta^2 B^{-1} q^k + O(\|p^k\|^3),$$

where q^k is now the m-dimensional column vector with components

$$q_i^k = \tfrac{1}{2} {}^tp^k {}^tT^k H_i(x^k) T^k p^k. \tag{15}$$

We thus obtain a second order approximation of the form

$$\tilde{x}^k(\theta) = \begin{bmatrix} \tilde{x}_B(\theta) \\ \tilde{x}_R(\theta) \end{bmatrix} = x^k - \theta T^k p^k - \theta^2 V^k q^k, \tag{16}$$

where

$$T^k = \begin{bmatrix} -(B^k)^{-1} D^k \\ I \end{bmatrix}, \qquad V^k = \begin{bmatrix} (B^k)^{-1} \\ 0 \end{bmatrix}.$$

From a geometric viewpoint, we can interpret this approximation as a move along the osculating parabola c^k to the ideal curve $\{x^k(\beta)|\beta > 0\}$, i.e., the parabola of origin x^k parameterized by θ as

$$\tilde{x}^k(\theta) = \theta t^k + \theta^2 v^k$$

in the 2-dimensional variety containing x^k and spanned by the vectors t^k and v^k, where

$$t^k = -T^k p^k, \qquad v^k = -V^k q^k.$$

This curve is the natural extension, for second order approximation, to the tangent t^k. (See Fig. 3.)

In general, however, the points $\tilde{x}^k(\theta)$ are not feasible, and a move back to $\mathscr{S}$ is again performed by altering the basic variables through a modified Newton's

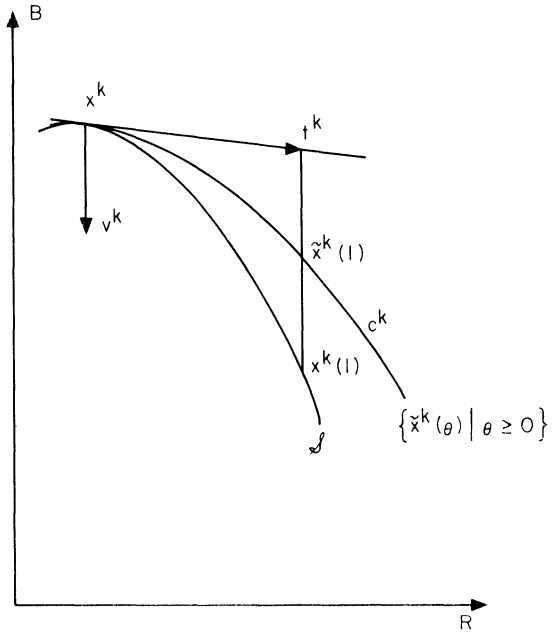

FIG. 3

method to solve the system

$$h[y, \tilde{x}_R^k(\theta)] = 0.$$

As mentioned in § 3.3, the restoration move can be a source of difficulty, since the modified Newton's method may fail to converge or may lead to a new feasible point which does not represent an improvement over x^k in the objective function. We handle these difficulties by successive halving of θ_k from the initial value $\theta_k = 1$ eventually finding a new feasible iterate x^{k+1} such that, given a scalar $\sigma \in (0, \frac{1}{2})$,

$$f(x^{k+1}) \leqq f(x^k) - \theta_k \frac{\sigma}{\|{}^t T^k L^k T^k\|} \|r^k\|^2. \tag{17}$$

We emphasize, however, that these difficulties are less frequent than with first order methods, since the approximations of the constraints used in the present scheme are valid to within second order.

We now present our algorithm. It is of the same form as the GRSD algorithm in § 3.3 except for Steps 3, 4 and 5.

GRN ALGORITHM (Generalized reduced Newton's method).

Step 0. Select a feasible $x^0 \in R^n$; set $k = 0$.

Step 1. Check regularity assumption.

Step 2. Compute λ^k, r^k.

Step 3. Stopping rule: if $r^k = 0$, then stop.

