CrossMark

ORIGINAL PAPER

# A linear-time algorithm for the trust region subproblem based on hidden convexity

Jiulin Wang[1] · Yong Xia[1]

**Abstract** We present a linear-time approximation scheme for solving the trust region subproblem (TRS). It employs Nesterov's accelerated gradient descent algorithm to solve a convex programming reformulation of (TRS). The total time complexity is less than that of the recent linear-time algorithm. The algorithm is further extended to the two-sided trust region subproblem.

**Keywords** Trust region subproblem · Hidden convexity · Proximal gradient algorithm · Linear time complexity

## 1 Introduction

Consider the classical trust region subproblem

$$\text{(TRS)} \quad \min \ f(x) = \frac{1}{2}x^T A x + b^T x$$
$$\text{s.t.} \ x^T x \leq 1,$$

where $A \in \mathbb{R}^{n \times n}$ is symmetric and possibly indefinite and $b \in \mathbb{R}^n$. (TRS) plays a great role in trust region methods [2,14] for nonlinear programming. Efficient numerical algorithms for (TRS) can be found in [5,8]. When $b = 0$, (TRS) reduces to the maximal eigenvector problem, which can be approximated in linear time [4]. Recently, the first

---

✉ Yong Xia
dearyxia@gmail.com

[1] State Key Laboratory of Software Development Environment, LMIB of the Ministry of Education School of Mathematics and System Sciences, Beihang University, 100191 Beijing, People's Republic of China

linear-time approximation algorithm for (TRS) is proposed [6] based on strong duality, approximate eigenvector oracle and the bisection scheme.

In this paper, we present a new and simple linear-time approximation scheme for solving (TRS). Actually, our algorithm works for a more general version of (TRS), the two-sided trust region subproblem:

$$\text{(TTRS)} \quad \min \ f(x) = \frac{1}{2}x^T A x + b^T x \tag{1}$$

$$s.t. \ \alpha \le x^T x \le 1, \tag{2}$$

where $0 \le \alpha \le 1$. (TTRS) was first introduced in [11]. For a recent survey on (TTRS), we refer to [10].

In the following, we study (TTRS) and keep in mind that (TRS) is a special case of (TTRS) when $\alpha = 0$. Our approximation scheme contains two main steps. First we establish a new convex programming reformulation for (TTRS), which is different from the existing convex reformulations [1,13]. Secondly, we employ Nesterov's accelerated gradient descent algorithm [9] to solve the convexified version of (TTRS). The total worst-case time complexity is even less than that of the recent linear-time algorithm [6], which only works for (TRS).

The remainder of this paper is as follows. Section 2 presents a new hidden convexity for (TTRS). Section 3 applies Nesterov's fast gradient method to solve the convexified formulation of (TTRS) and then analyzes the worst-case time complexity. The complexity is even less than the recent linear-time algorithm for (TRS). Conclusions are made in Sect. 4.

## 2 Hidden convexity

In this section, we present a new convex quadratic constrained quadratic programming reformulation for (TTRS).

Let $v(\cdot)$ be the optimal value of $(\cdot)$. Denote by $\lambda_1$ the minimal eigenvalue of $A$. Let $v \ne 0$ be the corresponding eigenvector. Without loss of generality, we assume $b^T v \le 0$.

**Theorem 1** (*TTRS*) *is equivalent to the following convex programming relaxation*

$$\text{(C)} \quad \min_{x,t} \frac{1}{2}x^T (A - \lambda_1 I) x + b^T x + \frac{1}{2}\lambda_1 t \tag{3}$$

$$s.t. \quad x^T x \le t, \ \alpha \le t \le 1, \tag{4}$$

*in the sense that* $v(TTRS) = v(C)$. *Moreover, for any optimal solution of* (C), *denoted by* $(x^*, t^*)$,

$$x^* + \frac{\sqrt{(x^{*T} v)^2 - v^T v(x^{*T} x^* - t^*)} - x^{*T} v}{v^T v} v \tag{5}$$

*globally solves* (TTRS).

