Journal of Inequalities and Applications
a SpringerOpen Journal

RESEARCH Open Access

Smoothing approximation to the lower order exact penalty function for inequality constrained optimization

Shujun Lian[1*] and Nana Niu[1]

*Correspondence:
lsjsd2003@126.com
[1] School of Management Science, Qufu Normal University, Rizhao, China

Abstract

For inequality constrained optimization problem, we first propose a new smoothing method to the lower order exact penalty function, and then show that an approximate global solution of the original problem can be obtained by solving a global solution of a smooth lower order exact penalty problem. We propose an algorithm based on the smoothed lower order exact penalty function. The global convergence of the algorithm is proved under some mild conditions. Some numerical experiments show the efficiency of the proposed method.

MSC: 90C30

Keywords: Lower order penalty function; Inequality constrained optimization; Exact penalty function; Smoothing method

1 Introduction

Consider the following inequality constrained optimization problem:

$$\begin{aligned} &\min f_0(x) \\ &\quad \text{s.t. } f_i(x) \leq 0, \quad i \in I = \{1, 2, \ldots, m\}, \end{aligned} \tag{P}$$

where $f_i : R^n \to R$, $i = 0, 1, \ldots, m$, are twice continuously differentiable functions. Throughout this paper, we use $X_0 = \{x \in R^n | f_i(x) \leq 0, i \in I\}$ to denote the feasible solution set.

This problem is widely applied in transportation, economics, mathematical programming, regional science, etc. [1–3], and it has received extensive attention on a related problem, for example, variational inequalities, equilibrium problem, minimizers of convex functions, etc. (see, e.g., [4–15]).

To solve problem (*P*), the penalty function methods have been introduced in many literature works (see, e.g., [16–24]). Zangwill [16] introduced the classical l_1 exact penalty function

$$F_1(x, q) = f_0(x) + q \sum_{i=1}^{m} \max\{f_i(x), 0\}, \tag{1.1}$$

where $q > 0$ is a penalty parameter, but it is not a smooth function. The corresponding penalty optimization problem is as follows:

$$\min_{x \in R^n} F_1(x,q). \tag{P_1}$$

The non-smoothness of the function restricts the application of a gradient-type or Newton-type algorithm to solving problem (P_1). In order to avoid this shortcoming, the smoothing of the l_1 exact penalty function is proposed in [17, 18].

In addition, to overcome the non-smoothness of the function, the following smooth penalty function is proposed:

$$F_2(x,q) = f_0(x) + q \sum_{i=1}^{m} \max\{f_i(x), 0\}^2. \tag{1.2}$$

However, the function is non-exact.

Recently, Wu *et al.* [20] proposed the following low order penalty function:

$$\varphi_{q,k}(x) = f_0(x) + q \sum_{i=1}^{m} \left(\max\{f_i(x), 0\}\right)^k, \quad k \in (0,1), \tag{1.3}$$

and proved that the low order penalty function is exact under mild conditions. But this penalty function is non-smooth, too. When $k = 1$, $\varphi_{q,k}(x)$ can be seen as the classical l_1 exact penalty function. The least exact penalty parameter corresponding to $k \in (0,1)$ is much less than that of the l_1 exact penalty function. This can avoid the defects of too large parameter ρ in the algorithm. Only for $k = \frac{1}{2}$, the smoothing of the lower order penalty function (1.3) is studied in [20] and [21]. In [24], a smoothing method of the low order penalty function (1.3) is given. We hope to study a new smoothing method for the low order penalty function (1.3) and compare it with the existing methods. With a different segmentation method, we will give a new piecewise smooth function and propose a new method to smooth the lower order penalty function (1.3) with $k \in [\frac{1}{2}, 1)$ in this paper.

The remainder of this paper is organized as follows. In Sect. 2, a new smoothing function is proposed. The error estimates are obtained among the optimal objective function values of the smoothed penalty problem, the non-smooth penalty problem, and the original problem. In Sect. 3, the corresponding algorithm is proposed to obtain an approximate solution to (P). The global convergence of the algorithm is proved. In Sect. 4, some numerical experiments are given to illustrate the efficiency of the algorithm. In Sect. 5, some conclusions are presented.

2 A smoothing penalty function

For the lower order penalty problem

$$\min_{x \in R^n} \varphi_{q,k}(x), \tag{LP}$$

in order to establish the global exact penalization, the following assumption is given in [20]. We will consider the smoothing method under the following assumption.

Assumption 2.1

(1) $f_0(x)$ satisfies the coercive condition

$$\lim_{\|x\|\to+\infty} f_0(x) = +\infty.$$

(2) The optimal solution set $G((P))$ is a finite set.

Under Assumption 2.1, problem (P) is equivalent to the following problem:

$$\begin{aligned} &\min f_0(x) \\ &\quad \text{s.t.}\, f_i(x) \le 0, \quad i \in I, \\ &\qquad x \in X, \end{aligned} \tag{P'}$$

where X is a box with $\operatorname{int}(X) \neq \emptyset$.

