数理解析研究所講究録
第 648 巻 1988 年 200-220

AN APPLICATION OF SEQUENTIAL QUADRATIC PROGRAMMING METHOD TO CONSTRAINED NONLINEAR LEAST SQUARES PROBLEMS
（制約条件付き最小二乗問題に対するＳＱＰ法の適用とそのソフトウェア）

システム計画研　八巻直一（Naokazu Yamaki）
東京理大・工　矢部 博（Hiroshi Yabe）
東京理大・理　高橋 悟（Satoru Takahashi）
専修大学・経営　本郷 茂（Shigeru Hongo）
青山学院女子短大　宮田雅智（Masanori Miyata）

1. Introduction

This paper is concerned with solving the constrained nonlinear least squares problem:

(1.1) minimize $f(x) = (1/2)\sum_{j=1}^{p}(r_j(x))^2$

subject to the inequality and equality constraints

(1.2) $g_i(x) \leqq 0, \quad i = 1,\ldots, m,$

(1.3) $h_j(x) = 0, \quad j = 1,\ldots, l,$

on the vector of variables $x \in R^n$, where the functions $r_j(x)(j = 1, \ldots, p)$, $g_i(x)(i = 1,\ldots, m)$, $h_j(j = 1,\ldots, l)$ are real and twice continuously differentiable.

In developing numerical methods for solving general nonlinear optimization problems, penalty function methods, augmented Lagrangian function methods, and gradient projection methods are well known [5]. Recent research has centered on implementing some form of a quasi-Newton technique (Han[8],[9], Powell[12], Yamashita[20]). This class of algorithms is known as the sequential quadratic programming (SQP) method or the successive quadratic programming method. The global convergence, and the local and superlinear convergence, of these methods have been investigated by a number of authors.

On the other hand, Dennis, Gay and Welsch[4] proposed the quasi-Newton based method for unconstrained nonlinear least squares problems, by using the special structure of the Hessian matrix. So, for solving constrained nonlinear least squares problems, we can combine Dennis, Gay and Welsch's strategy with the SQP method by using the special structure of the Hessian matrix of the Lagrangian function of the above problem.

In Section 2, we state the SQP algorithm and the global convergence of this method. Further, Goldfarb and Idnani's QP method is introduced. In Section 3, we summarize the quasi-Newton methods for solving unconstrained nonlinear least squares problems. Section 4 gives a modification of the SQP method for constrained nonlinear least squares problems. In Section 5, the numerical experiments are presented. Finally, the nonlinear optimization code ASNOP(Application System for Nonlinear Optimization Problems) is described. Through this paper, the subscript "k" denotes the iretation number.

2. Sequential Quadratic Programming Method

This section considers the general nonlinear programming problem:

(NLP) minimize $f(x)$

subject to the inequality and equality constraints

(2.1) $$g(x) \leqq 0, \quad g(x) = (g_1(x), \ldots , g_m(x))^T,$$

(2.2) $$h(x) = 0, \quad h(x) = (h_1(x), \ldots , h_l(x))^T$$

on the vector of variables $x \in R^n$, where the functions $f(x)$, $g_i(x)$ $(i = 1,\ldots,m)$ and $h_j(x)$ $(j = 1,\ldots,l)$ are real and twice continuously differentiable. Solving Problem NLP can be reduced to finding a Kuhn-Tucker point (x^*, λ^*, μ^*) which satisfies the following conditions:

(Kuhn-Tucker condition)

(2.3) $$\nabla_x L(x^*, \lambda^*, \mu^*) = \nabla f(x^*) + \nabla g(x^*)^T \lambda^* + \nabla h(x^*)^T \mu^* = 0,$$

(2.4) $g(x^*) \leqq 0,$

(2.5) $h(x^*) = 0,$

(2.6) $\lambda^* \geqq 0,$

(2.7) $(\lambda^*)^T g(x^*) = 0,$

where $L(x, \lambda, \mu)$ is a Lagrangian function of Problem NLP defined by

(2.8) $L(x, \lambda, \mu) = f(x) + \lambda^T g(x) + \mu^T h(x),$

λ, μ are vectors of Lagrangian multipliers, ∇f is the gradient vector of f, ∇g and ∇h are the Jacobian matrices of g and h, respectively.

In subsection 2.1 below, we describe the SQP method for solving Problem NLP and the global convergence, following Han[9]. Since the SQP method defines the search direction by solving the strictly convex QP subproblem on each iteration, an efficient QP solver must be chosen. In subsection 2.2, we recommend Goldfarb and Idnani's QP method and state some features of this.

2.1 SQP Algorithm

The SQP algorithm is the following:

(SQP Algorithm)

Starting with a point $x_1 \in R^n$, an $n \times n$ symmetric positive definite matrix B_1, and three numbers $\delta > 0$, $\omega \in (0, 0.5)$ and $\tau \in (0, 1)$, the algorithm proceeds, for k = 1, 2, ... , as follows:

Step 1. Having x_k and B_k, find the search direction d_k by solving the QP subproblem:

(QP subproblem)

(2.9) minimize $(1/2)d^T B_k d + \nabla f(x_k)^T d$

(2.10) subject to $g(x_k) + \nabla g(x_k)d \leqq 0,$

(2.11) $h(x_k) + \nabla h(x_k)d = 0,$

and choose λ_{k+1} and μ_{k+1} to be the optimal multiplier vectors for this problem.