*Proof* Let $\widetilde{x}$ be an optimal solution of (TTRS). Then, $(\widetilde{x}, \widetilde{t} := \widetilde{x}^T \widetilde{x})$ is a feasible solution of (C) whose objective value is equal to $v$(TTRS). Therefore, we have $v$(TTRS) $\geq v$(C). Now it is sufficient to show that the lower bound $v$(C) is attained for (TTRS).

Let $(x^*, t^*)$ be an optimal solution of (C). We assume $x^{*T} x^* < t^*$, otherwise, we are done. Define

$$x(\gamma) = x^* + \gamma v.$$

Then, $(x(\gamma), t^*)$ is a feasible solution of (C) for any $\gamma$ such that $x(\gamma)^T x(\gamma) \leq t^*$. Moreover, the objective value of $(x(\gamma), t^*)$ is decreasing when $\gamma \geq 0$ increases. Consequently, $(x(\gamma), t^*)$ remains optimal for $\gamma \geq 0$ and $x(\gamma)^T x(\gamma) \leq t^*$. Notice that the equation $x(\gamma)^T x(\gamma) = t^*$ in terms of $\gamma$ has a positive root $\frac{\sqrt{(x^{*T} v)^2 - v^T v(x^{*T} x^* - t^*)} - x^{*T} v}{v^T v}$. The proof is complete. $\square$

When $\alpha = 0$, Theorem 1 reduces to the following result for (TRS).

**Corollary 1** ([3], Theorem 1) *Let* $\mu = \min\{\lambda_1, 0\}$. *(TRS) is equivalent to the convex programming relaxation*

$$(C_0) \quad \min_{x^T x \leq 1} \frac{1}{2} x^T (A - \mu I) x + b^T x + \frac{1}{2} \mu$$

*in the sense that* $v$(TRS) $= v(C_0)$. *Moreover, when* $\lambda_1 < 0$, *for any optimal solution of* $(C_0)$, *denoted by* $x^*$,

$$x^* + \frac{\sqrt{(x^{*T} v)^2 - v^T v(x^{*T} x^* - 1)} - x^{*T} v}{v^T v} v \qquad (6)$$

*globally solves* (TRS).

To approximately calculate $\lambda_1$, we need the following estimation [6] on the maximal eigenvalue of $A$, denoted by $\lambda_{\max}(A)$, which is based on the analysis by Kuczynski and Wozniakowski [7].

**Theorem 2** [6,7] *Given a symmetric matrix* $A \in \mathbb{R}^{n \times n}$ *with* $\|A\| \leq \rho$, *parameters* $\varepsilon, \delta > 0$ *and let* $N(\geq n)$ *is an upper bound of the number of non-zero entries in* $A$, *the Lanczos method [4] runs in time* $O(\frac{N\sqrt{\rho}}{\sqrt{\varepsilon}} \log \frac{n}{\delta})$ *and returns a unit vector* $x \in \mathbb{R}^n$ *for which*

$$x^T A x \geq \lambda_{\max}(A) - \epsilon$$

*with probability at least* $1 - \delta$.

Notice that $\|\rho I - A\| \leq 2\rho$ and the maximal eigenvalue of $\rho I - A$ is equal to $\rho - \lambda_1$. Thus, applying Theorem 2 to $\rho I - A$ yields the following estimation on $\lambda_1$.

**Corollary 2** *Given a symmetric matrix $A \in \mathbb{R}^{n \times n}$ with $\|A\| \leq \rho$, parameters $\varepsilon, \delta > 0$ and let $N(\geq n)$ is an upper bound of the number of non-zero entries in A, the Lanczos method [4] runs in time $O(\frac{N\sqrt{\rho}}{\sqrt{\varepsilon}} \log \frac{n}{\delta})$ and returns a unit vector $\widetilde{v} \in \mathbb{R}^n$ for which*

