◆◆ Scientific
◆◆ Research

Fixed-Point Iteration Method for Solving the Convex Quadratic Programming with Mixed Constraints

Ruopeng Wang, Hong Shi, Kai Ruan, Xiangyu Gao
Department of Mathematics & Physics, Beijing Institute of Petrochemical Technology, Beijing, China
Email: wangruopeng@bipt.edu.cn

ABSTRACT

The present paper is devoted to a novel smoothing function method for convex quadratic programming problem with mixed constrains, which has important application in mechanics and engineering science. The problem is reformulated as a system of non-smooth equations, and then a smoothing function for the system of non-smooth equations is proposed. The condition of convergences of this iteration algorithm is given. Theory analysis and primary numerical results illustrate that this method is feasible and effective.

KEYWORDS

1. Introduction

Consider the following convex quadratic programming problem

$$\min \frac{1}{2} x^{\mathrm{T}} Q x + c^{\mathrm{T}} x$$

$$\text{s.t.} \quad \begin{cases} A_{11} x_I + A_{12} x_{II} \geq b_I \\ A_{21} x_I + A_{22} x_{II} = b_{II} \end{cases} \tag{1}$$

where $Q = \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{bmatrix}$ is an $n \times n$ symmetrical positive semi-definite partitioned matrix, $A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}$ is

an $m \times n$ partitioned matrix, $b = \begin{bmatrix} b_I \\ b_{II} \end{bmatrix}$ and $c = \begin{bmatrix} c_I \\ c_{II} \end{bmatrix}$ are vectors in R^m and R^n, respectively. And Q_{ij},

$A_{ij} (i, j = 1, 2)$ are the partitioned matrix of Q and A respectively.

As is well known, this problem models a large spectrum of applications in computer science. Operation research and engineering as the nonlinear programming is solved eventually based on the convex quadratic programming problem. Therefore, the study on its effective solvers becomes a prolonged research subject.

Common effective algorithm is proposed related to certain schemes of convex quadratic programming problem, see [1-5]. Moreover, these methods are not polynomial algorithm in [6]. And the theory and algorithm of quadratic programming are presented completely. According to the optimization condition (KKT) and the duality theory of quadratic programming, the fixed-point iteration method is obtained for equality constrained (see [7]) and inequality constrained (see [8]) convex quadratic programming problem respectively. In fact, equality and inequality can be converted with inputting artificial variables and slack variables. The main default is to increase the scale and numerical difficulty while introduces these variables. Motivated by smoothing methods of

[9-11] and fixed-point method of [7,8], the paper mainly concerns about the mixed constraint quadratic programming and the fixed-point iteration method is given. And also the rank of the coefficient matrix is not full. The method considering is fulfilled efficiently. Thus the proposing question in [8] is solved.

2. Scheme of Algorithm

According to the dual theory, the dual problem of primal problem (1) is

$$\max -\frac{1}{2} x^T Q x + b^T y$$

$$\text{s.t.} \begin{cases} Q_{11} x_I + Q_{12} x_{II} - A_{11}^T y_I - A_{21}^T y_{II} + c_I \geq 0 \\ Q_{21} x_I + Q_{22} x_{II} - A_{21}^T y_I - A_{22}^T y_{II} + c_{II} = 0 \end{cases} \tag{2}$$

where $y = \begin{bmatrix} y_I \\ y_{II} \end{bmatrix} \in R^m$.

Let $u = \begin{bmatrix} u_I \\ u_{II} \end{bmatrix}, u_I = \begin{bmatrix} x_I \\ y_I \end{bmatrix}, u_{II} = \begin{bmatrix} x_{II} \\ y_{II} \end{bmatrix}$, we know if $x^* = \begin{bmatrix} x_I^* \\ x_{II}^* \end{bmatrix} \in R^n$ is the optimal solution of (1),there exists a

vector $y^* = \begin{bmatrix} y_I^* \\ y_{II}^* \end{bmatrix} \in R^m$, an $n \times n$ symmetrical positive semi-definite partitioned matrix, and (x^*, y^*) is the optimal solution of (2).

Let us define the following notation

$$F = \left\{ u = (x_I, y_I, x_{II}, y_{II}) \in R^{n+m} \middle| u_I \geq 0 \right\}$$

$F^* = \left\{ u^* = (x_I^*, y_I^*, x_{II}^*, y_{II}^*) \middle| (x_I^*, x_{II}^*) \text{ is the optimal solution of (1) and } (x_I^*, y_I^*, x_{II}^*, y_{II}^*) \text{ is the optimal solution of (2) } \right\}$.

