optimization 20 (1989) 4, 421–450 Akademie-Verlag Berlin

On the Local and Global Convergence of a Reduced Quasi-NEWTON Method[1]

J. CH. GILBERT

International Institute for Applied Systems Analysis (IIASA), Laxenburg

Summary: In optimization in $\mathbb{R}^n$ with m nonlinear equality constraints, we study the local convergence of reduced quasi-NEWTON methods, in which the updated matrix is of order $n-m$. Furthermore, we give necessary and sufficient conditions for superlinear convergence (in one step) and we introduce a device to globalize the local algorithm. It consists in determining a step along an arc in order to decrease an exact penalty function and we give conditions so that asymptotically the step-size will be equal to one.

AMS 1980 Subject Classification:

Primary: 49 D 05; Secondary: 49 D 30, 65 K 05

Key words: Constrained optimization, successive quadratic programming, reduced quasi-NEWTON method, superlinear convergence, exact penalty function, arc search, step-size selection procedure, global convergence.

1. Introduction

Let ω be an open convex set in $\mathbb{R}^n$, f be a real-valued function on ω and c map ω to $\mathbb{R}^m$, where $m \leqslant n$. We shall suppose that f and c are functions of class C_b^ν with $\nu \geqslant 3$, i.e. f and c are supposed three times continuously differentiable with bounded derivatives on ω. We shall endow $\mathbb{R}^n$ with its canonical basis and with the Euclidean scalar product. We are interested in algorithms for solving the following minimization problem with equality constraints:

$$\min \{f(x) \mid x \in \omega,\ c(x) = 0\}\ . \tag{1.1}$$

In addition to the smoothness of f and c, we shall assume that c is a *submersion* on ω, that is to say that the $m \times n$ Jacobian matrix

$$A_x := A(x) \tag{1.2}$$

of partial derivatives of c at x is supposed surjective for all x in ω. If ω is "large" this is a very strong hypothesis, but it is usual to suppose that the gradients of the constraints are linearly independent at a solution of (1.1) and, therefore, this hypothesis is satisfied in a neighbourhood of a solution. Then, if x_* is a local minimizer for problem (1.1), there exists a unique LAGRANGE multiplier λ_* so that the

[1] Work supported in part by INRIA (Institut National de Recherche en Informatique et en Automatique), France and in part by the FNRS (Fonds National de la Recherche Scientifique), Belgium.

first order optimality conditions are satisfied at $x=x_*$ and $\lambda=\lambda_*$:

$$\begin{cases} c(x)=0\,, \\ \nabla f(x)+A_x^{\mathsf{T}}\lambda=0\,, \end{cases} \tag{1.3}$$

where $\nabla f(x)$ is the vector of partial derivatives of f at x. The quantity on the left hand side of the second equation is the first derivative with respect to x of the Lagrangian $l(x,\lambda):=f(x)+c(x)^{\mathsf{T}}\lambda$. The second order sufficient condition will also be assumed: the $n\times n$ Hessian matrix L_* of second derivatives with respect to x of l at (x_*,λ_*) is supposed positive definite in the null space $N(A_*)$ of $A_*:=A(x_*)$. For further references, we gather these hypotheses under the name of

Assumption A:

- f, c are in $C_b^v(\omega)$ with $v\geqq 3$,
- c is a submersion,
- (x_*,λ_*) satisfies (1.3) at $x=x_*$ and $\lambda=\lambda_*$,
- $h^{\mathsf{T}}L_*h>0$ for all h in $\mathbb{R}^n$ with $h\neq 0$ and $A_*h=0$.

Quasi-Newton methods, also called variable metric or secant methods, are methods for solving a system of nonlinear equations on $\mathbb{R}^N$, say $F(x_*)=0$. They generate a sequence of points (x_k) and a sequence of nonsingular matrices (J_k) of order N from the data of a point x_1, and a nonsingular matrix J_1 by the formula:

$$x_{k+1}=x_k-J_k^{-1}F(x_k)\,,$$

where J_k is updated at each iteration according to the following scheme:

$$\begin{aligned} &J_{k+1}=U(J_k,\gamma_k,\sigma_k)\,, \\ &\gamma_k:=F(x_{k+1})-F(x_k)\,, \\ &\sigma_k:=x_{k+1}-x_k\,. \end{aligned}$$

The rule U is designed in order that J_{k+1} will satisfy the secant equation $J_{k+1}\sigma_k=\gamma_k$ and then will improve the approximation by J_k of the Jacobian matrix $\nabla F(x_*)$ at the solution x_*. These methods are particularly attractive because second order derivatives need not be calculated and because a superlinear rate of convergence for (x_k) can be obtained (see [16]), i.e.

$$\frac{\|x_{k+1}-x_*\|}{\|x_k-x_*\|}\to 0 \quad \text{as} \quad k\to\infty\,. \tag{1.4}$$

Such a method could be used for solving system (1.3), but matrices of *order* $n+m$ should be updated. The aim of this paper is to introduce and study quasi-Newton methods that require the update of matrices of order $n-m$ only.

The *Successive Quadratic Programming* (SQP) method proposed by Wilson [38] and Han [28] improves the method described above with regard to the order of the updated matrices. In this method x_{k+1} is obtained from x_k by solving a quadratic programme with linear constraints:

$$\begin{cases} \min f'(x_k)\cdot(x-x_k)+\dfrac{1}{2}(x-x_k)^{\mathsf{T}}L_k(x-x_k)\,, \\ x\in\mathbb{R}^n \quad \text{and} \quad c(x_k)+c'(x_k)\cdot(x-x_k)=0\,, \end{cases} \tag{1.5}$$

where we have used a dot to separate the linear operators $f'(x_k)$ and $c'(x_k)$ from their argument $(x-x_k)$. The matrix L_k of *order* n is updated in order to approximate L_*, the Hessian of the Lagrangian. Under assumption A and the non singularity of L_k on $N(A_k)$, the solution of (1.5) can be written in the form (see [20]):

$$x_{k+1}=x_k-A_k^-c(x_k)-Z_k^-(Z_k^{-\mathsf{T}}L_kZ_k^-)^{-1}\left(g(x_k)-Z_k^{-\mathsf{T}}L_kA_k^-c(x_k)\right), \tag{1.6}$$

where A_k^- is any right inverse of $A_k:=A(x_k)$, Z_k^- is any $n\times(n-m)$ matrix whose columns form a basis of the tangent space $N(A_k)$ to the manifold $M_k:=c^{-1}(c(x_k))$ at x_k and $g(x_k)$ is the *reduced gradient* defined at $x\in\omega$ by

$$g(x):=Z_x^{-\mathsf{T}}\nabla f(x)\in\mathbb{R}^{n-m}. \tag{1.7}$$

The first part $(-A_k^-c(x_k))$ of the displacement in (1.6) is a *restoration step*, i.e. a Newton-like step for solving $c(x_*)=0$. It belongs to $R(A_k^-)$ which is a complementary space to $N(A_k)$ in $\mathbb{R}^n$. The second part of the displacement in (1.6) is a *minimization step* belonging to the tangent space $R(Z_k^-)=N(A_k)$ to M_k at x_k.

Let us consider for a while the case where the constraints are linear:

$$c(x):=Ax+b=0, \tag{1.8}$$

where A is an $m\times n$ surjective matrix and b is a vector in $\mathbb{R}^m$. As in the nonlinear case, let us introduce Z^-, an $n\times(n-m)$ injective matrix whose columns generate $N(A)$: $AZ^-=0$. Suppose that the first iterate belongs to the plane of the constraints: $c(x_1)=0$. Then, any point x satisfying the constraints (1.8) can be expressed by using a reduced variable u in $\mathbb{R}^{n-m}$ $(x=x_1+Z^-u)$ and the problem of minimizing f subject to the constraints (1.8) is equivalent to that of minimizing $\Phi(u):=f(x_1+Z^-u)$ on $\{u\in\mathbb{R}^{n-m}\mid x_1+Z^-u\in\omega\}$:

$$\min\{\Phi(u)\mid u\in\mathbb{R}^{n-m},\ x_1+Z^-u\in\omega\}. \tag{1.9}$$

By considering the optimality equation $\nabla\Phi(u_*)=Z^{-\mathsf{T}}\nabla f(x_1+Z^-u_*)=0$, a quasi-Newton method for solving problem (1.9) generates a sequence (u_k) in $\mathbb{R}^{n-m}$ and a sequence (G_k) of nonsingular matrices of order $n-m$. On the one hand, we have

$$u_{k+1}=u_k-G_k^{-1}\nabla\Phi(u_k).$$

By setting $x_k:=x_1+Z^-u_k$, we obtain

$$x_{k+1}=x_k-Z^-G_k^{-1}g(x_k), \tag{1.10}$$

where $g(x):=Z^{-\mathsf{T}}\nabla f(x)$ is the reduced gradient of f at x. On the other hand, G_k is updated as follows:

$$\begin{cases}G_{k+1}=U(G_k,\gamma_k,\sigma_k),\\ \gamma_k:=g(x_{k+1})-g(x_k),\\ Z^-\sigma_k:=x_{k+1}-x_k,\end{cases} \tag{1.11}$$

in order to approximate $\nabla^2\Phi(u_*)$, the Hessian of Φ at u_*, which is also the *reduced Hessian of* f *at* x_*:

$$Z^{-\mathsf{T}}\nabla^2 f(x_*)\,Z^-.$$

The conditions so that the sequence (u_k) will converge superlinearly can be satisfied and because of the injectivity of Z^-, the same conditions will assure the superlinear convergence of the sequence (x_k).

Algorithm (1.10)–(1.11) is a *reduced quasi-*NEWTON *method* because the order of the updated matrices is $n-m$ rather than n the SQP method. Our aim is to study such methods in the case of nonlinear constraints. They are particularly well adapted to problems with n large and $n-m$ small. That situation appears for example in the parametric identification of nonlinear sources in elliptic partial differential equations. If a finite element method is used to discretize the state equations (the constraints), m is large, say of the order of 1000, whereas the number $n-m$ of identifiable parameters is usually small: 2 or 3 in the example given by BLUM, GILBERT and THOORIS [3]. In that case, a reduced quasi-NEWTON method is usable while the SQP method is not because of the order of the matrices that should be updated. Another advantage in developing reduced quasi-NEWTON methods comes from the fact that, under Assumption A, the projected Hessian of the Lagrangian is positive definite at the optimum. Therefore, positive definite quasi-NEWTON approximations of that operator can be generated, in particular by the BFGS formula, which is at present widely believed to be the best update formula. Thus, we see that reduced quasi-NEWTON methods appear rather natural. So, it is important to generalize algorithm (1.10)–(1.11) in case the constraints are nonlinear.

This can be done by using the implicit function theorem in order to obtain a reduced objective function:

$$\Phi(u)=f(\xi(u)),$$

where $\xi: u\in V\subset\mathbb{R}^{n-m}\to\xi(u)\in\omega\subset\mathbb{R}^n$ is a parametric representation of the regular manifold $c^{-1}(0)$ around $x_*:=\xi(u_*)$. We have $c(\xi(u))=0$ for all u in the neighbourhood V of u_*. This is the basic idea of methods like the Generalized Reduced Gradient (GRG) method (ABADIE and CARPENTIER [1]). In fact, the parametric representation $\xi(u)$ is usually not known and this leads to several difficulties. Because the method asks the generated sequence (x_k) to be feasible ($c(x_k)=0$ for all k), and because this cannot be achieved exactly in practice, some criterion has to be introduced to decide when to stop the restoration steps, i.e. how well the equality $x_{k+1}=\xi(u_{k+1})$ has to be realized (MUKAI and POLAK [33]). Another difficulty appears when x_k is far from x_* and a step-size has to be introduced in the u-space in order to globalize the method. Indeed, every time a step-size is tried, an infinite number of restoration steps have to be done: see [18], [21], [33], [19].