$$(\lambda_1 \leq) \widetilde{v}^T A \widetilde{v} \leq \lambda_1 + \epsilon$$

*with probability at least $1 - \delta$.*

Define $\widetilde{\lambda}_1 = \widetilde{v}^T A \widetilde{v} - \epsilon$. Then,

$$A - \widetilde{\lambda}_1 I \succeq A - \lambda_1 I \succeq 0.$$

As an approximation of (C), we turn to solve the following convex programming problem

$$\text{(AC)} \quad \min_{x,t} \ g(x, t) := \frac{1}{2} x^T (A - \widetilde{\lambda}_1 I) x + b^T x + \frac{1}{2} \widetilde{\lambda}_1 t \tag{7}$$

$$\text{s.t.} \quad (x, t) \in X := \{(x, t) \mid x^T x \leq t, \ \alpha \leq t \leq 1\}. \tag{8}$$

It is not difficult to verify that

$$|v(\text{AC}) - v(\text{C})| \leq \epsilon. \tag{9}$$

## 3 Nesterov's accelerated gradient method and complexity

In this section, we employ Nesterov's accelerated gradient descent method [9] to solve (AC) (7), (8) and then study the worst-case time complexity.

Notice that $g(x, t)$ defined in (7) is a convex quadratic function. Then, the gradient of $g$, denoted by $\nabla g$, is Lipschitz continuous:

$$\|\nabla g(x_1, t_1) - \nabla g(x_2, t_2)\| \leq 2\rho \|(x_1, t_1) - (x_2, t_2)\|$$

where $\rho$ is an upper bound of $\|A\|$.

To solve (AC) (7), (8), we use the following algorithm in [12], which is a slightly generalized version of Nesterov's accelerated gradient method [9].

**Algorithm 1** 1. *Choose $\theta_0 = \theta_{-1} \in (0, 1]$, $(x_0, t_0) = (x_{-1}, t_{-1}) \in X$. Let $k := 0$.*
2. *Let*

$$(y_k, s_k) = (x_k, t_k) + \theta_k (\theta_{k-1}^{-1} - 1)((x_k, t_k) - (x_{k-1}, t_{k-1}))$$

*and $\nabla g_k = \nabla g(y_k, s_k)$. Update*

$$(x_{k+1}, t_{k+1}) = \arg \min_{(x,t) \in X} \nabla g_k^T (x, t) + \rho \|(x, t) - (y_k, s_k)\|^2. \tag{10}$$

3. *Choose $\theta_{k+1} \in (0, 1]$ satisfying*

$$\frac{1 - \theta_k}{\theta_{k+1}^2} \leq \frac{1}{\theta_k^2}.$$

*If the stopping criterion does not hold, update $k := k + 1$, and go to step 2.*

**Theorem 3** [9,12] *Let $\{(x_k, t_k)\}$ be generated by Algorithm 1, $(x^*, t^*)$ be the global minimizer of (AC) (7), (8), then*

$$g(x_k, t_k) - g(x^*, t^*) \leq O\left(\frac{\rho}{k^2}\right), \forall k. \tag{11}$$

*That is, Algorithm 1 returns an $\epsilon$-approximation solution of (AC) in $O(\sqrt{\rho/\varepsilon})$ iterations.*

In each iteration of Algorithm 1, the main computational cost is to solve (10). In the following, we show the solution details of (10). We first rewrite the objective function of (10) as follows

$$
\begin{aligned}
G(x, t) &:= \nabla g_k^T(x, t) + \rho \|(x, t) - (y_k, s_k)\|^2 \\
&= \rho x^T x + [(A - \widetilde{\lambda}_1 I - 2\rho I) y_k + b]^T x + \rho t^2 + (\widetilde{\lambda}_1/2 - 2\rho s_k)t + e \\
&:= \rho x^T x + c^T x + \rho t^2 + dt + e
\end{aligned}
$$

where $c = (A - \widetilde{\lambda}_1 I - 2\rho I) y_k + b$, $d = \widetilde{\lambda}_1/2 - 2\rho s_k$ and $e = \rho \|(y_k, s_k)\|^2$.

