Yokohama Publishers
ISSN 1349-8169 ONLINE JOURNAL
Pacific Journal of Optimization
Vol. 14, No. 2, pp. 369–380, 2018

AN ALTERNATING DIRECTIONS METHOD OF MULTIPLIERS FOR CONVEX QUADRATIC SECOND-ORDER CONE PROGRAMMING*

Xuewen Mu and Yaling Zhang

Abstract: An alternating directions method of multipliers for convex quadratic second-order cone programming is proposed. In the algorithm, at each iteration it first minimizes the dual augmented Lagrangian function with respect to the dual variables, and then with respect to the dual slack variables while keeping the other two variables fixed, and then finally it updates the Lagrange multipliers by adding a step size. Convergence result is given. Numerical results demonstrate that our method is efficient for the convex quadratic second-order cone programming.

Key words: *convex quadratic second-order cone programming, linear second-order cone programming , alternating directions method of multipliers, interior point method*

Mathematics Subject Classification: *90C25, 90C33, 90C53*

1 Introduction

In this paper, we consider a convex quadratic second-order cone programming (CQSOCP), which is defined by minimizing a convex quadratic function over the intersection of an affine set and the product of second-order cones. The primal convex quadratic second-order cone programming problem is defined as

$$\begin{array}{ll} \min & \frac{1}{2}x^TQx + c^Tx \\ \text{s.t.} & Ax = b \\ & x \in K, \end{array} \tag{1.1}$$

where Q is an $n \times n$ symmetric positive semidefinite matrix, $A \in R^{m\times n}$, $c \in R^n$,$b \in R^m$ and $x = [x_1, \cdots, x_N] \in R^{n_1} \times \cdots \times R^{n_N}$ is viewed as a column vector in $R^{n_1+\cdots+n_N}$ with $\sum\limits_{i=0}^{N} n_i = n$. In addition, $K = K^{n_1} \times K^{n_2} \times \cdots \times K^{n_N}$, and $x_i \in K^{n_i}$, where K^{n_i} is the n_i-dimensional second-order cone given by

$$K^{n_i} = \left\{ x_i = \left[\begin{array}{c} x_{i_1} \\ x_{i_0} \end{array} \right] \in R^{n_i-1} \times R : \|x_{i_1}\| \le x_{i_0} \right\}, \tag{1.2}$$

*This work was conducted while this author has been visiting Ohio University, Department of Mathematical. The work is supported by China Scholarship Council (CSC). This work was also supported by the National Science Foundations for Young Scientists of China (11101320, 61201297), and the Fundamental Research Funds for the Central Universities (K5051570013).

where $\|\cdot\|$ is the standard Euclidean norm.

The dual to primal CQSOCP (1.1) can be derived directly using Lagrangian method, which is described as

$$\begin{array}{ll} \max & b^T y - \frac{1}{2} x^T Q x \\ \text{s.t.} & A^T y + z - Qx = c \\ & z \in K, \end{array} \tag{1.3}$$

where $y \in R^n$ is the Lagrange multiplier.

Convex quadratic second-order cone programming is a nonlinear programming problem, which can be seen as a trust region subproblem or a convex quadratic subproblem in the trust region method and the successive quadratic programming method for the nonlinear second-order cone programming [24, 8]. Since Q is symmetric positive semidefinite, we can compute its positive semidefinite square root $Q^{\frac{1}{2}}$ by the singular value decomposition (SVD) method. Then the CQSOCP problem (1.1) can be equivalently transformed as the following linear second-order cone programming [25]

$$\begin{array}{ll} \min & t + c^T x \\ \text{s.t.} & Ax = b \\ & \sqrt{(t-1)^2 + 2\|Q^{\frac{1}{2}} x\|} \leq t + 1, \\ & x \in K \end{array} \tag{1.4}$$

In paper [24, 8], the authors use those well developed and publicly available softwares, based on interior point methods, such as SeDuMi [16] and SDPT3 [19] to solve the equivalent linear second-order cone programming (1.4).