On the other hand, some non-feasible reduced quasi-NEWTON methods have been developed recently. GABAY [20] has studied the following algorithm:

$$x_{k+1}=x_k+r_k^1+t_k^1, \tag{1.12}$$

$$r_k^1:=-A_k^-c(x_k), \tag{1.13}$$

$$t_k^1:=-Z_k^-G_k^{-1}g(x_k), \tag{1.14}$$

where A_k^- is any right inverse of A_k, Z_k^- is any $n\times(n-m)$ matrix whose columns form a basis of the tangent space $N(A_k)$, G_k is a nonsingular matrix of order $(n-m)$ and $g(x_k)$ is the reduced gradient of f at x_k. The *tangent step* t_k^1 in (1.12), tangent to the manifold M_k, has the same structure as the displacement in (1.10) except for the basis Z_k^-, which changes here at each iteration. The *restoration step* r_k^1 in (1.12) is introduced to improve the feasibility of the sequence. The displacement in (1.12) can also be deduced from the displacement (1.6) of the SQP method by dropping the last part of the minimization step and by considering G_k as an approximation of the projected Hessian $Z_k^{-\top}L(x_k, \lambda_k)\, Z_k^-$.

For their part, Coleman and Conn [11] have studied the following algorithm:

$$x_{k+1} = x_k + r_k^2 + t_k^2 , \tag{1.15}$$

$$r_k^2 := -A_k^- c\,(x_k + t_k^2) , \tag{1.16}$$

$$t_k^2 := -Z_k^- G_k^{-1} g(x_k) , \tag{1.17}$$

where Z_k^- is an $n\times(n-m)$ matrix whose columns form an orthogonal basis of $N(A_k)$ and A_k^- is the Penrose pseudo-inverse of A_k: $A_k^- := A_k^\top (A_k A_k^\top)^{-1}$. The relevant difference to algorithm (1.12)–(1.14) lies in the restoration step. Here, the constraints are evaluated at $x_k + t_k^2$, after the tangent step, rather than at x_k in algorithm (1.12)–(1.14).

The study of algorithms (1.12)–(1.14) and (1.15)–(1.17) showed that when the matrices G_k are suitably updated and the initial point x_1 is close to x_*, the sequence (x_k) generated by any of those algorithms converges to x_* superlinearly in two steps, that is to say:

$$\frac{\|x_{k+1} - x_*\|}{\|x_{k-1} - x_*\|} \to 0 \quad \text{as} \quad k \to \infty .$$

This rate of convergence is not so good as the rate (1.4) obtained with reduced quasi-Newton methods when the constraints are linear or with the SQP method. On the other hand, counter-examples have been given by Byrd [8] and Yuan [39] for which both the methods of Gabay and Coleman and Conn do not converge better than with a two step superlinear rate of convergence. Therefore, the question of the rate of convergence of reduced quasi-Newton methods seemed closed. However, Byrd [7] and Gilbert [22] have shown (independently) that the sequence $(x_k + t_k^2)$ of Coleman and Conn's algorithm converges superlinearly in one step. A similar result has also been obtained by Hoyer [30] who considers algorithm (1.15)–(1.17) when G_k is the exact reduced Hessian of the Lagrangian: $G_k = Z_k^{-\top} L(x_k, \lambda_k)\, Z_k^-$ and λ_k is a Lagrange multiplier estimate. This fact makes this method competitive. Indeed, the quasi-Newton version of Coleman and Conn's algorithm (see [13]) needs two linearizations of the constraints per iteration. Consequently, as mentioned by Byrd [7], it was thought that a superlinear step was made in this algorithm for every four constraint linearizations. The convergence result of Byrd [7], Hoyer [30] and Gilbert [22] shows however that one superlinear step is made for every two constraint

linearizations. Furthermore, it can be shown (see [25]) that the use of an update criterion allows linearization of the constraints only once per iteration. Therefore, the method makes a superlinear step for each constraint linearization.

In this paper, we first show how algorithm (1.15)–(1.17) can be obtained from a very general principle. If unconstrained optimization problems are related to the solution of nonlinear equations (via the optimality condition $f'(x_*)=0$), we see from (1.3) that constrained optimization problems are closely related to the solution of two coupled nonlinear equations:

$$c(x_*)=0 \text{ in } \mathbb{R}^m\,, \tag{1.18}$$

$$g(x_*)=0 \text{ in } \mathbb{R}^{n-m}\,. \tag{1.19}$$

The equation (1.19) expresses the vanishment of the reduced gradient defined in (1.7) and is obtained by projecting the second equation of (1.3) on the tangent space $N(A_*)$. A "decoupling" method for solving (1.18)–(1.19) is introduced in *Section 2*. At x_k, the first part of the step of the method consists in doing a NEWTON-like displacement for solving (1.18). This leads to a point y_k. Then, x_{k+1} is obtained by doing a NEWTON-like displacement for solving (1.19) from the point y_k:

$$y_k=x_k-A_*^-c(x_k)\,, \tag{1.20}$$

$$x_{k+1}=y_k-B_*^-g(y_k)\,. \tag{1.21}$$

In (1.20), A_*^- is a right inverse of $\nabla c(x_*)$ and in (1.21), B_*^- is a right inverse of $\nabla g(x_*)$. We shall show that only conditions on B_*^- have to be imposed in order to ensure the local quadratic convergence (in one step) of the process. In *Section 4*, we apply this algorithm to constrained optimization, when g in (1.19) has the special structure (1.7), and it takes the form of a reduced method. Its extension to reduced quasi-NEWTON method is then easily done. This presentation gives, in our opinion, some insight into method (1.15)–(1.17) and shows what the degrees of freedom are in the choice of the operators A_k^- in (1.16) and Z_k^- in (1.17). In fact, A_k^- may be any right inverse of A_k (with a smoothness hypothesis of $A_k^-=A(x_k)^-$ according to x_k) and not necessarily the PENROSE pseudo-inverse and the columns of Z_k^- may form any basis of $N(A_k)$ and not necessarily an orthogonal basis. This remark may be crucial in some applications like the one mentioned above where the "partitioned framework" (see *Section 3*) occurs naturally. In *Section 5*, we give a necessary and sufficient condition of superlinear convergence of the method that is weaker than the sufficient condition given by BYRD [7].

The globalization of the local method could then be done as in the paper by COLEMAN and CONN [12]. In *Section 6*, however, we examine another globalizing technique essentially based on the ideas of HAN [29] for the SQP method (see also [14]). We introduce the following exact penalty function:

$$\Theta_p(x)=f(x)+p\|c(x)\|_1\,, \tag{1.22}$$

where p is a large enough penalty parameter and $\|\cdot\|_1$ is the l_1 norm on $\mathbb{R}^m$. We look for x_* by minimizing Θ_p on ω. The idea is then to obtain a descent direction for Θ_p at the current iterate from the displacements calculated by the local algorithm (1.20)–(1.21). Contrary to what happens with the SQP method, our total displacement is not necessarily a descent direction for Θ_p any more. So, we shall introduce a descent arc, being inspired in this way by the work of Gabay [20] for algorithm (1.12)–(1.14) and Mayne and Polak [32] for the SQP method, although in those algorithms, the arc was introduced for other reasons. A search on the arc is done in order to decrease the penalty function Θ_p with the help of an Armijo-like criterion. This gives a theorem assuring the global convergence of the method. Furthermore, under natural conditions, the "Maratos effect" is avoided: the step-size is equal to one after a finite number of iterations. Therefore there is a smooth transition from the global to the local method that does not prevent the superlinear convergence from occurring.

If (v_k) is a sequence in a normed space $(E, \|\cdot\|_E)$ and (α_k) is a sequence of positive numbers, we shall say that (v_k) is a big O of (α_k) (we shall note $v_k = O(\alpha_k)$) if the sequence $(\|v_k\|_E/\alpha_k)$ is bounded and we shall say that (v_k) is a small o of (α_k) (we shall note $v_k = o(\alpha_k)$) if the sequence $(\|v_k\|_E/\alpha_k)$ converges to zero. We shall say that two positive real sequences (α_k) and (β_k) are *equivalent* (we shall note $\alpha_k \sim \beta_k$) if $\alpha_k = O(\beta_k)$ and $\beta_k = O(\alpha_k)$. We shall note v^i, the i-th component of a vector v in E. If A is a linear operator from $(E, \|\cdot\|_E)$ to $(F, \|\cdot\|_F)$, we shall note $\|A\| := \sup \{\|Av\|_F \mid \|v\|_E \leq 1\}$. If A and B are two square matrices of the same order, we shall note $A \leqq B$ when $(B - A)$ is positive semi definite.

This paper constitutes a revised version of a part of the INRIA report number RR-482 in which some techniques for updating the reduced matrix have also been investigated (see also [25]). A variant of the method is given in [23].

2. A Decoupling Method for Solving Two Nonlinear Coupled Equations

Let us consider the following coupled system of nonlinear equations:

$$\begin{cases} F(x) = 0 , \\ G(x) = 0 , \end{cases} \tag{2.1}$$

where F and G are supposed smooth and map $\mathbb{R}^n$ to $\mathbb{R}^m$ and $\mathbb{R}^{n-m}$ $(m < n)$ respectively. Let x_* be a solution of (2.1) and let us denote by A_* the $m \times n$ Jacobian matrix of F at x_* and B_* the $(n-m) \times n$ Jacobian matrix of G at x_*. We shall say that x_* is a *regular solution* of (2.1) if the Jacobian matrix of the system (2.1),

$$J_* := \begin{bmatrix} A_* \\ B_* \end{bmatrix}, \tag{2.2}$$

is nonsingular. This will be the case if and only if $N(A_*) \cap N(B_*) = \{0\}$. We would like to define a Newton-like method for solving (2.1) without having to inverse

the Jacobian $J(x_k)$ of (2.1) at x_k. For that, let us suppose that x_* is a regular solution. Then A_* and B_* are surjective and we can introduce a right inverse A_*^- of A_* and a right inverse B_*^- of B_*:

$$A_*A_*^- = I_m, \quad B_*B_*^- = I_{n-m} . \tag{2.3}$$

Two algorithms using A_*^- and B_*^- can be considered and we introduce them with the help of fixed point maps ξ_1 and ξ_2. The first one is

$$x_{k+1} = \xi_1(x_k) , \tag{2.4}$$

$$\xi_1(x) := x - A_*^- F(x) - B_*^- G(x) \tag{2.5}$$

and the second one is

$$x_{k+1} = \xi_2(x_k) := (\Psi \circ \Phi)(x_k) , \tag{2.6}$$

$$\Phi(x) := x - A_*^- F(x) , \tag{2.7}$$

$$\Psi(y) := y - B_*^- G(y) . \tag{2.8}$$

These algorithms are somewhat "ideal". Indeed, the matrices A_* and B_* are not known and so neither are A_*^- and B_*^-. But they are simpler to study than their implementable versions that will be introduced later.

The relations (2.3) do not determine the right inverses A_*^- and B_*^- completely. Therefore, we may try to choose them so that the sequences generated by algorithms (2.4)–(2.5) and (2.6)–(2.8) will have a good local behaviour. The next two propositions show that this is possible: we can get conditions on A_*^- and B_*^- in order to have $\xi_1'(x_*) = 0$ and $\xi_2'(x_*) = 0$, which will ensure a quadratic rate of convergence for both algorithms. We shall say that an n row matrix is a *basis* of a given subspace of $\mathbb{R}^n$ if it is injective and if its columns form a basis of that subspace.