Step 2. If the vectors x_k, λ_{k+1} and μ_{k+1} satisfy the Kuhn-Tucker condition of Problem NLP, then stop; otherwise, go to Step 3.

Step 3. Determine a step length α_k by the line search algorithm:

Step 3.1 Set $\beta_{k,1} = 1$ and $j = 1$.

Step 3.2 If

(2.12) $$\theta_\delta(x_k + \beta_{k,j}d_k) \leqq \theta_\delta(x_k) - \omega\beta_{k,j}d_k^T B_k d_k,$$

then set $\alpha_k = \beta_{k,j}$ and go to Step 4; otherwise, go to Step 3.3, where $\theta_\delta(x)$ is a line search objective function and is defined by

(2.13) $$\theta_\delta(x) = f(x)+\delta\max(0,g_1(x),\ldots,g_m(x),|h_1(x)|,\ldots,|h_l(x)|).$$

Step 3.3 Set $\beta_{k,j+1} = \tau\beta_{k,j}$, $j = j + 1$ and go to Step 3.2.

Step 4. Set $x_{k+1} = x_k + \alpha_k d_k$.

Step 5. Update B_k giving a symmetric positive definite matrix B_{k+1} by a quasi-Newton updating formula.

It should be noted that $\theta_\delta(x)$ in (2.13) is an exact penalty function and that the condition (2.12) is the extension of Armijo's rule to the nondifferentiable function [20]. Han[9] has shown that the line search algorithm in Step 3 terminates after a finite number of procedure. Furthermore, the following theorem shows the global convergence of the SQP method.

<u>Theorem 1.</u>(Han[9]) For given positive δ, let the level set at a starting point $\{ x \in R^n \mid \theta_\delta(x) \leqq \theta_\delta(x_1) \}$ be compact. Assume that, at each k, the QP subproblem is feasible and there holds

(2.14) $$\delta \geqq \|\lambda_k\|_1 + \|\mu_k\|_1.$$

Suppose that there exist positive constants ξ_1 and ξ_2 such that

(2.15) $\quad \xi_1 \| v \|_2^2 \leqq v^T B_k v \leqq \xi_2 \| v \|_2^2$

for any $v \in R^n$ and each $k \geqq 1$.

Then either the sequence $\{(x_k, \lambda_{k+1}, \mu_{k+1})\}$ generated by SQP Algorithm terminates at a Kuhn-Tucker point of Problem NLP or any accumulation point of the sequence is a Kuhn-Tucker point of Problem NLP.

In SQP Algorithm, the matrix B_k is an approximation to the Hessian matrix of $L(x, \lambda, \mu)$ with respect to x, in the sense that

(2.16) $$\begin{bmatrix} B_k & \nabla g(x_k)^T & \nabla h(x_k)^T \\ \nabla g(x_k) & 0 & 0 \\ \nabla h(x_k) & 0 & 0 \end{bmatrix} \sim \begin{bmatrix} \nabla_{xx} L_k & \nabla g(x_k)^T & \nabla h(x_k)^T \\ \nabla g(x_k) & 0 & 0 \\ \nabla h(x_k) & 0 & 0 \end{bmatrix},$$

where $\nabla_{xx} L_k = \nabla_{xx} L(x_k, \lambda_k, \mu_k)$, and Step 5 is the quasi-Newton update, which is to construct a symmetric positive definite matrix B_{k+1} such that the secant condition

(2.17) $\quad B_{k+1} s_k = y_k$

is satisfied, where

(2.18) $\quad s_k = x_{k+1} - x_k$

and

(2.19) $\quad y_k = \nabla_x L(x_{k+1}, \lambda_{k+1}, \mu_{k+1}) - \nabla_x L(x_k, \lambda_{k+1}, \mu_{k+1})$.

Typical updates are the BFGS and the DFP updates, which possess hereditary positive definiteness if and only if $s_k^T y_k > 0$. However, vectors s_k and y_k do not necessarily satisfy $s_k^T y_k > 0$ for the constrained minimization. If $s_k^T y_k \leqq 0$, we can no longer obtain a symmetric positive definite matrix which satisfies the secant condition. To maintain the positive definiteness of B_k, Powell[12] has recommend-ed the condition

(2.20) $B_{k+1}s_k = \eta_k, \quad \eta_k = \psi_k y_k + (1-\psi_k) B_k s_k$

instead of the original secant condition, where ψ_k is a parameter chosen such that $s_k^T(\psi_k y_k + (1-\psi_k) B_k s_k) > 0$. Powell has proposed the formula based on the BFGS update, which is expressed by

(2.21) $B_{k+1} = B_k - B_k s_k s_k^T B_k / s_k^T B_k s_k + \eta_k \eta_k^T / s_k^T \eta_k,$

where

(2.22) $\psi_k = 1 \quad \text{if} \quad s_k^T y_k \geqq 0.2\, s_k^T B_k s_k,$

(2.23) $\psi_k = 0.8\, s_k^T B_k s_k / s_k^T (B_k s_k - y_k),$ otherwise.