For linear second-order cone problems, interior-point methods have been well developed [10, 1, 12]. However, since at each iteration these solvers require to formulate and solve a dense Schur complement matrix, which for the CQSOCP amounts to a linear system of dimension $(m+n+3) \times (m+n+3)$. In addition, the transformed method needs compute the square root of semidefinite matrix Q by the SVD method. When n is large, because of the very large size and ill-conditioning of the linear system of equations, interior point method are difficult to solve the transformed linear second-order cone programming problem [25].

The alternating direction method has been an effective first-order approach for solving large optimization problems, such as, linear programming [3], linear semidefinite programming (LSDP) [23, 11], linear second-order cone programming [13], nonlinear convex optimization [18], and nonsmooth l_1 minimization arising from compressive sensing [20, 22]. Paper [17] proposes a modified alternating direction method for convex quadratically constrained quadratic semidefinite programs. The method is a primal alternating direction method. In the thesis [25], a semismooth Newton-CG augmented Lagrangian method is proposed for large scale convex quadratic symmetric cone programming. The method is also a primal alternating direction method. In paper [21], an alternating direction dual augmented Lagrangian method for solving linear semidefinite programming problems in standard form is presented and is extended to the SDP with inequality constraints and positivity constraints.

In the paper, an alternating directions method of multipliers (ADMM) for the CQSOCP problems is proposed. Our algorithm applies the alternating direction method within a dual augmented Lagrangian framework. At each iteration, the algorithm minimizes the augmented Lagrangian function for the dual CQSOCP problem sequentially, first with respect to the dual variables corresponding to the linear constraints, and then with respect to the dual slack variables, while in each minimization keeping the other variables fixed, after which it updates the primal variables. Numerical experiments on, for example, random convex quadratic second-order cone programming problems, show that our method is efficient.

2 The Projection on the Second-order Cone

We make the following assumption throughout our presentation.
Assumption 1.1 The matrix A has full row rank and the strictly feasible primal and dual points of CQSOCP exist.

Based on the strong duality theorem for general conic programming problems [15], the KKT condition is given as

$$\begin{cases} Ax = b, \\ A^T y + z - Qx = c, \\ x^T z = 0, x, z \in K. \end{cases} \tag{2.1}$$

For the purpose of studying the metric projection operator over second-order cone, we need some knowledge about Euclidean Jordan algebras, which can be found from the standard references [5, 14, 9].

Let $x_i = \begin{bmatrix} x_{i_1} \\ x_{i_0} \end{bmatrix} \in R^{n_i-1} \times R$ for $i = 1, 2, \cdots, N$, then the spectral decomposition of x_i associate with second-order cone K^{n_i} can be described as [5, 14, 9].

$$x_i = \lambda_1(x_i)c_1(x_i) + \lambda_2(x_i)c_2(x_i), i = 1, 2, \cdots, N,$$

where

$$\lambda_1(x_i) = x_{i_0} - ||x_{i_1}||, \quad \lambda_2(x_i) = x_{i_0} + ||x_{i_1}||$$

and

$$c_1(x_i) = \tfrac{1}{2}\begin{bmatrix} -w \\ 1 \end{bmatrix}, \quad c_2(x_i) = \tfrac{1}{2}\begin{bmatrix} w \\ 1 \end{bmatrix}$$

with $w = \frac{-x_{i_1}}{||x_{i_1}||}$ if $x_{i_1} \neq 0$, and any vector in R^{n_i-1} satisfying $||w|| = 1$ if $x_{i_1} = 0$.

Next we introduce the projection lemma over the second-order cone [5, 14, 9].

Lemma 2.1 ([5, 14, 9]). *For any $x_i = \begin{bmatrix} x_{i_1} \\ x_{i_0} \end{bmatrix} \in R^{n_i-1} \times R$, let $P_{K^{n_i}}(x_i)$ be the projection of x_i onto the second-order cone K^{n_i}, then we have*

$$P_{K^{n_i}}(x_i) = (\lambda_1(x_i))_+ c_1(x_i) + (\lambda_2(x_i))_+ c_2(x_i), i = 1, 2, \cdots, N. \tag{2.2}$$

where $s_+ := max(0, s)$ for $s \in R$.