For any $k \in (0,1)$, penalty problem (LP) is equivalent to the following penalty problem:

$$\min_{x\in X} \varphi_{q,k}(x). \tag{LP'}$$

Now we consider a new smoothing technique to the lower order penalty function (1.3). Let $p_k(t) = (\max\{t,0\})^k$, then

$$\varphi_{q,k}(x) = f_0(x) + q\sum_{i=1}^{m} p_k\bigl(f_i(x)\bigr). \tag{2.1}$$

Define a function $p_{k,\epsilon}(t)$ ($\epsilon > 0$) by

$$p_{k,\epsilon}(t) = \begin{cases} 0, & \text{if } t \le -\epsilon^k, \\ \frac{k}{2}\epsilon^{-1}(t+\epsilon^k)^2, & \text{if } -\epsilon^k < t < 0, \\ (t+\epsilon)^k + \frac{k}{2}\epsilon^{2k-1} - \epsilon^k, & \text{if } t \ge 0, \end{cases} \tag{2.2}$$

where $\frac{1}{2} \le k < 1$. It is easy to see that $p_{k,\epsilon}(t)$ is continuously differentiable and

$$\lim_{\epsilon\to 0^+} p_{k,\epsilon}(t) = p_k(t).$$

The following figure shows the process of function $p_{k,\epsilon}(t)$ approaching function $p_k(t)$.

Figure 1 shows the behavior of $p_{\frac{3}{4},0.01}(t)$ (represented by the dash and dot line), $p_{\frac{3}{4},0.001}(t)$ (represented by the dot line), $p_{\frac{3}{4},0.0001}(t)$ (represented by the dash line), and $p_{\frac{3}{4}}(t)$ (represented by the solid line).

Based on this, we consider the following continuously differentiable penalty function:

$$\varphi_{q,k,\epsilon}(x) = f_0(x) + q\sum_{i=1}^{m} p_{k,\epsilon}\bigl(f_i(x)\bigr), \tag{2.3}$$

where $\lim_{\varepsilon\to 0^+} \varphi_{q,k,\epsilon}(x) = \varphi_{q,k}(x)$.

The corresponding optimization problem to $\varphi_{q,k,\epsilon}(x)$ is as follows:

$$\min_{x\in X} \varphi_{q,k,\epsilon}(x). \tag{SP}$$

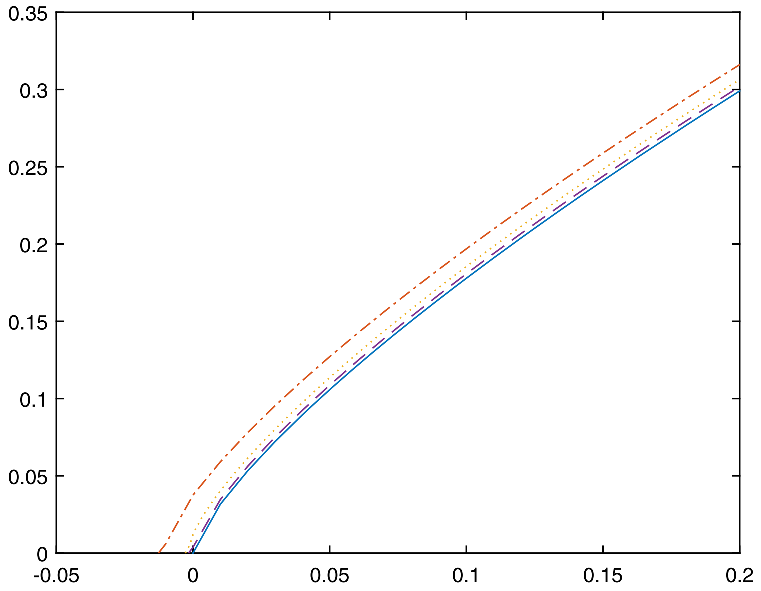

Figure 1 The behavior of $p_{k,\epsilon}(t)$ and $p_k(t)$

For problems (P), (LP'), and (SP), we have the following conclusion.

Lemma 2.1 *For any $x \in X$, $\epsilon > 0$, and $q > 0$, it holds that*

$$-\frac{k}{2}\epsilon^{2k-1}mq \le \varphi_{q,k}(x) - \varphi_{q,k,\epsilon}(x) < \epsilon^k mq, \quad k \in \left[\frac{1}{2}, 1\right).$$

Proof For all $i \in I$, it holds that

$$p_k\big(f_i(x)\big) - p_{k,\epsilon}\big(f_i(x)\big) = \begin{cases} 0, & \text{if } f_i(x) \le -\epsilon^k, \\ -\frac{k}{2}\epsilon^{-1}(f_i(x) + \epsilon^k)^2, & \text{if } -\epsilon^k < f_i(x) < 0, \\ f_i(x)^k - (f_i(x) + \epsilon)^k - \frac{k}{2}\epsilon^{2k-1} + \epsilon^k, & \text{if } f_i(x) \ge 0. \end{cases}$$

Set

$$F(t) = t^k - (t + \epsilon)^k, \quad t \ge 0.$$

Then

$$F'(t) = k\big[t^{k-1} - (t + \epsilon)^{k-1}\big].$$

It is easy to see that function $F(t)$ is monotonically increasing w.r.t. t due to that $k \in [\frac{1}{2}, 1)$. One has

$$-\epsilon^k \le f_i(x)^k - \big(f_i(x) + \epsilon\big)^k \le 0, \quad \text{if } f_i(x) \ge 0.$$

