Hindawi
Mathematical Problems in Engineering
Volume 2018, Article ID 5358191, 9 pages
https://doi.org/10.1155/2018/5358191

Research Article

A Simple Alternating Direction Method for the Conic Trust Region Subproblem

Honglan Zhu [1,2] and Qin Ni [1]

[1] *Department of Mathematics, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China*
[2] *Business School, Huaiyin Institute of Technology, Huaian 223003, China*

Correspondence should be addressed to Qin Ni; niqfs@nuaa.edu.cn

Received 23 September 2018; Accepted 27 November 2018; Published 18 December 2018

Academic Editor: Fazal M. Mahomed

A simple alternating direction method is used to solve the conic trust region subproblem of unconstrained optimization. By use of the new method, the subproblem is solved by two steps in a descent direction and its orthogonal direction, the original conic trust domain subproblem into a one-dimensional subproblem and a low-dimensional quadratic model subproblem, both of which are very easy to solve. Then the global convergence of the method under some reasonable conditions is established. Numerical experiment shows that the new method seems simple and effective.

1. Introduction

In this paper, we consider the unconstrained optimization problem

$$\min_{x\in R^n} \quad f(x), \tag{1}$$

where $f(x)$ is continuously differentiable. The trust region method is a very effective method for the unconstrained optimization problem (1) (see [1–6]). Traditional trust region methods are based on a quadratic model and the corresponding quadratic program subproblem is, at the kth iteration,

$$\min_{s\in R^n} \quad \varrho_k(s) = g_k^T s + \frac{1}{2} s^T B_k s, \tag{2}$$

$$\text{s.t.} \quad \|s\| \le \Delta_k, \tag{3}$$

where x_k is the current iterate point, $g_k = \nabla f(x_k)$, B_k is symmetric and an approximation to the Hessian of $f(x)$, $\|\cdot\|$ refers to the Euclidean norm, and Δ_k is the trust region radius at the kth iteration. There are many methods that can be used to solve the subproblems (2)-(3). The simple, low cost, and effective methods are dogleg methods (see [7–11]). Now, we recall the simple dogleg algorithm for solving trust region subproblem with the quadratic model as in [12].

Algorithm 1.

Step 0. Input the data of the kth iteration i.e., g_k, B_k and Δ_k.

Step 1. Compute $s_k^N = -B_k^{-1} g_k$. If $\|s_k^N\| \le \Delta_k$, then $s_* = s_k^N$, and stop.

Step 2. Compute $s_k^c = -(g_k^T g_k / g_k^T B_k g_k) g_k$. If $\|s_k^c\| \ge \Delta_k$, then $s_* = -\Delta_k g_k / \|g_k\|$, and stop. Otherwise, go to Step 3.

Step 3. Compute

$$\begin{aligned} d &= \left\|s_k^N - s_k^c\right\|^2, \\ e &= \left(s_k^N - s_k^c\right)^T s_k^c, \\ f &= \left\|s_k^c\right\|^2 - \Delta_k^2. \end{aligned} \tag{4}$$

then $s_* = s_k^c + \lambda(s_k^N - s_k^c)$, where $\lambda = (-e + \sqrt{e^2 - df})/d$.

In 1980, Davidon first proposed the conic model (see [13]). It is an alternative model to substitute the quadratic model. For optimization problems; if the objective function has a strong nonquadratic or its curvature changes severely, then the conic model is better than the quadratic model both the effect of data fitting and the result of numerical calculations. In addition, the conic model can supply enough freedom to make best use of both information of gradients and function values in iterate points. In view of these good properties of conic model, it has attracted wide attention of many scholars [14–28]. In [28], Ni proposed a new trust region subproblem and gave the optimality conditions for the trust region subproblems of a conic model. That is, at the kth iteration, the trial step s_k is computed by solving the following conic model trust region subproblem

$$\min_{s\in R^n} \quad \phi_k(s) = \frac{g_k^T s}{1 - a_k^T s} + \frac{s^T B_k s}{2\left(1 - a_k^T s\right)^2}, \tag{5}$$

$$\text{s.t.} \quad s \in S, \tag{6}$$

where

$$S = \left\{ s \mid \|s\| \le \Delta_k,\ \left|1 - a_k^T s\right| \ge \varepsilon_0 \right\}, \tag{7}$$

horizon vector $a_k \in R^n$, B_k is symmetric and positive semidefinite, and ε_0 $(0 < \varepsilon_0 < 1)$ is a sufficiently small positive number. We note that the conic model $\phi_k(s)$ has a denominator and the shape of the trust region S is irregular; therefore the conic trust region subproblems (5)-(7) are not easy to search for the descent point and difficult to solve. The trust region method often does not require the exact solution of trust region subproblem but only requires an approximate solution. The dogleg method for solving the trust region subproblem based on conic model is an approximate solution method; however its calculation is relatively complicated.

In this paper, we continue to study the subproblems (5)-(7). In order to find an easy way to solve, inspired by the alternating direction method of multipliers (ADMM), we consider obtaining the approximate solution by the two-step search in two orthogonal directions in the trust region S. ADMM is an algorithm that solves convex optimization problems by breaking them into smaller pieces, each of which are then easier to handle. Because of its significant efficiency and easy implementation, it has recently found wide application in a number of areas (see [29–45]).

In the following, we use the alternating orthogonal direction search method to find the approximate solution of the subproblems (5)-(7). The rest of this paper is organized as follows. In the next section, the motivation and description of the simple alternating direction search method are presented. In Section 3, we give the quasi-Newton method based on the conic model for solving unconstrained optimization problems and prove its global convergence properties. The numerical results are provided in Section 4.