Proposition 2.1: *Suppose that F and G are differentiable at x_*, a regular solution of (2.1). Then, the following statements are equivalent:*

(i) $\xi_1'(x_*) = 0$,

(ii) $R(A_*^-) = N(B_*)$ *and* $R(B_*^-) = N(A_*)$,

(iii) *for any right inverse A_*^m of A_* and any basis Z_*^- of $N(A_*)$, we have*

$$A_*^- = (I - Z_*^-(B_*Z_*^-)^{-1} B_*) A_*^m , \tag{2.9}$$

$$B_*^- = Z_*^-(B_*Z_*^-)^{-1} . \tag{2.10}$$

Proof: First, we prove (i)⇔(ii). Statement (i) is equivalent to

$$I = A_*^-A_* + B_*^-B_* . \tag{2.11}$$

The right hand side of (2.11) is equal to $[A_*^- B_*^-]\, J_*$. Then, (2.11) means that $[A_*^- B_*^-]$ is the inverse of J_* and therefore is equivalent to $I = J_*[A_*^- B_*^-]$, i.e. $A_*B_*^- = 0$ and $B_*A_*^- = 0$, which is equivalent to statement (ii) because the matrices A_*, B_*, A_*^- and B_*^- have full rank. Next, we prove (ii)⇒(iii). Let Z_*^- be any basis of $N(A_*)$: $R(Z_*^-) = N(A_*)$ and Z_*^- is injective. Because J_* is nonsingular, $B_*Z_*^-$ is nonsingular. Indeed, if u in $\mathbb{R}^{n-m}$ satisfies $B_*Z_*^-u = 0$, we have $J_*Z_*^-u = 0$ (because $A_*Z_*^- = 0$), then $Z_*^-u = 0$ because J_* is nonsingular and $u = 0$ because Z_*^-

is injective. Then, by multiplying (2.11) to the right by Z_*^-, we get $Z_*^- = B_*^-(B_* Z_*^-)$ and therefore (2.10). Formula (2.9) is obtained by multiplying (2.11) to the right by any right inverse $A_*^=$ of A_* and by using (2.10). It remains to prove (iii)⇒(ii). If we take $A_*^= = A_*^-$ in (2.9), we obtain $B_* A_*^- = 0$ and so $R(A_*^-) = N(B_*)$ because these spaces have the same dimension m. Because $A_* Z_*^- = 0$, (2.10) gives $A_* B_*^- = 0$ and so $R(Z_*^-) = N(A_*)$ because these spaces have the same dimension $n-m$. ■

Proposition 2.2: *Suppose that F and G are differentiable at x_*, a regular solution of (2.1). Then, the following statements are equivalent:*

(i) $\xi_2'(x_*) = 0$,

(ii) $R(B_*^-) = N(A_*)$,

(iii) *for any basis Z_*^- of $N(A_*)$, we have $B_*^- = Z_*^-(B_* Z_*^-)^{-1}$.*

Proof: The equivalence (i)⇔(ii) comes from $\xi_2'(x_*) = (I - B_*^- B_*)\ (I - A_*^- A_*)$ and that the spaces $N(A_*) = R\,(I - A_*^- A_*)$ and $R(B_*^-) = N\,(I - B_*^- B_*)$ have the same dimension $n-m$. To prove that (i) implies (iii), let Z_*^- be any basis of $N(A_*)$. By multiplying to the right both sides of

$$(I - B_*^- B_*)\ (I - A_*^- A_*) = 0$$

by Z_*^-, we get $Z_*^- = B_*^-(B_* Z_*^-)$ and therefore (iii) because $B_* Z_*^-$ is nonsingular (see the proof of Proposition 2.1). From (iii), we get $A_* B_*^- = 0$ by multiplying to the left $B_*^- = Z_*^-(B_* Z_*^-)^{-1}$ by A_* and $A_* B_*^- = 0$ is equivalent to (ii) because $R(B_*^-)$ and $N(A_*)$ have the same dimension $n-m$. ■

In statement (iii) of Proposition 2.1, we could equivalently have given to B_* the role of A_*. We also see that the right inverses A_*^- and B_*^- are completely determined by condition (i) of Proposition 2.1 and do not depend on the choice of $A_*^=$ and Z_*^- in (iii). Similarly, the right inverse B_*^- is completely determined by condition (i) of Proposition 2.2 and does not depend on the choice of Z_*^- in (iii).

From Proposition 2.1, we see that $\xi_1'(x_*) = 0$ if and only if $[A_*^- B_*^-]$ is the inverse of J_*. This means that algorithm (2.4)–(2.5) is in fact the "*ideal*" (with J_*^{-1} rather than $J(x_k)^{-1}$) Newton *method* for solving (2.1) (see the displacement in (2.5)), the method we wanted to avoid. On the other hand, Proposition 2.2 shows that algorithm (2.6)–(2.8) needs fewer conditions to have a good local behaviour than algorithm (2.4)–(2.5). The fact that no conditions are required on the right inverse A_*^- means that any solver of the first equation in (2.1) can be used in (2.7), independently of the second equation of (2.1), whereas this is not true for the solver B_*^- of the second equation of (2.1) that has to be adapted to the first equation.

The results of Propositions 2.1 and 2.2 have a geometrical interpretation. In the ideal Newton method, (x_k) will converge quadratically if the displacements $(-A_*^- F(x_k))$ and $(-B_*^- G(x_k))$ belong to the tangent space at x_* to the manifolds defined respectively by the pre-image of 0 by F and G. In method (2.6)–(2.8) only the second step $(-B_*^- G(y_k))$ has to belong to the tangent space $N(A_*)$; the first step is arbitrary (apart from the fact that A_*^- has to be a right inverse of A_*).

3. A Change of Coordinates

Before applying the results of the previous section to constrained optimization, let us give some examples of right inverses A_x^- of A_x and basis Z_x^- of $N(A_x)$ that are frequently used in practice. The formalism adopted here has been introduced by GABAY [19].

Once the injective matrices A_x^- and Z_x^- have been chosen, the columns of $[A_x^- Z_x^-]$ form a new basis of $\mathbb{R}^n$. Indeed, $R(A_x^-)$ is a complementary space of $N(A_x) = R(Z_x^-)$. To make a change of coordinates in that new basis, it is convenient to introduce the additional $(n-m)\times n$ matrix Z_x given by the following proposition.

Proposition 3.1: *Let A_x be an $m\times n$ $(m<n)$ surjective matrix, A_x^- be any right inverse of A_x and Z_x^- be any basis of $N(A_x)$. Then, there exists a unique $(n-m)\times n$ matrix Z_x such that*

$$Z_x A_x^- = 0 \,, \tag{3.1}$$

$$Z_x Z_x^- = I_{n-m} \,. \tag{3.2}$$

Furthermore, we have

$$I_n = A_x^- A_x + Z_x^- Z_x \,. \tag{3.3}$$

Proof: Existence and unicity of the matrix Z_x come from the nonsingularity of $[A_x^- Z_x^-]$ and (3.3) comes from the fact that $[A_x^{\mathsf{T}} Z_x^{\mathsf{T}}]^{\mathsf{T}}$ is the inverse of $[A_x^- Z_x^-]$. ■

The relation (3.1) shows that $N(Z_x) = R(A_x^-)$ (the matrices A_x^- and Z_x have full rank) and (3.2) shows that Z_x^- is a right inverse of Z_x. The equality (3.3) can be used to introduce a change of coordinates. Indeed, by applying it to a vector ξ of $\mathbb{R}^n$, we see that $A_x\xi$ are the coordinates of ξ in $R(A_x^-) = N(Z_x)$ and $Z_x\xi$ are the coordinates of ξ in $R(Z_x^-) = N(A_x)$.

A first choice of matrices A_x^- and Z_x^-, which is frequently made in constrained optimization, defines what could be called the *orthogonal framework*: A_x^- is the MOORE-PENROSE pseudo-inverse of A_x (see [2]) and Z_x^- is an orthogonal basis of $N(A_x)$ for the Euclidean scalar product. We have:

$$A_x^- = A_x^{\mathsf{T}}(A_x A_x^{\mathsf{T}})^{-1} \,, \tag{3.4}$$

$$Z_x^{-\mathsf{T}} Z_x^- = I_{n-m} \,.$$

Then $Z_x = Z_x^{-\mathsf{T}}$ is the unique matrix satisfying (3.1) and (3.2). We see that $R(A_x^-)$ is orthogonal to $N(A_x)$. This framework has been adopted by COLEMAN and CONN [11–13] and by BYRD [7].

Another choice of matrices A_x^- and Z_x^- can be made when a separation of variables occurs naturally, as in optimal control problems or in parameter identification problems. This is also the natural framework to introduce the GRG method (ABADIE and CARPENTIER [1]). It could be called the *partitioned framework*: A_x is supposed to be partitioned in two submatrices

$$A_x = [C_x D_x] \,, \tag{3.5}$$

where the $m \times m$ matrix C_x is nonsingular and D_x has dimension $m \times (n-m)$. The right inverse A_x^- is then taken as

$$A_x^- = \begin{bmatrix} C_x^{-1} \\ 0 \end{bmatrix} \tag{3.6}$$

and the basis of $N(A_x)$ is

$$Z_x^- = \begin{bmatrix} -C_x^{-1} D_x \\ I_{n-m} \end{bmatrix} . \tag{3.7}$$

Then $Z_x = [0\, I_{n-m}]$ is the unique matrix satisfying (3.1) and (3.2). This framework has been adopted by Hoyer [30].

In the following, we shall suppose that the choice of (A_x^-, Z_x^-) is a smooth function of x:

Assumption B:
The function $x \to (A_x^-, Z_x^-, Z_x)$ is bounded on ω and is in $C_b^{\nu-1}(\omega)$ with $\nu \geqq 3$.

This assumption is satisfied for A_x^- given by (3.4). With regard to Z_x^-, the question is more delicate, although the assumption can be satisfied locally by projection on $N(A_x)$ of a basis Z_*^- of $N(A_*)$ (see [26], [9]).

4. A Reduced Quasi-Newton Method for Constrained Optimization

In this section, we apply the results of Section 2 to constrained optimization. The first step consists in reducing the size of the optimality system (1.3). This can be done because the second optimality condition can be expressed by $n-m$ equations rather than n, in fact, by the vanishment of the $n-m$ coordinates of the orthogonal projection of $\nabla f(x_*)$ on $N(A_*)$. If Z_*^- is any basis of $N(A_*)$, the orthogonal projector on $N(A_*)$ is $Z_*^-(Z_*^{-\mathsf{T}} Z_*^-)^{-1} Z_*^{-\mathsf{T}}$. Then, the second equation of (1.3) is projected on $N(A_*)$ by multiplying it by $Z_*^{-\mathsf{T}}$. Using the definition (1.7) of the reduced gradient, the system (1.3) can be rewritten as follows:

$$\begin{cases} c(x_*) = 0 , \\ g(x_*) = 0 . \end{cases} \tag{4.1}$$

In order to apply the previous results, we need to calculate the first derivative of g at x_*. This can be done as in [37]:

$$\nabla g(x_*) = \nabla \left(Z_x^{-\mathsf{T}} (\nabla f(x) + A_x^{\mathsf{T}} \lambda_*) \right)(x_*) = Z_*^{-\mathsf{T}} L_* . \tag{4.2}$$

The Jacobian matrix of (4.1),

$$\begin{bmatrix} A_* \\ Z_*^{-\mathsf{T}} L_* \end{bmatrix} ,$$

is nonsingular because of the surjectivity of A_* and the second order sufficient condition which is equivalent to the positive definiteness of

$$G_* := Z_*^{-\mathsf{T}} L_* Z_*^- . \tag{4.3}$$

We shall note $H_* := G_*^{-1}$.

Now, let us apply algorithm (2.4)–(2.5) to the system (4.1). Using statement (iii) of Proposition 2.1 and (4.2), we get the following quadratically convergent algorithm:

$$x_{k+1} = x_k - (I - Z_*^- H_* Z_*^{-\mathrm{T}} L_*)\, A_*^- c(x_k) - Z_*^- H_* g(x_k)\,, \tag{4.4}$$

where A_*^- is any right inverse of A_* (playing the role of $A_*^=$ in (2.9)) and Z_*^- is any basis of $N(A_*)$. This is exactly the "ideal" SQP method: see (1.6). See also GOODMAN [27] for a related result. We shall not go further with this method.

If we apply algorithm (2.6)–(2.8) to the system (4.1), we get, using statement (iii) of Proposition 2.2 and (4.2):

$$\bar{y}_k = x_k - A_*^- c(x_k)\,, \tag{4.5}$$

$$\bar{x}_{k+1} = \bar{y}_k - Z_*^- G_*^{-1} g(\bar{y}_k)\,, \tag{4.6}$$

where A_*^- is any right inverse of A_* and Z_*^- is any basis of $N(A_*)$. The following lemma is a consequence of Proposition 2.2. This result has also been obtained by HOYER [30, Theorem 4.3] in the partitioned framework.