Step 4. Procedure "move along the osculating parabola":

compute $p^k = ({}^t T^k L^k T^k)^{-1} r^k$;

compute $q_i^k = \frac{1}{2} {}^t p^k {}^t T^k H_i^k T^k p^k$ for $i = 1, \cdots, m$.

Step 5. Set $\theta = 1$ and $\tilde{x}^k(\theta) = x^k - \theta T^k p^k - \theta^2 V^k q^k$;

Steps 6, 7. Restoration of the constraints.

Step 8. If $f[x^k(\theta)] > f(x^k) - \theta(\sigma/\|{}^tT^kL^kT^k\|)\|r^k\|^2$,
then set $\theta = \frac{1}{2}\theta$, go to Step 6.

Step 9. Set $\theta_k = \theta$ and $x^{k+1} = x^k(\theta_k)$.

Step 10. Set $k = k+1$ and go to Step 1.

Since $-p^k$ is a direction of descent, the test (17) is satisfied after at most a finite number of halvings of the original step size $\theta_k = 1$. This selection rule guarantees the convergence of a subsequence of $\{x^k\}$ to a critical point, as established in § 3.4.

We can also show that, after a finite number of iterations, no halving of the step length is necessary.

PROPOSITION 2. *Let $\{x^k\}$ be a sequence, constructed by the GRN algorithm, converging to x^*, an isolated local minimizer of* (P). *Assume that ${}^tT^*L^*T^*$ is positive definite. Then there exists an integer N, such that, for all $k > N$, we may take $\theta_k = 1$.*

Proof. An expansion of $f[\tilde{x}^k(1)]$ to second order gives

$$f[\tilde{x}^k(1)] = f(x^k) - \nabla f(x^k)(T^k p^k + V^k q^k) + \tfrac{1}{2}{}^t(T^k p^k + V^k q^k)F^k(T^k p^k + V^k q^k) + O(\|p^k\|^3).$$

Using (15), we obtain

$$f[\tilde{x}^k(1)] = f(x^k) - {}^tr^k p^k + \tfrac{1}{2}{}^t p^k {}^tT^k L^k T^k p^k + O(\|p^k\|^3).$$

By definition (16), we have

$$\|h[\tilde{x}^k(1)]\| = O(\|r^k\|^3).$$

Therefore, if $\|r^k\|$ is not too large, which occurs for k large enough since $r^k \to 0$, the modified Newton's method converges to a feasible point $x^k(1)$. We derive, using the definition (14) of p^k,

$$f[x^k(1)] = f(x^k) - \tfrac{1}{2}{}^tr^k({}^tT^kL^kT^k)^{-1}r^k + O(\|r^k\|^3). \tag{18}$$

For any $\sigma \in (0, \frac{1}{2})$, there exists an N large enough such that

$$f[x^k(1)] < f(x^k) - \frac{\sigma}{\|{}^tT^kL^kT^k\|}\|r^k\|^2 \quad \text{for all } k > N;$$

therefore the test of Step 8 is satisfied for the step length $\theta_k = 1$ and the new iterate $x^{k+1} = x^k(1)$. □

The estimate (18) shows that the choice $\theta_k = 1$ achieves, at least asymptotically, the best possible decrease in the objective along the parabola c^k.

THEOREM 3. *Assume that f and h_i are three times continuously differentiable. Assume that $\{x^k\}$ converges to x^*, an isolated local minimizer of* (P). *Then this convergence is of order at least* 2.

Proof. Let us introduce the error vector $y^k = x^k - x^*$ and partition it as $y^k = (w^k, z^k)$. Since x^{k+1} and x^* are feasible, we have

$$h(x^{k+1}) - h(x^*) = 0 = \nabla h(x^*)y^{k+1} + O(\|y^{k+1}\|^2) = \nabla_B h(x^*)w^{k+1} + \nabla_R h(x^*)z^{k+1} + O(\|y^{k+1}\|^2).$$