<u>2.2 Goldfarb and Idnani's QP Method</u>

Goldfarb and Idnani[7] have developed a numerically stable dual method for solving the strictly convex QP problem

(2.24) minimize $(1/2)d^T G\, d + a^T d$ with respect to $d \in R^n$

(2.25) subject to $C^T d + b \leqq 0,$

where G is an $n \times n$ symmetric positive definite matrix, C is an $n \times m$ matrix and a, b are n and m dimensional vectors, respectively. In this paper, we omit the details of their algorithm. The features of the algorithm are as follows:

(1) The unconstrained minimum of (2.24), $-G^{-1}a$, is used as a starting point in the primal space. The origin is used as a starting point in the dual space.

(2) The algorithm is based upon projections onto active sets of constraints.

(3) The algorithm iterates until primal feasibility (dual optimality) is achieved. It is very important that the origin in the dual space is always dual feasible, so that no Phase 1 procedure is required.

(4) The algorithm solves the QP problem (2.24) or indicates that it has no feasible solution in a finite number of steps.

(5) The numerical implementation is based upon the Cholesky factori-

zation $G = LL^T$ and the QR factorization $L^{-1}N = [Q_1 \mid Q_2][R^T \mid 0]^T$, where L is an n×n lower triangular matrix, q is the number of the current active constraints, R is a q×q upper triangular matrix, $[Q_1 \mid Q_2]$ is an n×n orthogonal matrix, and N is an n×q matrix whose columns are the normal vectors of the constraints in the current active set. Let the matrix P be $P = L^{-T}[Q_1 \mid Q_2]$. Whenever a new constraint is added to or deleted from the set of active constraints, the matrices P and R are updated by using the Givens rotation. Since q = 0 at a starting point, the matrix P is set to L^{-T} and the matrix R starts from the empty matrix.

Note that the above algorithm can be easily applied to the strictly convex QP problem with inequality and equality constraints. It is important to be able to make use of the final active set in the last outer iteration of SQP Algorithm, and if the starting point in the primal space is feasible, we can use it as the new search direction directly. So it is stated that this QP method is particularly suitable for the SQP method ([13],[18]).

If we use Powell's modified BFGS update (2.21) in the SQP method with Goldfarb and Idnani's QP algorithm, the matrix B_k is factorized in the triangular form. In which case, Gill, Murray and Saunders' technique [6] can be used. They have dealt with the Cholesky factorization of a symmetric rank one update

(2.26) $$L_{k+1}L_{k+1}^T = L_kL_k^T + \sigma_k v_k v_k^T,$$

where L_k and L_{k+1} are n×n lower triangular matrices, v_k is a vector and σ_k is a scalar. So we have the following procedure:

(Procedure A)

Step 1.Applying Gill, Murray and Saunders' technique to Powell's modified BFGS update (2.21) in twice, we have the Cholesky factor

L_{k+1} in $O(n^2)$ operations.

Step 2. The matrix L_{k+1}^{-T} is calculated in $O(n^3)$ operations by the backward substitution.

Step 3. In Goldfarb and Idnani's QP algorithm, $P = L_{k+1}^{-T}$ is used as an initial matrix.

3. Unconstrained Nonlinear Least Squares Problems

This section is concerned with the problem of minimizing a sum of squared nonlinear functions

$$f(x) = (1/2) \sum_{j=1}^{p} (r_j(x))^2, \tag{3.1}$$

where $r_j: R^n \rightarrow R$ are twice continuously differentiable for $j=1,\ldots,p$ and $p \geq n$. Most iterative methods for the above problem are variants of Newton's method. At the k-th iteration of Newton's method, the search direction d_k is computed by

$$\nabla^2 f(x_k)\, d_k = -\nabla f(x_k) \tag{3.2}$$

and the new point is generated by

$$x_{k+1} = x_k + d_k\ . \tag{3.3}$$

Here x_k is a current estimate of the minimum point x^* and ∇f, $\nabla^2 f$ are the gradient vector and the Hessian matrix of f, respectively, which are given by

$$\nabla f(x) = J(x)^T r(x), \tag{3.4}$$

$$\nabla^2 f(x) = J(x)^T J(x) + \sum_{j=1}^{p} r_j(x) \nabla^2 r_j(x), \tag{3.5}$$

where

$$r(x) = (r_1(x), \ \ldots\ , r_p(x))^T \tag{3.6}$$

and J is the $p \times n$ Jacobian matrix whose (i,j)-th element is $\partial r_i / \partial x_j$.