It is easy to verify that F is a close set.

Suppose that the feasible region F^* is not empty. A necessary and sufficient condition for a optimal solution is that:

1) Primal feasibility

$$A_{11} x_I + A_{12} x_{II} - b_I \geq 0$$

$$A_{21} x_I + A_{22} x_{II} - b_{II} = 0$$

2) Dual feasibility

$$Q_{11} x_I + Q_{12} x_{II} - A_{11}^T y_I - A_{21}^T y_{II} + c_I \geq 0$$

$$Q_{21} x_I + Q_{22} x_{II} - A_{21}^T y_I - A_{22}^T y_{II} + c_{II} = 0$$

3) Complementary condition

$$x_I^T \left[Q_{11} x_I + Q_{12} x_{II} - A_{11}^T y_I - A_{21}^T y_{II} + c_I \right] = 0$$

$$y_I^T \left[A_{11} x_I + A_{12} x_{II} - b_I \right] = 0$$

The scheme are rewritten below

$$u_I \geq 0, (Mu + q)_I \geq 0, u^T (Mu + q)_I = 0 \ (Mu + q)_{II} = 0 \tag{3}$$

Where

$$M = \begin{bmatrix} Q_{11} & -A_{11}^T & Q_{12} & -A_{21}^T \\ A_{11} & 0 & A_{12} & 0 \\ Q_{21} & -A_{12}^T & Q_{22} & -A_{22}^T \\ A_{21} & 0 & A_{22} & 0 \end{bmatrix}$$

$$q = \begin{bmatrix} q_I \\ q_{II} \end{bmatrix}, q_I = \begin{bmatrix} c_I \\ -b_I \end{bmatrix}, q_{II} = \begin{bmatrix} c_{II} \\ -b_{II} \end{bmatrix}$$

More formally, the following link fixed-point iteration problem is performed

$$u = \max\left(0, u - r\left(Mu + q\right)\right) \tag{4}$$

The component form is

$$u_j = \max\left(0, u - r\left(M_j u_j + q_j\right)\right) \tag{5}$$

Where $r > 0$.

The question (5) is non-smooth optimization problem, and the Newton-type methods cannot be applied directly. In the paper, we converted the problem into a smooth optimization.

At the beginning, we introduce a function mapping R into R^+, and have the following properties.

Property 1 Let $\theta : R \to R^+$ be a real function, and such that:

1) $\theta(t)$ is strictly convex and differentiable;

2) $\left.\dfrac{d\theta}{dt}\right|_{t=0} > 0$;

3) $\lim\limits_{t \to -\infty} \theta(t) = 0$;

4) $\lim\limits_{t \to +\infty} \dfrac{\theta(t)}{t} = 1$.

We assumed that $\theta(t)$ is a convex and differentiable function, which indicate one can apply the classical Newton-type algorithm directly. And the assumption $\theta'(0) > 0$ means that 0 is not a sub-gradient of θ at 0. The hypothesis (3) and (4) indicate the approximate function $\theta(t)$ has the same numerical properties as the max-function.

We consider the following function, and $\theta(t)$ describe as above

$$\theta_p(t) = p\theta\left(\frac{t}{p}\right) \tag{6}$$

For any $p \in (0,1]$, $\theta_p(t)$ satisfies the following properties:

Theorem 1 Let be a function given as (6), then for any arbitrary $p \in (0,1], \theta_p(t)$ is strictly convex and differentiable, and we have $\left.\dfrac{d\theta_p(t)}{dt}\right|_{t=0} > 0$.

Proof According to property 1 and (6), the convexity and differentiability are straightforward.

Nothing that

$$\left.\frac{d\theta_p(t)}{dt}\right|_{t=0} = \theta'\left(\frac{t}{p}\right) \geq \theta'(t)$$

The property (2) implies $\left.\dfrac{d\theta_p(t)}{dt}\right|_{t=0} > 0$.

Theorem 2 Let $\theta_p(t) : R \to R^+$ be a function given as (6), then $\lim\limits_{p \to +0} \theta_p(t) = \max\{0, t\}$. That is, the function $\theta_p(t)$ uniformly converges to the max-function, when the adjustable parameter p approaches 0.