Let $x = [x_1, \cdots, x_N] \in R^{n_1} \times \cdots R^{n_N}$, then the projection $P_K(x)$ of x over the cone K is described as

$$P_K(x) = \left[P_{K_{n_1}}(x_1), \cdots, P_{K_{n_N}}(x_N)\right] \in R^{n_1} \times \cdots R^{n_N}. \tag{2.3}$$

Next we will give the following important conclusion.

Lemma 2.2 ([13]). *Assume $z \in R^n$, then we have*

$$z_1 = P_K(z), z_2 = -P_{-K}(z) \Longleftrightarrow z = z_1 - z_2, z_1^T z_2 = 0, z_1, z_2 \in K. \tag{2.4}$$

In fact, Lemma 2.2 is a special case of the Theorem 3.2.5 in [7].

$\boxed{3}$ An Alternating Directions Method of Multipliers for CQSOCP

In this section, we give an alternating directions method of multipliers for convex quadratic second-order cone programming.

The augmented Lagrangian function for the dual CQSOCP is defined as:

$$L_\lambda(x,y,z) = \frac{1}{2}x^TQx - b^Ty + x^T(A^Ty + z - Qx - c) + \frac{1}{2\lambda}\|A^Ty + z - Qx - c\|_2^2 \tag{3.1}$$

where $x \in R^n$ and $\lambda > 0$.

Suppose the initial point of x, z is x^0, z^0, at the k-th iteration, the alternating directions method of multipliers solves problem (3.1) by the following

$$y^{k+1} = arg \min_{y \in R^m} L_\lambda(x^k, y, z^k), \tag{3.2}$$

$$z^{k+1} = arg \min_{z \in R^n} L_\lambda(x^k, y^{k+1}, z), \quad s.t. \;\; z \in K, \tag{3.3}$$

$$x^{k+1} = x^k + \rho\frac{1}{\lambda}(A^Ty^{k+1} + z^{k+1} - Qx^k - c), \tag{3.4}$$

where $\rho \in (0, \frac{1+\sqrt{5}}{2})$.

The first-order optimality conditions for (3.2) are

$$\nabla_y L_\lambda(x^k, y^{k+1}, z^k) = Ax^k - b + \frac{1}{\lambda}A(A^Ty^{k+1} + z^k - Qx^k - c)$$

Since AA^T is invertible by Assumption 1, we obtain $y^{k+1} = y(z^k, x^k)$, where

$$y(z,x) = -(AA^T)^{-1}(\lambda(Ax - b) + A(z - c - Qx)). \tag{3.5}$$

For problem (3.3), it is easily verified that it is equivalent to

$$\min_{z \in R^n} \parallel z - v^{k+1} \parallel_2^2, z \in K, \tag{3.6}$$

where $v^{k+1} = v(z^k, x^k)$ and the function $v(z,x)$ is defined as

$$v(z,x) = Qx + c - A^Ty(z,x) - \lambda x. \tag{3.7}$$

Hence, we obtain the solution $z^{k+1} = v_1^{k+1} = P_K(v^{k+1})$.

In Eq. (3.4), the Lagrange multiplier is updated by adding the step size, which is an efficient method to improve the numerical performance of the alternating directions method of multipliers. The method has been used in many alternating direction methods [18, 21, 2, 6]. It follows from the updating Eq. (3.4) that

$$\begin{aligned} x^{k+1} &= x^k + \rho\frac{1}{\lambda}(A^Ty^{k+1} + z^{k+1} - Qx^k - c) \\ &= (1-\rho)x^k + \rho\left(x^k + \frac{1}{\lambda}(A^Ty^{k+1} + z^{k+1} - Qx^k - c)\right) \\ &= (1-\rho)x^k + \rho\frac{1}{\lambda}(z^{k+1} - v^{k+1}) \\ &= (1-\rho)x^k + \rho\frac{1}{\lambda}v_2^{k+1}. \end{aligned}$$

where $v_2^{k+1} = -P_{-K}(v^{k+1})$. Let $\bar{x}^{k+1} = \frac{1}{\lambda} v_2^{k+1}$, we have

$$x^{k+1} = (1-\rho)x^k + \rho\bar{x}^{k+1}. \tag{3.8}$$

From the above observation, we give the alternating directions method of multipliers as follows:

The alternating directions method of multipliers

Given $x^0 \in K, z^0 \in K$, and $\lambda > 0, \rho(0, \frac{1+\sqrt{5}}{2})$.