2. Range of ε_0 and the Approximate Solution of the Subproblem

In this section, we will modify the range of ε_0 and give the motivation and description of the algorithm. We note that the conic model $\phi_k(s)$ has one more parameter a_k than $\varrho_k(s)$. Therefore, $\phi_k(s)$ can make full use of the existing function information to satisfy more interpolation conditions by using of the function values and the gradient values. All of these will improve the effectiveness of the algorithm. In general, a_k chooses a descent direction, such as g_{k-1}, g_k, or s_{k-1} (see [13–17]). For convenience, we omit the index k of a_k, g_k, and B_k in this section. Therefore, in this paper we assume that

$$a^T g < 0 \tag{8}$$

and B is positive (abbreviated as $B \succ 0$).

Let

$$B_{ag} = a^T B a - a^T a a^T g, \tag{9}$$

$$\bar{\varepsilon}_0 = \min\left\{1, \frac{a^T B a}{B_{ag}}\right\}. \tag{10}$$

From (8), we have $a \ne 0$ and

$$B_{ag} > 0. \tag{11}$$

Although in principle we are seeking the optimal solution of the subproblems (5)-(7), it is enough to find an approximate solution s_k in the feasible region and guarantee a sufficient reduction in the model and the global convergence. Therefore, in order to simplify this algorithm we choose ε_0 in (7) such that it satisfies

$$0 < \varepsilon_0 < \bar{\varepsilon}_0. \tag{12}$$

In the following, we consider the alternating direction search method to solve the subproblems (5)-(7) by making full use of the parameters a. The new method is divided into two steps. First, we search in the direction of a and then search in the direction y which is perpendicular to a.

Let

$$s = \tau a + y, \tag{13}$$

where $\tau \ge 0, a \ne 0, y \in R^n$, and $a^T y = 0$. Then, the solving process of subproblems (5)-(7) is divided into the following two stages.

In the first stage, set $y = 0$ and then $s = \tau a$. Substituting it into (5)-(7), we have

$$\min_{\tau\in R} \quad \rho(\tau) = \frac{\tau a^T g}{1 - \tau a^T a} + \frac{\tau^2 a^T B a}{2\left(1 - \tau a^T a\right)^2}, \tag{14}$$

$$\text{s.t.} \quad \tau \in \Omega, \tag{15}$$

where

$$\Omega = \left\{\tau \mid 0 \le \tau \le \tau_\Delta\right\} \cap \left\{\tau \mid \tau \le \tau_d \ or\ \tau \ge \tau_u\right\}, \tag{16}$$

$$\begin{aligned} \tau_\Delta &= \frac{\Delta}{\|a\|}, \\ \tau_d &= \frac{1 - \varepsilon_0}{\|a\|^2}, \\ \tau_u &= \frac{1 + \varepsilon_0}{\|a\|^2}. \end{aligned} \tag{17}$$

By the direct computation, we have that the derivative of $\rho(\tau)$ is

$$\rho'(\tau) = \frac{B_{ag}\tau + a^T g}{-\|a\|^6 (\tau - \tau_m)^3}, \tag{18}$$

where

$$\tau_m = \frac{1}{\|a\|^2}. \tag{19}$$

From (17), we know that $0 < \tau_d < \tau_m < \tau_u$ and then $\tau_m \notin \Omega$. Since $B_{ag} > 0$, then $\rho(\tau)$ has only one stationary point

$$\tau_{cp} = \frac{-a^T g}{B_{ag}}. \tag{20}$$

By simple calculation, the following lemmas can be easily obtained.

Lemma 2. *Suppose* $a^T g < 0, B \succ 0$ *and* $0 < \varepsilon_0 < \bar{\varepsilon}_0$. *Then* $\tau_{cp} < \tau_d$.

Proof. Combining with the definition of $\bar{\varepsilon}_0$ as in (10), we easily know that

$$0 < \varepsilon_0 < \frac{a^T Ba}{B_{ag}}. \tag{21}$$

From (17) and (20), we have

$$\tau_{cp} - \tau_d = \frac{\varepsilon_0 - \left(a^T Ba / B_{ag}\right)}{\|a\|^2} < 0. \tag{22}$$

□

Lemma 3. *Under the same conditions as Lemma 2, then* $\rho(\tau)$ *is monotonically increasing in the trust region* (τ_{cp}, τ_m); $\rho(\tau)$ *is monotonically decreasing for* $0 < \tau < \tau_{cp}$ *and* $\tau > \tau_m$.

Proof. From (19) and (20), we know that

$$\tau_m - \tau_{cp} = \frac{a^T Ba}{B_{ag}\|a\|^2} > 0. \tag{23}$$

Then combining with (18) we can obtain that the lemma obviously holds. □

Let τ_* be the solution of the subproblems (14)-(15). Then, we can get the following theorems by analysis.

Theorem 4. *Under the same conditions as Lemma 2, then the solution of the subproblems (14)-(15) is*

$$\tau_* = \begin{cases} \min\left\{\tau_{cp}, \tau_\Delta\right\}, & \text{if } \tau_\Delta \le \tau_d, \\ \tau_{cp}, & \text{if } \tau_\Delta > \tau_d. \end{cases} \tag{24}$$

Proof. (1) If $\tau_\Delta \le \tau_d$ then from (16) we know that $\Omega = [0, \tau_\Delta]$. From Lemmas 2 and 3, we can obtain that if $\tau_{cp} \le \tau_\Delta$ then $\tau_* = \tau_{cp}$ and if $\tau_\Delta < \tau_{cp}$ then $\tau_* = \tau_\Delta$. Therefore, $\tau_* = \min\{\tau_{cp}, \tau_\Delta\}$.

(2) If $\tau_d < \tau_\Delta < \tau_u$, then $\Omega = [0, \tau_d]$. From Lemmas 2 and 3, we can similarly get $\tau_* = \tau_{cp}$.