Lemma 4.1: *Suppose that Assumptions* A *and* B *are satisfied and let x_* be a solution of* (1.1). *There exists a positive constant C that depends only on f and c such that if x_k is sufficiently close to x_*, $\bar{y}_k$ is given by* (4.5) *and $\bar{x}_{k+1}$ is given by* (4.6), *we have*

$$\|\bar{x}_{k+1} - x_*\| \leqq C\, \|x_k - x_*\|^2\,. \tag{4.7}$$

From the quadratically convergent algorithm (4.5)–(4.6), a quasi-NEWTON method is easily introduced. In (4.6), G_* is replaced by an approximation G_k and Z_*^- is replaced by $Z(y_k)^-$, which intervenes in the calculation of the reduced gradient $g(y_k)$. If A_*^- in (4.5) is replaced by $A(x_k)^-$, the constraints will have to be linearized twice per iteration: at x_k to calculate $A(x_k)$ and at y_k to calculate the basis $Z(y_k)^-$. Since the constraints have to be linearized at y_k to calculate the reduced gradient in (4.6), we avoid one linearization of the constraints by replacing A_*^- in (4.5) by $A(y_{k-1})^-$. So we obtain the following local algorithm:

$$y_k = x_k - A(y_{k-1})^-\, c(x_k)\,, \tag{4.8}$$

$$x_{k+1} = y_k - Z(y_k)^-\, G_k^{-1} g(y_k)\,. \tag{4.9}$$

We shall note $H_k := G_k^{-1}$ and denote by r_k the restoration step and by t_k the tangent step:

$$r_k := -A(y_{k-1})^-\, c(x_k)\,, \tag{4.10}$$

$$t_k := -Z(y_k)^-\, G_k^{-1} g(y_k)\,. \tag{4.11}$$

We shall also use the total displacements

$$d_k := r_k + t_k\,, \tag{4.12}$$

$$e_k := t_k + r_{k+1}\,. \tag{4.13}$$

In practice, the algorithm cannot start with (4.8) from a point x_1 without knowing a point y_0. So, we shall suppose in the following that the algorithm starts with (4.9) from a point y_0 in ω.

This is really the same type of algorithm as Coleman and Conn's method (1.15)–(1.17) if we exchange in (4.8)–(4.9) y_k with x_k and x_{k+1} with $x_k + t_k$. However, our point of view shows that there is no reason to take a restoration step orthogonal to $N(A(y_{k-1}))$ in (4.10) or to calculate an orthogonal basis $Z(y_k)^-$ of $N(A(y_k))$ in (4.11). In particular, this validates the use of the partitioned framework (3.5)–(3.7) that occurs often in practice, see also [30]. We shall see that contrary to the sequence (y_k) in (4.8)–(4.9), which does not usually converge superlinearly in one step (see the examples given by Byrd [8] and Yuan [39]), the sequence (x_k) converges superlinearly as expected from the behaviour of the ideal algorithm (4.5)–(4.6).

In fact, it is not essential to reduce the size of the optimality system before applying algorithm (2.6)–(2.8). The same method (4.8)–(4.9) can be obtained when the method is applied to the optimality conditions (1.3), see [25]. In this case, $B_* = [L_* A_*^\top]$ and

$$B_*^- = \begin{bmatrix} Z_* G_*^{-1} Z_*^\top \\ A_*^{-\top} (I - L_* Z_*^- G_*^{-1} Z_*^{-\top}) \end{bmatrix},$$

where A_*^- is any right inverse of A_* and Z_*^- is any basis of $N(A_*)$. Furthermore, this derivation of the algorithm gives an iteration scheme for the Lagrange multipliers (λ_k):

$$\lambda_{k+1} = -A(y_k)^{-\top} \nabla f(y_k) + A(y_k)^{-\top} L_k Z(y_k)^- G_k^{-1} g(y_k), \tag{4.14}$$

where L_k is an approximation of L_*. Therefore, if L_k in (4.14) is correctly updated, we shall also have $\lambda_{k+1} - \lambda_* = o(\|x_k - x_*\|)$; indeed, λ_k does not intervene in the iteration. Formula (4.14) simplifies the one obtained when quasi-Newton method is applied to (1.3), which can be written (see [20]):

$$\lambda_{k+1} = -A(x_k)^{-\top} (I - L_k Z(x_k)^- G_k^{-1} Z(x_k)^{-\top}) (\nabla f(x_k) - L_k A(x_k)^- c(x_k)).$$

Algorithm (4.8)–(4.9) is a *reduced quasi-*Newton *method* because the only matrix to update is the approximation G_k of G_* and it is of order $n-m$. Unfortunately, this is no longer the case when the sequence (λ_k) is generated by (4.14) since L_k intervenes in the formula and L_k is of order n.

In the next two propositions, we study the local linear convergence of algorithm (4.8)–(4.9).

Lemma 4.2: *Suppose that Assumptions* A *and* B *are satisfied. Then, there exist positive constants* δ, C_1, C_2 *and* C_3 *that depend only on* f, c *and* ω *such that on the one hand,* $\|G_k - G_*\| \leq \delta$ *implies that* G_k *is nonsingular with* $\|G_k^{-1}\| \leq C_3$ *and on the other hand,*

$$\|y_{k-1} - x_*\| \leq (1 + C_1)\,\delta, \tag{4.15}$$

$$\|x_k - x_*\| \leq \delta, \tag{4.16}$$

$$\|G_k - G_*\| \leq \delta, \tag{4.17}$$

with $0 < \delta \leqq \bar{\delta}$ *imply that* y_k *and* x_{k+1} *are well defined by* (4.8)–(4.9) *and satisfy*

$$\|y_k - x_*\| \leqq (1 + C_1)\,\delta\,, \tag{4.18}$$

$$\|x_{k+1} - x_*\| \leqq C_2 \delta\,\|x_k - x_*\|\,. \tag{4.19}$$

Proof: We shall denote by C_i $(i = 1, 2, \ldots)$ any positive constant that depends only on f, c and ω. According to Assumptions A and B, there exists a positive constant C_1 such that

$$\max\left(\|A(y)\|, \|A(y)^-\|, \|Z(y)^-\|, \|g'(y)\|\right) \leqq C_1\,, \tag{4.20}$$

for all y in ω and

$$\max\left(\|\bar{y}_k - x_*\|, \|y_k - x_*\|\right) \leqq (1 + C_1)\,\|x_k - x_*\|\,, \tag{4.21}$$

if y_{k-1} and x_k are in ω and $\bar{y}_k$ and y_k are calculated from x_k by (4.5) and (4.8) respectively. Both inequalities in (4.21) are obtained from (4.5) and (4.8) by using TAYLOR's expansion of $c(x_k)$. Let ε, η and $\bar{\delta}$ be three fixed positive constants such that

$$B(x_*, \varepsilon) \subset \omega\,, \tag{4.22}$$

$$\|G_*^{-1}\| \leqq \eta\,, \tag{4.23}$$

$$\bar{\delta} < \frac{1}{\eta}\,, \tag{4.24}$$

$$(1 + C_1)\,\bar{\delta} < \varepsilon\,, \tag{4.25}$$

where $B(x_*, \varepsilon)$ denotes the ball of radius ε centered at x_*. These constants ε, η and $\bar{\delta}$ depend only on f, c and ω. If G_k satisfies $\|G_k - G_*\| \leqq \delta$ then, by (4.24) and (4.23), G_k is nonsingular and satisfies (see for example [36, Chap. II, Theorem 62]:

$$\|G_k^{-1}\| \leqq \frac{1}{\frac{1}{\eta} - \delta} =: C_3\,. \tag{4.26}$$

This proves the first part of the lemma.

For the second part, let us suppose that inequalities (4.15)–(4.17) are satisfied with δ in $]0, \bar{\delta}]$. According to (4.15), (4.25) and (4.22), y_{k-1} belongs to ω and according to (4.16), (4.25) and (4.22), x_k belongs to ω. Therefore y_k is well defined by (4.8) and we have (4.21). This inequality and (4.16) show (4.18). Now, according to (4.21), (4.16), (4.25) and (4.22), y_k and $\bar{y}_k$ belong to ω. So, $\bar{x}_{k+1}$ and x_{k+1} are well defined by (4.6) and (4.9) respectively. From (4.26), (4.23), (4.17) and $G_k^{-1} - G_*^{-1} = -G_k^{-1}\,(G_* - G_k)\,G_*^{-1}$, we deduce:

$$\|G_k^{-1} - G_*^{-1}\| \leqq \|G_k^{-1}\|\,\|G_*^{-1}\|\,\|G_k - G_*\| \leqq C_3 \eta \delta =: C_4 \delta\,.$$

Let C_5 be the constant given by Lemma 4.1. Then, TAYLOR expansions give easily the following inequalities:

$$\|x_{k+1} - x_*\| \leqq \|x_{k+1} - \bar{x}_{k+1}\| + \|\bar{x}_{k+1} - x_*\|\,,$$

$$\|x_{k+1} - \bar{x}_{k+1}\| \leqq C_6\,\|y_k - \bar{y}_k\| + C_7\,\|y_k - x_*\|^2 + C_4 C_8 \delta\,\|y_k - x_*\|\,,$$

$$\|y_k - \bar{y}_k\| \leqq C_9\,\|y_{k-1} - x_*\|\,\|x_k - x_*\|\,.$$

By combining these inequalities with (4.7), (4.15), (4.16) and (4.21), we get (4.19) with $C_2 = C_5 + C_6 C_9 (1+C_1) + C_7 (1+C_1)^2 + C_4 C_8 (1+C_1)$. ■

Theorem 4.3: *Suppose that Assumptions* A *and* B *are satisfied. Then, there exists a positive constant* C *that depends only on* f, c *and* ω *such that if* $\varkappa$ *is a real number in* $]0, 1[$ *and if*

$$\|y_0 - x_*\| \leqq C\varkappa \,, \tag{4.27}$$

$$\|G_k - G_*\| \leqq C\varkappa \quad \text{for all subscripts } k \,, \tag{4.28}$$

then algorithm (4.8)–(4.9) *generates from* y_0 *a sequence* (x_k) *in* ω *that converges linearly to* x_* *and*

$$\|x_{k+1} - x_*\| \leqq \varkappa \|x_k - x_*\| \,, \tag{4.29}$$

for all subscripts k.

Proof: Let δ, C_1, C_2 and C_3 be the positive constants given by Lemma 4.2. Then, if G_{k-1} satisfies $\|G_{k-1} - G_*\| \leqq \delta$, we have $\|G_{k-1}^{-1}\| \leqq C_3$. By expanding $g(y_{k-1})$ about x_* (with (4.2)), (4.9) shows that for $k \geqq 1$, we have

$$\|x_k - x_*\| \leqq (1 + C_4) \|y_{k-1} - x_*\| \,, \tag{4.30}$$

where C_4 is a positive constant that depends only on f, c and ω. Then, the theorem can be proved with $C := \min(\delta, 1/C_2)/(1+C_4)$. Indeed, if (4.27) and (4.28) are satisfied, we see, with the help of (4.30), that (4.15)–(4.17) are satisfied for $k = 1$ and $\delta := (1+C_4) C\varkappa \leqq \delta$:

$$\|y_0 - x_*\| \leqq C\varkappa \leqq \delta \leqq (1+C_1)\,\delta \,,$$
$$\|x_1 - x_*\| \leqq (1+C_4) \|y_0 - x_*\| \leqq \delta \,,$$
$$\|G_1 - G_*\| \leqq C\varkappa \leqq \delta \,.$$

We can then apply Lemma 4.2 and because $C_2\delta \leqq \varkappa$, (4.19) shows that (4.29) is satisfied for $k = 1$. The fact that $\varkappa$ is less than 1, (4.28) and (4.18) for $k = 1$ show that (4.15)–(4.17) are still satisfied for $k = 2$. So we can conclude by induction. ■

The next proposition gives some useful estimates and equivalences.