For $k = 0, 1, 2, \cdots$, then

Step 1. Compute $y^{k+1} = y(x^k, z^k)$ according to (3.5).

Step 2. Compute v^{k+1} and its projection, then set $z^{k+1} = P_K(v^{k+1})$.

Step 3. Compute $\bar{x}^{k+1} = \frac{1}{\lambda}(z^{k+1} - v^{k+1})$.

Step 4. Update the Lagrange multiplier by $x^{k+1} = (1-\rho)x^k + \rho\bar{x}^{k+1}$.

If $AA^T = I$, step (3.5) is very inexpensive. If $AA^T \neq I$, we can compute AA^T and its inverse (or its Cholesky factorization). If computing the Cholesky factorization of AA^T is very expensive, the iterative method in paper [2, 4] is used to solve the system of linear equations corresponding to (3.5).

4 The Convergence Result

Coupled with the convergence results of the alternating direction methods for variational inequalities in paper [6] and for the linear semidefinite programming in paper [21], we give the convergence analysis of the alternating direction method of multipliers for CQSOCP.

Similar to the conclusions in [21, 6], we obtain the following Lemma.

Lemma 4.1. *Let* (x^*, y^*, z^*) *be a primal and dual optimal solution of* (1.1) *and* (1.3), $\rho \in (0, \frac{1+\sqrt{5}}{2})$ *and* $T = 2 - \frac{1}{2}(1+\rho-\rho^2)$. *Then we have*

$$\begin{aligned}
&\|x^{k+1} - x^*\|^2 + \frac{\rho}{\lambda^2}\|z^{k+1} - z^*\|^2 + \frac{\rho(T-\rho)}{\lambda^2}\|A^T y^{k+1} + z^{k+1} - Qx^k - c\|^2 \\
&\leq \|x^k - x^*\|^2 + \frac{\rho}{\lambda^2}\|z^k - z^*\|^2 + \frac{\rho(T-\rho)}{\lambda^2}\|A^T y^k + z^k - Qx^{k-1} - c\|^2 \\
&- \frac{\rho(1+\rho-\rho^2)}{3\lambda^2}\left(\|z^{k+1} - z^k\|^2 + \|A^T y^{k+1} + z^{k+1} - Qx^k - c\|^2\right)
\end{aligned}$$

Hence, we obtain

$$\lim_{k\to\infty}\left(\|z^{k+1} - z^k\|^2 + \|A^T y^{k+1} + z^{k+1} - Qx^k - c\|^2\right) = 0. \tag{4.1}$$

To prove Lemma 4.1, we can use the similar analysis as in paper [6]. Now, we give the convergence result.

Theorem 4.2. *The sequence* $\{(x^k, y^k, z^k)\}$ *generated by the alternating directions method of multipliers converges to a primal and dual optimal solution* (x^*, y^*, z^*) *of* (1.1) *and* (1.3).

Proof. From (3.8), we have

$$\|x^{k+1} - \bar{x}^{k+1}\| = |\rho - 1| \cdot \|\bar{x}^{k+1} - x^k\| = \frac{|\rho-1|}{\lambda}\|A^T y^{k+1} + z^{k+1} - Qx^k - c\|. \tag{4.2}$$

It follows from From (3.5) and (3.7) that

$$\|A\bar{x}^{k+1} - b)\| = \|\frac{1}{\lambda}A(z^{k+1} - v^{k+1}) - b\|$$

$$
\begin{aligned}
&\le \|\frac{1}{\lambda}A(z^{k+1}-Qx^k-c)+\frac{1}{\lambda}AA^Ty(x^k,z^k)+Ax^k-b\| \\
&\le \|\frac{1}{\lambda}A(z^{k+1}-Qx^k-c)-(Ax^k-b)-\frac{1}{\lambda}A(z^k-c-Qx^k)+Ax^k-b\| \\
&\le \frac{1}{\lambda}\|A(z^{k+1}-z^k)\|. \qquad (4.3)
\end{aligned}
$$