Since h is uniformly regular, we get

$$\|w^{k+1}\| = O(\|z^{k+1}\|).$$

According to Proposition 2, we have, for k large enough,

$$x_R^{k+1} = x_R^k - ({}^tT^k L^k T^k)^{-1} r^k.$$

The study of this iterative process shows that its convergence is of second order. Hence

$$\|z^{k+1}\| = O(\|z^k\|^2).$$

Since

$$\|y^{k+1}\| \leqq \|z^{k+1}\| + \|w^{k+1}\| = O(\|y^k\|^2),$$

we obtain the rate of convergence of order 2:

$$\|x^{k+1} - x^*\| \leqq c\|x^k - x^*\|^2. \qquad \square$$

Recently Mangasarian [22] has proposed a Newton's method for nonlinearly constrained minimization which exhibits quadratic convergence. The role of the Lagrange multipliers is also central to his approach, although he uses more general Lagrangian functions than in this paper. Feasibility is not required at each iteration, but it is necessary to compute the inverse of an $n \times n$ matrix.

6. Numerical experience. The GRN algorithm has been tested on the quadratic problem described in § 3.5. Convergence was quite rapid. From the same starting point as used before, the problem was solved in 3 iterations, yielding a value for the solution with 7 exact digits.

A nonquadratic test problem in 5 variables and 3 constraints was also run. From an initial approximation defined as the solution rounded to one decimal place, full precision was achieved after a single iteration.

7. Conclusion. The algorithms presented in this paper are of both practical and theoretical interest for nonlinearity constrained minimization. On the practical side, our methods efficiently generalize to this class of problems the appealing convergence properties of the optimum steepest descent method, the variable metric method, and Newton's method for unconstrained minimization. Computational results indicate that they can provide significant savings in computer time as compared to the existing schemes, particularly when the constraints are highly nonlinear. There is, of course, room for further improvement. The requirement of maintaining feasibility may cause excessive time expenditure in the restoration phase of each iteration if a high degree of accuracy is demanded in the satisfaction of the constraints. It is, however, possible to adaptively improve the accuracy requirements of the restoration phase as the minimization procedure progresses. Further investigation of such restoration schemes might, therefore, lead to faster computational performance. Other areas for further research include the extension of our methods to problems with inequality constraints and the development of effective rules for updating the partition between basic and independent variables.

On the theoretical side, our study has shown how to fully exploit the viewpoint associated with the implicit function theorem in order to define computational algorithms for the solution of nonlinear programming problems. One of the key observations in this perspective is the necessity of constructing a sequence of approximate Lagrange multipliers $\{\lambda^k\}$ simultaneously with the sequence of approximate solutions $\{x^k\}$. The resulting interplay of Lagrangian methods in a primal framework should lead to useful new results.

REFERENCES

[1] J. ABADIE, J. CARPENTIER AND C. HENSGEN, *Generalization of the Wolfe reduced gradient method to the case of nonlinear constraints*, ES/TIMS Joint European Meeting, Warsaw, Poland, 1966 (available from Electricité de France, EDF Note HR 7595, April 5th 1967).

[2] J. ABADIE AND J. GUIGOU, *Gradient réduit généralisé*, EDF Note HI 069/02, April 15th, 1969.

[3] ———, *Numerical experiments with the* GRG *method*, Integer and Nonlinear Programming, J. Abadie, ed., North-Holland, Amsterdam, 1970, pp. 529–536.

[4] M. ALTMAN, *A generalized gradient method for the conditional minimum of a functional*, Bull. Acad. Polon. Sci. Sér. Sci. Math., 14 (1966), pp. 445–451.

[5] L. ARMIJO, *Minimization of functions having Lipschitz-continuous first partial derivatives*, Pacific J. Math., 16 (1966), pp. 1–3.

[6] A. CAUCHY, *Méthode générale pour la résolution des systèmes d'équations simultanées*, C. R. Acad. Sci. Paris, 25 (1847), pp. 536–538.

[7] A. R. COLVILLE, *A comparative study on nonlinear programming codes*, IBM Tech. Rep. 320-2949, New York Scientific Center, 1968.