It follows that

$$-\frac{k}{2}\epsilon^{2k-1} \le p_k\big(f_i(x)\big) - p_{k,\epsilon}\big(f_i(x)\big) \le \epsilon^k, \quad \text{if } f_i(x) \ge 0.$$

When $-\epsilon^k < f_i(x) < 0$, one has

$$-\frac{k}{2}\epsilon^{2k-1} < p_k\bigl(f_i(x)\bigr) - p_{k,\epsilon}\bigl(f_i(x)\bigr) < 0.$$

So,

$$-\frac{k}{2}\epsilon^{2k-1} \le p_k\bigl(f_i(x)\bigr) - p_{k,\epsilon}\bigl(f_i(x)\bigr) < \epsilon^k, \quad \forall i \in I. \tag{2.4}$$

It follows from (2.1), (2.3), and (2.4) that

$$-\frac{k}{2}\epsilon^{2k-1}mq \le \varphi_{q,k}(x) - \varphi_{q,k,\epsilon}(x) < \epsilon^k mq$$

by the fact that $q > 0$. □

Theorem 2.1 *For a positive sequence $\{\varepsilon_j\}$, which converges to 0 as $j \to \infty$, assume that x^j is an optimal solution to $\min_{x\in X} \varphi_{q,k,\epsilon_j}(x)$ for some given $q > 0$, $k \in [\frac{1}{2}, 1)$. If $\bar{x}$ is an accumulating point of sequence $\{x^j\}$, then $\bar{x}$ is an optimal solution to $\min_{x\in X} \varphi_{q,k}(x)$.*

Proof It follows from Lemma 2.1 that

$$-\frac{k}{2}\epsilon_j^{2k-1}mq \le \varphi_{q,k}(x) - \varphi_{q,k,\epsilon_j}(x) < \epsilon_j^k mq, \quad \forall x \in X. \tag{2.5}$$

Since x_j is a solution to $\min_{x\in X} \varphi_{q,k,\epsilon_j}(x)$, one has

$$\varphi_{q,k,\epsilon_j}(x_j) \le \varphi_{q,k,\epsilon_j}(x), \quad \forall x \in X. \tag{2.6}$$

It follows from (2.5) and (2.6) that

$$\begin{aligned}
\varphi_{q,k}(x_j) &< \varphi_{q,k,\epsilon_j}(x_j) + \epsilon_j^k mq \\
&\le \varphi_{q,k,\epsilon_j}(x) + \epsilon_j^k mq \\
&\le \varphi_{q,k}(x) + \epsilon_j^k mq + \frac{k}{2}\epsilon_j^{2k-1}mq.
\end{aligned}$$

Letting $j \to \infty$ yields

$$\varphi_{q,k}(\bar{x}) \le \varphi_{q,k}(x).$$

Thus, $\bar{x}$ is an optimal solution to $\min_{x\in X} \varphi_{q,k}(x)$. □

Theorem 2.2 *Let $x^*_{q,k} \in X$ be an optimal solution of problem (LP'), and $\bar{x}_{q,k,\epsilon} \in X$ be an optimal solution of problem (SP) for some $q > 0$, $k \in [\frac{1}{2}, 1)$, and $\epsilon > 0$. Then*

$$-\frac{k}{2}\epsilon^{2k-1}mq \le \varphi_{q,k}\bigl(x^*_{q,k}\bigr) - \varphi_{q,k,\epsilon}(\bar{x}_{q,k,\epsilon}) < \epsilon^k mq.$$

Proof Under the hypothetical conditions, it holds that $\varphi_{q,k}(x^*_{q,k}) \le \varphi_{q,k}(x)$ and $\varphi_{q,k,\epsilon}(\bar{x}_{q,k,\epsilon}) \le \varphi_{q,k,\epsilon}(x)$, $\forall x \in X$.

Therefore, by Lemma 2.1, one has

$$-\frac{k}{2}\epsilon^{2k-1}mq \le \varphi_{q,k}\bigl(x^*_{q,k}\bigr) - \varphi_{q,k,\epsilon}\bigl(x^*_{q,k}\bigr) \le \varphi_{q,k}\bigl(x^*_{q,k}\bigr) - \varphi_{q,k,\epsilon}(\bar{x}_{q,k,\epsilon})$$

and

$$\varphi_{q,k}\bigl(x^*_{q,k}\bigr) - \varphi_{q,k,\epsilon}(\bar{x}_{q,k,\epsilon}) \le \varphi_{q,k}(\bar{x}_{q,k,\epsilon}) - \varphi_{q,k,\epsilon}(\bar{x}_{q,k,\epsilon}) < \epsilon^k mq.$$

This completes the proof. □

Corollary 2.1 *Suppose that Assumption 2.1 holds, and for any $x \in G((P))$, there exists $\lambda^* \in R^m_+$ such that the pair (x^*, λ^*) satisfies the second order sufficient condition (in [20]). Let $x^* \in X$ be an optimal solution of problem (P) and $\bar{x}_{q,k,\epsilon} \in X$ be an optimal solution of problem (SP) for some $q > 0$, $k \in [\frac{1}{2}, 1)$, and $\epsilon > 0$. Then there exists $q^* > 0$ such that, for any $q > q^*$,*

$$-\frac{k}{2}\epsilon^{2k-1}mq \le f_0\bigl(x^*\bigr) - \varphi_{q,k,\epsilon}(\bar{x}_{q,k,\epsilon}) < \epsilon^k mq.$$