From (3.8) and (4.3), we have

$$
\begin{aligned}
\|Ax^{k+1}-b\| &= \|Ax^{k+1}-A\bar{x}^{k+1}+A\bar{x}^{k+1}-b)\| \\
&\le \|Ax^{k+1}-A\bar{x}^{k+1}\|+\|A\bar{x}^{k+1}-b)\| \\
&\le \|A\|_2\|x^{k+1}-\bar{x}^{k+1}\|+\tfrac{1}{\lambda}\|A\|_2\|z^{k+1}-z^k\|.
\end{aligned} \qquad (4.4)
$$

We obtain from (3.8) that

$$
\begin{aligned}
\|\langle x^{k+1},z^{k+1}\rangle\| &= |1-\rho|\cdot\|\langle x^k,z^{k+1}\rangle\| \\
&\le |1-\rho|\cdot\big(\|\langle x^k-\bar{x}^{k+1},z^{k+1}\rangle\|+\|\langle\bar{x}^{k+1},z^{k+1}\rangle\|\big) \\
&= |1-\rho|\cdot\|\langle x^k-\bar{x}^{k+1},z^{k+1}\rangle\| \\
&\le |1-\rho|\cdot\|x^k-\bar{x}^{k+1}\|\cdot\|z^{k+1}\|. \qquad (4.5)
\end{aligned}
$$

In addition, we have

$$
\begin{aligned}
&\|A^Ty^{k+1}+z^{k+1}-Qx^{k+1}-c\| \\
&= \|A^Ty^{k+1}+z^{k+1}-Qx^k-c+Q(x^k-x^{k+1})\| \\
&\le \|A^Ty^{k+1}+z^{k+1}-Qx^k-c\|+\|Q(x^k-x^{k+1})\| \\
&\le \|A^Ty^{k+1}+z^{k+1}-Qx^k-c\|+\|Q\|_2\|(x^k-x^{k+1})\|. \qquad (4.6)
\end{aligned}
$$

It follows from (3.4), (4.1) and (4.2) that

$$
\begin{aligned}
\lim_{k\to\infty}\|x^{k+1}-x^k\| &= 0, \\
\lim_{k\to\infty}\|x^{k+1}-\bar{x}^{k+1}\| &= 0, \\
\lim_{k\to\infty}\|\bar{x}^{k+1}-x^k\| &= 0. \qquad (4.7)
\end{aligned}
$$

Combining (4.3)-(4.7), we obtain

$$
\begin{aligned}
\lim_{k\to\infty}\|Ax^{k+1}-b\| &= 0, \\
\lim_{k\to\infty}\|A^Ty^{k+1}+z^{k+1}-Qx^{k+1}-c\| &= 0, \\
\lim_{k\to\infty}\|\langle x^{k+1},z^{k+1}\rangle\| &= 0.
\end{aligned}
$$

We obtain from (4.7) that $x^{k+1}\in K$, and it is obvious that $z^{k+1}\in K$. So we know that $\{(x^k,y^k,z^k)\}$ convergent to a primal and dual optimal solution (x^*,y^*,z^*) of (1.1) and (1.3). □

5 Simulation Experiments

In this section we present computational results by comparing the alternating directions method of multipliers with the interior point method. The interior point method is used to solve the transformed linear second-order cone programming problems (1.4). All the algorithms are run in the MATLAB 7.0 environment on an Inter Core processor 1.80 GHz personal computer with 2.00 GB of Ram.

The test problems are formulated by random method as follows:

Step 1. Given the values of $n, m, N, n_i, i = 1, 2, \cdots, N$ with $\sum_{i=0}^{N} n_i = n$.

Step 2. Generate a random matrix $\tilde{Q} \in R^{n\times n}$ with $density = 0.2$, and set $Q = \tilde{Q}^T\tilde{Q}$. At the same time, generate a random matrix $A \in R^{m\times n}$ with full row rank.

Step 3. Given $x = [x_1, x_2, \cdots, x_N] \in R^{n_1} \times \cdots \times R^{n_N}$, then generate the random vector x_i and make it be an interior point of second-order cone K_{n_i} for $i = 1, 2, \cdots, N$.

Step 4. Generate a random vector $y \in R^m$, and generate z and make it be an interior point of second-order cone K.