Proof For t fixed, we can compute

$$\lim\limits_{p \to +0} \theta_p(t) = \lim\limits_{p \to +0} p\theta\left(\frac{t}{p}\right)$$

If $t < 0$, then $\lim\limits_{p \to +0} p\theta\left(\dfrac{t}{p}\right) = 0$; otherwise, change variable $z \equiv \dfrac{t}{p}$,

$$\lim\limits_{p \to +0} \theta_p(t) = \lim\limits_{z \to +\infty} t \cdot \frac{\theta(z)}{x} = t$$

Hence, $\lim_{p \to +0} \theta_p(t) = \max\{0, t\}$ for all t .

From the discussion above, we introduce the following function in the paper

$$\theta(t) = \ln\left(1 + e^t\right)$$

and

$$\theta_p(t) = p\theta\left(\frac{t}{p}\right) = p\ln\left(1 + e^{\frac{t}{p}}\right)$$

It is easily known that satisfies the properties (1) to (4), together with theorem 1 and 2. Thus, the max-function in (3) replaced by $\theta_p(t) = p\ln\left(1 + e^{\frac{t}{p}}\right)$, and the fixed-point iteration which basing on smoothing function is obtained.

Algorithm
Step 1 Set a initial point $u^0 \in R^{m+n}$, choose $\varepsilon > 0$, $0 < p_0 \le 1$, $l \in (0,1)$, let $k = 0$;
Step 2 Apply fixed-point algorithms to solve

$$u_j^{k+1} = \theta_{i,p}\left(u_j^k - r_k\left(M_j u_j^k + q_j\right)\right)$$

$$j = 1, 2, \cdots, n+m \tag{7}$$

Step 3 Terminal condition: $\left|\left(u^k\right)^{\mathrm{T}}\left(M_j u^k + q\right)\right| < \varepsilon$, stop; otherwise, go to Step 4;

Step 4 Let $p_{k+1} = lp_k$, $k = k+1$, go to step 2.
The Jacobian matrix in (7) is shown

$$B = V\left(I - rM^{\mathrm{T}}\right) \tag{8}$$

Where

$$V = \begin{pmatrix} v_1 & & \\ & \ddots & \\ & & v_{n+m} \end{pmatrix}, \quad v_j = \frac{e^{u\frac{k}{j} - r_k\left(M_j u_j^k + q_j\right)}{p}}{1 + e^{u\frac{k}{j} - r_k\left(M_j u_j^k + q_j\right)}{p}}$$

3. Convergence Analyses

Now we discuss the convergence of the algorithm. To begin with, two simple lemmas are introduced that is the basis of our theoretical analysis. Since their proof can be found in reference [12], we here omit the proof due to space.

Lemma 1 [12] A necessary and sufficient condition for the fixed-point iteration method to be convergent is that the spectral radius of Jacobian matrix B satisfies $\rho(B) < 1$, where the spectral radius $\rho(B)$ be defined as $\rho(B) = \sqrt{\max \lambda_{B^{\mathrm{T}} B}}$, and denotes the eigenvalue of matrix $B^{\mathrm{T}} B$.

Lemma 2 [12] Suppose M is a matrix, whose eigenvalues can be ordered from the smallest to largest, *i.e.* $|\lambda_1| \le |\lambda_2| \le \cdots \le |\lambda_{n+m}|$, and V is diagonal matrix with nonnegative diagonal elements, then the maximum norm of the matrix VM 's eigenvalue such that $|\bar{\lambda}| \le \max_j\{|v_j|\}|\lambda_j|$.

Theorem 3 suppose $\lambda_1, \lambda_2, \cdots, \lambda_{n+m}$ be the eigenvalues of the Jacobian matrix M , if one of the following conditions is true, then we can select the proper parameter r_k makes the algorithm convergent:
(1) The real part of λ_j is positive, that is $\mathrm{Re}\,\lambda_j > 0$ $(j = 1, 2, \cdots, n+m)$;
(2) All of the eigenvalues $\lambda_j(j = 1, 2, \cdots, n+m)$ of matrix M are real.
Proof Suppose $\mu_j(j = 1, 2, \cdots, n+m)$ are eigenvalues of the Jacobian matrix in (8), and matrix M 's eigenvalues denote as $\lambda_j(j = 1, 2, \cdots, n+m)$.
Let $\lambda = x + y\mathrm{i}$ (where $\mathrm{i} = \sqrt{-1}$).

