# A Modified SQP Algorithm for Mathematical Programming

Vasile MORARU
*Technical University of Moldova*
*Bd. Stefan cel Mare, 168, Md-2004, Chisinau, Moldova*
*moraru@mail.utm.md*

***Abstract*—Recent efforts in mathematical programming have been focused a popular sequential quadratic programming (SQP) method. In this paper, a method for mathematical programs with equalities and inequalities constraints is presented, which solves two subproblems at each iterate, one a linear programming subproblem and the other is a quadratic programming (QP) subproblem. The considered method assures that the QP subproblem, is consistent.**

*Index Terms*—**constrained optimization, merit function, Sequential Quadratic Programming, super linear convergence.**

## I. INTRODUCTION

We consider the mathematical programming problem with general equality and inequality constraints

$$\left.\begin{array}{l} f(x) \to \min \\ \text{subject to} \\ h(x) = 0, \\ g(x) \le 0, \end{array}\right\} \qquad (1.1)$$

where the objective function $f : \Re^n \to \Re$, and the constraint functions $h : \Re^n \to \Re^m, g : \Re^n \to \Re^p$ are assumed to be twice continuously differentiable.

We briefly will describe the notation used in this paper. All vectors are column vectors. The subscript notation $x_i$ referees to an element of the vector *x*. A superscript *k* is used to denote iteration numbers. Superscript "*T*" denotes transposition. $\Re^n$ denotes the space of *n*-dimensional real column vectors.

We denote by $x^*$ a local solution of the problem (1.1). The Lagrangian function associated with the problem (1.1) is defined by

$$L(x, l, m) = f(x) + l^T h(x) + m^T g(x),$$

where $l \in \Re^m, m \in \Re^p$ are vectors of Lagrange multipliers. Assume that a *Linear Independence Constraint Qualification* (*LICQ*) condition holds at ; then multipliers $l^*$ and $m^* \ge 0$ exist such that [1]:

$$\left.\begin{array}{l} \nabla_x L(x^*, l^*, m^*) = 0, \\ (l^*)^T g(x^*) = 0, \\ h(x^*) = 0, \\ g(x^*) \le 0. \end{array}\right\} \qquad (1.2)$$

A primal-dual solution $(x^*, l^*, m^*)$ is said to be a Karush-Kuhn-Tucher (*KKT*) triple.

The basic idea of the typical sequential quadratic programming (*SQP*) is as follows [2]. Let the current *KKT* point be $(x^{(k)}, l^{(k)}, m^{(k)})$. A new approximation $(x^{(k+1)}, l^{(k+1)}, m^{(k+1)})$ to the solution is the procedure:

$$x^{(k+1)} = x^{(k)} + a_k d^{(k)},\ l^{(k+1)} = l^{QP},\ m^{(k+1)} = m^{QP},$$

where $d^{(k)}$ is a search direction which minimizes a quadratic model subject to the linearized constraints

$$\left.\begin{array}{l} \frac{1}{2} d^T B_k d + \left[\nabla f(x^{(k)})\right]^T d \to \min \\ h_i(x^{(k)}) + \left[\nabla h_i(x^{(k)})\right]^T d = 0, i = 1,2,\mathbf{K}, m, \\ g_i(x^{(k)}) + \left[\nabla g_i(x^{(k)})\right]^T d \le 0, i = 1,2,\mathbf{K}, p, \end{array}\right\} \qquad (1.3)$$

and $(l^{QP}, m^{QP})$ are taken as the Lagrange multipliers for (1.3). $a_k$ is the step size along the direction chosen to reduce the value of the merit function [3-5]:

$$F_{c_k}(x) = f(x) + c_k \max\{0, |h_1(x)|, \mathbf{K}, |h_m(x)|, g_1(x), \mathbf{K}\, g_p(x)\},$$

where $c_k > \sum_{j=1}^{m} |l_j^*| + \sum_{j=1}^{p} m_j^*$ is a penalty parameter.

The matrix $B_k$ is a symmetric approximation to the Hessian of the Lagrangian function [6, 7]:

$$B_k \approx \nabla_{xx}^2 L(x^{(k)}, l^{(k)}, m^{(k)}).$$

In traditional *SQP* method, the quadratic program (1.3) may be inconsistent; the feasible set of (1.3) may be empty. This is serious limitation of the *SQP* method. Several techniques for evitation of the inconsistency phenomen of the linearized constraints of the quadratic programming problem (1.3) were proposed [8-13]. Recently, in [11,12], modifications of the SQP method were proposed where at each step two subproblems are resolved: one linear programming problem or one linear square problem and one quadratic programming problem.