(3) If $\tau_\Delta \ge \tau_u$, then $\Omega = [0, \tau_d] \cup [\tau_u, \tau_\Delta]$. From Lemmas 2 and 3, we know that

$$\tau_* = \arg\min\left\{\rho\left(\tau_{cp}\right), \rho\left(\tau_\Delta\right)\right\}. \tag{25}$$

Because of $\tau_\Delta \ge \tau_u$, then from (17) we have $1 - \Delta\|a\| \le -\varepsilon_0 < 0$. Since $a^T g < 0$, then

$$\rho\left(\tau_\Delta\right) = \frac{\Delta a^T g}{\|a\|\left(1 - \Delta\|a\|\right)} + \frac{\Delta^2 a^T Ba}{2\|a\|^2\left(1 - \Delta\|a\|\right)^2} > 0. \tag{26}$$

Then $\tau_* = \tau_{cp}$. The theorem is proved. □

It is worth noting that if $\tau_* = \tau_\Delta$ then from (17) we have $\|\tau_* a\| = \Delta$. Therefore, for this case we set $s_* = \tau_* a$ and exit the calculation of subproblem. Otherwise, we know that $\tau_* a$ is inside the trust region S. Then, we should carry out the calculation of the second stage below.

We set $s = \tau_* a + y$ and substitute it into $\phi_k(s)$. Then the subproblems (5)-(7) become

$$\min_{y \in R^n} \quad \psi(y) = \frac{g^T\left(\tau_* a + y\right)}{1 - \tau_* a^T a} + \frac{\left(\tau_* a + y\right)^T B\left(\tau_* a + y\right)}{2\left(1 - \tau_* a^T a\right)^2}, \tag{27}$$

$$\text{s.t.} \quad \|y\| \le \tilde{\Delta}, \quad a^T y = 0, \tag{28}$$

where

$$\tilde{\Delta} = \sqrt{\Delta^2 - \left(\tau_*\right)^2\|a\|^2}. \tag{29}$$

In order to remove the equality constraint in (28), we use the null space technique. That is, for $a \neq 0$ then there exist $n - 1$ mutually orthogonal unit vectors $q, q, \ldots, q_{n-1}$ orthogonal to the parameter vector a. Set $Q = [q, q, \ldots, q_{n-1}]$ and $y = Qu$, where $u \in R^{n-1}$. Then (27)-(28) can be simplified as the following subproblem:

$$\min_{u \in R^{n-1}} \quad \tilde{\psi}(u) = \tilde{g}^T u + \frac{1}{2} u^T \tilde{B} u, \tag{30}$$

$$\text{s.t.} \quad \|u\| \le \tilde{\Delta}, \tag{31}$$

where

$$\tilde{g} = \frac{Q^T g}{1 - \tau_* a^T a} + \frac{\tau_* Q^T Ba}{\left(1 - \tau_* a^T a\right)^2}, \quad \tilde{B} = \frac{Q^T BQ}{\left(1 - \tau_* a^T a\right)^2} \tag{32}$$

Set $g_k = \widetilde{g}$, $B_k = \widetilde{B}$, and $\Delta_k = \widetilde{\Delta}$. By Algorithm 1, we can obtain the solution u_* of the subproblems (30)-(31). Then $y_* = Qu_*$ and $s_* = \tau_* a + y_*$. Thus, the subproblems (5)-(7) are solved approximately.

Now we could give the alternating direction search method for solving the conic trust region subproblems (5)-(7) as follows.

Algorithm 5. Given a, g, B, and Δ,

Step 1. If $a^T g \geq 0$, then set $a = 0$. Then solve the subproblems (5)-(7) by Algorithm 1 to get s_k, and stop.

Step 2. Compute B_{ag} and $\bar{\varepsilon}_0$ by (9) and (10). let $\varepsilon_0 = 0.9\bar{\varepsilon}_0$.

Step 3. Compute τ_Δ, τ_d and τ_{cp} by (17) and (20).

Step 4. Compute τ_* by (24).

Step 5. If $\tau_* = \tau_\Delta$, then $s_k = \tau_\Delta a$, and stop; otherwise, compute Q, $\widetilde{\Delta}$, $\widetilde{g}$, and $\widetilde{B}$ by (29) and (32).

Step 6. Set $g_k = \widetilde{g}$, $B_k = \widetilde{B}$ and $\Delta_k = \widetilde{\Delta}$. Then solve the subproblems (30)-(31) by Algorithm 1 to get u_*.

Step 7. Set $y_* = Qu_*$ and $s_k = \tau_* a + y_*$, and stop.

In order to discuss the lower bound of predicted reduction in each iteration, we define the following predicted reduction:

$$\begin{aligned} \operatorname{pred}(s) &= \phi(0) - \phi(s), \\ \operatorname{pred}_1(\tau) &= \rho(0) - \rho(\tau), \end{aligned} \tag{33}$$

$$\begin{aligned} \operatorname{pred}_2(y) &= \psi(0) - \psi(y), \\ \operatorname{pred}_3(u) &= \widetilde{\psi}(0) - \widetilde{\psi}(u) \end{aligned} \tag{34}$$

Now we should prove the following theorem to guarantee the global convergence of the algorithm proposed in the next section.

Theorem 6. *Under the same conditions as Lemma 2, if s_k is obtained from the above Algorithm 5, then there exists a positive constant c_3 such that*

$$pred(s_k) \geq \frac{1}{2} c_3 \|g\| \min\left\{\Delta, \frac{\|g\|}{\|B\|}\right\}. \tag{35}$$

Proof. (1) If s_k is obtained by Step 1 of Algorithm 5, then from Nocedal and Wright [46] we have

$$\operatorname{pred}(s_k) \geq \frac{1}{2} c_1 \|g\| \min\left\{\Delta, \frac{\|g\|}{\|B\|}\right\}, \tag{36}$$

where $c_1 \in (0, 1]$.