Proposition 4.4: *Suppose that Assumptions* A *and* B *are satisfied. Let* (G_k) *be a sequence of nonsingular matrices of order* $n - m$. *Let* (x_k) *in* ω *and* (y_k) *in* ω *be the sequences generated by algorithm* (4.8)–(4.9) *starting from a point* y_0 *in* ω. *If* (x_k) *and* (y_k) *converge to a solution* x_* *of* (1.1), *we have*

$$r_k = -A_*^- A_* (x_k - x_*) + o(\|x_k - x_*\|) \,, \tag{4.31}$$

$$y_k - x_* = Z_*^- Z_* (x_k - x_*) + o(\|x_k - x_*\|) \,. \tag{4.32}$$

If moreover (G_k) *and* (G_k^{-1}) *are bounded, we have*

$$\|d_k\| \sim \|x_k - x_*\| \,, \tag{4.33}$$

$$\|e_k\| \sim \|y_k - x_*\| \,. \tag{4.34}$$

Proof: From the definition (4.10) of r_k, the expansion of $c(x_k)$ about x_* and the convergence of (y_k), we get (4.31). Then, using identities (3.3) and (4.31) on $y_k =$

$=x_k+r_k$, we obtain (4.32). Next, using the boundedness of (G_k^{-1}) and (4.2), we see that $t_k=-Z_*^- G_k^{-1} Z_*^{-\mathrm{T}} L_* (y_k-x_*)+o(\|y_k-x_*\|)$. Finally, with (4.12), (4.31) and (4.32), this gives

$$d_k=-(A_*^- A_*+Z_*^- G_k^{-1} G_* Z_*)(x_k-x_*)+o(\|x_k-x_*\|) .$$

This estimate shows that $d_k=O(\|x_k-x_*\|)$. To prove the $x_k-x_*=O(\|d_k\|)$, we only have to show that the operator in square brackets is nonsingular with bounded inverse. If this were not the case, there would exist a subsequence $\mathbb{K}$ of subscripts and a sequence $(\xi_k \mid k\in\mathbb{K})$ in $\mathbb{R}^n$ such that:

$$\|\xi_k\|=1 \text{ for } k \text{ in } \mathbb{K} , \tag{4.35}$$

$$(A_*^- A_*+Z_*^- G_k^{-1} G_* Z_*)\,\xi_k \to 0 \quad \text{for } k \text{ in } \mathbb{K} . \tag{4.36}$$

By multiplying (4.36) by A_* (resp. Z_*), we would obtain $A_*\xi_k\to 0$ (resp. $G_k^{-1}G_*Z_*\xi_k\to 0$, from which we would deduce $Z_*\xi_k\to 0$ because of the boundedness of (G_k) and the nonsingularity of G_*). Finally, with (3.3), we would have $\xi_k\to 0$, which would contradict (4.35). So, (4.33) is proved. The proof of (4.34) is similar and is based on the estimate

$$e_k=-(A_*^- A_*+Z_*^- G_k^{-1} Z_*^{-\mathrm{T}} L_*)(y_k-x_*)+o(\|y_k-x_*\|) . \quad \blacksquare$$

5. Conditions for Superlinear Convergence

Theorem 4.3 has an immediate corollary, which states that if in addition to (4.28), the sequence (G_k) converges to G_* then (x_k) converges to x_* superlinearly (see for example the argument in the proof of corollary 3.5 in HAN [28]). However, this assumption on (G_k') is usually not satisfied when these matrices are generated by quasi-NEWTON formulas. Assuming that (x_k) converges to x_*, the next theorem gives necessary and sufficient conditions on (G_k) to have the superlinear convergence of (x_k). It is the analogue of Theorem 2.2 of DENNIS and MORÉ [15], valid for quasi-NEWTON methods in optimization without constraints.

Theorem 5.1: *Suppose that Assumptions* A *and* B *are satisfied and that* (y_k) *and* (x_k) *are generated in* ω *from a point* y_0 *by algorithm* (4.8)–(4.9) *with a sequence* (G_k) *of nonsingular matrices. Suppose that* (x_k) *and* (y_k) *converge to* x_*. *Then, the following statements are equivalent:*

(i) (x_k) *converges superlinearly,*

(ii) $g(y_{k+1})=o(\|x_k-x_*\|)$,

(iii) $(G_k-G_*)\,Z(y_k)\,t_k=o(\|x_k-x_*\|)$.

Proof: The estimate (4.32) shows that

$$y_k-x_*=O(\|x_k-x_*\|) . \tag{5.1}$$

Using $A(y_k)-A_*=O(\|y_k-x_*\|)$, $t_k\to 0$ (because $t_k=x_{k+1}-y_k$), (4.32) and $A_*Z_*^-=0$, we get:

$$\begin{aligned} &A_*(x_{k+1}-x_*) \\ &=A_*(y_k-x_*)+A(y_k)\,t_k+o(\|y_k-x_*\|)=o(\|x_k-x_*\|) . \end{aligned} \tag{5.2}$$

According to (3.3), it remains to estimate $Z_*(x_{k+1}-x_*)$. This will depend on the quality of the tangent step t_k.

Let us first prove the equivalence (i)$\Leftrightarrow$(ii). With (4.2) and (4.32), we have

$$g(y_{k+1})=Z_*^{-\top}L_*(y_{k+1}-x_*)+o(\|y_{k+1}-x_*\|)=G_*Z_*(x_{k+1}-x_*)+o(\|x_{k+1}-x_*\|).$$

Then, (ii) is clear from (i). If (ii) is satisfied, this estimate and the nonsingularity of G_* give

$$Z_*(x_{k+1}-x_*)=o(\|x_{k+1}-x_*\|)+o(\|x_k-x_*\|).$$

This estimate, (5.2) and identity (3.3) show (i).

Now, let us show that in any of the situations (i), (ii) or (iii), we have

$$t_k=O(\|x_k-x_*\|). \tag{5.3}$$

This estimate is clear when (G_k^{-1}) is bounded, but we do not suppose this here. Writing $t_k=(x_{k+1}-x_*)-(y_k-x_*)$ and using (5.1), we see that (5.3) is clearly satisfied when (i) is true and therefore when (ii) is true. When (iii) is satisfied, we have

$$G_*Z(y_k)\,t_k=G_kZ(y_k)\,t_k+o(\|x_k-x_*\|)=-g(y_k)+o(\|x_k-x_*\|).$$

Then expanding $g(y_k)$ about x_* and using (5.1) and the nonsingularity of G_*, we get

$$Z(y_k)\,t_k=O(\|x_k-x_*\|).$$

But $t_k=Z(y_k)^-Z(y_k)\,t_k$, therefore (5.3) is still satisfied.

Now, from (5.1) and (5.3), it follows that $x_{k+1}-x_*=y_k-x_*+t_k=O(\|x_k-x_*\|)$, $y_{k+1}-y_k=(y_{k+1}-x_*)-(y_k-x_*)=O(\|x_{k+1}-x_*\|)+O(\|x_k-x_*\|)=O(\|x_k-x_*\|)$ and, with (4.31) and (5.2),

$$r_{k+1}=o(\|x_k-x_*\|). \tag{5.4}$$

Let us now prove the equivalence (ii)$\Leftrightarrow$(iii). Expanding $g(y_{k+1})$ about y_k and using (4.2), (5.3) and (5.4), we have

$$g(y_{k+1})=g(y_k)+Z_*^{-\top}L_*t_k+o(\|x_k-x_*\|).$$

But $g(y_k)=-G_kZ(y_k)\,t_k$ and $t_k=Z(y_k)^-Z(y_k)\,t_k=Z_*^-Z(y_k)\,t_k+o(\|x_k-x_*\|)$. So, we obtain

$$g(y_{k+1})=-(G_k-G_*)\,Z(y_k)\,t_k+o(\|x_k-x_*\|).$$

The equivalence (ii)$\Leftrightarrow$(iii) follows. ■

In statement (ii) of Theorem 5.1, $g(y_{k+1})$ could be replaced by $g(x_{k+1})$, but the reduced gradient is not evaluated at x_{k+1} in the algorithm. Statement (iii) is equivalent to

$$(H_k-H_*)\,g(y_k)=o(\|x_k-x_*\|),$$

which is based on the gap between the inverse of the reduced Hessians. Statement (iii) can also be replaced by many other equivalent estimates. For example, $(G_k-G_*)\,Z_*(x_k-x_*)=o(\|x_k-x_*\|)$. The advantage of (iii) is that it does not re-

quire the boundedness of the sequences (G_k) or (G_k^{-1}). If this boundedness is assumed, Proposition 4.4 shows that the estimates can be made in relation to $\|d_k\|$ rather than $\|x_k - x_*\|$.

Condition (4.28) and condition (iii) of Theorem 5.1 show the advantage of reduced quasi-NEWTON methods over the SQP method with regard to the approximation of the Hessian of the Lagrangian. Indeed, a necessary and sufficient condition for the SQP method to generate superlinearly convergent sequences is that

$$Z_*^{-\mathrm{T}} (L_k - L_*) (x_k - x_*) = o(\|x_k - x_*\|),$$

where L_k is the updated approximation of L_*. This famous result can be found in [4] and [34]. Therefore, in the SQP method, the $(n-m)\times n$ matrix $Z_*^{-\mathrm{T}} L_*$ has to be correctly approximated and not only the projected Hessian of the Lagrangian $Z_*^{-\mathrm{T}} L_* Z_*^{-}$ as in reduced methods.

As a final remark, let us mention that, if we suppose that the equivalence (4.34) holds, the sufficient condition given by Byrd [7] can be written

$$(G_k - G_*) Z(y_k) t_k = o(\|y_k - x_*\|) \tag{5.5}$$

and is therefore stronger than condition (iii) of Theorem 5.1. However, this estimate (5.5) is satisfied in practice with the update schemes currently proposed: COLEMAN and CONN [13, Theorem 3.6] proved it for their algorithm and it is proved in [22] that

$$(G_k - G_*) Z(y_k) t_k = o(\|t_k\|) \tag{5.6}$$

(which implies (5.5)) for both of the algorithms proposed. But those schemes are not completely satisfactory and condition (iii) of Theorem 5.1 may become useful in other circumstances. We shall see in the next section that, for the globalizing technique proposed, the strongest condition (5.6) allows avoidance of the MARATOS effect, i.e. allows the unit step-size to be accepted asymptotically.

6. Globalization of the Algorithm

In order to globalize the local algorithm (4.8)–(4.9), we introduce a step-size ϱ_k. For that, we consider the following *exact penalty function*:

$$\Theta_p(x) = f(x) + p\|c(x)\|_1, \tag{6.1}$$

where p is a positive *penalty parameter* and $\|\cdot\|_1$ is the l_1-norm on $\mathbb{R}^m$. Other norms than the l_1-norm can be used in (6.1): see [6]. If p is taken to be greater than $\|\lambda_*\|_\infty$ (here it is the dual norm of the one used in (6.1) that is relevant), the *feasible* minimizers of (1.1) and (6.1) are the same (see [17], for example). It is therefore natural to look for x_* by minimizing Θ_p. For this, we need to calculate descent directions of this non-differentiable function. On that point, a crucial observation has been made by HAN [29]: the displacement d_k^{SQP} of the

SQP method is a descent direction of Θ_p at x_k (under some hypotheses). Therefore a better approximation x_{k+1} of the solution x_* will be obtained by taking

$$x_{k+1} = x_k + \varrho_k d_k^{SQP} ,$$

where ϱ_k gives the step-size and is obtained from some rule using Θ_p as a "merit" function.

Let us try to use the same globalizing technique for our algorithm. Is there any descent direction of Θ_p among the displacements r_k, t_k, d_k, and e_k given by (4.10)–(4.13)? The inconvenience of r_k and therefore of d_k and e_k is that this displacement is calculated by using two different points y_{k-1} and x_k that can be far from each other when x_k is far from x_*. So, it is difficult to see when these directions are descent directions for Θ_p. On the other hand, t_k uses only the point y_k in its definition and if G_k is positive definite, it is certainly a descent direction of Θ_p at y_k. Indeed, this displacement is tangent to $c^{-1}(c(y_k))$ at y_k and $f'(y_k) \cdot t_k$ is negative. Therefore at the first order, the first term of the right hand side of (6.1) will decrease while the second term will remain constant. These remarks lead us to define a *descent arc* of Θ_p at y_k, tangent to t_k:

$$y_k(\varrho) = y_k + \varrho t_k + \varrho^a r_{k+1}, \quad a > 1 . \tag{6.2}$$

Let us note that search arcs have already been proposed by Mayne and Polak [32] to cope with the Maratos effect in the SQP method (see further) and by Gabay [20] also to avoid the Maratos effect in algorithm (1.12)–(1.14).

This globalizing technique based on the arc (6.2) gives priority to the minimization step t_k; and this is due to the asymmetry of the local method (4.8)–(4.9). This priority can be harmful in certain circumstances but it can be suppressed by adding a restoration step to the local method (see [23]).

The point y_{k+1} is then obtained from y_k by selecting a particular value ϱ_k of ϱ:

$$y_{k+1} := y_k(\varrho_k) . \tag{6.3}$$

The step-size ϱ_k will be determined here so that the following Armijo-like criterion will be satisfied:

$$\beta \in]0, 1[, \tag{6.4}$$

$$\varrho_k := \beta^{l_k} , \tag{6.5}$$

where l_k is the smallest non-negative integer such that

$$\Theta_p(y_k(\beta^{l_k})) \leqq \Theta_p(y_k) + \beta^{l_k} \alpha f'(y_k) \cdot t_k - \beta^{a l_k} \alpha \, (p - \|\lambda(y_k)\|_\infty) \, \|c(y_k)\|_1 . \tag{6.6}$$

In this inequality, α is a real number chosen in $]0, 1/2[$ for reasons that will be clear at the end of this section. The exponent (al_k) of β in the last term of (6.6) takes into account the curvature of the search path (6.2). The vector $\lambda(y_k)$ is an approximation at y_k of the Lagrange multiplier λ_*. It is defined by

$$\lambda(y) := -A(y)^{-T} \nabla f(y) . \tag{6.7}$$

It is just the first term of (4.14). So, usually, $(\lambda(y_k))$ will not converge superlinearly.