The presented method in this paper was announced in [14] and it is similar to the one given in [9] .At each iteration, two subproblems are solved – one is a linear programming; the other is a quadratic subproblem. Our algorithm is distinct from the one proposed in [9] in two important ways. Firstly, in both linear and quadratic programming problems, beside the inequality constraints of the problem (PNL), we also consider the equality ones. Secondly, at each iterate the linear programming subproblem is deferent from the one is [9]; we consider the local behavior of all constraints.

## II. THE MODIFIED SQP METHOD

We consider the following linear programming subproblem:

$$\left.\begin{array}{c} j\,(y,z)=\sum_{i=1}^{m} y_i+\sum_{i=1}^{p} z_i \rightarrow \min \\ -y_i \le h_i\left(x^{(k)}\right)+\left[\nabla h_i\left(x^{(k)}\right)\right]^T d \le y_i,\, i=1,\mathbf{K},m, \\ g_i\left(x^{(k)}\right)+\left[\nabla g_i\left(x^{(k)}\right)\right]^T d \le z_i,\, i=1,2,\mathbf{K},p, \\ y_i \ge 0, z_i \ge 0, \forall i. \end{array}\right\} \quad (2.1)$$

Let $\tilde{d}^{(k)}, \tilde{y}_i^{(k)}, \tilde{z}_i^{(k)}$ be the solution of (2.1). If $x^{(k)}$ is feasible, we have $\tilde{d}^{(k)}=0$. Now we consider the following modified quadratic programming (*MQP*) subproblem:

$$\left.\begin{array}{c} \frac{1}{2} d^T B_k d+\left[\nabla f\left(x^{(k)}\right)\right]^T d \rightarrow \min \\ -\tilde{y}_i^{(k)} \le h_i\left(x^{(k)}\right)+\left[\nabla h_i\left(x^{(k)}\right)\right]^T d \le \tilde{y}_i^{(k)},\, i=1,\mathbf{K},m \\ g_i\left(x^{(k)}\right)+\left[\nabla g_i\left(x^{(k)}\right)\right]^T d \le \tilde{z}_i^{(k)},\, i=1,2,\mathbf{K},p. \end{array}\right\} (2.2)$$

Notice that $\tilde{d}^{(k)}$ is feasible solution of (2.2) so, the feasible region of this subproblem is nonempty. Let $d^{(k)}$ be the solution of *MQP* (2.2). If matrix $B_k$ is positive definite, $d^{(k)}$ is unique and is a descendent direction of $F_{c_k}(x)$.

We now describe the proposed algorithm.

**Step 0**. *Given the initial approximate* $x^{(0)} \in \Re^n$, $l^{(0)} \in \Re^m, m^{(0)} \in \Re^p$ *a* $n \times n$ *symmetric positive definite matrix* $B_0$, *an initial penalty parameter* $c_0 > 0$ *and the scalars* $b \in \left(0,\frac{1}{2}\right), g \in (0,1)$; $k := 0$;

**Step1.** *Solve subproblem (*2.1*) to obtain* $\tilde{d}^{(k)}, \tilde{y}_i^{(k)}, \tilde{z}_i^{(k)}$. *If* $\tilde{d}^{(k)} = 0$ *and* $\exists i\ \tilde{y}_i^{(k)} > 0$ *or*, $\tilde{z}_i^{(k)} > 0$, *stop;*

**Step 2.** *Solve subproblem (*2.2*) to generate* $d^{(k)}$. *If* $d^{(k)} = 0$, *stop*;

**Step 3.** *Choose the penalty parameter* $c_k$ *such that*

$$c_k > \sum_{j=1}^{m}\left| l_j^{MPQ}\right| + \sum_{j=1}^{p} m_j^{MPQ} \ ;$$

**Step 4.** *Select the smallest positive integer s such that*

$$F_{c_k}\left(x^{(k)}+g^s d^{(k)}\right) \le F_{c_k}\left(x^{(k)}\right) - bg^s\left(d^{(k)}\right)^T B_k d^{(k)}.$$

*Let* $a_k = g^s$ *and*

$$x^{(k+1)} = x^{(k)} + a_k d^{(k)},\ l^{(k+1)} = l^{MQP},\ m^{(k+1)} = m^{MQP}\ ;$$

**Step 5.** *Choose a symmetric positive definite matrix* $B_{k+1}$. *Set* $k := k+1$. *Go to Step 1.*

The matrix $B_k$ can be calculated using the technique from [6]. This guaranty that they are positive definite and they approximate the Hessian matrix $\nabla_{xx}^2 L\left(x^{(*)}, l^{(*)}, m^{(*)}\right)$ on the "tangent" subspace of active constraints, and that the $\left\{x^{(k)}\right\}$ superlinear converge to $\left\{x^{(*)}\right\}$.