Since the cost of providing the complete Hessian matrix is often expensive, methods have been derived which use only the first derivative information. For example, the Gauss-Newton method and the Levenberg-Marquardt method are well known. Since these methods neglect the second part of the Hessian matrix of f, they can be expected to perform well when the residuals at x^* are small or the functions r_i are close to linear. However, when the residuals at x^* are very large and the functions are rather nonlinear, these methods may perform poorly [3,Chapter 10].

On the other hand, the quasi-Newton approximation to the second part of the Hessian matrix has been considered [3],[11]. Recently, the robust algorithms have been proposed by Biggs[1] and Dennis, Gay and Welsch[4]. Since the nonlinear least squares algorithms usually calculate the Jacobian matrix J analytically or numerically, the portion $J(x)^TJ(x)$ of $\nabla^2 f(x)$ is always readily available, so we only have to approximate the second part of $\nabla^2 f(x)$. Thus, for unconstrained nonlinear least squares problems, it has been considered that the search direction can be computed by

$$(3.7) \qquad (J(x_k)^TJ(x_k) + A_k)d_k = -J(x_k)^Tr_k,$$

where $r_k = r(x_k)$ and the matrix A_k is the k-th approximation to the second part of the Hessian matrix of f[11]. The matrix A_k is updated such that the new matrix A_{k+1} satisfies the secant condition

$$(3.8) \qquad A_{k+1}s_k = u_k, \quad u_k = y_k - J(x_{k+1})^TJ(x_{k+1})s_k$$

or

$$(3.9) \qquad A_{k+1}s_k = v_k, \quad v_k = (J(x_{k+1}) - J(x_k))^Tr_{k+1}.$$

The first is proposed by Broyden and Dennis (BD) [2], and follows directly from the usual secant condition. The second is proposed by Biggs[1] and by Dennis, Gay and Welsch (DGW) [4], and is based on the special structure of the second part of the Hessian matrix. The Hessian matrices of r_j (j=1,...,p) at x_{k+1} can be approximated by the form, in the direction of s_k,

(3.10) $\nabla^2 r_j(x_{k+1})\, s_k \sim \nabla r_j(x_{k+1}) - \nabla r_j(x_k), \quad j = 1,\ldots,p.$

We therefore have the secant condition (3.9). The quasi-Newton methods based on (3.7) are called the special purpose quasi-Newton methods.

Special purpose quasi-Newton updates are usually of rank one or of rank two. Broyden and Dennis gave the following update:

(i)the BD update

(3.11) $$A_{k+1} = A_k + ((u_k - A_k s_k)s_k^T + s_k(u_k - A_k s_k)^T)/s_k^T s_k - ((u_k - A_k s_k)^T s_k/(s_k^T s_k)^2)s_k s_k^T.$$

Biggs and Dennis et al. have used scaling techniques and proposed the following updates:

(ii)the Biggs update

(3.12) $$A_{k+1} = \beta_k A_k + (v_k - \beta_k A_k s_k)(v_k - \beta_k A_k s_k)^T/(v_k - \beta_k A_k s_k)^T s_k,$$

(3.13) $$\beta_k = r_{k+1}^T r_k \,/\, r_k^T r_k,$$

(iii)the DGW update

(3.14) $$A_{k+1} = \beta_k A_k + ((v_k - \beta_k A_k s_k)y_k^T + y_k(v_k - \beta_k A_k s_k)^T)/s_k^T y_k - \{s_k^T(v_k - \beta_k A_k s_k)/(s_k^T y_k)^2\}y_k y_k^T,$$

(3.15) $$\beta_k = \min(\,|\, s_k^T v_k \,/ s_k^T A_k s_k \,|\,,\ 1\,),$$

where β_k is a scaling parameter.

The Biggs and the DGW methods are robust for both cases of large and small residual problems. When the residual at the minimum point is large, information of the term $\sum_{j=1}^{p} r_j(x)\nabla^2 r_j(x)$ is included in the matrix A_{k+1}, so these algorithms perform as well as the BD method. On the other hand, for the very small or the zero residual problems, these algorithms perform almost as well as the Gauss-Newton method.

This is based on the fact that if the residual $\| r(x_{k+1}) \|_2$ at x_{k+1} becomes zero, then the parameter β_k is zero and the new matrix A_{k+1} becomes a zero matrix.

4. Constrained Nonlinear Least Squares Problems

Consider the constrained nonlinear least squares problem:

(NLS)

(4.1) minimize $f(x) = (1/2)r(x)^T r(x) = (1/2) \sum_{j=1}^{p} (r_j(x))^2$

subject to the inequality and equality constraints

(4.2) $g(x) \leqq 0, \quad g(x) = (g_1(x), \ldots, g_m(x))^T,$

(4.3) $h(x) = 0, \quad h(x) = (h_1(x), \ldots, h_l(x))^T$

on the vector of variables $x \in R^n$, where the functions $r_j(x)(j = 1, \ldots, p)$, $g_i(x)(i = 1,\ldots, m)$, $h_j(j = 1,\ldots, l)$ are real and twice continuously differentiable, and $r(x)$ is a residual vector given by

(4.4) $r(x) = (r_1(x), \ldots, r_p(x))^T.$

The Lagrangian function of the above is represented as

(4.5) $L(x, \lambda, \mu) = (1/2)r(x)^T r(x) + \lambda^T g(x) + \mu^T h(x),$

and the Hessian matrix of this with respect to x is formed by

(4.6) $\nabla_{xx} L = J(x)^T J(x) + \sum_{j=1}^{p} r_j(x) \nabla^2 r_j(x) + \sum_{i=1}^{m} \lambda_i \nabla^2 g_i(x) + \sum_{j=1}^{l} \mu_j \nabla^2 h_j(x),$

where $J(x)$ is a Jacobian matrix of $r(x)$.