Proof It follows from Corollary 2.3 (in [20]) that $x^* \in X$ is an optimal solution of problem (LP'). By Theorem 2.2, one has

$$-\frac{k}{2}\epsilon^{2k-1}mq \le \varphi_{q,k}\bigl(x^*\bigr) - \varphi_{q,k,\epsilon}(\bar{x}_{q,k,\epsilon}) < \epsilon^k mq.$$

Since $\sum_{i=1}^m p_k(f_i(x^*)) = 0$, it holds that

$$\varphi_{q,k}\bigl(x^*\bigr) = f_0\bigl(x^*\bigr) + q\sum_{i=1}^m p_k\bigl(f_i\bigl(x^*\bigr)\bigr) = f_0\bigl(x^*\bigr).$$

This completes the proof. □

Definition 1 For $\epsilon > 0$, if $x \in X$ is such that

$$f_i(x) \le \epsilon, \quad i = 1, 2, \ldots, m,$$

then $x \in X$ is an ϵ-feasible solution of problem (P).

Theorem 2.3 *Let $x^*_{q,k} \in X$ be an optimal solution of problem (LP'), and $\bar{x}_{q,k,\epsilon} \in X$ be an optimal solution of problem (SP) for some $q > 0$, $k \in [\frac{1}{2}, 1)$, and $\epsilon > 0$. If $x^*_{q,k}$ is a* feasible *solution of problem (P), and $\bar{x}_{q,k,\epsilon}$ is an ϵ-feasible solution of problem (P), then*

$$-\frac{k}{2}\epsilon^{2k-1}mq \le f_0\bigl(x^*_{q,k}\bigr) - f_0(\bar{x}_{q,k,\epsilon}) < \left(2^k\epsilon^k + \frac{k}{2}\epsilon^{2k-1}\right)mq.$$

Proof By (2.1), (2.3), and Theorem 2.2, one has

$$\begin{aligned}
-\frac{k}{2}\epsilon^{2k-1}mq &\le \varphi_{q,k}\left(x^*_{q,k}\right) - \varphi_{q,k,\epsilon}(\bar{x}_{q,k,\epsilon}) \\
&= f_0\left(x^*_{q,k}\right) + q\sum_{i=1}^{m} p_k\left(f_i\left(x^*_{q,k}\right)\right) - \left(f_0(\bar{x}_{q,k,\epsilon}) + q\sum_{i=1}^{m} p_{k,\epsilon}\left(f_i(\bar{x}_{q,k,\epsilon})\right)\right) \\
&< \epsilon^k mq.
\end{aligned}$$

Since $\sum_{i=1}^{m} p_k(f_i(x^*_{q,k})) = 0$, it holds that

$$\begin{aligned}
-\frac{k}{2}\epsilon^{2k-1}mq + q\sum_{i=1}^{m} p_{k,\epsilon}\left(f_i(\bar{x}_{q,k,\epsilon})\right) &\le f_0\left(x^*_{q,k}\right) - f_0(\bar{x}_{q,k,\epsilon}) \\
&< \epsilon^k mq + q\sum_{i=1}^{m} p_{k,\epsilon}\left(f_i(\bar{x}_{q,k,\epsilon})\right).
\end{aligned} \tag{2.7}$$

Note that

$$f_i(\bar{x}_{q,k,\epsilon}) \le \epsilon, \quad i \in I.$$

Thus, it follows from (2.2) that

$$0 \le q\sum_{i=1}^{m} p_{k,\epsilon}\left(f_i(\bar{x}_{q,k,\epsilon})\right) \le \left(2^k\epsilon^k + \frac{k}{2}\epsilon^{2k-1} - \epsilon^k\right)mq. \tag{2.8}$$

By (2.7) and (2.8), one has

$$-\frac{k}{2}\epsilon^{2k-1}mq \le f_0\left(x^*_{q,k}\right) - f_0(\bar{x}_{q,k,\epsilon}) < \left(2^k\epsilon^k + \frac{k}{2}\epsilon^{2k-1}\right)mq. \qquad \square$$

Theorems 2.1 and 2.2 show that an optimal solution of (*SP*) is also an approximate optimal solution of (*LP*′) when the error ϵ is sufficiently small. By Theorem 2.3, an optimal solution of (*SP*) is an approximately optimal solution of (*P*) if the optimal solution of (*SP*) is an ϵ-feasible solution of (*P*).

3 A smoothing method

Based on the discussion in the last section, we can design an algorithm to obtain an approximate optimal solution of (*P*) by solving (*SP*).

Algorithm 3.1

Step 1. Take x^0, $\epsilon_0 > 0$, $0 < a < 1$, $q_0 > 0$, $b > 1$, $\epsilon > 0$, and $k \in [\frac{1}{2}, 1)$, let $j = 0$ and go to Step 2.