Note that

$$|1 - r\lambda| \leq 1 \Leftrightarrow r\left(x^2 + y^2\right) \leq 2x$$

If $\lambda = 0$, for all positive number of r; while $\lambda \neq 0$, we know $r \leq \dfrac{2x}{x^2 + y^2}$.

1) If $\lambda_j \left(j = 1, 2, \cdots, n + m\right)$ is positive, we let $r = \dfrac{2 \min_j \left\{\mathrm{Re}\,\lambda_j\right\}}{\max_j \left\{\mathrm{Re}\left|\lambda_j\right|^2\right\}}$, and have $\left|1 - r\lambda_j\right| \leq 1$

$\left(j = 1, 2, \cdots, n + m\right)$;

2) If all of the eigenvalues $\lambda_j \left(j = 1, 2, \cdots, n + m\right)$ of matrix M are real, we let $r = \dfrac{2}{\max_j \left\{\mathrm{Re}\left|\lambda_j\right|\right\}}$, and

have $\left|1 - r\lambda_j\right| \leq 1$ $\left(j = 1, 2, \cdots, n + m\right)$.

For all discussed above, associate with Lemma 1, we know that the matrix M has the eigenvalues of $\mu_j \left(j = 1, 2, \cdots, n + m\right)$, such that

$$\left|\mu_j\right| \leq \max_j \left\{\left|v_j\right|\right\}\left|1 - r\lambda_j\right| \leq \max_j \left\{\left|v_j\right|\right\} < 1$$

According to the theorem above, the algorithm is always convergent if the parameter r is chosen properly. Thus the recent relevant results in reference [8] are extended.

4. Numerical Experiment and Conclusion

In this section, we implement algorithm for solving system of inequalities in MATLAB7.0 in order to demonstrate the behavior of the algorithm. Due to page limit, we omit to test the effectiveness of our method in generating highly quadratic programming problems.

Example 1 Consider

$$\min z = x_1^2 + 2x_2^2 + x_3^2 - 2x_1 x_2 + x_3$$

$$\text{s.t.} \begin{cases} x_1 + x_2 + x_3 = 4 \\ 2x_1 - x_2 + x_3 = 2 \\ x_1 - 2x_2 - x_3 \leq 4 \\ x_1, x_2, x_3 \geq 0 \end{cases}$$

The exact optimal solution is $x^* = \left(\dfrac{21}{11}, \dfrac{43}{22}, \dfrac{3}{22}\right)^{\mathrm{T}}$, and the optimal value of objective function is $z^* = \dfrac{175}{44}$.

Example 2 Consider

$$\max z = x_1^2 + x_2^2 + x_3^2$$

$$\text{s.t.} \begin{cases} 3x_1 + 2x_2 + x_3 = 6 \\ 2x_1 - 4x_2 + x_3 \leq -8 \\ x_1, x_2, x_3 \geq 0 \end{cases}$$

The exact optimal solution is $x^* = \left(\dfrac{1}{2}, \dfrac{9}{4}, 0\right)^{\mathrm{T}}$, and the optimal value of objective function is $z^* = \dfrac{55}{4}$.

The results are shown in **Table 1**.

From the table, we can see that the algorithm is very simple and only applies the traditional fixed-point method. The implementation of this algorithm is rather easy.

Example 3 Consider

$$\min \frac{1}{2} x^{\mathrm{T}} Q x + c^{\mathrm{T}} x$$

Table 1. Experiments and calculated data.

Example	Optimal Solution	Objective Value
1	$\begin{pmatrix} 1.909091 \\ 1.954546 \\ 0.136364 \end{pmatrix}$	3.977273
2	$\begin{pmatrix} 0.500000 \\ 2.250000 \\ 0.000000 \end{pmatrix}$	13.750000

Table 2. Experiments and calculated data.