For Problem NLS, Takahashi et al.[17] try to combine the SQP method in Section 2 with the special purpose quasi-Newton method in Section 3, that is, they approximate the second term of the Hessian (4.6) by A_k, and the remainders by C_k, respectively. Then we have

(4.7) $B_k = J(x_k)^T J(x_k) + A_k + C_k.$

In the field of curve-fitting and parameter estimation, the sample size p is large but the number of variables is rather small, so it is not expensive, in practice, to store the matrices A_k and C_k. We can use directly the Dennis, Gay and Welsch update (3.14) for approximating the second term of (4.6). Noting that the matrix B_k must be positive definite for the global convergence of the SQP method, we may try to impose the requirement that the matrix A_k maintains the positive definiteness. So, we have an another update

(4.8) $A_{k+1}=\beta_k A_k + \tau_k\{(1+\beta_k \tau_k s_k^T A_k s_k)\nu_k \nu_k^T - \beta_k(A_k s_k \nu_k^T + \nu_k s_k^T A_k)\}$,

(4.9) $\nu_k = \psi_k v_k + (1-\psi_k)\beta_k A_k s_k$,

where v_k is defined by (3.9),

(4.10) $\tau_k = 0$ if $|s_k^T \nu_k| \leqq \rho$, ρ is a small positive constant,

(4.11) $\tau_k = 1/s_k^T \nu_k$, otherwise,

and

(4.12) $\psi_k = 1$ if $s_k^T v_k \geqq 0.2\ \beta_k s_k^T A_k s_k$,

(4.13) $\psi_k = 0.8\ \beta_k s_k^T A_k s_k / s_k^T(\beta_k A_k s_k - v_k)$, otherwise.

Further, for the matrix C_k, we obtain the secant condition

(4.14) $C_{k+1} s_k = u_k$,

where

(4.15) $u_k = [\nabla g(x_{k+1}) - \nabla g(x_k)]^T \lambda_{k+1} + [\nabla h(x_{k+1}) - \nabla h(x_k)]^T \mu_{k+1}$.

Various updating formulae which satisfy the above condition can be obtained. Considering the positive definiteness of the matrix B_k in (4.7), we propose Powell's modified BFGS and the modified DFP updates

(4.16) $C_{k+1} = C_k - C_k s_k s_k^T C_k / s_k^T C_k s_k + q_k q_k^T / s_k^T q_k$

and

(4.17) $C_{k+1} = C_k + (1 + s_k^T C_k s_k / s_k^T q_k) q_k q_k^T / s_k^T q_k - (C_k s_k q_k^T + q_k s_k^T C_k)/s_k^T q_k$,

respectively, where

(4.18) $q_k = \phi_k u_k + (1-\phi_k)\ C_k s_k$

and

(4.19) $\phi_k = 1$ if $s_k^T u_k \geqq 0.2\, s_k^T C_k s_k$,

(4.20) $\phi_k = 0.8\, s_k^T C_k s_k / s_k^T (C_k s_k - u_k)$, otherwise.

Now we present the SQP algorithm for solving Problem NLS:

(NLSSQP Algorithm)

Starting with an n-dimensional vector x_1, $n \times n$ symmetric matrices A_1, C_1, and three numbers $\delta > 0$, $\omega \in (0, 0.5)$ and $\tau \in (0, 1)$, the algorithm proceeds, for $k = 1,2,\ldots$, as follows:

Step 1. Having x_k, A_k and C_k, find the search direction d_k by solving the QP subproblem:

(QP subproblem)

(4.21) minimize $(1/2)d^T(J(x_k)^T J(x_k) + A_k + C_k)\, d + \nabla f(x_k)^T d$

(4.22) subject to $g(x_k) + \nabla g(x_k) d \leqq 0$,

(4.23) $h(x_k) + \nabla h(x_k) d = 0$,

and choose λ_{k+1} and μ_{k+1} to be the optimal multiplier vectors for this problem.

Step 2. If the vectors x_k, λ_{k+1} and μ_{k+1} satisfy the Kuhn-Tucker condition of Problem NLS, then stop; otherwise, go to Step 3.

Step 3 and 4 are the same procedures as SQP Algorithm.

Step 5. Update A_k and C_k giving A_{k+1} and C_{k+1} by the formulae (3.14), (4.16) (or (4.17)).