We shall define again

$$x_{k+1} := y_k + t_k \,. \tag{6.8}$$

Now, we have to examine in what conditions inequality (6.6) can be realized with a large enough l_k. This is the subject of the following lemma.

Lemma 6.1: *Suppose that Assumptions* A *and* B *are satisfied and that a point* y_k *is given in* ω *such that* x_{k+1} *and* $x_{k+1}+r_{k+1}$ *will also be in* ω. *Suppose that* α *is in*]0, 1[*and that there exist positive constants* $\underline{p}$, $\bar{p}$ *and* h *such that*

$$\begin{aligned} &\underline{p} + \|\lambda(y_k)\|_\infty \leqq p \leqq \bar{p} \,, \\ &hI \leqq G_k^{-1} \leqq h^{-1} I \,. \end{aligned} \tag{6.9}$$

Then the rule (6.4)–(6.6) *allows determination of a positive step-size* ϱ_k. *If, moreover,* M *is a positive constant such that*

$$\|c(y_k)\|_1 \leqq M \,, \tag{6.10}$$

then, there exists a positive real $\underline{\varrho}$ *that depends only on* f, c, $\underline{p}$, $\bar{p}$, h, α, β *and* M *such that*

$$\varrho_k \geqq \underline{\varrho} > 0 \,.$$

Proof: We shall denote by C_i $(i = 1, 2, \ldots)$ any positive constant. Using $c'(y_k) \cdot t_k = 0$, TAYLOR's theorem gives

$$\|c(x_{k+1}) - c(y_k)\|_1 \leqq C_1 \|t_k\|^2 \,, \tag{6.11}$$

where C_1 depends only on c. Using (6.11) and $0 < \varrho \leqq 1$ and applying again TAYLOR's theorem, we get

$$\begin{aligned} f(y_k + \varrho t_k + \varrho^a r_{k+1}) \leqq{}& f(y_k) + \varrho f'(y_k) \cdot t_k + \varrho^a c(y_k)^{\mathrm{T}} \lambda(y_k) \\ &+ \varrho^a C_2 \|t_k\|^2 + C_3 \|\varrho t_k + \varrho^a r_{k+1}\|^2 \,, \end{aligned} \tag{6.12}$$

$$\begin{aligned} \|c(y_k + \varrho t_k + \varrho^a r_{k+1})\|_1 \leqq{}& (1 - \varrho^a) \|c(y_k)\|_1 \\ &+ \varrho^a C_1 \|t_k\|^2 + C_4 \|\varrho t_k + \varrho^a r_{k+1}\|^2 \,, \end{aligned} \tag{6.13}$$

where C_2, C_3 and C_4 depend only on f and c. Supposing ϱ in]0, 1], we get from (6.12) and (6.13):

$$\begin{aligned} \Theta_p(y_k + \varrho t_k + \varrho^a r_{k+1}) \leqq{}& \Theta_p(y_k) + \varrho f'(y_k) \cdot t_k - \varrho^a (p - \|\lambda(y_k)\|_\infty) \|c(y_k)\|_1 \\ &+ C_5 (\varrho^a + \varrho^2) \|t_k\|^2 + C_6 \varrho^{2a} \|r_{k+1}\|^2 \,, \end{aligned} \tag{6.14}$$

where C_5 and C_6 depend only on f, c and $\bar{p}$. From the definition (4.10) of r_{k+1} and (6.11), we have

$$\|r_{k+1}\| \leqq C_7 \|c(y_k)\|_1 + C_8 \|t_k\|^2 \,, \tag{6.15}$$

where C_7 and C_8 depend only on c.

Now, let us suppose that (6.6) is not true for a given $\varrho = \beta^l$ in]0, 1]. Then, with (6.9), (6.14) and (6.15), we get

$$\begin{aligned} \varrho (1 - \alpha) \left(-f'(y_k) \cdot t_k\right) + \varrho^a (1 - \alpha) \underline{p} \|c(y_k)\|_1 \leqq{}& C_5 (\varrho^a + \varrho^2) \|t_k\|^2 \\ &+ C_6 \varrho^{2a} \|r_{k+1}\|^2 \,. \end{aligned}$$

But $-f'(y_k)\cdot t_k = g(y_k)^T G_k^{-1} g(y_k) \geqq h\|g(y_k)\|^2$ and $\|t_k\| \leqq C_9\|g(y_k)\|$ where C_9 depends only on c and h. Then, with (6.15), the last inequality becomes

$$\varrho\|g(y_k)\|^2 + \varrho^a\|c(y_k)\|_1 \leqq C_{10}\,(\varrho^a + \varrho^2)\,\|g(y_k)\|^2 + C_{11}\varrho^{2a}\|c(y_k)\|_1^2\,, \tag{6.16}$$

where C_{10} and C_{11} depend only on f, c, $\underline{p}$, $\bar{p}$, h and α. This inequality shows that ϱ cannot be arbitrarily small if $\|g(y_k)\| + \|c(y_k)\|_1 \neq 0$ (if $\|g(y_k)\| + \|c(y_k)\|_1 = 0$, $\varrho = \beta^l = 1$ clearly satisfies (6.6) because then $t_k = 0$, $r_{k+1} = 0$ and therefore $y_k(\varrho) = y_k$). Indeed, otherwise letting ϱ converge to 0 in (6.16) previously devided by ϱ, we would obtain $g(y_k) = 0$ and then dividing (6.16) by ϱ^a and taking the limit on ϱ would give $c(y_k) = 0$. This proves the first part of the lemma.

For the second part, let us suppose that rule (6.4)–(6.6) gives a step-size ϱ_k smaller than 1. Then (6.6) is not satisfied with $\varrho = \varrho_k/\beta$ and we have inequality (6.16) for this ϱ. Using (6.10), we obtain:

$$\varrho_k\|g(y_k)\|^2 + \varrho_k^a\|c(y_k)\|_1 \leqq C_{12}\varrho_k^b\,(\varrho_k\|g(y_k)\|^2 + \varrho_k^a\|c(y_k)\|_1)\,,$$

where $b := \min(1, a-1)$ and C_{12} depends only on f, c, $\underline{p}$, $\bar{p}$, α, β, h and M. Because $\varrho_k\|g(y_k)\|^2 + \varrho_k^a\|c(y_k)\|_1 \neq 0$ (otherwise $\varrho_k = 1$), the last inequality proves the second part of the lemma with $\underline{\varrho} = C_{12}^{-1/b}$. ■

Inequality (6.9) shows that the penalty parameter p has to be large enough to ensure the decrease of Θ_p along the arc (6.2) and that its lower bound depends on the current point y_k. So, sometimes it will be necessary to update the penalty parameter, which we shall denote by p_k. We shall suppose that the adapting rule of p_k will satisfy the following three conditions:

$$p_k \geqq \|\lambda(y_k)\|_\infty + \underline{p}, \quad \text{for every } k\,; \tag{6.17}$$

there exists a subscript K such that for every k greater than K,

$$(p_{k-1} \geqq \|\lambda(y_k)\|_\infty + \underline{p}) \text{ implies that } p_k = p_{k-1}\,; \tag{6.18}$$

$$(p_k) \text{ is bounded if and only if } p_k \text{ is modified finitely often}\,. \tag{6.19}$$

In (6.17) and (6.18), $\underline{p}$ is a given positive constant. Condition (6.18) means that eventually (for $k \geqq \bar{K}$), p_k is modified only if it is necessary to have (6.17). So $(p_k \mid k \geqq K)$ is an increasing sequence. An example of an adapting rule satisfying these conditions is given by Mayne and Polak [32]:

if $p_{k-1} \geqq \|\lambda(y_k)\|_\infty + \underline{p}$,
then $p_k := p_{k-1}$,
else $p_k := \max(\delta p_{k-1}, \|\lambda(y_k)\|_\infty + \underline{p})$,

where δ is a given constant greater than 1.

We are now able to state the algorithm that globalizes the local method (4.8)–(4.9).

Algorithm RQN:

1. Choose a convergence tolerance $\varepsilon > 0$, $\beta \in]0, 1[$, $\alpha \in]0, 1/2[$ and $a > 1$.
2. Choose y_0 in ω and a symmetric positive definite matrix G_0 of order $n-m$.

3. Let $k:=0$.
4. Repeat:
   4.1. Linearize the constraints at y_k: choose a right inverse $A(y_k)^-$ of $\nabla c(y_k)$ and a basis $Z(y_k)^-$ of $N(\nabla c(y_k))$ according to assumption B.
   4.2. Evaluate $\lambda(y_k):=-A(y_k)^{-\mathsf{T}}\nabla f(y_k)$ and $g(y_k):=Z(y_k)^{-\mathsf{T}}\nabla f(y_k)$.
   4.3. If $k\geq 1$ then evaluate the symmetric positive definite matrix G_k by updating G_{k-1}.
   4.4. Tangent step: evaluate $t_k:=-Z(y_k)^- G_k^{-1} g(y_k)$ and $x_{k+1}:=y_k+t_k$.
   4.5. Restoration step: evaluate $c(x_{k+1})$ and $r_{k+1}:=-A(y_k)^- c(x_{k+1})$.
   4.6. If $\|g(y_k)\|+\|c(x_{k+1})\|\leq\varepsilon$ then stop.
   4.7. Adapt p_k according to (6.17)–(6.19).
   4.8. Search a point y_{k+1} from y_k along the arc (6.2) in order to decrease the penalty function (6.1) (with $p=p_k$) with the help of the rule (6.3)–(6.6).
   4.9. Next iteration: set $k:=k+1$.

In the partitioned framework (see Section 3), only one linear system has to be solved at the step 4.2. Indeed, if $A(y_k)=[C(y_k)\ D(y_k)]$, $\lambda(y_k)$ is obtained by solving

$$C(y_k)^{\mathsf{T}}\lambda(y_k)=-\nabla_1 f(y_k)\,,$$

where $\nabla_1 f(y_k)$ is the vector formed by the first m components of $\nabla f(y_k)$. Then $g(y_k)=D(y_k)^{\mathsf{T}}\lambda(y_k)+\nabla_2 f(y_k)$, where $\nabla_2 f(y_k)$ is the vector formed by the last $n-m$ components of $\nabla f(y_k)$.

The important question of the update of the matrices (G_k), which is mentioned at step 4.3 of Algorithm RQN, has been investigated by Coleman and Conn [13] and by Gilbert [22], [25]. See Nocedal and Overton [34] for algorithm (1.12)–(1.14). G_{k+1} is obtained from G_k by the BFGS formula using two vectors γ_k and σ_k in $\mathbb{R}^{n-m}$:

$$G_{k+1}=G_k-\frac{G_k\sigma_k\sigma_k^{\mathsf{T}}G_k}{\sigma_k^{\mathsf{T}}G_k\sigma_k}+\frac{\gamma_k\gamma_k^{\mathsf{T}}}{\gamma_k^{\mathsf{T}}\sigma_k}\,. \tag{6.20}$$

Therefore, G_{k+1} satisfies the secant equation:

$$\gamma_k=G_{k+1}\sigma_k\,. \tag{6.21}$$

The point now is to choose adequately the vectors γ_k and σ_k so that G_{k+1} will approximate $G_*:=Z_*^{-\mathsf{T}}L_*Z_*^-$. This form of G_* and formula (4.2) suggest taking γ_k as the difference of two reduced gradients. A first possible choice consists in taking:

$$\gamma_k=g(x_{k+1})-g(y_k)\,, \tag{6.22}$$

$$\sigma_k=-G_k^{-1}g(y_k)\,. \tag{6.23}$$

Then, if we suppose that (y_k) and (x_k) converge to x_*, Taylor's theorem gives:

$$\gamma_k=G_*\sigma_k+o(\|\sigma_k\|)\,. \tag{6.24}$$

This relation and (6.21) show that γ_k and σ_k are correctly chosen. Unfortunately, this choice needs an additional linearization of the constraints at x_{k+1} in order to

calculate the reduced gradient at this point. This may be avoided by taking:

$$\gamma_k = g(y_{k+1}) - g(y_k) , \tag{6.25}$$

$$\sigma_k = -\varrho_k G_k^{-1} g(y_k) . \tag{6.26}$$

But in this case, (6.24) will not be necessarily satisfied any more, which means that updating G_k by formula (6.20) would deteriorate the matrix. Therefore, an update criterion of the form

$$\|\varrho_k^a r_{k+1}\| \leqq \mu_k \|\varrho_k t_k\| , \tag{6.27}$$

where (μ_k) is an appropriate sequence converging to zero, has to be introduced. When (6.27) is satisfied, it is not too difficult to show that the estimate (6.24) is still valid with γ_k and σ_k given by (6.25) and (6.26). The crucial point is now to choose correctly the sequence (μ_k) so that when (6.27) is *not* satisfied the superlinear rate of convergence of (x_k) can be preserved. A good choice for μ_k is

$$\mu_k = \mu \|e_{k--}\| ,$$

where μ is a small enough constant and $(k--)$ is the subscript of the last but one iteration at which (6.27) was satisfied, i.e. at which G_{k--} was updated by formula (6.20).