Step 2. Solve $\min_{x\in R^n} \varphi_{q_j,k,\epsilon_j}(x)$ starting at x^j. Let x^{j+1} be the optimal solution (x^{j+1} can be obtained by a quasi-Newton method).

Step 3. Let $\epsilon_{j+1} = a\epsilon_j$, $q_{j+1} = bq_j$, and $j = j + 1$, then go to Step 2.

Remark Since $0 < a < 1$ and $b > 1$, let $a^{2k-1}b < 1$, as $j \to +\infty$, the sequence $\{\epsilon_j\}$ is gradually decreased to 0, the sequence $\{q_j\}$ is gradually increased to $+\infty$ and $\{\epsilon_j^{2k-1}q_j\}$ is gradually decreased to 0.

Under some mild conditions, the following conclusion shows the global convergence of Algorithm 3.1.

Theorem 3.1 *Suppose that Assumption 2.1 holds, and for any* $\epsilon \in (0, \epsilon_0]$, $q \in [q_0, +\infty)$, *the solution set of* $\min_{x\in R^n} \varphi_{q,k,\epsilon}(x)$ *is nonempty. If* $\{x^{j+1}\}$ *is the sequence generated by Algorithm 3.1 satisfying* $a^{2k-1}b < 1$, *and the sequence* $\{\varphi_{q_j,k,\epsilon_j}(x^{j+1})\}$ *is bounded, then*

(1) $\{x^{j+1}\}$ *is bounded.*

(2) *Any limit point of* $\{x^{j+1}\}$ *is an optimal solution of* (*P*).

Proof (1) It follows from (2.3) that

$$\varphi_{q_j,k,\epsilon_j}\left(x^{j+1}\right) = f_0\left(x^{j+1}\right) + q_j \sum_{i=1}^{m} p_{k,\epsilon_j}\left(f_i\left(x^{j+1}\right)\right), \quad j = 0, 1, 2, \ldots. \tag{3.1}$$

By hypothesis, there exists some number L such that

$$L > \varphi_{q_j,k,\epsilon_j}\left(x^{j+1}\right), \quad j = 0, 1, 2, \ldots. \tag{3.2}$$

For the sake of contradiction, suppose that $\{x^{j+1}\}$ is unbounded. Without loss of generality, we assume that $\|x^{j+1}\| \to \infty$ as $j \to \infty$.

By (2.2), (3.1), and (3.2), one has

$$L > f_0\left(x^{j+1}\right), \quad j = 0, 1, 2, \ldots,$$

which results in a contradiction with Assumption 2.1(1).

(2) Without loss of generality, we assume $x^{j+1} \to x^*$ as $j \to \infty$.

To prove x^* is the optimal solution of (*P*), it is only needed to show that $x^* \in X_0$ and $f_0(x^*) \le f_0(x)$, $\forall x \in X_0$.

To show that $x^* \in X_0$, we outline a proof by contradiction. We presuppose that $x^* \notin X_0$, then there exist $\delta_0 > 0$, $i_0 \in I$, and the subset $J \subset N$ such that

$$f_{i_0}\left(x^{j+1}\right) \ge \delta_0 > \epsilon_j, \quad \forall j \in J,$$

where N is the natural number set.

By Step 2, (2.2), and (2.3), for any $x \in X_0$, one has

$$\begin{aligned} f_0\left(x^{j+1}\right) + q_j\left((\delta_0 + \epsilon_j)^k + \frac{k}{2}\epsilon_j^{2k-1} - \epsilon_j^k\right) &\le \varphi_{q_j,k,\epsilon_j}\left(x^{j+1}\right) \\ &\le \varphi_{q_j,k,\epsilon_j}(x) \\ &\le f_0(x) + m\frac{k}{2}\epsilon_j^{2k-1}q_j. \end{aligned}$$

It follows that

$$f_0\left(x^{j+1}\right) + q_j\left((\delta_0 + \epsilon_j)^k - \epsilon_j^k\right) \le f_0(x) + (m-1)\frac{k}{2}\epsilon_j^{2k-1}q_j, \quad \forall x \in X_0,$$

which contradicts with $q_j \to +\infty, \epsilon_j \to 0$, and $\epsilon_j^{2k-1}q_j \to 0$, as $j \to \infty$. Then we have that $x^* \in X_0$.

Next, we show that $f_0(x^*) \le f_0(x)$, $\forall x \in X_0$.

For this, by Step 2, (2.2), and (2.3), it holds that

$$f_0\left(x^{j+1}\right) \le \varphi_{q_j,k,\epsilon_j}\left(x^{j+1}\right) \le \varphi_{q_j,k,\epsilon_j}(x) \le f_0(x) + m\frac{k}{2}\epsilon_j^{2k-1}q_j, \quad \forall x \in X_0.$$

Letting $j \to \infty$ yields that

$$f_0\left(x^*\right) \le f_0(x).$$

Therefore, any limit point of $\{x^{j+1}\}$ is an optimal solution of (P). □

4 Numerical examples

In this section, we will do some numerical experiments to show the efficiency of Algorithm 3.1.