Though the two expressions relating to constraints in (4.6) are approximated by only one matrix C_k in (4.7), partition of C_k can be considered, that is,

(4.24) $C_k^1 \sim \sum_{i=1}^{m} \lambda_k^{(i)} \nabla^2 g_i(x_k)$ and $C_k^2 \sim \sum_{j=1}^{l} \mu_k^{(j)} \nabla^2 h_j(x_k)$, where $\lambda_k^{(i)}$ and $\mu_k^{(j)}$ denote the i-th element of λ_k and the j-th element of μ_k, respectively. In which case, a similar secant condition to (4.14) is obtained for C_{k+1}^1 and C_{k+1}^2, and we have some updating formulae. Further, considering the expression (4.6) and the linearity of functions, we have the following strategies:

(1)When all the model functions are linear (i.e., $r(x) = Mx - b$, where M is an $m \times n$ constant matrix and b is an m-dimensional constant vector), we set $A_k = 0$ at each step in NLSSQP Algorithm.

(2)When all the constraints are linear (i.e., $g_i(x) = a_i^T x - z_i$ and $h_j(x) = p_j^T x - w_j$, where a_i, z_i, p_j and w_j are constant vectors), we set $C_k = 0$ at each step in NLSSQP Algorithm.

(3)When all the model functions and constraints are linear (i.e., $r(x) = Mx - b$, $g_i(x) = a_i^T x - z_i$ and $h_j(x) = p_j^T x - w_j$), Problem NLS becomes the QP problem

(4.25) $\quad$ minimize $\quad (1/2)x^T(M^TM)\,x - b^TMx + (1/2)b^Tb$

(4.26) $\quad$ subject to $\quad a_i^T x - z_i \leq 0, \quad i = 1,\ldots, m,$

(4.27) $\quad\quad\quad\quad\quad\quad\quad p_j^T x - w_j = 0, \quad j = 1,\ldots, l.$

If the matrix M is of full rank, we can use Goldfarb and Idnani's QP method [15].

5. Numerical Experiments and Discussions

This section presents the results of some numerical experiments. The performance of NLSSQP Algorithm is compared with SQP Algorithm. The algorithms were coded in FORTRAN 77 and the double precision arithmetic was used. The convergence criteria used in all the algorithms are as follows:

$$\|\nabla_x L(x_k, \lambda_{k+1}, \mu_{k+1})\|_\infty \leqq \varepsilon,$$
$$g_i(x_k) \leqq \varepsilon, \quad i = 1, \ldots, m,$$
$$(5.1) \qquad |h_j(x_k)| \leqq \varepsilon, \quad j = 1, \ldots, l,$$
$$\lambda_{k+1}^{(i)} \geqq -\varepsilon, \quad i = 1, \ldots, m,$$
$$|\lambda_{k+1}^{(i)} g_i(x_k)| \leqq \varepsilon, \quad i = 1, \ldots, m,$$

where $\lambda_{k+1}^{(i)}$ denotes the i-th element of λ_{k+1} and ε is a tolerance.

For a comparison between SQP Algorithm and NLSSQP Algorithm, the run was made on an NEC PC-9801 personal computer. Both algorithms used Goldfarb and Idnani's QP method, and three parameters δ, ω and τ were set to 10, 0.1 and 0.5, respectively. In SQP Algorithm, Powell's modified BFGS update (2.21) with $B_1 = I$ and Procedure A were used. In NLSSQP Algorithm, the updating formulae (3.14), (4.16) and (4.17) with $A_1 = O$, $C_1 = I$ were used, and the Cholesky factorization of the matrix (4.7) was done. A total of five problems were solved. The starting points used and the optimal points can be found in Hock and Schittkowski[10]. The test problems are as follows:

Problem 1 (No.57); n = 2, m = 3, l = 0, p = 44.

Minimize $\sum_{j=1}^{44} \{ b_j - x_1 - (0.49 - x_1)\exp(-x_2(a_j - 8))\}^2$,

a_j and b_j (j=1,...,44) are given in [10],

subject to $x_1 x_2 - 0.49\, x_2 + 0.09 \leqq 0$, $0.4 \leqq x_1$, $-4 \leqq x_2$.

Problem 2 (No.65); n = 3, m = 7, l = 0, p = 3.

Minimize $(x_1 - x_2)^2 + (x_1 + x_2 - 10)^2/9 + (x_3 - 5)^2$

subject to $x_1^2 + x_2^2 + x_3^2 \leqq 48$,

$-4.5 \leqq x_i \leqq 4.5$ (i=1,2), $-5 \leqq x_3 \leqq 5$.

Problem 3 (No.70); $n = 4$, $m = 9$, $l = 0$, $p = 19$.