Step 5. We obtain b and c by computing $b = Ax$ and $c = A^Ty + z - Qx$, respectively.

From Step 3-5, we know that the test problems have strick primal and dual feasible points, which is a sufficient condition for the optimal condition (2.1).

The first set of test problems includes 16 small scale CQSOCP problems, which is shown in Table 1. In Table 1,3, an entry of the form "20×5" in the "SOC" column means that there are 20 5-dimensional second-order cones, and the "ratio" denotes the ratio between the number of the second-order cones and the value of n.

Table 1: The test problems with small scale

Problems	m	n	SOC	ratio
P01	40	100	1×100	1.00%
P02	40	100	$1 \times 40; 20 \times 3$	21.00%
P03	40	100	20×5	20.00%
P04	40	100	$1 \times 4; 32 \times 3$	33.00%
P05	120	200	1×200	0.50%
P06	120	200	$1 \times 100; 1 \times 4; 32 \times 3$	17.00%
P07	120	200	40×5	20.00%
P08	120	200	$1 \times 5; 65 \times 3$	33.00%
P09	200	400	1×400	0.25%
P10	200	400	$1 \times 200; 1 \times 5; 65 \times 3$	16.75%
P11	200	400	80×5	20.00%
P12	200	400	$1 \times 4; 132 \times 3$	33.25%
P13	300	600	1×600	0.16%
P14	300	600	$1 \times 400; 1 \times 5; 65 \times 3$	11.16%
P15	300	600	120×5	20.00%
P16	300	600	200×3	33.33%

As is known to all, the interior point methods have been proved to be one of the most efficient class of methods for SOCP. Here the Matlab program codes for the interior point method is designed from the software package by SeDuMi [16]. In the SeDuMi software, we set the desired accuracy parameter $pars.eps = 10^{-6}$.

In the alternating directions method of multiplier, we define

$$pinf = \frac{\|Ax-b\|}{1+\|b\|}, dinf = \frac{\|Qx+c-z-A^Ty\|}{1+\|c\|}, gap = \frac{|x^TQx+b^Ty-c^Tx|}{1+|b^Ty|+|c^Tx|}.$$

We stop our algorithm when

$$Accuracy = max\{pinf, dinf, gap\} \leq \epsilon$$

for $\epsilon > 0$. Here we set $\lambda = 0.6, \rho = 0.4, \epsilon = 10^{-6}$. We choose the initial point $x^0 = 0_m, y^0 = 0_n$, where 0_n and 0_m are n dimension and m dimension vectors whose elements are zeros, respectively.

For the first set of test problems, the iteration number and average CPU time are used to evaluate the performances of the dual alternating direction method and the interior point method by SeDuMi. The test results are shown in Table 2. In the Table 2,4, "Time" represents the average CPU time (in seconds), and "Iter." denotes the average number of iteration. In addition, "ADMM" represents the alternating directions method of multipliers. In Table 4, "/ " denotes that the method doesn't work in our personal computer because the method is "out of memory".

Table 2: The results for the test problems with small scale

Problems	ADMM		SeDuMi	
	Iter.	Time	Iter.	Time
P01	40	0.0625	10	0.2812
P02	78	0.1093	12	0.3437
P03	82	0.1406	14	0.3593
P04	91	0.2031	14	0.4375
P05	58	0.1875	10	0.7656
P06	92	0.4375	15	1.2968
P07	104	0.5000	16	1.3593
P08	119	0.7343	17	1.4062
P09	59	0.7968	13	4.8906
P10	104	1.3593	16	6.3906
P11	119	1.5468	17	6.6093
P12	142	2.2500	18	7.0000
P13	56	1.7812	13	13.0312
P14	103	3.0468	18	18.4531
P15	129	3.1406	18	20.9062
P16	162	4.9062	19	21.6250

Table 2 shows the alternating directions method of multiplier costs less CPU time than the interior point method by SeDuMi. But, the iteration number of the interior point method is less than that of the alternating direction method of multipliers.

In addition, Table 1 gives different kinds test problem, including the problems with only one large second-order cone, such as P01,P05,P09,P13, the problems with many small second-order cones, such as P04,P08,P12,P16, and the problems with one large second-order cone and some small second-order cones, such as P02,P06,P10,P14. The test results in Table 2 show that the dual alternating direction method can solve different kinds of convex quadratic second-order cone programming with appropriate iteration number and CPU time.