## III. ALGORITHM FOR QUADRATIC PROGRAMMING

The efficiency of proposed SQP algorithm depends on the efficiency of the algorithm of solving quadratic programming sub problems (2.2). There are a great number of algorithms of solving quadratic programming problems. A relative complete bibliography of these methods can be found in [15]. In case we have a quadratic programming problem with a great number of variables and constraints, it is necessary to effectuate a relatively great number of arithmetical operations to find the solution of the problem. In such situations, it is more convenient to solve a finite succession of quadratic programming problems without any restriction or with simple restrictions, instead of solving the considered problem. Such a method is presented in [16].

We consider the quadratic programming problem with the following form:

$$f(x) = \frac{1}{2} x^T H x + g^T x \rightarrow \min \qquad (3.1)$$

$$\text{subject to linear constraints} \quad Ax \le b,$$

where $H$ is a symmetric matrix , positive definite of the $n \times n$ dimension, $A$ is a $m \times n$ matrix; $g$, $x$ and $b$ are column vectors, $g, x \in \Re^n, b \in \Re^m$.

It is well known (see for example [2, 17]) that the optimal solution $x_*$ of the problem (1.1) is defined by the relation:

$$x^* = -H^{-1}\left(A^T l^* + g\right). \qquad (3.2)$$

The Lagrange multipliers vector

$$l_* = \left(l_*^1, l_*^2, \mathbf{K}\ l_*^m\right)^T$$

is the solution of the dual problem:

$$j\,(l) = \frac{1}{2} l^T D l + c^T l \quad \rightarrow \min \qquad (3.3)$$

$$\text{subject to} \ \ l^i \ge 0,\ i = 1,2,\mathbf{K}\, m,$$

where $D = AH^{-1}A^T, c = b + AH^{-1}g$.

If the inverse matrix $H^{-1}$ is well known, then the quadratic programming problem (3.1) would be equivalent to the problem (3.3) which has simple constraints. A method of solving the problem (3.1) is proposed in [16], in which the inversion of matrixes is avoided. The method results from joining the solutions of some systems of linear equations with the same matrix $H$ with the method of selecting active constraints

The method proposed for solving the quadratic programming problem (3.1) consists in the following:

**Step 1.** *The free minim point of the quadratic function x0 is determined:*

$$f(x) = \frac{1}{2} x^T H x + g^T x \qquad (3.4)$$

*The calculation of the minim $x^0$ comes to the solution of the system of linear equations: $Hx = -g$; however, some methods of unconstrained optimization, such as that of conjugate directions can be also utilized.*

**Step 2**.*The vector*

$$d = Ax^0 = \left[a_1^T x^0, a_2^T x^0, \ldots, a_m^T x^0\right]^T$$

*is determined by the point $x^0$, where $a_i^T$ is the line i of the matrix A. If $d \le b$ then $x^* = x^0$ is the most propitious solution and the problem (3.1) is solved, otherwise it is passed to the following step.*

**Step 3**.*We computing the free minim points $x_1, x_2, \ldots, x_m$ of the respective quadratic functions:*

$$f_i(x) = \frac{1}{2} x^T Hx - a_i^T x \ , i=1,2,\ldots,m \qquad (3.5)$$

*This may be achieved by solving the linear equation systems: $Hx^i = a_i$, i=1, 2,…, m with the same matrix H, or by applying other methods of unconstrained optimization.*

**Step 4**. *Using the solutions obtained above, at Step 3, we make the matrix $W = (w_{ij})$ with the dimensions mxm. This matrix has the elements $w_{ij} = a_i^T x^j, 1 \le i, j \le m$.*

**Step 5**. *The quadratic programming problem is solved with simple constraints:*

$$\min_l \left\{ j(l) = \frac{1}{2} l^T W l + (d+b)^T l \,\middle|\, l \ge 0 \right\} \qquad (3.6)$$

**Step 6**. *The optimum solution is found so:*

$$x^* = x^0 - \sum_{i=1}^{m} l_i^* x^i, \qquad (3.7)$$

*where $l^* = \left(l_1^*, l_2^*, \ldots, l_m^*\right)^T$ - is the optimum in (3.6)*

The validity of this algorithm is justified by the following theorems and lemmas.