Minimize $\sum_{j=1}^{19} (F_1 + F_2 - F_j^{obs})^2$,

$$F_1 = (1+1/(12x_2))^{-1}[x_3 b^{x_2} (x_2/6.2832)^{0.5}(c_j/7.658)^{x_2 - 1} \exp(x_2 - bc_j x_2/7.658)],$$

$$F_2 = (1+1/(12x_1))^{-1}[(b/x_4)^{x_1} (x_1/6.2832)^{0.5}(c_j/7.658)^{x_1 - 1} (1 - x_3)\exp(x_1 - bc_j x_1/(7.658x_4))],$$

$b = x_3 + (1 - x_3)x_4$, c_j and F_j^{obs} are given in [10],

subject to $x_3 + (1 - x_3)x_4 \geqq 0$,

$0.00001 \leqq x_i \leqq 100$ $(i=1,2,4)$, $0.00001 \leqq x_3 \leqq 1$.

Problem 4 (No.79); $n = 5$, $m = 0$, $l = 3$, $p = 5$.

Minimize $(x_1 - 1)^2 + (x_1 - x_2)^2 + (x_2 - x_3)^2 + (x_3 - x_4)^4 + (x_4 - x_5)^4$

subject to $x_1 + x_2^2 + x_3^3 - 2 - 3(2)^{0.5} = 0$,

$x_2 - x_3^2 + x_4 + 2 - 2(2)^{0.5} = 0$, $x_1 x_5 = 2$.

Problem 5 (No.100); $n = 7$, $m = 4$, $l = 0$.

Minimize $(x_1 - 10)^2 + 5(x_2 - 12)^2 + x_3^4 + 3(x_4 - 11)^2 + 10x_5^6 + 7x_6^2 + x_7^4 - 4x_6x_7 - 10x_6 - 8x_7$

subject to $2x_1^2 + 3x_2^4 + x_3 + 4x_4^2 + 5x_5 \leqq 127$,

$7x_1 + 3x_2 + 10x_3^2 + x_4 - x_5 \leqq 282$,

$23x_1 + x_2^2 + 6x_6^2 - 8x_7 \leqq 196$,

$4x_1^2 + x_2^2 - 3x_1x_2 + 2x_3^2 + 5x_6 - 11x_7 \leqq 0$.

Problems 3 and 5 were solved using the values 10^{-3}, 10^{-4}, respectively, for the tolerance in the convergence criterion, and the others were solved using $\varepsilon = 10^{-6}$. Since Problem 3 (Himmelblau's problem) is printed incorrectly in [10], we referred directly to Himmelblau's book(Applied Nonlinear Programming,1972). Note that Problem 5(Wong's problem) can be reduced to the nonlinear least squares problem($p = 9$):

$$(5.2)\qquad f(x) = (x_1 - 10)^2 + 5(x_2 - 12)^2 + (x_3^2)^2 + 3(x_4 - 11)^2 + 10(x_5^3)^2 + 5(x_6 - 1)^2 + (x_7^2 - 2)^2 + 2(x_7 - x_6)^2 + 2(x_7 - 2)^2.$$

Table 1 displays the numerical results. These results suggest that NLSSQP Algorithm is very promising.

Table 1. Comparison between SQP Algorithm and NLSSQP Algorithm

	SQP with Update (2.21)			NLSSQP with (3.14) and (4.16)			NLSSQP with (3.14) and (4.17)		
Problem	IT	FE	QP	IT	FE	QP	IT	FE	QP
Problem 1	16	21	11	16	17	6	16	17	6
Problem 2	12	18	12	12	19	11	12	19	11
Problem 3	55	279	16	13	16	3	13	16	3
Problem 4	13	13	40	9	10	30	9	10	30
Problem 5	15	33	32	10	13	21	8	13	18

IT: the number of iterations.
FE: the number of objective function evaluations.
QP: the total number of inner iterations required in the QP algorithm.

Since we use the DGW update (3.14),the matrix B_k in (4.7) is not necessarily positive definite. In fact, for Problem 1, the positive definiteness of B_k was broken. When the positive definiteness was broken at t-th iteration, we tried to reset the matrix (4.7) as follows;

Reset 1: Set $B_t = J(x_t)^T J(x_t)$.

Reset 2: Set $B_t = J(x_t)^T J(x_t)$ and $A_t = 0$.

Reset 3: Set $B_t = J(x_t)^T J(x_t) + 10^{-2} I$.

Reset 4: Set $B_t = J(x_t)^T J(x_t) + 10^{-2} I$ and $A_t = 0$.

Reset 5: Set $B_t = J(x_t)^T J(x_t) + C_t$.

Reset 6: Set $B_t = J(x_t)^T J(x_t) + C_t$ and $A_t = 0$.

Table 2 displays the numerical results of NLSSQP Algorithm with (3.14) and (4.16) by using the above resetting rules. These results suggest that performance of the algorithm is sensitive to the choice of resetting rule. Note that the best result of Problem 1 in Table 2 is included in the appropriate column of Table 1.