The update scheme at step 4.3 of Algorithm RQN is expected to generate a sequence of nonsingular matrices G_k satisfying

$$hI \leqq G_k^{-1} \leqq h^{-1} I , \tag{6.28}$$

for some positive constant h. This property is really not easy to obtain. However, using the same type of arguments that are used in unconstrained optimization, it can be proved either in a local framework (when (x_0, G_0) is supposed to be close to (x_*, G_*) and $\varrho_k = 1$) or when it is assumed that (x_k) and (y_k) converge to x_* with

$$\sum_{k=0}^{\infty} \|x_k - x_*\| < +\infty \quad \text{and} \quad \sum_{k=0}^{\infty} \|y_k - x_*\| < +\infty .$$

See [25].

The next theorem gives some global convergence result for Algorithm RQN under hypothesis (6.28).

Theorem 6.2: *Suppose that Assumptions* A *and* B *are satisfied and that f is bounded from below on ω. Let (x_k), (y_k) and (G_k) be the sequences generated by Algorithm* RQN *with α in* $]0, 1[$. *Suppose that (x_k) and (y_k) are in ω and that the matrices G_k are nonsingular and satisfy* (6.18) *with a positive constant h independent of k. Then, either (p_k) is unbounded and $(y_k \mid p_k \neq p_{k-1})$ has no accumulation point in ω, or (p_k) is bounded and*

$$\|g(y_k)\| + \|c(y_k)\|_1 \to 0 . \tag{6.29}$$

Proof: Suppose first that (p_k) is unbounded and let $\mathbb{K}$ be the subsequence of the subscripts $k \geqq K$ (K given in (6.18)) for which $p_k \neq p_{k-1}$. By (6.18),

$$p_{k-1} < \|\lambda(y_k)\|_\infty + \underline{p} ,$$

for k in $\mathbb{K}$. Because $(p_k \mid k \geqq K)$ is an increasing sequence, we see from this inequality that $\|\lambda(y_k)\|_- \to \infty$ for $k \to \infty$ in $\mathbb{K}$. Therefore $(y_k \mid p_k \neq p_{k-1})$ has no accumulation point in ω (here, we use the continuity of $y \to \lambda(y)$ and therefore, the surjectivity of $\nabla c(y)$ and assumption B are strongly invoked).

Now, let us suppose that (p_k) is bounded. From (6.19), p_k is constant when k is great enough. Let us say that $p_k = p$ for $k \geqq K_1$. So, at each iteration the same penalty function Θ_p decreases. The function f being bounded from below, we get

$$p\|c(y_k)\|_1 \leqq \Theta_p(y_{K_1}) - \inf f, \quad \text{for} \quad k \geqq K_1 .$$

Therefore, $(\|c(y_k)\|_1)$ is bounded and we can apply Lemma 6.1, which states the existence of a positive lower bound $\underline{\varrho}$ for the sequence (ϱ_k). From (6.28), we get $(-f'(y_k) \cdot t_k) \geqq h\|g(y_k)\|^2$ and then, with (6.17), (6.6) can be written

$$\underline{\varrho}\alpha h\|g(y_k)\|^2 + \underline{\varrho}^a \alpha \underline{p}\|c(y_k)\|_1 \leqq \Theta_p(y_k) - \Theta_p(y_{k+1}), \quad \text{for} \quad k \geqq K_1 .$$

But $(\Theta_p(y_k))$ converges (a decreasing bounded from below sequence). Therefore, taking the limit on k in this inequality shows that $\|g(y_k)\|$ and $\|c(y_k)\|_1$ converge to zero. ■

The last problem we tackle concerns the admissibility of the unit step-size. When $\varrho_k = 1$ is accepted by (6.6), Algorithm RQN proceeds like the local method (4.8)–(4.9) and superlinear convergence of (x_k) will occur when the reduced Hessian G_* is correctly approached by G_k (see Theorem 5.1, statement (iii)). It is known that this admissibility property is not satisfied when the SQP method is globalized with the penalty function (6.1) and the technique described at the beginning of this section. This has been called the "MARATOS effect" of the SQP method (see MARATOS [31]) and several remedies have been proposed to overcome this drawback: see [20], [10], [32] and [5]. This inconvenience is not shared with our algorithm. In fact, when $c(y_k) = 0$, which is a favourable situation for the appearance of the MARATOS effect, the total displacement $e_k = t_k + r_{k+1}$ is exactly the same as that of the SQP method with the MAYNE and POLAK's correction.

Let (x_k) in ω, (y_k) in ω and (G_k) be the sequences generated by Algorithm RQN and suppose that (y_k) converges to a solution x_* of (1.1). Let $\mathbb{K}$ be a subsequence of subscripts. We are interested in finding conditions under which ϱ_k will be equal to 1 for all but finitely many subscripts k in the subsequence $\mathbb{K}$. The following four properties will be meaningful:

$$\|G_k - G_*\| \leqq M \quad \text{for} \quad k \text{ in } \mathbb{K}, \tag{6.30}$$

$$(G_k - G_*) Z_* t_k = o(\|t_k\|) \quad \text{for} \quad k \text{ in } \mathbb{K}, \tag{6.31}$$

$$t_k = O(\|r_{k+1}\|) \quad \text{for} \quad k \text{ in } \mathbb{K}, \tag{6.32}$$

$$\varrho_k < 1 \quad \text{and} \quad t_k = o(\|r_k\|) \quad \text{for} \quad k \text{ in } \mathbb{K}. \tag{6.33}$$

Properties (6.30) and (6.31) concern the approximation of the reduced Hessian G_* by G_k. Property (6.30) is very strong when M is small and is usually not satis-

fied when second order derivatives are not calculated. Property (6.31) recalls condition (iii) of Theorem 5.1, which is when (G_k) and (G_k^{-1}) are bounded:

$$(G_k - G_*) Z_* t_k = o(\|d_k\|) . \tag{6.34}$$

Therefore, (6.31) is usually stronger than (6.34) and, in fact, is satisfied by some subsequences of subscripts when (G_k) is updated by the BFGS formula (see the discussion following the proof of Theorem 5.1). Property (6.32) concerns the comparison of the tangent step and the restoration step.

The next theorem shows that for the subsequences $\mathbb{K}$ for which (6.30) with M small enough or (6.31) or (6.32) is satisfied, the rule (6.3)–(6.6) will give $\varrho_k = 1$ for all but finitely many k in $\mathbb{K}$. Therefore, the unit step-size will be admissible either when G_* is correctly approximated by G_k (properties (6.30) and (6.31)) or when t_k is of the same order of magnitude as r_{k+1} (property (6.32)).

Property (6.33) is more particular. In concrete algorithms using the update scheme (6.20), (6.25) and (6.26) with the update criterion (6.27), neither of the properties (6.31)–(6.33) is satisfied for the entire sequence. These properties are satisfied only for subsequences $\mathbb{K}$. The result obtained in Theorem 6.3 with property (6.33) is then used to prove that only property (6.31) or (6.32) may occur in the considered algorithm (see [25]).

Theorem 6.3: *Suppose that Assumptions* A *and* B *are satisfied. Let* (x_k), (y_k) *and* (G_k) *be the sequences generated by Algorithm* RQN *with* α *in* $]0, 1/2[$. *Suppose that* (x_k) *and* (y_k) *are in* ω, *that* (y_k) *converges to* x_* *and that the matrices* G_k *are nonsingular and satisfy* (6.28) *with a positive constant* h *independent of* k. *Let* $\mathbb{K}$ *be a subsequence of subscripts. Then,*

(i) *there exists a positive constant* $\bar{M}$ *that depends only on* c, α *and* h *such that if* (6.30) *is satisfied with* $M < \bar{M}$ *then* $\varrho_k = 1$ *for all but finitely many* k *in* $\mathbb{K}$,

(ii) *if* (6.31) *or* (6.32) *is satisfied then* $\varrho_k = 1$ *for all but finitely many* k *in* $\mathbb{K}$,

(iii) *if* (6.33) *is satisfied, then* $r_{k+1} = o(\|r_k\| \, \|t_k\|)$ *for* k *in* $\mathbb{K}$.

Proof: Since (y_k) converges, Proposition 6.2 shows that (p_k) is bounded and by (6.19), p_k is modified finitely often. So we can suppose that $p_k = p$ for all k. By Taylor's theorem, we expand $\Theta_p(y_k + t_k + r_{k+1})$ at the second order in t_k and the first order in r_{k+1}. First, note that because (y_k) converges to x_* and (G_k^{-1}) is bounded, (x_k) converges to x_*. Then, we have

$$c(x_{k+1}) = c(y_k) + \frac{1}{2} c''(x_*) \cdot t_k^2 + o(\|t_k\|^2) . \tag{6.35}$$

We also have

$$f(y_k + e_k) = f(y_k) + f'(y_k) \cdot t_k + c(x_{k+1})^{\mathsf{T}} \lambda(y_k) + \frac{1}{2} f''(x_*) \cdot t_k^2 + o(\|t_k\|^2) + o(\|r_{k+1}\|)$$

and using the estimate (6.35), we get

$$f(y_k + e_k) = f(y_k) + f'(y_k) \cdot t_k + c(y_k)^{\mathsf{T}} \lambda(y_k) + \frac{1}{2} t_k^{\mathsf{T}} L_* t_k + o(\|t_k\|^2) + o(\|r_{k+1}\|) . \tag{6.36}$$

On the other hand, expanding $c(y_k+e_k)$ about y_k we obtain

$$c(y_k+e_k)=c(y_k)-c(x_{k+1})+\frac{1}{2}c''(y_k)\cdot t_k^2+o(\|t_k\|^2)+o(\|r_{k+1}\|)$$

and using again (6.35), we get

$$c(y_k+e_k)=o(\|t_k\|^2)+o(\|r_{k+1}\|)\,. \tag{6.37}$$

Let us define

$$\Delta_k:=f'(y_k)\cdot t_k-(p-\|\lambda(y_k)\|_\infty)\,\|c(y_k)\|_1\,,$$

which is negative by (6.28) and (6.17). Finally, (6.36) and (6.37) give

$$\Theta_p(y_k+e_k)\leqq\Theta_p(y_k)+\Delta_k+\frac{1}{2}t_k^{\mathsf T}L_*t_k+o(\|t_k\|^2)+o(\|r_{k+1}\|)\,.$$

But $t_k=Z_*^{-}Z_*t_k+o(\|t_k\|)$ and the boundedness of (G_k) allows us to write $g(y_k)=$ $=-G_kZ_*t_k+o(\|t_k\|)$. Therefore, using $f'(y_k)\cdot t_k=-g(y_k)^{\mathsf T}G_k^{-1}g(y_k)=-t_k^{\mathsf T}Z_*^{\mathsf T}G_kZ_*t_k+$ $+o(\|t_k\|^2)$, we obtain

$$\begin{aligned}&\Theta_p(y_k+e_k)-\Theta_p(y_k)-\alpha\Delta_k\\&\leqq\left(\frac{1}{2}-\alpha\right)\Delta_k-\frac{1}{2}t_k^{\mathsf T}Z_*^{\mathsf T}(G_k-G_*)\,Z_*t_k+o(\|t_k\|^2)+o(\|r_{k+1}\|)\,.\end{aligned}\tag{6.38}$$

Using this inequality we now prove the theorem. Suppose that the step-size ϱ_k is different from 1 for infinitely many k in a subsequence $\mathbb{K}$, say for $k\in\mathbb{K}'\subset\mathbb{K}$. Then, according to Armijo's rule (6.4)–(6.6), the left hand side of (6.38) is positive and we have for $k\in\mathbb{K}'$:

$$-\Delta_k\leqq-\frac{1}{1-2\alpha}t_k^{\mathsf T}Z_*^{\mathsf T}(G_k-G_*)\,Z_*t_k+o(\|t_k\|^2)+o(\|r_{k+1}\|)\,.$$

Using the inequality $C_1\|t_k\|\leqq\|g(y_k)\|$ (where C_1 is a positive constant that depends only on c and h), property (6.28), inequality (6.17), the definition (4.10) of r_{k+1} and (6.35), we can obtain a lower bound for the left hand side:

$$\begin{aligned}&hC_1^2\|t_k\|^2+\underline{p}\|c(y_k)\|_1\\&\leqq-\frac{1}{1-2\alpha}t_k^{\mathsf T}Z_*^{\mathsf T}(G_k-G_*)\,Z_*t_k+o(\|t_k\|^2)+o(\|c(y_k)\|_1)\,.\end{aligned}\tag{6.39}$$

Now, if one of the properties (6.30) with $M<\bar{M}:=(1-2\alpha)\,hC_1^2/\|Z_*\|^2$ or (6.31) or (6.32) is verified for $k\in\mathbb{K}$, this inequality leads to

$$\|t_k\|^2+\|c(y_k)\|_1=o(\|t_k\|^2)+o(\|c(y_k)\|_1)\,,$$

for $k\in\mathbb{K}'$, which shows that $\mathbb{K}'$ cannot be infinite. This proves statements (i) and (ii) of the theorem.