The second set of test problems includes 15 medium scale problems, which is shown in Table 3. For the second set of test problems, the test results are shown in Table 4. In addition, for the second set of test problems, we set $\lambda = 0.8$, and two different parameters $\rho = 0.05$ and $\rho = 1.25$. Then, we test the performances of two different parameters.

Table 3: The test problems with medium scale

Problems	m	n	SOC	ratio
P21	400	1000	1×1000	0.10%
P22	400	1000	$1 \times 200; 160 \times 5$	16.10%
P23	400	1000	$1 \times 4; 332 \times 3$	33.30%
P24	600	2000	1×2000	0.05%
P25	600	2000	$1 \times 1000; 1 \times 4; 332 \times 3$	16.70%
P26	600	2000	$1 \times 5; 665 \times 3$	33.33%
P27	800	3000	1×3000	0.03%
P28	800	3000	600×5	20.00%
P29	800	3000	1000×3	33.33%
P30	1000	4000	1×4000	0.025%
P31	1000	4000	$1 \times 1000; 1000 \times 3$	25.02%
P32	1000	4000	$1 \times 4; 1332 \times 3$	33.32%
P33	2000	5000	1×5000	0.02%
P34	2000	5000	$1 \times 2000; 1000 \times 3$	20.02%
P35	2000	5000	$1 \times 5; 1665 \times 3$	33.32%

Table 4: The results for the test problems with medium scale

Problems	ADMM($\rho = 0.05$)		ADMM($\rho = 1.25$)		SeDuMi	
	Iter.	Time	Iter.	Time	Iter.	Time
P21	280	14.0156	17	1.4531	15	68.6865
P22	310	16.9531	57	3.7500	19	88.4983
P23	407	23.1875	94	6.3906	20	118.2052
P24	273	56.8125	17	6.7656	20	566.2341
P25	331	70.3125	47	12.6562	20	672.5000
P26	404	91.8906	86	21.4218	21	729.1400
P27	274	122.4843	23	18.0781	19	1791.6900
P28	347	162.5781	74	35.2500	22	2174.4545
P29	389	183.8593	78	39.8906	23	2517.3500
P30	272	208.9062	17	24.3125	/	/
P31	323	254.4218	61	53.0156	/	/
P32	395	306.0937	81	66.4531	/	/
P33	279	379.7968	21	49.9218	/	/
P34	340	437.7343	55	83.7343	/	/
P35	364	490.2812	110	144.6718	/	/

The results in Table 4 show the interior point method by SeDuMi doesn't work because of " out of memory " in our personal computer when $n+m > 5000$, but the alternating directions method of multipliers is still efficient because the alternating directions method of multipliers needs less memory space than the interior point method. In addition, the results in Table 4 show that the parameter ρ is much important for alternating directions method of multipliers, and the different parameters result in different simulation results. The alternating directions method of multipliers with $\rho = 1.25$ costs less CPU time than the method with $\rho = 0.05$. We also conclude the iteration number of the method with $\rho = 1.25$ is less than that of the method with $\rho = 0.05$.

6 Conclusion

The paper proposes the alternating directions method of multipliers for convex quadratic second-order cone programming problems. The alternating directions method of multipliers is a first-order method. The random simulation results show that our method is efficient for some convex quadratic second-order cone programming problems.

References

[1] F. Alizadeh and D. Goldfarb, Second-order cone programming, *Math. Program.* 95 (2003) 3–51.

[2] G. Chen and M. Teboulle, A proximal-based decomposition method for convex minimization problems, *Math. Program.* 64 (1994) 81–101.

[3] J. Eckstein, and D.P. Bertsekas, An alternating direction method for linear programming, LIDS-P, Cambridge, MA, Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, 1967.

[4] J. Eckstein and D.P. Bertsekas, On the Douglas–Rachford splitting method and the proximal point algorithm for maximal monotone operators, *Math. Program.* 55 (1992) 293–318.

[5] U. Faraut and A. Koranyi, *Analysis on Symmetric Cone*, Oxford University Press: New York, 1994.