***Lemma 3.1** The matrix W is symmetric and semi positive definite with the diagonal elements $w_{ii} > 0$, i=1,2,…,m. If $m \le n$ and the vectors $a_1, a_2, \ldots, a_m$ are linear independent then $\det(W) \ne 0$ (the inverse matrix $W^{-1}$ exists) and the matrix W is positive definite.*

***Proof:*** For any *i*, *j* is true:

$$w_{ij} = a_i^T x^j = \left(x^j\right)^T Hx^i = a_j^T H^{-1} Hx^i = a_j^T x^i = w_{ji},$$

and as follows $W^T = W$. If we note $X = \left[x^1, x^2, \ldots x^m\right]$- a matrix with its dimensions *mxm*, its columns are the vectors $x^1, x^2, \ldots x^m$. Then according to that was written above we can write: $W = X^T HX$. So $W$ is a positive semi definite matrix with $w_{ii} = \left(x^i\right)^T Hx^i > 0$, because the matrix $H$ is as being positive definite. If we have a system of linear independent vectors $\{a_i\}_i^m$ then the $rank(X) = m$ and as a consequence *rank(W)=m* and $W \succ 0$. The lemma has been proved.

***Lemma 3.2** The quadratic programming problems (3.3) and (3.6) are equivalent.*

***Proof:*** According to the proposed algorithm $HX = A^T$ and so $W = X^T HX = X^T A^T = AH^{-1}A^T$. We also notice that $d = AH^{-1}g$ (Step1), as a consequence we have $c = b + AH^{-1}g = b + d$. The lemma has been proved.

***Consequence:*** The quadratic programming problem (3.6) has a unique optimum solution.

We notice that $x_*$ that has been calculated using formula (3.7) is the same as that one calculated using formula (3.2). This justifies the proposed algorithm

## IV. CONCLUSION

In this paper, we have presented a SQP method which we solve two subproblems: one a linear programming and the other is a quadratic programming (QP) subproblem. The following main results are obtained. First, the proposed method can assure that the QP subproblem is consistent. Second, since solving a linear programming is very easy, so the method can be implemented without difficulty.

## REFERENCES


[1] Bertsekas D.P. Constrained optimization and Lagrange multiplier methods. Academic Press, 1982.

[2] Boggs P., Tolle J. Sequential quadratic programming. Acta Numerica, 4, 1995, pp. 1-51.

[3] Pschenichny B.N. Algorithms for general problems of mathematical programming. Kibernetica, **6**, 1970, pp.120-125.

[4] Pshenichniy B. N., Danilin Iu. M. *Numerical methods in extremal problems.* (In Russian) –M.: Nauka, 1975, 319 p.

[5] Han S. P. A globally convergent method for nonlinear programming. Journal of Optimization Theory and Applications 15, 1977, pp.319-342.

[6] Gould N., Toint Ph. SQP methods for large-scale nonlinear programming. In System modeling and optimization, methods, theory and applications (Powel M. and Scholtes S. editors), Kluwer, 2000, pp.149-178.

[7] Moraru V. Quasi-Newton methods for solving nonlinear programming problems. Computer Science Journal of Moldova, 3, No.3, 1995, pp.263-277.

[8] Schittkowski K. The nonlinear programming method of Wilson, Han and Powell with an augmented Lagrange type line search function. Part 1 : convergence analysis. Numer. Math. 38, 1981, pp. 83-114.

[9] Burke J. V., Han S. P. A robust SQP method. Mathematical Programming, 43, 1989, pp. 277-303.

[10] Zhang J., Zhang X. *A robust SQP method for optimization with inequality constraints.* Journal of Computational Mathematics, **21,** No. 2, 2003, pp. 247-256.

[11] Liu X., Yuan Y. A robust algorithm for optimization with general equality and inequality constraints. SIAM Journal on Scientific and Computing, **22,** No. 2, 2000, pp. 517-534.

[12] Wan Z. Global convergence of a modified SQP method for mathematical programs with inequalities and equalities constraints. Journal of Inequalities in Pure and Applied Mathematics, **5,** No.2, 2004, Art. 37.

[13] Byrd R.H., Gould N., Nocedal J., Waltz R. *On the convergence of successive linear-quadratic programming algorithms.* Mathematical Programming, **100,** No. 1, 2004, pp. 27-48.

[14] Moraru V. *A modified SQP method for optimization with equalities and inequalities constraints.* Proceedings of the 4th International Conference on “Microelectronics and Computer Science”, Volume II, September 15-17,2005, pp.275-278.

[15] Moraru V. (1997) *An Algorithm for Solving Quadratic Programming Problems.* Computer Science Journal of Moldova, vol.5, No. 2, pp.223-235.

[16] Moraru V. *Primal-Dual Method for Solving Convex Quadratic Programming Problems.* Computer Science Journal of Moldova, Volume 8, No.2, 2000, pp.209-220.

[17] Bazaraa M.C., Shetty C.M. (1979) *Nonlinear Programming. Theory and Algorithms.* John Wiley \& Sons, Inc.