Now, (10) is equivalent to the following convex program:

$$\min \rho x^T x + c^T x + \rho t^2 + dt \tag{12}$$
$$\text{s.t. } x^T x \leq t, \tag{13}$$
$$t^2 - (\alpha + 1)t + \alpha \leq 0. \tag{14}$$

It is sufficient to solve the KKT system of (12)–(14), which reads as follows:

$$2\rho x + c + 2\mu_1 x = 0, \tag{15}$$
$$2\rho t + d - \mu_1 + 2\mu_2 t - (\alpha + 1)\mu_2 = 0, \tag{16}$$
$$\mu_1(x^T x - t) = 0, \tag{17}$$
$$\mu_2(t^2 - (\alpha + 1)t + \alpha) = 0, \tag{18}$$
$$x^T x \leq t, \ \alpha \leq t \leq 1, \ \mu_1 \geq 0, \ \mu_2 \geq 0, \tag{19}$$

where $\mu_1$ and $\mu_2$ are the Lagrangian multipliers corresponding to (13) and (14), respectively.

Notice that $\rho > 0$, $\mu_1 \geq 0$ and $\mu_2 \geq 0$. It follows from (15) and (16) that

$$x(\mu_1) = \frac{-c}{2\rho + 2\mu_1}, \tag{20}$$

$$t(\mu_1, \mu_2) = \frac{\mu_1 + (\alpha + 1)\mu_2 - d}{2\rho + 2\mu_2}. \tag{21}$$

Consider the following different cases.

(a) Suppose $\mu_1 = 0$ and $\mu_2 = 0$. If $x(0)^T x(0) \leq t(0, 0)$ and $\alpha \leq t(0, 0) \leq 1$, then $(x(0), t(0, 0))$ is the optimal solution of (10).
(b) Suppose $\mu_1 = 0$ and $\mu_2 > 0$. According to (18), $t(0, \mu_2) = \alpha$ or 1. Hence, $\mu_2$ is solved. If $x(0)^T x(0) \leq t(0, \mu_2)$, then $(x(0), t(0, \mu_2))$ is the optimal solution of (10).
(c) Suppose $\mu_1 > 0$ and $\mu_2 = 0$. According to (17), we have $x(\mu_1)^T x(\mu_1) = t(\mu_1, 0)$. Consequently, $\mu_1$ is obtained by solving this cubic equation. If $\alpha \leq t(\mu_1, 0) \leq 1$, then $(x(\mu_1), t(\mu_1, 0))$ is the optimal solution of (10).
(d) Suppose $\mu_1 > 0$ and $\mu_2 > 0$. It follows from (17) to (18) that $x(\mu_1)^T x(\mu_1) = t(\mu_1, \mu_2)$, and $t(\mu_1, \mu_2) = \alpha$ or 1. The first equation is cubic and the second is linear. Therefore, combining both yields a cubic function of $(\mu_1)$, which also has an explicit solution. For the obtained $(\mu_1, \mu_2)$, if $x(\mu_1)^T x(\mu_1) \leq t(\mu_1, \mu_2)$ and $\alpha \leq t(\mu_1, \mu_2) \leq 1$, then $(x(\mu_1), t(\mu_1, \mu_2))$ is the optimal solution of (10).

From above, we can see that (10) is modeled in $O(N)$ time and then solved in $O(n)$ time, where $N(\geq n)$ is an upper bound of the number of non-zero entries in $A$. According to Theorem 3, we have the following estimation.

**Lemma 1** *The time complexity of approximately solving (AC) is*

$$O\left(N\sqrt{\frac{\rho}{\varepsilon}}\right). \tag{22}$$

Furthermore, combining (9), Theorem 1, Corollary 2, and Lemma 1 yields the following main result.