(2) If s_k is obtained by Step 5 of Algorithm 5, then $s_k = \tau_\Delta a$ and $\tau_* = \tau_\Delta$, where τ_* as defined in (24). By computation, we have

$$\begin{aligned} \operatorname{pred}(s_k) &= \operatorname{pred}_1(\tau_\Delta) = -\rho(\tau_\Delta) \\ &= \frac{-\Delta\left(\Delta B_{ag} + 2\|a\| a^T g - \Delta \|a\|^2 a^T g\right)}{2\|a\|^2 (1 - \Delta\|a\|)^2}. \end{aligned} \tag{37}$$

From (24), we know that if $\tau_* = \tau_\Delta$ then $\tau_\Delta \leq \tau_{cp}$ and $\tau_\Delta \leq \tau_d$. And then from (17) and (20), we can obtain

$$\Delta B_{ag} + \|a\| a^T g \leq 0 \tag{38}$$

and

$$1 - \Delta\|a\| \geq \varepsilon_0 > 0. \tag{39}$$

Combining with (37)-(39), we know that

$$\operatorname{pred}(s_k) \geq \frac{-\Delta a^T g}{2\|a\|(1 - \Delta\|a\|)} \geq \frac{\epsilon\Delta\|g\|}{2}, \tag{40}$$

where

$$\epsilon = \frac{-a^T g}{\|a\|\|g\|}. \tag{41}$$

For $a^T g < 0$, then $0 < \epsilon \leq 1$ holds obviously.

(3) If s_k is obtained by Step 7 of Algorithm 5, then $s_k = \tau_* a + Qu_*$, where $\tau_* \neq \tau_\Delta$. From (24), we know that $\tau_* = \tau_{cp}$. Combining with (33) and (34), we have

$$\operatorname{pred}(s_k) = \operatorname{pred}_1(\tau_{cp}) + \operatorname{pred}_3(u_*). \tag{42}$$

Because u_* is obtained by Algorithm 1, then from [46] we have

$$\operatorname{pred}_3(u_*) \geq \frac{1}{2} c_2 \|\widetilde{g}\| \min\left\{\widetilde{\Delta}, \frac{\|\widetilde{g}\|}{\|\widetilde{B}\|}\right\} \tag{43}$$

where $c_2 \in (0, 1]$ and $\widetilde{\Delta}$, $\widetilde{g}$, and $\widetilde{B}$ are defined by (29) and (32). Thus,

$$\begin{aligned} \operatorname{pred}(s_k) &\geq \operatorname{pred}_1(\tau_{cp}) \\ &= -\frac{\tau_{cp} a^T g}{1 - \tau_{cp}\|a\|^2} - \frac{\tau_{cp}^2 a^T B a}{2\left(1 - \tau_{cp}\|a\|^2\right)^2} \\ &= \frac{\left(a^T g\right)^2}{2a^T B a} \geq \frac{\epsilon^2 \|g\|^2}{2\|B\|}, \end{aligned} \tag{44}$$

where the second equality is from (20) and the last inequality is from (41).

Therefore, the theorem follows from (36), (40), and (44) with

$$c_3 = \min\left\{c_1, \epsilon^2\right\}. \tag{45}$$

□

3. The Algorithm and Its Convergence

In this section, we propose a quasi-Newton method with a conic model for unconstrained minimization and prove its convergence under some reasonable conditions. In order to solve the problem (1), we approximate $f(x)$ with a conic model of the form

$$m_k(s) = f_k + \frac{g_k^T s}{1 - a_k^T s} + \frac{1}{2}\frac{s^T B_k s}{\left(1 - a_k^T s\right)^2}, \quad (46)$$

where $f_k = f(x_k)$, $g_k = \nabla f(x_k)$, $B_k \in R^{n\times n}$, and $a_k \in R^n$ are parameter vectors.

Now we give the simple alternating direction trust region algorithm based on conic model (46).

Algorithm 7.

Step 0. Choose parameters $\epsilon, \varepsilon, \in (0,1)$, $0 < \eta_1 < \eta_2 < 1$, $0 < \delta_1 < 1 < \delta_2$, and $\overline{\Delta} > 0$; give a starting point $x_0 \in R^n$, $B_0 \in R^{n\times n}$, $a_0 \in R^n$, and an initial trust region radius $\Delta_0 \in (0, \overline{\Delta}]$; set $k = 0$.

Step 1. Compute f_k and g_k. If $\|g_k\| < \varepsilon$, and then stop with x_k as the approximate optimal solution; otherwise go to Step 2.

Step 2. Set $a = a_k$, $g = g_k$, $B = B_k$, and $\Delta = \Delta_k$. Then solve the subproblem (5)-(7) by Algorithm 5 to get one of the approximate solution s_k.

Step 3. Compute the ratio of predicted reduction and the actual reduction

$$r_k = \frac{ared(s_k)}{pred(s_k)}, \quad (47)$$

where

$$ared(s_k) = f(x_k) - f(x_k + s_k), \quad (48)$$

$$pred(s_k) = -\frac{g_k^T s_k}{1 - a_k^T s_k} - \frac{1}{2}\frac{s_k^T B_k s_k}{\left(1 - a_k^T s_k\right)^2}. \quad (49)$$

Step 4. If $r_k \le \eta_1$, then set $\Delta_k = \delta_1 \Delta_k$ and go to Step 2. If $r_k > \eta_1$, then set $x_{k+1} = x_k + s_k$ and choose the new trust region bound satisfying

$$\Delta_k = \begin{cases} \min\left\{\delta_2 \Delta_k, \overline{\Delta}\right\}, & \text{if } r_k \ge \eta_2, \|s_k\| = \Delta_k, \\ \Delta_k, & \text{otherwise.} \end{cases} \quad (50)$$

Step 5. Generate a_{k+1} and B_{k+1}; set $k = k+1$, and go to Step 1.