It remains to prove (iii). With (6.33), inequality (6.39) is valid for $k\in\mathbb{K}$. And as $t_k=o(\|r_k\|)$, it implies for $k\in\mathbb{K}$:

$$\|t_k\|^2+\|c(y_k)\|_1=o(\|r_k\|\,\|t_k\|)+o(\|t_k\|^2)+o(\|c(y_k)\|_1)\,,$$

from which we deduce

$$\|t_k\|^2 + \|c(y_k)\|_1 = o(\|r_k\| \, \|t_k\|) .$$

Then, this estimate, the definition of r_{k+1} and (6.35) give

$$r_{k+1} = o(\|r_k\| \, \|t_k\|) + o(\|t_k\|^2) .$$

Therefore, using $t_k = o(\|r_k\|)$, we get the estimate in (iii). ■

7. Conclusion

In this paper, we have studied the local and global convergence of a variable metric algorithm for equality constrained optimization in which the order of the updated matrices is $n-m$. This reduced method can be seen as making a link between GRG-like methods which are feasible methods ($c(x_k)=0$ for all k) with reduced matrices (of order $n-m$) and the SQP method, which is an unfeasible method with full matrices (of order n). The studied algorithm is indeed an unfeasible method with reduced matrices. The algorithm inherits also the good properties of both methods (reduced metrics, superlinear convergence and unfeasibility) and shows, in particular, that *locally* only one restoration step is necessary to obtain the superlinear convergence of GRG-like methods when the reduced matrices are correctly approximated.

The global convergence is obtained by Han's technique to globalize the SQP method. The l_1 penalty function is used as a merit function and is decreased along an arc-shaped search path. Conditions for the asymptotic admissibility of the unit step-size are given that turn out to be satisfied in practice.

An important facet of the method has not been tackled here and is reported elsewhere (Gilbert [25]). This concerns the update of the reduced matrices G_k. This one is based on a secant equation using the change in the reduced gradient g. The fact that the gradient of g at x_* (see (4.2)) is not equal to G_* (and cannot be equal because $\nabla g(x_*)$ is an $(n-m)\times n$ matrix while G_* is of order $n-m$) leads to an alternative. Either the reduced gradient is evaluated twice per iteration, at y_k and x_{k+1}, or it is evaluated only once per iteration, at y_k. In the first case, the change $g(x_{k+1})-g(y_k)$ is used in the secant equation and the matrices G_k are updated at each iteration but with the inconvenience of having to linearize the constraints twice per iteration: see Coleman and Conn [13] and Gilbert [25]. In the second case, the change $g(y_{k+1})-g(y_k)$ is used in the secant equation but usually the matrices G_k can no longer be updated at each iteration. An update criterion has to be introduced in order to decide when an update is appropriate. Despite this, the superlinear convergence can be achieved either in a local framework (see [34] for algorithm (1.12)–(1.14)) or in a global framework (see [25]) for Algorithm RQN of Section 6).

Acknowledgements: This work was supported in part during 1984–1985 by the Centre d'Etudes Nucléaires, Fontenay-aux-Roses (France) under a grant from the Commission of the European Community and in part by the Institut National de Recherche en Informatique et en Automatique (INRIA), Le Chesnay (France). I would like to thank those organizations and to express my gratitude to J. F. Bonnans for helpful discussions and suggestions and for his friendly and constant advice.

References

[1] Abadie, J.; J. Carpentier: Generalization of the Wolfe reduced gradient method to the case of nonlinear constraints. *In:* R. Fletcher (ed.): Optimization. Academic Press, London, 1969.

[2] Ben-Israel, A.; T. N. E. Greville: Generalized inverses: theory and applications. John Wiley & Sons, 1974.

[3] Blum, J.; J. Ch. Gilbert; B. Thooris: Parametric identification of the plasma current density from the magnetic measurements and the pressure profile, code IDENTC. Report of JET contract number JT3/9008. Centre d'Etudes Nucléaires (DRFC), B.P. 6, 92265 Fontenay-aux-Roses (France), 1985.

[4] Boggs, P. T.; J. W. Tolle; P. Wang: On the local convergence of quasi-Newton methods for constrained optimization. SIAM Journal on Control and Optimization, **20** (1982) 2, 161–171.

[5] Bonnans, J. F.: Asymptotic admissibility of the unity stepsize in exact penalty methods I: equality-constrained problems. Rapport de recherche de l'INRIA RR-273, 78153 Le Chesnay, France, 1984.

[6] Bonnans, J. F.; D. Gabay: Une extension de la programmation quadratique successive. *In:* A. Bensoussan, J. L. Lions (eds.): Lecture Notes in Control and Information Sciences, **63** (1984) 16–31, Springer-Verlag.

[7] Byrd, R. H.: On the convergence of constrained optimization methods with accurate Hessian information on a subspace. Report CU-CS-270-84. Department of Computer Science, University of Colorado, Boulder, Colorado 80309.

[8] Byrd, R. H.: An example of irregular convergence in some constrained optimization methods that use the projected Hessian. Mathematical Programming, **32** (1985) 232–237.

[9] Byrd, R. H.; R. B. Schnabel: Continuity of the null space basis and constrained optimization. Mathematical Programming, **35** (1986) 32–41.

[10] Chamberlain, R. M.; C. Lemaréchal; H. C. Pedersen; M. J. D. Powell: The watchdog technique for forcing convergence in algorithms for constrained optimization. Mathematical Programming Study, **16** (1982) 1–17.

[11] Coleman, T. F.; A. R. Conn: Nonlinear programming via an exact penalty function: asymptotic analysis. Mathematical Programming, **24** (1982) 123–136.

[12] Coleman, T. F.; A. R. Conn: Nonlinear programming via an exact penalty function: global analysis. Mathematical Programming, **24** (1982) 137–161.

[13] Coleman, T. F.; A. R. Conn: On the local convergence of a quasi-Newton method for the nonlinear programming problem. SIAM Journal on Numerical Analysis, **21** (1984) 4, 755–769.

[14] Danilin, Y. M.; B. N. Pschenichny: Numerical methods in extremal problems. MIR, Moscow, 1965. (English translation, 1978).

[15] Dennis, J. E.; J. J. Moré): A characterization of superlinear convergence and its application to quasi-Newton methods. Mathematics of Computation, **28** (1974) 126, 549–560.

[16] Dennis, J. E.; J. J. Moré: Quasi-Newton methods, motivation and theory. SIAM Review, **19** (1977) 46–89.

[17] Fletcher, R.: Practical methods of optimization. Vol. 2: Constrained Optimization. Wiley, 1981.

[18] Gabay, D.: Efficient convergence of implementable gradient algorithms and stepsize selection procedures for constrained minimization. *In:* F. Gelenbe, K. D. Potier (eds.): International Computing Symposium 1975. North-Holland Publishing Company, 1975.

[19] Galay, D.: Minimizing a differentiable function over a differentiable manifold. Journal of Optimization Theory and Applications, **37** (1982) 2, 171–219.

[20] Gabay, D.: Reduced quasi-Newton methods with feasibility improvement for nonlinearly constrained optimization. Mathematical Programming Study, **16** (1982) 18–44.

[21] Gabay, D.; D. G. Luenberger: Efficiently converging minimization methods based on the reduced gradient. SIAM Journal on Control and Optimization, **14** (1976) 1, 42–61.

[22] Gilbert, J. Ch.: Une méthode à métrique variable réduite en optimisation avec contraintes d'égalité non linéaires. Rapport de recherche de l'INRIA RR-482.78153 Le Chesnay, France, 1986.

[23] Gilbert, J. Ch.: Une méthode de quasi-Newton réduite en optimisation sous contraintes avec priorité à la restauration. *In:* A. Bensoussan, J. L. Lions (eds.): Lecture Notes in Control and Information Sciences **83** (1986) 40–53, Springer-Verlag.

[24] Gilbert, J. Ch.: Sur quelques problèmes d'identification et d'optimisation rencontrés en physique des plasmas. Thèse de doctorat de l'université Paris VI, 1986.

[25] Gilbert, J. Ch.: Mise à jour de la métrique dans les méthodes de quasi-Newton réduites en optimisation avec contraintes d'égalité. Mathematical Modelling and Numerical Analysis, **22** (1988) 251–288.

[26] Gill, P. E.; W. Murray; M. A. Saunders; G. W. Stewart; M. H. Wright: Properties of a representation of a basis for the null space. Mathematical Programming, **33** (1985) 172–186.

[27] Goodman, J.: Newton's method for constrained optimization. Mathematical Programming, **33** (1985) 162–171.

[28] Han, S. P.: Superlinearly convergent variable metric algorithms for general nonlinear programming problems. Mathematical Programming, **11** (1976) 263–282.

[29] Han, S. P.: A globally convergent method for nonlinear programming. Journal of Optimization Theory and Applications, **22** (1977) 3, 297–309.

[30] Hoyer, W.: Variants of the reduced Newton method for nonlinear equality constrained optimization problems. Optimization, **17** (1986) 757–774.

[31] Maratos, N.: Exact penalty function algorithms for finite dimensional and control optimization problems. Ph. D. Thesis, University of London, 1978.

[32] Mayne, D. Q.; E. Polak: A superlinearly convergent algorithm for constrained optimization problems. Mathematical Programming Study, **16** (1982) 45–61.

[33] Mukai, H.; E. Polak: On the use of approximations in algorithms for optimization problems with equality and inequality constraints. SIAM Journal on Numerical Analysis, **15** (1978) 4, 674–693.

[34] Nocedal, J.; M. L. Overton: Projected Hessian updating algorithms for nonlinearly constrained optimization. SIAM Journal on Numerical Analysis, **22** (1985) 5, 821–850.

[35] Powell, M. J. D.: The convergence of the variable metric methods for nonlinearly constrained optimization calculations. *In:* O. L. Mangasarian, R. R. Meyer, S. M. Robinson (eds.): Nonlinear Programming 3, Academic Press, 1978, 27–63.

[36] Schwartz, L.: Cours d'analyse. Tome I. Hermann, 1981.
[37] Stoer, J.: Principles of sequential quadratic programming methods for solving nonlinear programs. Proceedings of the NATO ASI on Computational Mathematical Programming, Bad Windsheim, Germany, 1984.
[38] Wilson, R. B.: A simplicial algorithm for concave programming. Ph. D. thesis. Graduate School of Business Administration, Harvard Univ., Cambridge, MA, 1963.
[39] Yuan, Y.: An only 2-step Q-superlinear convergence example for some algorithms that use reduced Hessian approximations. Mathematical Programming, **32** (1985) 224–231.

Received November 1987, revised October 1988

J. Ch. Gilbert
INRIA
Domaine de Voluceau, Recquencourt
F - 78153 - Le Chesnay Cedex
France