**Theorem 4** *Given the parameters $\epsilon > 0$, $\delta > 0$, with probability at least $1 - \delta$ over the randomization of an approximate eigenvector oracle, we can find an $\epsilon$-approximation solution of (TTRS), $\widetilde{x} \in \mathbb{R}^n$, satisfying $\alpha \leq \widetilde{x}^T \widetilde{x} \leq 1$ and*

$$f(\widetilde{x}) \leq v(TTRS) + \epsilon,$$

*in total time*

$$O\left(\frac{N\sqrt{\|A\|}}{\sqrt{\varepsilon}} \log \frac{n}{\delta}\right). \tag{23}$$

We notice that Hazan and Koren's approximation scheme [6] is particular to (TRS). To the best of our knowledge, whether it can be extended to solve (TTRS) remains open. Nevertheless, even for (TRS), our complexity (23) is clearly less than Hazan and Koren's [6]:

$$O\left(\frac{N\sqrt{\beta}}{\sqrt{\varepsilon}}\log\frac{\beta}{\varepsilon}\log\left(\frac{n}{\delta}\log\frac{\beta}{\varepsilon}\right)\right),$$

where $\beta = \max(\|A\| + \|b\|, 1)$.

## 4 Conclusions

Recently, the first linear-time algorithm for the trust region subproblem (TRS) is proposed based on strong duality, approximate eigenvector oracle and the bisection scheme. In this paper, we present a new linear-time approximation scheme for solving the trust region subproblem (TRS). It trivially employs Nesterov's accelerated gradient descent algorithm to solve a convex quadratic constrained quadratic programming reformulation of (TRS). However, the total time complexity of our scheme is less than that of the recent one for (TRS). Our approach is further extended to the two-sided trust region subproblem. The future work will include more extensions to the other nonconvex quadratic programming problems.

## References

1. Ben-Tal, A., Teboulle, M.: Hidden convexity in some nonconvex quadratically constrained quadratic programming. Math. Program. **72**, 51–63 (1996)
2. Conn, A.R., Gould, N.I., Toint, P.L.: Trust Region Methods. Society for Industrial and Applied Mathematics, Philadelphia (2000)
3. Flippo, O.E., Jansen, B.: Duality and sensitivity in nonconvex quadratic optimization over an ellipsoid. Eur. J. Oper. Res. **94**, 167–178 (1996)
4. Golub, G.H., Van Loan, C.F.: Matrix Computations. Johns Hopkins University Press, Baltimore (1989)
5. Gould, N.I.M., Lucidi, S., Roma, M., Toint, P.L.: Solving the trust-region subproblem using the Lanczos method. SIAM J. Optim. **9**(2), 504–525 (1999)
6. Hazan, E., Koren, T.: A linear-time algorithm for trust region problems. Math. Program. **158**(1), 363–381 (2016)
7. Kuczynski, J., Wozniakowski, H.: Estimating the largest eigenvalue by the power and Lanczos algorithms with a random start. SIAM J. Matrix Anal. Appl. **13**(4), 1094–1122 (1992)
8. Moré, J.J., Sorensen, D.C.: SIAM J. Sci. Statist. Comput. Computing a trust region step **4**(3), 553–572 (1983)
9. Nesterov, YuE: A method for unconstrained convex minimization problem with the rate of convergence $O(1/k^2)$. Soviet Math. Dokl. **27**(2), 372–376 (1983)
10. Pong, T.K., Wolkowicz, H.: Generalizations of the trust region subproblem. Comput. Optim. Appl. **58**(2), 273–322 (2014)
11. Stern, R., Wolkowicz, H.: Indefinite trust region subproblems and nonsymmetric eigenvalue perturbations. SIAM J. Optim. **5**(2), 286–313 (1995)

12. Tseng, P.: On accelerated proximal gradient methods for convex–concave optimization (unpublished manuscript) (2008)
13. Wang, S., Xia, Y.: Strong duality for generalized trust region subproblem: S-lemma with interval bounds. Optim. Lett. **9**(6), 1063–1073 (2015)
14. Yuan, Y.: Recent advances in trust region algorithms. Math. Program. **151**(1), 249–281 (2015)