The choice of parameter a_{k+1} in the cone model method is crucial. In general, a_{k+1} and B_{k+1} are chosen to satisfy certain interpolation conditions, which means that the conic model function interpolates both the function values and the gradient values of the objective function at x_k and x_{k+1}. The choice of the parameters a_{k+1} and B_{k+1} can refer to [13–17] and [47–49], respectively. In this paper, we are not prepared to study the specific iterative formulas of a_{k+1} and B_{k+1} in depth and directly adopt the choice of a_{k+1} in [16] and the choice of B_{k+1} in [49].

From (49) and Theorem 6, we have

$$\text{pred}(s_k) \ge \frac{1}{2} c_3 \|g_k\| \min\left\{\Delta_k, \frac{\|g_k\|}{\|B_k\|}\right\}, \quad (51)$$

where c_3 as defined by (45). The following theorem guarantees that the Algorithm 7 is globally convergent.

Theorem 8. *Under the same conditions as Lemma 2, suppose that the level set*

$$L(x_0) = \{x \mid f(x) \le f(x_0)\} \quad (52)$$

and the sequence $\{\|a_k\|\}$, $\{\|g_k\|\}$, and $\{\|B_k\|\}$ is all uniformly bounded, B_k is symmetric and positive definite, and f is twice continuously differentiable in $L(x_0)$. Then for any $\varepsilon > 0$, Algorithm 7 terminates in finite number of iterations, that is,

$$\lim_{k\to\infty} \|g_k\| = 0. \quad (53)$$

Proof. We give the proof by contradiction. Suppose that there is $\varepsilon_1 > 0$ such that

$$\|g_k\| \ge \varepsilon_1, \quad \forall k. \quad (54)$$

From the hypothesis, we have

$$\begin{aligned} \|a_k\| &\le \overline{a}, \\ \|g_k\| &\le \overline{g}, \\ 0 < \|B_k\| &\le \overline{B}. \end{aligned} \quad (55)$$

Combining with (51)-(55), we have

$$\text{pred}(s_k) \ge \frac{1}{2} c_3 \min\left\{\Delta_k, \frac{\varepsilon_1}{\overline{B}}\right\} \ge \frac{1}{2}\zeta\Delta_k, \quad (56)$$

where the first inequality follows from

$$\min\{p, q\} \ge \frac{pq}{p+q}, \quad \forall p, q > 0, \quad (57)$$

and the second inequality is from $\Delta_k \le \overline{\Delta}$ and

$$\zeta = \frac{c_3 \varepsilon_1}{\varepsilon_1 + \overline{B}\,\overline{\Delta}}. \quad (58)$$

From Step 4 of Algorithm 7 and (56), we obtain that for all k

$$f_k - f_{k+1} \ge \eta_1 \text{pred}(s_k) \ge \frac{1}{2}\eta_1 \zeta \Delta_k. \quad (59)$$

Since $f(x)$ is bounded from below and $f_{k+1} < f_k$, then we have

$$\infty > \sum_k (f_k - f_{k+1}) \ge \sum_k \left(\frac{1}{2}\eta_1 \zeta \Delta_k\right), \quad (60)$$

which implies that

$$\sum_{k=1}^{\infty} \Delta_k < \infty \tag{61}$$

And then

$$\begin{aligned} \lim_{k \longrightarrow \infty} \Delta_k &= 0, \\ \lim_{k \longrightarrow \infty} \|s_k\| &= 0. \end{aligned} \tag{62}$$

On the other hand, from (55) and (62) we can get

$$\frac{1}{1 - a_k^T s_k} = 1 + a_k^T s_k + o\left(\|s_k\|\right), \tag{63}$$

$$\frac{s_k^T B_k s_k}{2\left(1 - a_k^T s_k\right)^2} = \frac{1}{2}\left(s_k\right)^T B_k s_k + o\left(\|s_k\|^2\right). \tag{64}$$

Then from (55) to (63), we have

$$\begin{aligned} &\left|f_k - f\left(x_k + s_k\right) - \operatorname{pred}\left(s_k\right)\right| = \Big| f_k - f\left(x_k + s_k\right) \\ &\quad + \left(1 + a_k^T s_k\right) g_k^T s_k + \frac{1}{2} s_k^T B_k s_k + o\left(\|s_k\|^2\right)\Big| \\ &= \Big| -\frac{1}{2} s_k^T \nabla^2 f\left(x_k + \vartheta_k s_k\right) s_k + a_k^T s_k g_k^T s_k + \frac{1}{2} s_k^T B_k s_k \\ &\quad + o\left(\|s_k\|^2\right)\Big| \le \frac{1}{2}\left(M_1 + 2\bar{a}\,\bar{g} + \bar{B} + O(1)\right)\|s_k\|^2 \\ &\le \frac{1}{2}\left(Q + O(1)\right)\left(\Delta_k\right)^2, \end{aligned} \tag{65}$$

where $\vartheta_k \in (0,1)$ and $Q = M_1 + 2\bar{a}\,\bar{g} + \bar{B}$. Combining with (56) and (65), we can get that

$$\left|r_k - 1\right| = \left|\frac{f_k - f\left(x_k + s_k\right)}{\operatorname{pred}\left(s_k\right)} - 1\right| \le \frac{Q + O(1)}{\zeta}\Delta_k. \tag{66}$$