Iterations	Time	Objective Value
10	5.2s	0.00000

$$Ax \geq b$$

And

$$Q = \begin{pmatrix} 100.0429 & 0.9681 & 1.0672 & 1.4239 & 0.2477 & 0.7917 \\ 0.9681 & 100.0988 & 1.1956 & 0.7964 & 0.3543 & 0.8045 \\ 1.0672 & 1.1956 & 100.0919 & 0.7258 & 0.3490 & 1.2315 \\ 1.4239 & 0.7964 & 0.7258 & 101.7046 & 1.1653 & 0.8368 \\ 0.2477 & 0.3543 & 0.3490 & 1.1653 & 101.3110 & 1.1122 \\ 0.7917 & 0.8045 & 1.2315 & 0.8368 & 1.1122 & 100.3080 \end{pmatrix}$$

$$A = \begin{pmatrix} 6.3842 & 6.3349 & 9.7925 & 0.8606 & 6.8983 & 4.7373 \\ 5.1643 & 2.8327 & 0.6622 & 0.5958 & 1.4803 & 2.5038 \\ 1.1316 & 3.9409 & 7.2566 & 7.8306 & 1.1021 & 1.6560 \\ 9.2422 & 9.8371 & 7.7039 & 7.6555 & 6.6527 & 4.9634 \end{pmatrix}$$

$$c = \begin{pmatrix} 4.5647 & 0.1850 & 8.2141 & 4.4470 & 6.1543 & 7.9194 \end{pmatrix}^{\mathrm{T}}$$

$$b = \begin{pmatrix} -4.6091 & -3.6910 & -0.8813 & -2.0285 \end{pmatrix}^{\mathrm{T}}$$

The exact optimal solution is

$$x^* = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}^{\mathrm{T}},$$

and the optimal value of objective function is $z^* = 0$.

Appling the method proposed, the iterations consuming time are shown in **Table 2**.

Acknowledgements

REFERENCES

[1] D. Goldfarb and A. Idinabi, "A Numerical Stable Dual Method for Solving Strictly Convex Quadratic Programs," *Mathematical Programming*, Vol. 27, No. 2, 1983, pp. 1-33. http://dx.doi.org/10.1007/BF02591962

[2] Y. Ye and E. Tse, "An Extension of Kamarker's Algorithm to Convex Quadratic Programming," *Mathematical Programming*, Vol. 47, No. 4, 1989, pp. 157-179. http://dx.doi.org/10.1007/BF01587086

[3] J. L. Zhang and X. S. Zhang, "A Predictor-Corrector Method for Convex Quadratic Programming," *Journal of System Science and Mathematical Science*, Vol. 23, No. 3, 2003, pp. 353-366.

[4] R. D. Monteiro, I. Adler and M. G. Resende, "A Polynomial-Time Primal-Dual Affine Scaling Algorithm for Linear and Con-

vex Quadratic Programming and Its Powerseries Extension," *Mathematics of Operations Research*, Vol. 15, No. 2, 1990, pp. 191-214. http://dx.doi.org/10.1287/moor.15.2.191

[5]   M. W. Zhang and C. C. Huang, "A Primal-Dual Infeasible Interior Point Algorithm for Convex Quadratic Programming Problem with Box Constraints," *Journal of Engineering Mathematics*, Vol. 18, No. 2, 2001, pp. 85-90.

[6]   Y. X. Yuan and W. Y. Sun, "Optimization Theory and Method," Science Press, 2002, pp. 111-117.

[7]   R. P. Wang, "Fixed Iterative Method for Solving the Equality Constrained Convex Quadratic Programming Problem," *Journal of Beijing Institute of Petro-Chemical Technology*, Vol. 16, No. 3, 2008, pp. 64-66.

[8]   R. P. Wang, "Fixed Iterative Method for Solving the Inequality Constrained Quadratic Programming Problem," *Journal of Beijing Institute of Petro-Chemical Technology*, Vol. 15, No. 1, 2007, pp. 1-4.

[9]   X. S. Li, "An Effective Algorithm for Non-differentiable Problem," *Science in China* (*Series A*), Vol. 23, No. 3, 1994, pp. 353-366.

[10] G. Q. Chen and Y. Q. Chen, "An Entropy Function Method for Nonlinear Complementary Problems," *Acta Scientiarum Naturalium Universitatis NeiMongol*, Vol. 31, No. 5, 2000, pp. 447-451.

[11] H. W. Tang and L. W. Zhang, "An Entropy Function Method for Nonlinear Programming Problems," *Chinese Science Bulletin*, Vol. 39, No. 8, 1994, pp. 682-684.

[12] Q. Y. Li, Z. Z. Mo and L. Q. Qi, "Numerical Method for System of Nonlinear Equations," Beijing Science Press, Beijing, 1999.