Table 2. NLSSQP Algorithm with (3.14), (4.16) and Resetting

	Reset 1 or Reset 2			Reset 3 or Reset 4			Reset 5 or Reset 6		
Problem	IT	FE	QP	IT	FE	QP	IT	FE	QP
Problem 1	20	27	17	16	17	6	20	27	15

6. Nonlinear Optimization Code ASNOP

The increasing importance of nonlinear optimization arising in practical situations, e.g., engineering design, operations research and economics, requires the development of suitable optimization software. In recent years, a lot of effort has been made to implement efficient and reliable optimization programs. Schittkowski has introduced, in [14], 20 different optimization codes in 26 versions and tested them extensively from different points of view.

We have been developing the nonlinear optimization code ASNOP since 1982, at the Information Science Research Center of Aoyama Gakuin University. The numerical methods included in ASNOP are as follows;

(1)The quasi-Newton methods (e.g., BFGS method and DFP method) for general unconstrained optimization.

(2)Goldfarb and Idnani's method for quadratic programming.

(3)The augmented Lagrangian function method for general constrained optimization.

(4)The SQP method, described in [18], for general constrained nonlinear optimization.

(5)The NLSSQP method, described in this paper, for constrained nonlinear least squares problems.

The details can be found in [16] and [19].

REFERENCES

[1]M.C.Bartholomew-Biggs: The estimation of the Hessian matrix in nonlinear least squares problems with non-zero residuals, Mathematical Programming, Vol.12, pp.67-80 (1977).

[2]J.E.Dennis,Jr.: Some computational techniques for the nonlinear least squares peoblems, in "Numerical Solution of Systems of Nonlinear Algebraic Equations", G.D.Byrne and C.A.Hall (eds.), pp.157-183, Academic Press, New York (1973).

[3]J.E.Dennis,Jr. and R.B.Schnabel: Numerical Methods for Unconstrained Optimization and Nonlinear Equations, Prentice-Hall, New Jersey (1983).

[4]J.E.Dennis,Jr., D.M.Gay and R.E.Welsch: An adaptive nonlinear least-squares algorithm, ACM Transactions on Mathematical Software, Vol.7, pp.348-368 (1981).

[5]R.Fletcher: Practical Methods of Optimization Vol.2 - Constrained Optimization, John Wiley & Sons, Chichester (1981).

[6]P.E.Gill, W.Murray and M.A.Saunders: Methods for computing and modifying the LDV factors of a matrix, Mathematics of Computation, Vol.29, pp.1051-1077 (1975).

[7]D.Goldfarb and A.Idnani: A numerically stable dual method for solving strictly convex quadratic programs, Mathematical Programming, Vol.27, pp.1-33 (1983).

[8]S.P.Han: Superlinearly convergent variable metric algorithms for general nonlinear programming problems, Mathematical Programming, Vol.11, pp.263-282 (1976).

[9]S.P.Han: Variable metric methods for minimizing a class of non-differentiable functions, ibid., Vol.20, pp.1-13 (1981).

[10]W.Hock and K.Schittkowski:Test Examples for Nonlinear Programming Codes, Lecture Notes in Economics and Mathematical Systems, Vol.187, Springer-Verlag, Berlin (1981).

[11]L.Nazareth: Some recent approaches to solving large residual nonlinear least squares problems, SIAM Review, Vol.22,pp.1-11(1980).

[12]M.J.D.Powell:A fast algorithm for nonlinearly constrained optimization calculations, in "Numerical Analysis", G.A.Watson (ed.), Lecture Notes in Mathematics, Vol.630, pp.144-157, Springer-Verlag, Berlin (1978).

[13]M.J.D.Powell: On the quadratic programming algorithm of Goldfarb and Idnani, Mathematical Programming Study,Vol.25,pp.46-61(1985).

[14]K.Schittkowski: Nonlinear Programming Codes, Lecture Notes in Economics and Mathematical Systems, Vol.183, Springer-Verlag, Berlin (1980).

[15]J.Stoer: On the realization of the quadratic programming algorithm of Goldfarb and Idnani, in "Vistas in Applied Mathematics", A.V.Balakrishnan, A.A.Dorodnitsyn and J.L.Lions (eds.), pp.167-180, Optimization Software, Inc., New York (1986).

[16]S.Takahashi, S.Hongo, M.Miyata, N.Yamaki and H.Yabe:An appication system for nonlinear optimization problems (III), Aoyama Computer Science, Vol.13, No.1, pp.53-79, in Japanese (1985).
[17]S.Takahashi,N.Yamaki and H.Yabe: Some modifications of sequential quadratic programming method for constrained optimization, TRU Mathematics, Vol.23, pp.281-295 (1987).
[18]H.Yabe,S.Takahashi and N.Yamaki: Application of factorized quasi-Newton updates to successive quadratic programming method for constrained optization, Transactions of Information Processing Society of Japan, Vol.27, pp.1147-1154, in Japanese (1986).
[19]N.Yamaki,S.Hongo, M.Miyata, S.Takahashi and H.Yabe: An appication system for nonlinear optimization problems (IV), Aoyama Computer Science, Vol.14, No.2, pp.105-121, in Japanese (1986).
[20]H.Yamashita: Quadratic programming approximation for nonlinear optimization, in Proceedings of the 4-th Mathematical Programming Symposium Japan, pp.35-58 (1983).