From (62) we have $r_k \longrightarrow 1$. Hence, there is a sufficiently large positive number K such that $\forall k \ge K$ and

$$r_k > \eta_1 \tag{67}$$

holds. From Step 4 of Algorithm 7, it follows that

$$\Delta_{k+1} \ge \Delta_k, \quad \forall k \ge K, \tag{68}$$

which is a contradiction to (62). The theorem is proved. □

4. Numerical Tests

In this section, Algorithm 7 is tested with some standard test problems from [16, 50]. The names of the 16 test problems are listed in Table 1. All the computations are carried out in Matlab R2016b on a microcomputer in double precision arithmetic. These tests use the same stopping criterion $\|g_k\| \le 10^{-5}$. The columns in the Tables have the following meanings: No. denotes the numbers of the test problems; n is the dimension of the test problems; Iter is the number of iterations; nf is the number of function evaluations performed; ng is the number of gradient evaluations; f_k is the final objective function value; $\|g\|$ is the Euclidean norm of the final gradient; CPU(s) denotes the total iteration time of the algorithm in seconds.

Table 1: Test functions.

No.	Problem	No.	Problem
1	Cube	2	Penalty-I
3	Beale	4	Conic
5	Extended powell	6	Variably Dimensioned
7	Rosenbrock	8	Extended Trigonometric
9	Tridiagonal Exponential	10	Brent
11	Troesch	12	Cragg and Levy
13	Broyden Tridiagonal	14	Brown
15	Discrete Boundary Value	16	Extended Trigonometric

The parameters in these algorithms are

$$\begin{aligned} a_0 &= 0, \\ B_0 &= I, \\ \varepsilon_0 &= \epsilon = 10^{-5}, \\ \Delta_0 &= 1, \\ \bar{\Delta} &= 10, \\ \eta_1 &= 0.01, \\ \eta_2 &= 0.75, \\ \delta_1 &= 0.5, \\ \delta_2 &= 2. \end{aligned} \tag{69}$$

In order to analyze the effectiveness of our new algorithm, we compare Algorithm 7 with the alternating direction trust region method based on conic model (abbreviated as ADCTR) in [12]. The numerical results of ADCTR and Algorithm 7 are listed in Table 2. We note that the optimal value of these test problems is $f_* = 0$. From Table 2, we can see that the performance of Algorithm 7 is feasible and effective. For the above 16 problems, Algorithm 7 is better than the ADCTR for 13 tests and is somewhat bad for 4 tests, and the two algorithms are same in efficiency for the other 1 tests. Therefore, it seems that Algorithm 7 is better than algorithm ADCTR in [12].

5. Conclusions

The algorithm ADCTR and Algorithm 7 are similar; that is, the idea of alternating direction method is used to solve the conic trust region subproblem. However, Algorithm 7 in this paper takes into account the special property that the parameter vector a is generally taken as the descending direction. Thus, under the assumption of $a^T g < 0$, the calculation of Algorithm 7 is simpler to calculate and has

Table 2: Numerical results.

No.	n	Starting point	Algorithm	Iter	nf/ng	f_k	$\|g\|$	CPU (s)
1	2	(2,1.5)	ADCTR	52	53/43	1.0377e-15	1.9477e-06	0.064681
			Algorithm 7	56	57/37	7.7243e-13	5.0126e-06	0.049091
2	2	(1,2)	ADCTR	10	11/11	9.0831e-06	8.9419e-06	0.048493
			Algorithm 7	8	9/9	9.0831e-06	8.9461e-06	0.033654
3	2	(2,-2)	ADCTR	18	19/18	9.0379e-15	9.3925e-07	0.053525
			Algorithm 7	16	17/17	7.3180e-16	4.3875e-08	0.039889
4	2	(-1,1)	ADCTR	16	17/13	1.1407e-12	2.1360e-06	0.050445
			Algorithm 7	9	10/10	9.8953e-13	1.9895e-06	0.038502
5	4	(3,-1,0,1)	ADCTR	41	42/34	4.8648e-09	4.5887e-06	0.062011
			Algorithm 7	107	108/83	9.8794e-12	3.9079e-06	0.054692
6	4	0.25∗(3,2,1,0)	ADCTR	32	33/29	2.3856e-14	3.0965e-07	0.066287
			Algorithm 7	27	28/27	3.6994e-13	5.7059e-06	0.059686
7	2	(-1.2,1)	ADCTR	50	51/49	1.5486e-14	5.4101e-06	0.064227
			Algorithm 7	47	48/40	2.4987e-19	2.1924e-08	0.048847
8	4	0.25∗(1,1,1,1)	ADCTR	47	48/34	7.9158e-15	4.1153e-07	0.076865
			Algorithm 7	28	29/22	3.0282e-04	8.2932e-06	0.085045
9	4	1.5∗(1,1,1,1)	ADCTR	7	8/8	8.1577e-12	4.5905e-06	0.058505
			Algorithm 7	6	7/7	5.2383e-13	2.5151e-06	0.054092
10	4	10∗(1,1,1,1)	ADCTR	81	82/58	5.8024e-18	4.6604e-07	0.089702
			Algorithm 7	56	57/40	7.1113e-18	2.9344e-07	0.077486
11	4	(-1,-1,-1,-1)	ADCTR	64	65/52	9.9974e-14	4.4986e-06	0.082793
			Algorithm 7	51	52/41	2.0655e-13	5.6223e-06	0.060827
12	4	(-1,-1,-1,-1)	ADCTR	48	49/43	1.1247e-08	5.2578e-06	0.068215
			Algorithm 7	128	129/125	4.3892e-08	6.6956e-06	0.054621
13	4	(-1,-1,-1,-1)	ADCTR	35	36/19	1.4498e-11	5.0442e-06	0.063276
			Algorithm 7	38	39/19	1.4483e-11	6.4239e-06	0.047787
14	4	(0,1,0,1)	ADCTR	91	92/52	0.1998e-06	2.5916e-07	0.089294
			Algorithm 7	63	64/41	1.3996e-05	5.7261e-07	0.052852
15	4	-0.08∗(2, 3,3,2)	ADCTR	23	24/15	2.0042e-12	8.2898e-06	0.061544
			Algorithm 7	19	20/12	2.3574e-12	6.3180e-06	0.043328
16	4	0.25∗(1,1,1,1)	ADCTR	14	15/15	3.0282e-04	4.9068e-06	0.048488
			Algorithm 7	14	15/11	3.6700e-04	8.1833e-06	0.049998

shorter CPU time, better calculation effect, and also global convergence.