Example 4.1 Consider the following optimization problem considered in [18, 22, 23]:

$$\begin{aligned} &\min f_0(x) = x_1^2 + x_2^2 + 2x_3^2 + x_4^2 - 5x_1 - 5x_2 - 21x_3 + 7x_4 \\ &\quad \text{s.t.}\, f_1(x) = 2x_1^2 + x_2^2 + x_3^2 + 2x_1 + x_2 + x_4 - 5 \le 0, \\ &\qquad f_3(x) = x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_1 - x_2 + x_3 - x_4 - 8 \le 0, \\ &\qquad f_3(x) = x_1^2 + 2x_2^2 + x_3^2 + 2x_4^2 - x_1 - x_4 - 10 \le 0. \end{aligned}$$

For this problem, we let $k = \frac{3}{4}$, $\epsilon_0 = 0.01$, $a = 0.01$, $q_0 = 1$, $b = 2$, $\epsilon = 10^{-16}$. With different starting points, numerical results of Algorithm 3.1 are shown in Tables 1, 2, and 3.

From Tables 1, 2, 3, we know that the obtained approximate optimal solutions are similar, which shows that the numerical result of Algorithm 3.1 does not depend on the section of the starting points for this example. In [18], the objective function value $f_0(x^*) = -44.23040$ was obtained in the forth iteration. From the numerical results given in [22], we know that the optimal solution is $x^* = (0.1585001, 0.8339736, 2.014753, -0.959688)$ with the objective function value $f_0(x^*) = -44.22965$. In [23], the objective function value obtained in

Table 1 Numerical results for Example 4.1 with $x^0 = (0, 0, 0, 0)$

j	x^{j+1}	q_j	ϵ_j	$f_1(x^{j+1})$	$f_2(x^{j+1})$	$f_3(x^{j+1})$	$f_0(x^{j+1})$
0	(0.185009, 0.804369, 2.015460, –0.952409)	1	0.01	–4.797079	–0.00109	–2.028111	–44.225926
1	(0.169902, 0.835670, 2.008151, –0.965196)	2	0.0001	–9.748052	–9.337847	–1.883271	–44.231252

Table 2 Numerical results for Example 4.1 with $x^0 = (2, 0, 3.5, 0)$

j	x^{j+1}	q_j	ϵ_j	$f_1(x^{j+1})$	$f_2(x^{j+1})$	$f_3(x^{j+1})$	$f_0(x^{j+1})$
0	(0.169693, 0.835634, 2.008291, –0.965082)	1	0.01	–9.502428	–8.676884	–1.883244	–44.231403

Table 3 Numerical results for Example 4.1 with $x^0 = (2, 2, 2, 0.5)$

j	x^{j+1}	q_j	ϵ_j	$f_1(x^{j+1})$	$f_2(x^{j+1})$	$f_3(x^{j+1})$	$f_0(x^{j+1})$
0	(0.169691, 0.835633, 2.008294, –0.965080)	1	0.01	–9.502279	–8.676796	–1.883249	–44.231403

Table 4 Numerical results for Example 4.2 with $k = \frac{3}{4}$

j	x^{j+1}	q_j	ϵ_j	$f_1(x^{j+1})$	$f_2(x^{j+1})$	$f_0(x^{j+1})$
0	(3.4217, 2.7082)	2	0.001	4.1300	–0.0053	–15.2492
1	(0.8022, 1.1978)	20	0.000001	0.0000	–0.4066	–7.1999

Table 5 Numerical results for Example 4.2 with $k = \frac{3}{5}$

j	x^{j+1}	q_j	ϵ_j	$f_1(x^{j+1})$	$f_2(x^{j+1})$	$f_0(x^{j+1})$
0	(4.0607, 3.0227)	2	0.001	5.0834	–0.0153	–16.0434
1	(0.8027, 1.1971)	20	0.000001	–0.0003	–0.4086	–7.1992

Table 6 Numerical results for Example 4.2 with $k = \frac{8}{9}$

j	x^{j+1}	q_j	ϵ_j	$f_1(x^{j+1})$	$f_2(x^{j+1})$	$f_0(x^{j+1})$
0	(2.6356, 2.3168)	2	0.001	2.9523	–0.0020	–13.7027
1	(0.8005, 1.1995)	20	0.000001	0.0000	–0.4015	–7.2000

the 25th iteration is $f_0(x^*) = -44$. Hence, the numerical results obtained by Algorithm 3.1 are better than the numerical results given in [18, 22, 23] for this example.

Example 4.2 Consider the following problem considered in [17]:

$$\begin{aligned} \min f_0(x) &= -2x_1 - 6x_2 + x_1^2 - 2x_1x_2 + 2x_2^2 \\ \text{s.t.}\, f_1(x) &= x_1 + x_2 - 2 \le 0, \\ f_2(x) &= -x_1 + 2x_2 - 2 \le 0, \\ x_1, x_2 &\ge 0. \end{aligned}$$

For this problem, we let $x^0 = (0,0)$, $\epsilon_0 = 0.001$, $a = 0.001$, $q_0 = 2$, $b = 10$, $\epsilon = 10^{-16}$. With different k, numerical results of Algorithm 3.1 are shown in Tables 4, 5, and 6.