[6] B.S. He, H. Yang and S.L. Wang, Alternating direction method with self-adaptive penalty parameters for monotone variational inequalities, *J. Optim. Theory Appl.* 106 (2000) 337–356.

[7] J.-B. Hiriart-Urruty and C. Lemarȇchal, Convex analysis and minimization algorithms. I. Fundamentals. in *Grundlehren der Mathematischen Wissenschaften*, Fundamental Principles of Mathematical Sciences, vol. 305. Springer: Berlin, 1993.

[8] H. Kato and M. Fukushima, An SQP-type algorithm for nonlinear second-order cone programs, *Optimization Letters* 1 (2007) 129–144.

[9] Lingchen Kong, L. Tuncel, N.H. Xiu, Clarke generalized Jacobian of the projection onto symmetric cones, *Set-Valued Var. Anal.* 17 (2009) 135–151.

[10] M.S. Lobo, L. Vandenberghe, S. Boyd and H. Lebret, Application of Second Order Cone Programming, *Linear Algebra Appl.* 284 (1998) 193–228.

[11] J. Malick, J. Povh, F. Rendl and A. Wiegele, Regularization methods for semidefinite programming, *SIAM J Optim.* 20 (2009) 336–356.

[12] R.D.C. Monteiro, T. Tsuchiya, Polynomial convergence of primal-dual algorithms for the second-order cone program based on the MZ-family of directions, *Math. Program.* 88 (2000) 61–83.

[13] X. Mu, and Y. Zhang, An alternating direction method for second-order conic programming, *Comp. Oper. Res.* 40 (2013) 1752–1757.

[14] J.V. Outrata and D.F. Sun, On the coderivative of the projection operator onto the second-order cone, *Set-Valued Var. Analy.* 6 (2008) 999–1014.

[15] A. Ruszczynski, *Nonlinear Optimization, Princeton*, Princeton University Press: New Jersey, 2006.

[16] J.F. Sturm, Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones, *Optim. Methods Software* 11-12 (1999) 625–653.

[17] J. Sun and S. Zhang, A modified alternating direction method for convex quadratically constrained quadratic semidefinite programs, *European J. Oper. Res.* 207 (2010) 1210–1220.

[18] P. Tseng, Alternating projection-proximal methods for convex programming and variational inequalities, *SIAM J. Optim.* 7 (1997) 951-965.

[19] R.H. Tütüncü, K.C. Toh and M.J. Todd, Solving semidefinite-quadratic-linear programs using SDPT3, *Math. Program.* 95 (2003) 189–217.

[20] Y. Wang, J. Yang, W. Yin and Y. Zhang, A new alternating minimization algorithm for total variation image reconstruction, *SIAM J. Imaging Sci.* 1 (2008) 248–272.

[21] Z. Wen, D. Goldfarb and W. Yin, Alternating direction augmented Lagrangian methods for semidefinite programming, *Math. Prog. Comp.* 2 (2010) 203–230.

[22] J. Yang, Y. Zhang and W. Yin, An efficient tvl1 algorithm for deblurring multichannel images corrupted by impulsive noise, *SIAM J. Sci. Comput.* 31 (2008) 2842–2865.

[23] Z. Yu, Solving semidefinite programming problems via alternating direction methods, *J. Comput. Appl. Math.* 193 (2006) 437–445.

[24] X. Zhang, Z. Liu and S. Liu, A trust region SQP-filter method for nonlinear second-order cone programming, *Comp. Math. Appl.* 63 (2012) 1569–1576.

[25] X.Y. Zhao, A semismooth Newton-CG augmented Lagrangian method for large scale linear and convex quadratic SDPs. Ph.D. Thesis, National University of Singapore, 2009.

Manuscript received
revised
accepted for publication

Xuewen Mu
School of Mathematics and Statistics
Xidian University, Xi'an, 710071, China
E-mail address: xdmuxuewen@hotmail.com; xwmu@xidian.edu.cn

Yaling Zhang
School of Mathematics and Statistics
Xidian University, Xi'an 710071, China;

School of Computer Science, Xi'an Science and Technology University
Xi'an 710054, China
E-mail address: zyldella@xust.edu.cn