However, there are still many aspects worthy of further study, for example, weakening the positive definite condition of B_k, using algorithms to solve large-scale problems, calculation of convergence rate, and so on.

Data Availability

All data generated or analysed during this study are included within the article.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by National Natural Science Foundation of China (11071117, 11771210) and the Natural Science Foundation of Jiangsu Province (BK20141409, BK20150420).

References

[1] M. J. Powell and Y. Yuan, "A trust region algorithm for equality constrained optimization," *Mathematical Programming*, vol. 49, no. 2, pp. 189–211, 1990.

[2] A. Vardi, "A trust region algorithm for equality constrained minimization: convergence properties and implementation," *SIAM Journal on Numerical Analysis*, vol. 22, no. 3, pp. 575–591, 1985.

[3] P. T. Boggs, R. H. Byrd, and R. B. Schnabel, "A stable and efficient algorithm for nonlinear orthogonal distance regression," *SIAM Journal on Scientific Computing*, vol. 8, no. 6, pp. 1052–1078, 1987.

[4] P. L. Toint, "Global convergence of a class of trust-region methods for nonconvex minimization in Hilbert space," *IMA Journal of Numerical Analysis (IMAJNA)*, vol. 8, no. 2, pp. 231–252, 1988.

[5] J. Z. Zhang and D. T. Zhu, "Projected quasi-Newton algorithm with trust region for constrained optimization," *Journal of Optimization Theory and Applications*, vol. 67, no. 2, pp. 369–393, 1990.

[6] M. El-Alem, "A robust trust-region algorithm with a nonmonotonic penalty parameter scheme for constrained optimization," *SIAM Journal on Optimization*, vol. 5, no. 2, pp. 348–378, 1995.

[7] M. J. D. Powell, "A hybrid method for nonlinear equations," *Numerical Methods for Nonlinear Algebraic Equations*, pp. 87–114, 1970.

[8] J. Dennis and H. H. Mei, "Two new unconstrained optimization algorithms which use function and gradient values," *Journal of Optimization Theory and Applications*, vol. 28, no. 4, pp. 453–482, 1979.

[9] L. Zhang and Z. Q. Tang, "The hybrid dogleg method to solve subproblems of trust region," *Journal of Nanjing Normal University*, vol. 24, no. 1, pp. 28–32, 2001.

[10] J. Z. Zhang, X. J. Xu, and D. T. Zhu, "A nonmonotonic dogleg method for unconstrained optimization," *SIAM Journal on Scientific and Statistical Computing*, vol. 8, no. 6, pp. 1052–1078, 1987.

[11] Y. Zhao and C. Xu, "A new trust region dogleg method for unconstrained optimization," *Applied Mathematics-A Journal of Chinese Universities Series B*, vol. 15, no. 1, pp. 83–92, 2000.

[12] H. L. Zhu, Q. Ni, and C. Y. Dang, "A new alternating direction trust region method based on conic model for solving unconstrained optimization," https://arxiv.org/abs/1812.01935.

[13] W. C. Davidon, "Conic approximations and collinear scalings for optimizers," *SIAM Journal on Numerical Analysis*, vol. 17, no. 2, pp. 268–281, 1980.

[14] R. Schnabel, "Conic methods for unconstrained minimization and tensor methods for nonlinear equations," *Mathematical Programming the State of the Art*, vol. 21, no. 1, pp. 417–438, 1982.

[15] D. C. Sorensen, "Newton's method with a model trust region modification," *SIAM Journal on Numerical Analysis*, vol. 19, no. 2, pp. 409–426, 1982.

[16] M. F. Zhu, Y. Xue, and F. S. Zhang, "A quasi-Newton type trust region method based on the conic model," *Numerical Mathematics*, vol. 17, no. 1, pp. 36–47, 1995 (Chinese).

[17] X. P. Lu and Q. Ni, "A quasi-Newton trust region method with a new conic model for the unconstrained optimization," *Applied Mathematics and Computation*, vol. 204, no. 1, pp. 373–384, 2008.

[18] C. X. Xu and X. Y. Yang, "Convergence of conic quasi-Newton trust region methods for unconstrained minimization," *Mathematical Application*, vol. 11, no. 2, pp. 71–76, 1998.

[19] Y.-X. Yuan, "A review of trust region algorithms for optimization," *ICIAM*, vol. 99, no. 1, pp. 271–282, 2000.

[20] D. M. Gay, "Computing optimal locally constrained steps," *SIAM Journal on Scientific and Statistical Computing*, vol. 2, no. 2, pp. 186–197, 1981.

[21] J.-M. Peng and Y.-X. Yuan, "Optimality conditions for the minimization of a quadratic with two quadratic constraints," *SIAM Journal on Optimization*, vol. 7, no. 3, pp. 579–594, 1997.

[22] W. Sun and Y. X. Yuan, "A conic trust-region method for nonlinearly constrained optimization," *Annals of Operations Research*, vol. 103, pp. 175–191, 2001.

[23] S. Di and W. Sun, " A trust region method for conic model to solve unconstraind optimizaions ," *Optimization Methods and Software*, vol. 6, no. 4, pp. 237–263, 2007.