From Tables 4, 5, 6, we can see that almost the same approximate optimal solutions are obtained for different k in this example. The objective function value is similar to the objective function value $f_0(x^*) = -7.2000$ with $x^* = (0.8000, 1.2000)$ obtained in the forth iteration in [17].

Example 4.3 Consider the following problem considered in [24] and [25] (Test Problem 6 in Sect. 4.6):

$$\begin{aligned} \min f_0(x) &= -x_1 - x_2 \\ \text{s.t.}\, f_1(x) &= x_2 - 2x_1^4 + 8x_1^3 - 8x_1^2 - 2 \le 0, \\ f_2(x) &= x_2 - 4x_1^4 + 32x_1^3 - 88x_1^2 + 96x_1 - 36 \le 0, \\ 0 \le x_1 &\le 3, \qquad 0 \le x_2 \le 4. \end{aligned}$$

For this problem, we set $k = \frac{2}{3}$, $x^0 = (0,0)$, $\epsilon_0 = 0.01$, $a = 0.01$, $q_0 = 5$, $b = 2$, $\epsilon = 10^{-16}$. The numerical results of Algorithm 3.1 are shown in Table 7.

We set $k = \frac{8}{9}$, $x^0 = (1.0, 1.5)$, $\epsilon_0 = 0.1$, $a = 0.1$, $q_0 = 5$, $b = 3$, $\epsilon = 10^{-16}$. The numerical results of Algorithm 3.1 are shown in Table 8.

Table 7 Numerical results for Example 4.3 with $x^0 = (0,0)$

j	x^{j+1}	q_j	ϵ_j	$f_1(x^{j+1})$	$f_2(x^{j+1})$	$f_0(x^{j+1})$
0	(2.329795, 3.133729)	5	10^{-2}	–0.047009	–0.043471	–5.463524
1	(2.329238, 3.173320)	10	10^{-4}	–0.002868	–0.006501	–5.502557
2	(2.329452, 3.177637)	20	10^{-6}	–0.000302	–0.001176	–5.507089
3	(2.329626, 3.177558)	40	10^{-8}	–0.001802	–0.000436	–5.507185

Table 8 Numerical results for Example 4.3 with $x^0 = (1.0, 1.5)$

j	x^{j+1}	q_j	ϵ_j	$f_1(x^{j+1})$	$f_2(x^{j+1})$	$f_0(x^{j+1})$
0	(2.330261, 3.061875)	5	10^{-1}	–0.1226776	–0.1131323	–5.392137
1	(2.329664, 3.161611)	15	10^{-2}	–0.018055	–0.016207	–5.491275
2	(2.329639, 3.171941)	45	10^{-3}	–0.007524	–0.005993	–5.501580
3	(2.329560, 3.177804)	135	10^{-4}	–0.001013	–0.000503	–5.507363
4	(2.329593, 3.177793)	405	10^{-5}	–0.001297	–0.000357	–5.507386
5	(2.329622, 3.177781)	1215	10^{-6}	–0.001544	–0.000234	–5.507403

Table 9 Numerical results for Example 4.3 with $x^0 = (2, 0.5)$

j	x^{j+1}	q_j	ϵ_j	$f_1(x^{j+1})$	$f_2(x^{j+1})$	$f_0(x^{j+1})$
0	(2.330460, 3.179900)	2	10^{-5}	–0.006287	0.005832	–5.510360
1	(2.329672, 3.179735)	20	10^{-8}	–0.000001	0.001957	–5.509408
2	(2.329672, 3.179735)	200	10^{-11}	–0.000000	0.001957	–5.509407
3	(2.329541, 3.178391)	2000	10^{-14}	–0.000000	–0.000000	–5.507933

We set $k = \frac{3}{4}, x^0 = (2, 0.5), \epsilon_0 = 0.00001, a = 0.001, q_0 = 2, b = 10, \epsilon = 10^{-16}$. The numerical results of Algorithm 3.1 are shown in Table 9.

In [24], with three different starting points, similar numerical results are given with $k = \frac{2}{3}$. The optimal solution $(2.329517, 3.178421)$ is given with the objective function value -5.507938. In [25], the optimal solution $(2.3295, 3.1783)$ is given with the objective function value -5.5079. The numerical results of Example 4.3 are similar to the numerical results of [24] and [25] in this example.

From Tables 7, 8, 9, we can see that we need to adjust the parameters q_0, ϵ_0, a, b to get the better numerical results with different k and x^0. Usually, ϵ_0 may be $0.5, 0.1, 0.01, 0.001$, or smaller digits, and $a = 0.5, 0.1, 0.01$, or 0.001. q_0 may be $1, 2, 3, 5, 10, 100$, or larger digits, and $b = 2, 3, 5, 10$, or 100.

5 Concluding remarks

In this paper, we proposed a method to smooth the lower order exact penalty function with $k \in [\frac{1}{2}, 1)$ for inequality constrained optimization. Furthermore, we proved that the algorithm based on the smoothed penalty functions is globally convergent under mild conditions. The given numerical experiments show that the algorithm is effective.

Funding

This work was supported by the National Natural Science Foundation of China (71371107 and 61373027) and the Natural Science Foundation of Shandong Province (ZR2016AM10)

Competing interests

The authors declare that they have no competing interests.