[8] G. DANTZIG, *Linear Programming and Extensions*, Princeton University Press, Princeton, N.J., 1963.

[9] D. DAVIES, *Some practical methods of optimization*, Integer and Nonlinear Programming, J. Abadie, ed., 1970, pp. 87–117.

[10] J. DIEUDONNE, *Foundations of Modern Analysis*, Academic press, New York, 1960.

[11] P. FAURE AND P. HUARD, *Résolution de programmes mathématiques à fonction non linéaire par la méthode du gradient réduit*, Rev. Française Recherche Opérationnelle, 9 (1965), pp. 167–206.

[12] A. V. FIACCO AND G. P. MCCORMICK, *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*, Wiley, New York, 1968.

[13] R. FLETCHER AND M. J. D. POWELL, *A rapidly convergent descent method for minimization*, Comput. J., 6 (1963), pp. 163–168.

[14] R. FLETCHER AND C. M. REEVES, *Function minimization by conjugate gradients*, Ibid., 7 (1964), pp. 149–154.

[15] D. GABAY, *Efficient convergence of reduced gradient algorithms for nonlinearly constrained optimization*, Ph.D. dissertation, Dept. of Engineering-Economic Systems, Stanford University, Palo Alto, Calif., 1973.

[16] J. GREENSTADT, *On the relative efficiencies of gradient methods*, Math. Comp., 21 (1967), pp. 360–367.

[17] L. V. KANTOROVITCH, *Functional analysis and applied mathematics*, Uspehi Mat. Nauk., 3 (1948), pp. 89–185; English transl., Nat. Bur. of Standards Rep. No. 1509, 1952.

[18] L. V. KANTOROVITCH AND G. P. AKILOV, *Functional Analysis in Normed Spaces*, Fizmatgiz, Moscow, 1959; English transl., Pergamon Press–Macmillan, New York, 1964.

[19] H. J. KELLEY AND J. L. SPEYER, *Accelerated gradient projection*, Symposium on Optimization, Nice, 1969, A. V. Balakrishnan, ed., Springer-Verlag, Berlin, 1970, pp. 151–158.

[20] E. S. LEVITIN AND B. T. POLYAK, *Constrained Minimization Methods*, Ž. Vyčisl. Mat. i Mat. Fiz., 6 (1966), pp. 787–823 = USSR Comp. Math. and Math. Phys., 6 (1968), pp. 1–50.

[21] D. G. LUENBERGER, *Introduction to Linear and Nonlinear Programming*, Addison-Wesley, Reading, Mass., 1973.

[22] O. L. MANGASARIAN, *Unconstrained Lagrangians in nonlinear programming*, Tech. Rep. 174, Computer Sciences Department, University of Wisconsin, Madison, 1973.

[23] A. MIELE, H. Y. HUANG AND J. C. HEIDEMAN, *Sequential gradient restoration algorithm for the minimization of constrained functions—Ordinary conjugate gradient versions*, J. Optimization Theory Appl., 4 (1969), pp. 213–243.

[24] B. A. MURTAGH AND R. W. H. SARGENT, *A constrained minimization method with quadratic convergence*, Optimization, R. Fletcher, ed., Academic Press, London, 1969, pp. 215–246.

[25] G. P. MCCORMICK, *The variable reduction method for nonlinear programming*, Management Sci., 17, (1970), pp. 146–160.

[26] S. S. OREN AND D. G. LUENBERGER, *The self-scaling variable metric algorithm* (SSVM), Fifth Hawaiian Internat. Conf. on System Sciences, 1972.

[27] J. B. ROSEN, *The gradient projection method for nonlinear programming. Part I: Linear constraints*; *Part II: Nonlinear constraints*, J. Soc. Indust. Appl. Math., 8 (1960), pp. 181–217; 9 (1961), pp. 514–532.

[28] R. WOLFE, *Methods of nonlinear programming*, Recent Advances in Mathematical Programming, R. L. Graves and P. Wolfe, eds., McGraw Hill, New York, 1963, pp. 67–86.