[24] H. Gourgeon and J. Nocedal, "A conic algorithm for optimization," *SIAM Journal on Scientific Computing*, vol. 6, no. 2, pp. 253–267, 1985.

[25] X. T. Zhu, M. Xi, W. Y. Sun, and J. B. S. Raimundo, "A new nonmonotone BB-TR method based on simple conic model for large scale unconstrained optimization," *Numerical Mathematics A Journal of Chinese Universities*, vol. 38, no. 2, pp. 172–192, 2016.

[26] X. Zhang, J. Wen, and Q. Ni, "Subspace trust-region algorithm with conic model for unconstrained optimization," *Numerical Algebra, Control and Optimization*, vol. 3, no. 2, pp. 223–234, 2013.

[27] L. Zhao and W. Sun, "A conic affine scaling method for nonlinear optimization with bound constraints," *Asia-Pacific Journal of Operational Research*, vol. 30, no. 3, pp. 1–30, 2013.

[28] Q. Ni, "Optimality conditions for trust-region subproblems involving a conic model," *SIAM Journal on Optimization*, vol. 15, no. 3, pp. 826–837, 2005.

[29] D. Gabay and B. Mercier, "A dual algorithm for the solution of nonlinear variational problems via finite element approximation," *Computers & Mathematics with Applications*, vol. 2, no. 1, pp. 17–40, 1976.

[30] D. M. Gay, "Algorithm 611: Subroutines for Unconstrained Minimization Using a Model/Trust-Region Approach," *ACM Transactions on Mathematical Software*, vol. 9, no. 4, pp. 503–524, 1983.

[31] G. Chen and M. Teboulle, "A proximal-based decomposition method for convex minimization problems," *Mathematical Programming*, vol. 64, no. 1–3, pp. 81–101, 1994.

[32] J. Eckstein and M. Fukushima, "Some reformulations and applications of the alternating direction method of multipliers," in *Large Scale Optimization*, pp. 115–134, Springer, 1994.

[33] B. S. He, L.-Z. Liao, D. R. Han, and H. Yang, "A new inexact alternating directions method for monotone variational inequalities," *Mathematical Programming*, vol. 92, no. 1, pp. 103–118, 2002.

[34] S. Kontogiorgis and R. R. Meyer, "A variable-penalty alternating directions method for convex optimization," *Mathematical Programming*, vol. 83, no. 1, pp. 29–53, 1998.

[35] K. Zhang, J. Li, Y. Song, and X. Wang, "An alternating direction method of multipliers for elliptic equation constrained optimization problem," *Science China Mathematics*, vol. 60, no. 2, pp. 361–378, 2017.

[36] M. H. Xu, "Proximal alternating directions method for structured variational inequalities," *Journal of Optimization Theory and Applications*, vol. 134, no. 1, pp. 107–117, 2007.

[37] Lu Li, Xingyu Wang, and Guoqiang Wang, "Alternating Direction Method of Multipliers for Separable Convex Optimization of Real Functions in Complex Variables," *Mathematical Problems in Engineering*, vol. 2015, Article ID 104531, 14 pages, 2015.

[38] C. H. Chen, B. S. He, Y. Ye et al., "The direct extension of ADMM for multi-block convex minimization problems is not necessarily convergent," *Mathematical Programming*, vol. 155, no. 1-2, pp. 57–79, 2016.

[39] M. L. Goncalves, J. G. Melo, and R. D. Monteiro, "Improved pointwise iteration-complexity of a regularized ADMM and of a regularized non-Euclidean HPE framework," *SIAM Journal on Optimization*, 2016.

[40] J. Bai, J. Li, F. Xu, and H. Zhang, “Generalized symmetric ADMM for separable convex optimization,” *Computational optimization and applications*, vol. 70, no. 1, pp. 129–170, 2018.

[41] B. He and X. Yuan, “A class of ADMM-based algorithms for three-block separable convex programming,” *Computational Optimization and Applications*, vol. 70, no. 3, pp. 1–36, 2018.

[42] M. Hong and Z.-Q. Luo, “On the linear convergence of the alternating direction method of multipliers,” *Mathematical Programming*, vol. 162, no. 1-2, Ser. A, pp. 165–199, 2017.

[43] Y. T. Sun and J. L. Zhao, “An alternating directions method for structured split feasibility problems,” *Journal on Numerical Methods and Computer Applications*, vol. 39, no. 1, pp. 20–27, 2018 (Chinese).

[44] H. B. Zhu, Y. Gao, and Y. Hou, “Real-time pricing for demand response in smart grid based on alternating direction method of multipliers,” *Mathematical Problems in Engineering*, vol. 2018, Article ID 8760575, 10 pages, 2018.

[45] X.-R. Lv, Y. Li, and Y.-C. He, “Efficient impulsive noise mitigation for OFDM systems using the alternating direction method of multipliers,” *Mathematical Problems in Engineering*, vol. 2018, Article ID 4968682, 11 pages, 2018.

[46] J. Nocedal and S. J. Wright, *Numerical Optimization*, Science Press, Beijing, China, 2006.

[47] M. J. D. Powell, “Algorithms for nonlinear constraints that use Lagrange functions,” *Mathematical Programming*, vol. 14, no. 1, pp. 224–248, 1978.

[48] M. Al-Baali, “Damped techniques for enforcing convergence of quasi-Newton methods,” Taylor and Francis, Inc., vol. 29, no. 5, pp. 919–936, 2014.

[49] Q. Ni, *Optimization Method and Program Design*, Science Press, Beijing, China, 2009.

[50] J. J. Moré, B. S. Garbow, and K. E. Hillstrom, “Testing unconstrained optimization software,” *ACM Transactions on Mathematical Software*, vol. 7, no. 1, pp. 17–41, 1981.