Authors' contributions

SL and NN drafted the manuscript. SL revised it. All authors read and approved the final manuscript.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 11 March 2018 Accepted: 5 June 2018 Published online: 11 June 2018

References

1. Wang, C.W., Wang, Y.J.: A superlinearly convergent projection method for constrained systems of nonlinear equations. J. Glob. Optim. **40**, 283–296 (2009)
2. Qi, L.Q., Wang, F., Wang, Y.J.: Z-eigenvalue methods for a global polynomial optimization problem. Math. Program. **118**, 301–316 (2009)
3. Qi, L.Q., Tong, X.J., Wang, Y.J.: Computing power system parameters to maximize the small signal stability margin based on min-max models. Optim. Eng. **10**, 465–476 (2009)
4. Chen, H.B., Wang, Y.J.: A family of higher-order convergent iterative methods for computing the Moore–Penrose inverse. Appl. Math. Comput. **218**, 4012–4016 (2011)
5. Wang, G., Che, H.T., Chen, H.B.: Feasibility-solvability theorems for generalized vector equilibrium problem in reflexive Banach spaces. Fixed Point Theory Appl. **2012**, Article ID 38 (2012)
6. Wang, G., Yang, X.Q., Cheng, T.C.E.: Generalized Levitin–Polyak well-posedness for generalized semi-infinite programs. Numer. Funct. Anal. Optim. **34**, 695–711 (2013)
7. Wang, Y.J., Qi, L.Q., Luo, S.L., Xu, Y.: An alternative steepest direction method for optimization in evaluating geometric discord. Pac. J. Optim. **10**, 137–149 (2014)
8. Wang, Y.J., Caccetta, L., Zhou, G.L.: Convergence analysis of a block improvement method for polynomial optimization over unit spheres. Numer. Linear Algebra Appl. **22**, 1059–1076 (2015)
9. Wang, Y.J., Liu, W.Q., Caccetta, L., Zhou, G.: Parameter selection for nonnegative l_1 matrix/tensor sparse decomposition. Oper. Res. Lett. **43**, 423–426 (2015)
10. Sun, M., Wang, Y.J., Liu, J.: Generalized Peaceman–Rachford splitting method for multi-block separable convex programming with applications to robust PCA. Calcolo **54**, 77–94 (2017)
11. Sun, H.C., Sun, M., Wang, Y.J.: Proximal ADMM with larger step size for two-block separable convex programming and its application to the correlation matrices calibrating problems. J. Nonlinear Sci. Appl. **10**(9), 5038–5051 (2017)
12. Konnov, I.V.: The method of pairwise variations with tolerances for linearly constrained optimization problems. J. Nonlinear Var. Anal. **1**, 25–41 (2017)
13. Qin, X., Yao, J.C.: Projection splitting algorithms for nonself operators. J. Nonlinear Convex Anal. **18**, 925–935 (2007)
14. Luu, D.V., Mai, T.T.: Optimality conditions for Henig efficient and superefficient solutions of vector equilibrium problems. J. Nonlinear Funct. Anal. **2018**, Article ID 18 (2018)
15. Chieu, N.H., Lee, G.M., Yen, N.D.: Second-order subdifferentials and optimality conditions for C1-smooth optimization problems. Appl. Anal. Optim. **1**, 461–476 (2017)
16. Zangwill, W.I.: Nonlinear programming via penalty functions. Manag. Sci. **13**(5), 334–358 (1967)
17. Wu, Z.Y., Lee, H.W.J., Bai, F.S., Zhang, L.S., Yang, X.M.: Quadratic smoothing approximation to l_1 exact penalty function in global optimization. J. Ind. Manag. Optim. **1**, 533–547 (2005)
18. Lian, S.J.: Smoothing approximation to l_1 exact penalty function for inequality constrained optimization. Appl. Math. Comput. **219**(6), 3113–3121 (2012)
19. Lian, S.J., Zhang, L.S.: A simple smooth exact penalty function for smooth optimization problem. J. Syst. Sci. Complex. **25**(5), 521–528 (2012)
20. Wu, Z.Y., Bai, F.S., Yang, X.Q., Zhang, L.S.: An exact lower order penalty function and its smoothing in nonlinear programming. Optimization **53**, 51–68 (2004)
21. Meng, Z.Q., Dang, C.Y., Yang, X.Q.: On the smoothing of the squareroot exact penalty function for inequality constrained optimization. Comput. Optim. Appl. **35**, 375–398 (2006)
22. Lian, S.J., Han, J.L.: Smoothing approximation to the square-order exact penalty functions for constrained optimization. J. Appl. Math. **2013**, Article ID 568316 (2013)
23. Lasserre, J.B.: A globally convergent algorithm for exact penalty functions. Eur. J. Oper. Res. **7**, 389–395 (1981)
24. Lian, S.J., Duan, Y.Q.: Smoothing of the lower-order exact penalty function for inequality constrained optimization. J. Inequal. Appl. **2016**, Article ID 185 (2016). https://doi.org/10.1186/s13660-016-1126-9
25. Flondas, C.A., Pardalos, P.M.: A Collection of Test Problems for Constrained Global Optimization Algorithms. Springer, Berlin (1990)